HHS Public Access

Author manuscript

Curr Protoc Bioinformatics. Author manuscript; available in PMC 2020 September 01.

Published in final edited form as:

Curr Protoc Bioinformatics. 2019 September ; 67(1): e81. doi:10.1002/cpbi.81.

Ploidy and purity adjusted DNA allele specific analysis using CLONETv2

Davide Prandi[1], **Francesca Demichelis**[1,2,3]

[1]Department of Cellular, Computational and Integrative Biology (CIBIO), University of Trento, Trento Italy [2]Englander Institute for Precision Medicine, New York Presbyterian Hospital - Weill Cornell Medicine, New York, New York [3]Department of BioMedical Research, University of Bern, 3012 Bern, Switzerland

Abstract

High throughput DNA sequencing technology provides base level and statistically rich information about the genomic content of a sample. In the context of cancer research and of precision oncology, thousands of genomes from paired tumor and matched normal samples are being profiled and processed to determine somatic copy number changes and single nucleotide variations. Higher order informative analyses, as allele specific copy number assessments or subclonality quantification, require reliable estimates of tumor DNA ploidy and tumor cellularity. CLONETv2 provides a complete set of functions to process matched normal and tumor pairs using patient specific genotype data, is independent of low-level tools (e.g., aligner, segmentation, mutation caller) and offers high-level functions to compute allele specific copy number from segmented data and to identify subclonal population in the input sample. CLONETv2 is applicable to whole genome, whole exome and targeted sequencing data generated from both tissue and liquid biopsy samples.

Keywords

Cancer Genomics; Purity; Ploidy; Allele specific Analysis; Clonality

INTRODUCTION

Massive sequencing efforts, as by The Cancer Genome Atlas (TCGA) and the International Cancer Genome Consortium (ICGC), have generated a comprehensive collection of sequenced genomes of cancer patients, opening a new era for genomics. Advanced analyses of genomic sequencing data require accurate estimation of DNA cellularity (purity, 1 – DNA admixture) and tumor ploidy to allow for proper comparative computations. DNA admixture refers to the amount of non-cancer cells in a tumor sample, while ploidy represents the average number of chromosomes set in a cell. Human healthy cells are diploid, whereas tumor cells often demonstrate a dramatically variable number of ploidy also dependent on the tumor type (Chunduri and Storchova, 2019; Danielsen et al., 2016). The impact on tumor evolution and prognosis of ploidy changes is yet not clear, but recent pan-cancer studies have shed some light. In a primary tumor pan-cancer cohort from the TCGA project, cell

proliferation and immune evasion, two hallmarks of cancer, resulted deregulated in high aneuploidy samples (Davoli et al., 2017; Taylor et al., 2018). In a pan-cancer cohort of 9,692 patients with advanced disease, aneuploidy also associated with poor survival (Bielski et al., 2018).

A recent review (Aran et al., 2015) highlighted the importance of purity estimation in analyzing sequencing data. For instance, phylogenetic reconstruction of tumor evolution from multi-sample DNA sequencing data from a single patient strictly relies on the quantification of the variant allelic fraction (VAF) of single nucleotide variants (SNV) (Gundem et al., 2015) that is affected by both DNA admixture (normal cell dilute SNV VAFs) and by ploidy (polyploidy increases the total number of alleles) of each tumor sample. The same issues also affect the determination of the absolute number of copies of a genomic segment in a tumor sample (Carter et al., 2012). Many computational methods identify somatic copy number aberrations from the relative DNA amount between a tumor and its matched normal sample, but fair estimation of the integer number of copies of each allele requires purity and ploidy adjustments (Bao et al., 2014).

These considerations call for the development of computational tools to quantify tumor purity and ploidy. In the pre-sequencing era, several tools were developed for high-density single-nucleotide polymorphisms (SNP) array data (e.g., (Carter et al., 2012; Van Loo et al., 2010)), where typically the tumor over control signal ratio (hereafter LogR) and the abundance of allele specific signal (B allele frequency, BAF) distributions are jointly analyzed to infer DNA admixture and ploidy. However, array based tools are limited by the number of the assayed genomic bases (mainly in the range 0,5–2M of sites) and by the signal dynamic range. Next generation sequencing platforms overcome these limitations while preserving the same data feature to exploit (Aran et al., 2015); allelic fraction (AF) of inherited heterozygous SNP loci (hereafter called *informative SNPs*) and sequencing coverage resemble the BAF and the LogR data of SNP arrays, respectively. The statistically richer data offered by sequencing enables more complex analysis such as allele specific copy number and clonality estimates.

In general, available methods to estimate ploidy and DNA admixture adopt a global approach and distribution of AFs and LogR values are conjunctly used to infer DNA admixture and ploidy. Intuitively, the AF of informative SNPs is distributed around 0.5 in a 100% admixed tumor sample (up to reference mapping bias (Degner et al., 2009)) and lower AFs imply lower DNA admixture. LogR data is used as a covariate, as the AF also depends on the number of available alleles. If no tumor cell subpopulations are present (i.e., the copy number profile of a tumor sample is homogeneous, i.e., the ratio of subclonal deletions/amplifications is low), global inference approaches well capture the DNA admixture content. However, in the presence of complex genomic events, such as chromothripsis (Stephens et al., 2011) or chromoplexy (Baca et al., 2013), or upon multiple treatments that diversify the tumor cell population, global approaches are suboptimal.

CLONET (Prandi et al., 2014) (CLONality Estimate in Tumor) is a stand-alone tool specifically designed with a local approach to clonality estimation to handle heterogeneous tumor samples. Briefly, consider a tumor sample T with a hemizygous deletion HeD and the

set of informative SNPs *S* lying in the *HeD*. The AF value of SNPs in *S* is the convolution of the AF of the different cell populations composing *T*. If *HeD* is subclonal, the tumor sample comprises three main cell populations: i. non-tumor cells contributing to DNA admixture, with expected AFs of SNPs in *S* around 0.5; ii. tumor cells not harboring the deletion thus the AFs of SNPs in *S* cannot be distinguished from non-tumor cells; iii. tumor cells with *HeD*, where the AFs could either be equal to 1 (the deleted allele harbor the alternative base) or equal to 0 (the deleted allele harbor the reference allele). Based on the observation that apparent DNA admixture is higher in subclonal deletions with respect to clonal deletions, CLONET estimates DNA admixture at each hemizygous deletion and then identifies the most clonal deletions to finally nominate sample DNA admixture. This results in a more accurate DNA admixture estimation that would otherwise be overestimated in tumors with a significant fraction of subclonal deletions.

Here, we present CLONETv2, an R package (Team, 2017) available at The Comprehensive R Archive Network (https://cran.r-project.org/), that includes significant improvements over the original CLONET implementation. This is the result of its application to several clinical cohorts, including tissue and plasma samples, and on a variety of sequencing platforms, as whole genome, whole exome, and targeted sequencing panels. In Carreira et al. (Carreira et al., 2014), CLONET was used to estimate DNA admixture from a custom sequencing panel of about 40kb designed to analyze circulating tumor DNA of plasma samples from metastatic patients and the algorithm was modified to improve sensitivity in samples with less than 10% of tumor cells. In Beltran et al. (Beltran et al., 2016), CLONET was extended to provide allele specific copy number data from whole exome sequencing experiments; for each genomic segment in each study cohort tumor, the study reports the number of copies of each allele using ploidy, DNA admixture, LogR and the AF of informative SNPs. In Faltas et al. (Faltas et al., 2016), clonality analysis capability of CLONET was improved to account for complex allele specific combinations and single nucleotide variants (SNVs). Since its initial conception and application to whole genome sequencing data (Baca et al., 2013; Prandi et al., 2014), CLONET improvements have been used in several studies (including (Beltran et al., 2015; Boysen et al., 2015; Cancer Genome Atlas Research, 2015; Mu et al., 2017)). Here, we present a documented CLONETv2 version to uniformly highlight the approach features and propose it as an R package to make the tool available to a broader audience.

BASIC PROTOCOL 1

Compute beta table—All reads of a human DNA next-generation sequencing experiment that map within a genomic segment derive from either one of the parental chromosomes of origin. Reads can be split into two sets: a *copy number neutral* set that contains equal number of reads from the maternal and the paternal chromosomes, and an *active reads* set that includes sequences from only one parent. Generally speaking, given two random reads, it is impossible to determine whether or not they represent the same allele; however, if the two reads span an informative SNP then the allele of origin can be identified. For reads over informative SNPs, the number of reads (local coverage) supporting the reference or the alternative SNP represents the number of copies and the origin of the alleles present in the tumor sample. Each informative SNP can be characterized by its allelic fraction (AF) that

depends on the genomic context. For instance, let us consider the two informative SNPs within a mono-allelic deletion of the genomic segment A in Figure 1A. At position p_1, only the alternative allele is present and the AF is 1, while at position p_n the alternative allele is deleted and the AF is 0. Instead, in the wild type genomic segment B, the AF values of informative SNPs at positions p_{n+1} and p_m are distributed around 0.5, as both alleles equally contribute to the local coverage. Now, the percentage of neutral reads (called *beta*) at p_1 and p_n is equal to 0, regardless of which allele is deleted, whereas at wild type genomic positions p_{n+1} and p_m approximates 1, as no active reads are present. Overall, SNPs within somatically aberrant segments are easier to characterize using the beta values as opposed to the AFs, as the former is independent from the deleted allele. In a heterogeneous tumor sample, the distributions of AFs and betas result from the convolution of the distribution observed in basic wild type and mono-allelic deleted segments. As an example, Figure 1B depicts the distribution of the AF and the associated beta of the informative SNPs in genomic segments A and B in case of a normal cell, while Figure 1C and 1D show how distributions change in tumor cells with mono-allelic deletion of only genomic segment A, or of both A and B, respectively. Figure 1E considers tumor sample with one *Normal cell* (Figure 1B) and nine *Tumor cells 1* (Figure 1C). The DNA admixture is 1/10 and the AF could assume values around 1/11 or 10/11, while beta is 2/11. Genomic segment B is not deleted and therefore the AF and the beta are as in *Normal cell*. Figure 1F mimics a more complex situation involving one *Normal Cell* (Figure 1B), three *Tumor cells 1* (Figure 1C), and six *Tumor cells 2* (Figure 1D). The AF and the beta of informative SNPs in genomic segment A are as in Figure 1E, but only the six *Tumor cells 2* carry the mono-allelic deletion of genomic segment B. In this case, the AF distribution modes are centered in 4/14 and in 10/14, depending on the depleted base, while beta is 8/14. The full characterization of beta is described in (Prandi et al., 2014), while in Beltran et al [PMID: 26855148] we defined CLONET master equations that describe allele specific copy number of maternal and paternal alleles, cnM and cnP, as a function of the percentage of neutral reads *beta*, the log2 ratio values adjusted by ploidy *LogRp*, and the DNA admixture *G*

$$\begin{cases} cnM = \dfrac{(2-beta)\left(beta\ 2^{LogRp} - G\right) + 2G(1-beta)}{(1-G)beta} \\ cnP = \dfrac{beta\ 2^{LogRp} - G}{1-G} \end{cases} \quad 1$$

where maternal and paternal allele are arbitrarily assigned. Figure 2 sketches the transformation of the log2 ratio space implied by Equation 1. Figure 2A reports the histogram of the log2 ratio signal in a tumor sample; peaks in the distribution correspond to different copy number states while deviations from the position of the expected peaks (below) depend on ploidy and DNA admixture values. It is difficult to identify the peak that corresponds to wild type segments using only log2 ratio signal. When expanding the mono-dimensional LogR space with beta (Figure 2B), segments that contribute to the same peak along the LogR dimension form different clusters in the beta vs LogR space. Of note, the beta vs LogR plot still reflects ploidy and DNA admixture while the cnM and cnP space (see

Equation 1) allows for straightforward interpretation of copy number and clonality status of each genomic segment.

The function `compute_beta_table` estimates the beta of a genomic segment as described in Carreira et al. (Carreira et al., 2014). The function `compute_beta_table` includes the following input:

- `seg_tb`: a table resulting from DNA segmentation. For each genomic segment, the table reports chromosome, start/end position, and the log2 ratio of the tumor over the normal coverage, as defined in the Circular Binary Segmentation algorithm (Olshen et al., 2004);
- `pileup_normal`, `pileup_tumor`: two tables reporting allelic fraction and coverage of SNPs in normal and matched tumor samples, respectively. For each SNP, each table reports genomic coordinates (chromosome and position), allelic fraction, and coverage;
- `min_af_het_snps`, `max_af_het_snps`: for each SNP in the `pileup_normal` table, set minimum and maximum allelic fraction to consider the SNP as informative;
- `min_required_snps`: minimum number of informative SNPs in a genomic segment from `seg_tb` to retain the segment;
- `min_coverage`: minimum mean coverage of informative SNPs to retain a segment.

As output, the function `compute_beta_table` extends the input table `seg_tb`. For each segment in `seg_tb`, the function `compute_beta_table` returns the following values:

- `beta`: estimated value for the input segment;
- `nsnps`: number of informative SNPs in the input segment;
- `cov`: mean coverage of informative SNPs in the input segment;
- `n_beta`: estimated value for the input segment considering the matched normal sample. This value is expected to be 1, except for germline CNV or sequencing related errors.

The interpretation of the function `compute_beta_table` output is not an easy task due to the identifiability problem, i.e. more than one combination of ploidy and DNA admixture fit the observed data (Li and Xie, 2014). However, upon definition of ploidy and DNA admixture, Equation 1 completely defines the absolute copy number of both alleles. We will exploit this capability in Support Protocol 2, where Equation 1 is used to plot expected beta and log2 ratio against estimated values. The optional parameter `plot_stats` of the `compute_beta_table` function plots useful summary statistics for sanity check of the output. In particular, when `plot_stats` is `TRUE`, the function returns:

- number of processed segments: the number of segments in the input `seg_tb` table;

- number of segments with a valid beta estimate: the number input segments for which beta value is computed. This value is affected by the number of informative SNPs and their mean coverage;
- quantiles of input segment lengths: the quantiles of the distribution of the length of the input segments. The expected distribution depends on the segmentation algorithm used to produce `seg_tb` table. However, small values result in a low number of informative SNPs, while large segments may indicate under-segmentation that in turn affects beta estimates;
- quantiles of informative SNPs input segment coverage: the quantiles of the distribution of the mean coverage of the input segments. Expected coverage depends on the sequencing experiment, but low value may indicate problems with the input sample;
- quantiles of number of informative SNPs per input segment: the quantiles of the distribution of the number of informative SNPs in the input segments. Expected number of informative SNPs per kb is about 0.33 (based on common SNPs); therefore this value combined with input segment lengths gives information about the quality of the pileup data.

Necessary Resources

Hardware—A 64-bit computer running Linux with at least 8 GB of RAM

Software—The library has been tested with R version 3.5.2 and R libraries parallel 3.5.2, ggplot2 3.1.0, sets 1.0–18, arules 1.6–3, ggrepel 0.8.0.

Step annotations

1. Prepare tumor and normal pileups as described in SUPPORT PROTOCOL 1 or with other computational tools. The output of this step comprises two files tumor.pileup and normal.pileup.
2. Prepare tumor segmented data in file tumor_segments.txt and with columns compatible with parameter `seg_tb` described above.
3. Run R from command line

```
$ R
```

4. Install CLONETv2 the first time

```
> install.packages("CLONETv2")
```

5. Load the library

```
> library(CLONETv2)
```

6. Load input files

```
> seg_tb <- read.table(system.file("sample.seg", package =
"CLONETv2"),header = T, as.is=T)
> pileup_tumor <- read.table(system.file("sample_tumor_pileup.tsv", package
= "CLONETv2"),header = T, as.is=T)
> pileup_normal <- read.table(system.file("sample_normal_pileup.tsv",
package = "CLONETv2"),header = T, as.is=T)
```

7. Compute beta for each input segment with default parameters

```
> bt <- compute_beta_table(seg_tb, pileup_tumor, pileup_normal)
```

8. Compute beta activating plot_stats parameter

```
> bt <- compute_beta_table(seg_tb, pileup_tumor, pileup_normal, plot_stats=T)
```

This results in the following output:

```
Computed beta table of sample "sample1"
 Number of processed segments: 65
 Number of segments with valid beta: 49 (75%)
 Quantiles of input segment lengths:
  0%  :     2860
  25% : 17504185
  50% : 38004799
  75% : 59311449
  100%:147311449
 Quantiles of input segment coverage:
  0%  : 47.0000
  25% :137.7893
  50% :168.3820
  75% :186.6769
  100%:695.6145
 Quantiles of number of informative SNPs per input segment:
  0%  :  0
  25% : 12
  50% : 99
```

```
75% :213
100%:404
```

SUPPORT PROTOCOL 1

Prepare pileup data—This protocol describes the steps to prepare pileup data from a set of SNPs and matched tumor and normal bam files (Li et al., 2009). Tables `pileup_normal` and `pileup_tumor` report allelic fraction and coverage for a set of SNP positions. Candidate SNP positions can be downloaded directly from dbSNP ftp server (http://ftp.ncbi.nlm.nih.gov/snp/). We suggest starting from the largest possible set of SNPs, as the larger the number of informative SNPs, the more reliable the CLONETv2 estimates. Pileups from bam files can be obtained with several tools with no effect on the function `compute_beta_table`. Here we describe how to prepare pileups using ASEQ (Romanel et al., 2015) a tool freely available at http://demichelislab.eu/tools/ASEQ.

Necessary Resources

Hardware—A 64-bit computer running Linux with at least 8 GB of RAM

Software—ASEQ, curl

Input files—BAM files `tumor.bam` and `normal.bam` including aligned reads from genomic sequencing experiments of matched tumor and normal DNA samples, respectively. VCF (Degner et al., 2009) file `known_snp_positions.vcf` reporting known SNPs positions. ASEQ requires that the input VCF only lists SNPs, i.e. columns ALT and REF must contain one of the values A, C, G, or T. ASEQ parameters includes

- `mrq`: minimum read quality. ASEQ does not consider as part of the pileup reads with read quality < mrq;
- `mbq`: minimum base quality. ASEQ does not consider as part of the pileup bases with quality < mbq;
- `mdc`: minimum depth of coverage. ASEQ output only reports positions with coverage ≥ mdc;
- `threads`: number of threads available for ASEQ computation.

Step annotations

1. Download and uncompress the last version of ASEQ

```
$ curl http://demichelislab.unitn.it/lib/exe/fetch.php?media=aseq-v1.1.11-linux64.tar.gz > aseq-v1.1.11-linux64.tar.gz
$ tar xvf aseq-v1.1.11-linux64.tar.gz
```

ASEQ code will be available in subfolder `binaries/linux64/`.

2. Download and uncompress ASEQ examples

```
$ curl http://demichelislab.unitn.it/lib/exe/fetch.php?media=aseq-
examples.tar.gz > aseq-examples.tar.gz
$ tar xvf aseq-examples.tar.gz
```

ASEQ examples will available in subfolder `examples/VCF_samples/`.

3. Run ASEQ on example data 1.

```
$ ./binaries/linux64/ASEQ mode=PILEUP
vcf=examples/VCF_samples/sample1.vcf
bam=examples/BAM_samples/sample1.bam mbq=20 mrq=20 mdc=1
threads=1 out=.
```

ASEQ produces file `sample1.PILEUP.ASEQ` reporting allelic fraction and read coverage in bam file `sample1.bam` for each position in vcf file `sample1.vcf`. Parameters `mbq=20` and `mrq=20` tell ASEQ to ignore, respectively, bases and reads with quality less than 20. Parameter `mdc=1` instructs ASEQ to ignore positions in bam file with no reads. The parameters and the format of the output file `.PILEUP.ASEQ` are compatible with pileup data required in Basic Protocol 1.

BASIC PROTOCOL 2 (optional)

Compute ploidy—Segmentation algorithms partition input genomic space into segments with homogenous coverage. Given a pair of matched tumor and normal samples, the LogR value of a genomic segment is the log2 of the ratio between the tumor and the normal sample coverage within the segment. To account for different mean coverage in different sequencing experiments, LogR is normalized over the ratio between the mean tumor and the mean normal coverage; this applies both to whole genome and whole exome data. In the case of higher coverage in the tumor sample, without normalization the ratio between the mean tumor and the mean normal coverage is X, a wild type segment would have LogR=log2(X), while the expected value is 0 (i.e. same number of alleles between tumor and normal). The normalization would however introduce a bias whenever the difference in the mean coverage between the tumor and the normal sample is due to an abnormal number of alleles in the tumor - aneuploidy genome. In this case, the normalization leads to a shift in the LogR signal. Figure 3A shows an example of diploid genome sample with 127x and 69x mean tumor and mean normal coverage, respectively. The LogR signal is centered in 0 as expected (green line). Figure 3B highlights a more complex case: tumor and normal mean coverage are comparable (125x and 117x, respectively), but the position of the wild type segments (orange line) is shifted with respect to the expected value (green line). The shift is representative of the total number of alleles in the genome and ploidy can be estimated as

$$Ploidy = 2 \times 2^{-log_2(LogR\ shift)} \quad 2$$

Proof (Equation 2) is reported in CLONET paper (Prandi et al., 2014). Sample in Figure 3A has LogR shift 0 and Ploidy 2, while sample in Figure 3B has LogR shift of −0.34 and Ploidy 2.53.

The function `compute_ploidy` builds on this definition and is implemented to identify wild type genomic segments and to estimate how far the LogRs deviate from 0. The key step in the search is to restrict the genomic segments space to those with beta equal to 1, i.e., have an equal number of maternal and paternal copies. In Figure 3B, this step excludes segments with LogR around 0, as their beta is significantly lower than 1 and represent segments with copy number 3 (see BASIC PROTOCOL 4). In this context, green line in Figure 3B is centered on wild type segments while turquoise vertical line identifies segments copy number four. The function `compute_ploidy` includes the following input parameters:

- `beta_table`: a table created using function described in BASIC PROTOCOL 1;
- `max_homo_dels_fraction` (default 0.05): homozygous deletions can provide a confounding factor in the determination of sample ploidy. The parameter sets a percentage of genomic segments that will not be used for ploidy computation as putative homozygous deletion. Overestimating this value does not affect ploidy computation;
- `beta_limit_for_neutral_reads` (default 0.90): in theory, neutral reads correspond to beta equal to 1, but experimental noise lowers this value. Only segments with beta above the parameter are used to compute ploidy;
- `min_coverage` (default 20): only genomic segments with average coverage at least min_coverage are used to compute DNA admixture;
- `min_required_snps` (default 10): only genomic segments covering at least `min_required_snps` informative SNPs are considered for DNA admixture computation.

The function returns the ploidy for the input sample.

Necessary Resources

Hardware—A 64-bit computer running Linux with at least 4 GB of RAM

Software—The library has been tested with R version 3.5.2 and R libraries parallel 3.5.2, ggplot2 3.1.0, sets 1.0–18, arules 1.6–3, ggrepel 0.8.0.

Step annotations

1. Run R from command line

```
$ R
```

2. Compute beta table as described in BASIC PROTOCOL 1
3. Compute ploidy from beta table `bt`

```
> pl <- compute_ploidy(bt)
```

BASIC PROTOCOL 3

Compute DNA admixture—DNA admixture is defined as the percentage of non-tumor cells in a tumor sample. DNA admixture is an important confounding factor in genomic analysis, as it dilutes somatic aberration signal across all genomic and molecular alterations. Relevant to genomic analyses, it dilutes somatic copy number aberration (SCNA) and single nucleotide variant (SNV) signal. In a 100% pure tumor sample, the expected coverage across a mono-allelic (i.e. hemizygous) deletion should be about half of coverage of wild type segments and therefore the LogR should be equal to −1 (as log2(½)). However, if the purity is 50%, then only half of the total number of cells harbor the hemizygous deletion and the expected LogR is around −0.415 (log2(¾)). Similarly, the value of beta of a genomic segment varies depending on the level of DNA admixture. In BASIC PROTOCOL 1, we saw that the beta of a hemizygous deletion in a 100% pure sample is 0, as no neutral reads are present. However, 50% of admixture would increase beta to $\frac{2}{3}$, as for each tumor active read there are two neutral read from the admixed cells. The original CLONET manuscript (Prandi et al., 2014) describes the equations that define the expected LogR and beta corresponding to the spectrum of tumor admixture. The function `compute_dna_admixture` searches the (LogR, beta) space defined by the function `compute_beta_table` (BASIC PROTOCOL 1) for a value of DNA admixture that better explains the observed value in the `beta_table`. The function `compute_dna_admixture` also requires ploidy value as computed by the function `compute_ploidy` (BASIC PROTOCOL 2) to account for the shift in LogR values due to possible aneuploidy tumor genomes. The function `compute_dna_admixture` has the following input parameters:

- `beta_table`: a table created using function described in BASIC PROTOCOL 1;
- `ploidy_table`: a table created using function described in BASIC PROTOCOL 2;
- `min_coverage` (default 20): only genomic segments with average coverage at least `min_coverage` are used to compute DNA admixture;
- `min_required_snps` (default 10): only genomic segments covering at least `min_required_snps` informative SNPs are considered for DNA admixture computation;
- `error_tb`: the number of informative SNPs and the coverage of the considered segment affect the accuracy of the estimation of beta of a genomic. Table error_tb reports for each combination of number of informative SNPs and coverage the expected error around beta estimate. CLONETv2 embeds a pre-computed error_tb (details in (Prandi et al., 2014)) previously tested in several studies (Beltran et al., 2015; Beltran et al., 2016; Faltas et al., 2016). However, specific experimental settings, as ultra-deep targeted sequencing or low-pass whole genome sequencing, may require an ad-hoc table.

The function returns the estimated DNA admixture for the input sample as well as minimum and maximum DNA admixture accounting for errors around beta estimates.

Necessary Resources

Hardware—A 64-bit computer running Linux with at least 4 GB of RAM

Software—The library has been tested with R version 3.5.2 and R libraries parallel 3.5.2, ggplot2 3.1.0, sets 1.0–18, arules 1.6–3, ggrepel 0.8.0.

Step annotations

1. Run R from command line

```
$ R
```

2. Compute beta table as described in BASIC PROTOCOL 1
3. Compute ploidy table as described in BASIC PROTOCOL 2
4. Given beta table bt and ploidy pl

```
> adm <- compute_dna_admixture(bt, pl)
```

SUPPORT PROTOCOL 2

Visualize and interpret beta table, ploidy, and DNA admixture—BASIC PROTOCOL 1 describes how to derive the value of beta for a genomic segment. A tumor sample is then described as a set of (beta, LogR) values extending the usual LogR space and enabling the computation of ploidy and DNA admixture in BASIC PROTOCOLS 2 and 3, respectively. To help interpreting the results of the first three BASIC PROTOCOLS, CLONETv2 provides the function `check_ploidy_and_admixture` that plots beta vs LogR space for a given samples. Figures 4A and 4B shows the values of beta against the LogR of the same samples presented in Figures 3A and 3B, respectively. For each genomic segment, the plot reports the LogR as well as the beta computed by function `compute_beta_table`. To help the user, the function predicts expected (beta, LogR) given the input ploidy and DNA admixture level following the equations described in CLONET paper (Prandi et al., 2014). Predicted values are computed for different combination of allele specific copy number (see BASIC PROTOCOL 4) and represented as red circles. Comparing the expected (red circles) and the observed (gray dots) values helps the interpretation of the estimates. For instance, segments with LogR near 0 in Figure 3B cannot be wild type, as their betas are around 0.8, a value compatible with 3 DNA copies.

Necessary Resources

Hardware—A 64-bit computer running Linux with at least 4 GB of RAM

Software—The library has been tested with R version 3.5.2 and R libraries parallel 3.5.2, ggplot2 3.1.0, sets 1.0–18, arules 1.6–3, ggrepel 0.8.0.

Step annotations

1. Run R from command line

```
$ R
```

2. Follow BASIC PROTOCOLS 1 to 3 to compute beta table bt, ploidy table pl, and DNA admixture table adm, respectively.

3. Compute basic beta vs LogR plot

```
> check_plot <- check_ploidy_and_admixture(bt,pl,adm)
```

4. Check plot is a ggplot object (H., 2009) that can be customized by the user (e.g. font size, color, line width). Final plot is printed with command

```
> print(check_plot)
```

BASIC PROTOCOL 4

Compute allele specific copy number—Figure 3 suggests a relation between the values (beta, LogR) of a genomic segment and its allele specific copy number. Consider a 100% pure tumor sample and a genomic segment with wild type LogR, i.e. the log2 ratio is equal to 0; then beta could be either equal to 1 (if one copy of both the maternal and the paternal allele are present) or equal to 0 (if the two alleles come from the same parent, case of copy neutral loss of heterozygosity (CN-LOH)). The approach is generalized in Beltran et al. (Beltran et al., 2016) by defining the exact equations that relates (LogR, beta) to allele specific copy number, given ploidy and DNA admixture. Figure 5A shows an example in which CLONETv2 identifies three classes of loss of heterozygosity: the well-characterized hemizygous deletions and the CN-LOH, and the less common Gain-LOH, where one allele is lost but the total copy number (LogR value) is compatible with a DNA gain. Mapping (LogR, beta) space to allele specific copy number space (Figure 5B) simplifies interpretation the genomic landscape of a sample. Of note, allele specific copy number signal in Figure 5B does not contain information about the ploidy and purity of the original sample, making it easy to compare samples with different ploidy and purity values. The example highlights the novelty and the power of allele specific copy number analysis. Function `compute_allele_specific_scna_table` transforms (LogR, beta) pairs into allele specific copy number pairs (cnA, cnB). The function requires estimates of purity and ploidy and has the following parameters:

- `beta_table`: a table created using function described in BASIC PROTOCOL 1;

- `ploidy_table`: a table created using function described in BASIC PROTOCOL 2;
- `admixture_table`: a table created using function described in BASIC PROTOCOL 3;
- `error_tb`: same error_tb used in function `compute_dna_admixture` of BASIC PROTOCOL 3;
- `allelic_imbalance_th` (default 0.5): function `compute_allele_specific_scna_table` also returns integer values cnA.int and cnB.int for cnA and cnB, respectively. Value cnA.int is the round of cnA if | cnA.int - cnA| < allelic_imbalance_th, otherwise cnA.int is not defined. The same for cnB.

The function `compute_allele_specific_scna_table` extends input `beta_table` with allele specific copy number related columns:

- `log2.corr`: LogR value adjusted by ploidy and purity, i.e., the LogR value the segment would have in a diploid 100% pure tumor sample;
- `cnA`, `cnB`: number of copies of major (cnA) and minor (cnB) allele. The values do not contain information about ploidy and purity. Indeed, cnA + cnB equals 2*2^log2.corr;
- `cnA.int`, `cnB.int`: integer number of copies of major and minor alleles, respectively.

Necessary Resources

Hardware—A 64-bit computer running Linux with at least 4 GB of RAM

Software—The library has been tested with R version 3.5.2 and R libraries parallel 3.5.2, ggplot2 3.1.0, sets 1.0–18, arules 1.6–3, ggrepel 0.8.0.

Step annotations

1. Run R from command line

```
$ R
```

2. Follow BASIC PROTOCOLS 1 to 3
3. Given beta table bt, ploidy table pl, and DNA admixture table adm

```
> as_tb <- compute_allele_specific_scna_table(bt, pl, adm)
```

BASIC PROTOCOL 5

Compute somatic copy numbers clonality—A somatic aberration is clonal if all the tumor cells present the aberration. Suppose a 100% pure tumor sample with mono-allelic deletions of genomic segments D_1 and D_2, with 100% and 50% clonality, respectively, i.e. all tumor cells harbor D_1 deletion, but only 50% harbor D_2 deletion. Expected LogR is log2(½)=−1 for D_1 and log2(¾) for D_2. Note that expected LogR for D_2 is the same that would result considering a clonal deletion in a 50% pure sample (see BASIC PROTOCOL 3). This is because, in genomic region D_2, the reads sequenced from cells not harboring the deletion cannot be distinguished from those of non-tumor cells DNA admixture. The same consideration holds for the expected proportion of neutral reads beta. CLONET equations (Carreira et al., 2014) build on this intuition and define a map from (LogR, beta) pairs to the clonality of somatic copy number aberrations. However, fluctuations in the level of coverage that introduce noise in the LogR signal, as well as limitations in the sensitivity of the inference of beta due to the number of available informative SNPs make it difficult to compare the clonality levels of aberrations across different tumor samples. To facilitate the clonality comparisons, the function `compute_scna_clonality_table` returns a minimum and maximum estimated clonality value and a discretized clonality status. The function considers DNA admixture level, distribution of LogR values, and errors around beta estimates and assigns to each genomic segment a minimum and a maximum observed clonality. Lower and upper bound for clonality are used to assign to define the segment clonality status, among *clonal*, *uncertain.clonal*, *uncertain.subclonal*, *subclonal*, and *not.analysed*. Clonal and subclonal statuses correspond to more reliable clonality calls, while uncertain prefix is used when clonality estimate can be affected by the noise of the input data. For instance, Figure 6 reports the example of a tumor sample with two clusters of hemizygous deletions: clonal in (−0.6, 0.45) and subclonal in (−0.25, 0.8). Segments in (−0.9, 0.53) correspond to a region with subclonal homozygous deletion, 20% of the tumor cells lack both alleles while the other 80% retain one allele. Uncertain clonality status calls refer to segments at (−0.45, 0.58) and at (−0.63, 0.51); compared to clonal segments, the former shows markedly different beta but borderline LogR (uncertain.subclonal), the latter only shows small deviation in beta (uncertain.clonal segment). Not.analysed segments include wild type segments and aberrant segments with (LogR, beta) values that do not fit CLONETv2 model. The function `compute_scna_clonality_table` takes a beta table and the associated estimates of purity and ploidy together with the following parameters:

- `beta_table`: a table created using function described in BASIC PROTOCOL 1;
- `ploidy_table`: a table created using function described in BASIC PROTOCOL 2;
- `admixture_table`: a table created using function described in BASIC PROTOCOL 3;
- `error_tb`: same `error_tb` used in function compute_dna_admixture of BASIC PROTOCOL 3. Error around beta is propagated to clonality estimate and used in its discretization;

- `clonality_threshold` (default=0.85): function `compute_scna_clonality_table` returns minimum and maximum clonality for input genomic segments. Clonality_threshold is used to discretize clonality as described in (Prandi et al., 2014);
- `beta_threshold` (default=0.9): input beta values below `beta_theshold` are marked as potentially aberrant and used for clonality estimates.

The function `compute_scna_clonality_table` extends input `beta_table` with clonality related columns:

- `clonality`: real value representing the estimated percentage of tumor cells with uniform copy number for a given genomic segment;
- `clonality.min`, `clonality.max`: real values representing minimum and maximum estimated clonality given the distribution of beta and LogR values;
- `clonality.status`: discretized clonality.

Necessary Resources

Hardware—A 64-bit computer running Linux with at least 4 GB of RAM

Software—The library has been tested with R version 3.5.2 and R libraries parallel 3.5.2, ggplot2 3.1.0, sets 1.0–18, arules 1.6–3, ggrepel 0.8.0.

Step annotations

1. Run R from command line

```
$ R
```

2. Follow BASIC PROTOCOLS 1 to 3
3. Given beta table `bt`, ploidy table `pl`, and DNA admixture table `adm`

```
> clonality_tb <- compute_scna_clonality_table(bt, pl, adm)
```

BASIC PROTOCOL 6

Compute single nucleotide variants clonality—Each single nucleotide variant (SNV) is characterized by the variant allele fraction (VAF), i.e., the proportion of reads supporting the alternative allele; the VAF is intuitively representative of the amount of tumor DNA harboring the mutation (no alternative read is expected from the admixed normal cells). Therefore, low VAF values correspond to low clonality. In a 100% pure diploid sample, a clonal mono-allelic SNV within a wild type genomic segment is expected to show a VAF of 0.5 (for simplicity we here ignore the reference mapping bias (Degner et al., 2009)) where, in the same setting, an SNV that is present in the 60% of the tumor cells is expected to show a VAF of 0.3. However, several technical and biological factors influence VAF value,

including DNA admixture, ploidy, and somatic copy number status. In (Faltas et al., 2016), we extended the original implementation to deal with SNVs in the context of allele specific copy number. SNV VAF ranges over a finite set of values dictated by the DNA copy number state; for instance, a clonal SNV in a copy number aberrant segment (CN = 3) in a 100% pure diploid sample may have VAF equal to $\frac{1}{3}$, $\frac{2}{3}$, or 1, depending on the number of alleles harboring the mutation. By utilizing the sample admixture estimate and the its lower and upper bound (function `compute_dna_admixture`), we first estimate the minimum and maximum clonality and next, as for SCNA, a discretize clonality value (*clonal*, *uncertain.clonal*, *uncertain.subclonal*, and *subclonal)* is assigned. Figure 7A shows an example of SNVs clonality (y axis) distributions per discretized class (x axis) regardless of the copy number of the genomic segments harboring the SNVs (Figure 7B). Given a tumor sample, the function `compute_snv_clonality` takes as input the following parameters:

- `snv_read_count`: a table reporting in each row the genomic coordinates of an SNV together with number of reference and alternative reads covering the mutated position;
- `beta_table`: a table created using function described in BASIC PROTOCOL 1;
- `ploidy_table`: a table created using function described in BASIC PROTOCOL 2;
- `admixture_table`: a table created using function described in BASIC PROTOCOL 3;
- `error_tb`: same `error_tb` used in function compute_dna_admixture of BASIC PROTOCOL 3. Error around beta is propagated to assess clonality estimate boundary and in turn used for its discretization;
- `error_rate` (default=0.05): fraction of SNVs to exclude based on adjusted VAF distribution.

The function `compute_snv_clonality` extends input table `snv_read_count` with clonality related columns:

- `cnA`, `cnB`: allele specific copy number of the genomic segment containing the SNV;
- `t_af_corr`: tumor VAF adjusted for ploidy, admixture, and allele specific copy number;
- `SNV.clonality`: percentage of tumor cells harboring the SNV;
- `SNV.clonality.status`: discretized `SNV.clonality`.

Necessary Resources

Hardware—A 64-bit computer running Linux with at least 4 GB of RAM

Software—The library has been tested with R version 3.5.2 and R libraries parallel 3.5.2, ggplot2 3.1.0, sets 1.0–18, arules 1.6–3, ggrepel 0.8.0.

Step annotations

1. Run R from command line

```
$ R
```

2. Follow BASIC PROTOCOLS 1 to 3

3. Read a SNV table `snv_reads` with columns `rc_ref_tumor` and `rc_alt_tumor` for reference and alternative read counts, respectively;

```
> read.table(system.file("sample_snv_read_count.tsv", package =
"CLONETv2"),header = T, as.is=T, comment.char = "", check.names = F,
na.strings = "-")
```

4. Given beta table `bt`, ploidy table `pl`, and DNA admixture table `adm`

```
> snv_clonality_tb <- compute_snv_clonality("sample1", snv_reads, bt, pl,
adm)
```

GUIDELINES FOR UNDERSTANDING RESULTS

We present a complete R package to compute allele specific data from next generation sequencing experiments of paired tumor and matched normal DNA samples. CLONETv2 works on preprocessed data (does not work on bam or fastq files), including segmented genomic profiles and pileups of relevant genomic positions. This makes CLONETv2 more flexible with respect to other tools as ABSOLUTE (Carter et al., 2012), that requires segmented data from HAPSEG (bundled with ABSOLUTE) or FACETS (Shen and Seshan, 2016) that integrates LogR segmentation with allele specific analysis. The advantage is that CLONETv2 allows the user to choose the segmentation solution that best fits the study needs. As a didactic example, we ran CLONETv2 BASIC PROTOCOLS 1 to 3 on the sample from Figure 4A (that shows segments from CNVkit (Talevich et al., 2016)), on segmented data computed with EXCAVATOR2 (D'Aurizio et al., 2016) (Figure 8A) or with FACETS (Figure 8B). EXCAVATOR2 and CNVkit data in this space distribute similarly, although the former shows noisier signal, and ploidy and DNA admixture estimates perfectly match. On the contrary, on this specific example, FACETS estimates are different as expected given for instance a set of segments with LogR around −0.75 and beta equal to 1.

The central notion introduced with CLONETv2 is the proportion of neutral reads beta calculated in BASIC PROTOCOL 1. This value expands the one dimension LogR space returned by the segmentation algorithms to the two dimensions beta vs LogR space; an example of its utility is offered in Figure 3B where CLONETv2 resolves an ambiguous LogR profile from Figure 3B by utilizing beta values (see SUPPORT PROTOCOL 2). However, as more complex genomic profiles may require inspection of output estimates, we designed a function `check_ploidy_and_admixture` to help the user in the interpretation

of complex copy number data. Figure 9A shows beta vs LogR plot of a sample *S* that CLONETv2 defines as ploidy equals to 2.01 (diploid) and low DNA admixture. The unique feature of function `check_ploidy_and_admixture` is the ability to plot the expected position of a genomic segment in the beta vs LogR space, given ploidy and DNA admixture (red circles). In Figure 9A, green circles highlight the genomic segments that are not explained by estimated ploidy and DNA admixture and compatible with subclonality, as in Figure 6. However, an alternative interpretation is possible, where sample *S* is aneuploidy, and no wild type segments are present throughout the tumor genome; segments in (1,0) (Figure 9A) are rather CN-LOH (as depicted in Figure 9B, due to a shift in the LogR signal (see BASIC PROTOCOL 2)) and, therefore, wild type segments are expected at coordinates (−0.67, 1). Applying LogR shift equation (see BASIC PROTOCOL 2), ploidy results 3.14 and in turn function `compute_dna_admixture` estimates a DNA admixture value of 0.42. Subclonal copy number segments (green circles, Figure 9A) are then classified as clonal (red circles with green border, Figure 9B). Given the observed data, both interpretations are plausible. The allele specific plots (Figures 9C–D for Figures 9A–B, respectively), transparent to ploidy and DNA admixture values, may provide additional information to contextualize the two scenarios. The first one (Figure 9C) represents a tumor where exactly half of the cells harbor exactly the same set of subclonal hemizygous deletions, subclonal CN-LOH, and subclonal gain (green circles). The second one (Figure 9D) suggests genomic events that included whole genome duplication (or duplication of several chromosomal arms) exemplified by numerous allele specific copy number (2, 2) and CN-LOH (2,0).

Importantly, CLONETv2 computations are agnostic to gene models to avoid across studies constrains. To facilitate gene level focuses analysis outputs of functions `compute_allele_specific_scna_table` and `compute_scna_clonality_table` can be lifted using any gene model that includes chromosome, start and end position information; tables reporting allele specific copy number and clonality values are compatible with BED format (Quinlan and Hall, 2010) and can be intersected with common gene models from, e.g., Ensemble (Zerbino et al., 2018).

COMMENTARY

Background Information

Tumor ploidy and normal DNA admixture fraction are critical parameters in cancer genomic analysis, as incorrect estimations may compromise any downstream analysis (see example in Figure 9). CLONETv2 provides a reliable and flexible environment to process matched tumor and normal samples together with function `check_ploidy_and_admixture` that helps user evaluating reliability of estimates. Of note, CLONETv2 is neither bound to a specific copy number caller nor specific gene models. Finally, CLONETv2 is distributed as an R package and downstream processing, including allele specific copy number and subclonality estimation, can be easily integrated into broader analysis pipelines.

Critical Parameters

CLONETv2 default parameters have been tested in a variety of studies spanning tissue and plasma samples in different tumor settings. However, data analysis from specific

experimental conditions or analysis prerequisites would benefit from tweaking CLONETv2 parameters. Parameter `min_coverage` is common to many CLONETv2 functions; it is used to filter out the genomic segments with low mean coverage at informative SNPs; `min_required_snps` filters out segments with too few informative SNPs. Higher values of `min_coverage` and `min_required_snps` correspond to more reliable results but at the same time to fewer segments to be used to compute allele specific copy number and clonality. The optimal trade-off between reliability and extensiveness of the analysis is study dependent. For instance, an ultra-deep sequencing experiment (e.g., mean coverage > 5000x) would benefit from `min_coverage` higher than 20 (the default value), in fact it corresponds to the 0.4% of the expected coverage and is can hardly be distinguished from the background experimental noise. On the contrary, low pass whole genome sequencing experiments (coverage >4x) require a lower `min_coverage` by design.

A second critical parameter is `error_table`, a table reporting error around beta estimate for different combinations of coverage and number of informative SNPs. CLONETv2 has an error table in bundle, obtained simulating different inputs to function `compute_beta_table` with combinations of values for the coverage and the number of informative SNPs. If for a genomic segment, the number of informative SNPs and the mean coverage are not reported in the `error_table`, CLONETv2 uses the nearest available pair of values as previously described (Prandi et al., 2014).

Troubleshooting

CLONETv2 offers a robust framework for the genomic analysis of somatic copy number data together with the possibility to manually curate estimates (see SUPPORT PROTOCOL 2). However, some specific cases could prevent CLONETv2 from completing the analysis.

Figure 10A shows the beta vs LogR plot of a tumor sample with an uncommon profile. Profile presents genomic segments with all betas close 1 (alleles equally represent parental chromosome of origin) and LogR range in the interval (−0.5, 0.5), corresponding to approximately a loss of half copy and the gain of one copy. Moreover, the cloud of beta values around 0.75 within the same LogR range does not fit any CLONETv2 model. This data is either the result of uneven sequence read coverage (Wang et al., 2017)) that affects both LogR signal and AF of informative SNPs, or the representation of a large number of subclonal populations with diverse ploidy and somatic copy number profiles. Altogether, the information from the segmented data and pileup of informative SNPs is not sufficient to disentangle such case, and such data should be not included in any downstream analysis.

A second problematic case is presented in Figure 10B. All segments show LogR around 0 and beta close to 1, i.e., all genomic segments have wild type copy number. This beta vs LogR profile data is compatible with two very different situations: (i) a copy number quite tumor sample, i.e., no deletions or amplifications are detected; (ii) a near 100% DNA admixed tumor sample, i.e., almost all the cells in the sample are non-tumor. The first interpretation points to a potentially interesting case while the second highlights limitations either in the sample of origin or in the preparation. As for the case in Figure 10A,

CLONETv2 cannot distinguish between the two interpretations and the sample should not be considered.

ACKNOWLEDGEMENT

The authors like to thank the European Research Council for Consolidator grant ID 648670 (F.D.), and the National Cancer Institute (NIH) for R01 CA125612–05A1 and for the SPORE in Prostate Cancer P50-CA211024 (F.D.).

LITERATURE CITED

Aran D, Sirota M, and Butte AJ (2015). Systematic pan-cancer analysis of tumour purity. Nature communications 6, 8971.

Baca SC, Prandi D, Lawrence MS, Mosquera JM, Romanel A, Drier Y, Park K, Kitabayashi N, MacDonald TY, Ghandi M, et al. (2013). Punctuated evolution of prostate cancer genomes. Cell 153, 666–677. [PubMed: 23622249]

Bao L, Pu M, and Messer K (2014). AbsCN-seq: a statistical method to estimate tumor purity, ploidy and absolute copy numbers from next-generation sequencing data. Bioinformatics 30, 1056–1063. [PubMed: 24389661]

Beltran H, Eng K, Mosquera JM, Sigaras A, Romanel A, Rennert H, Kossai M, Pauli C, Faltas B, Fontugne J, et al. (2015). Whole-Exome Sequencing of Metastatic Cancer and Biomarkers of Treatment Response. JAMA Oncol 1, 466–474. [PubMed: 26181256]

Beltran H, Prandi D, Mosquera JM, Benelli M, Puca L, Cyrta J, Marotz C, Giannopoulou E, Chakravarthi BV, Varambally S, et al. (2016). Divergent clonal evolution of castration-resistant neuroendocrine prostate cancer. Nat Med 22, 298–305. [PubMed: 26855148]

Bielski CM, Zehir A, Penson AV, Donoghue MTA, Chatila W, Armenia J, Chang MT, Schram AM, Jonsson P, Bandlamudi C, et al. (2018). Genome doubling shapes the evolution and prognosis of advanced cancers. Nature genetics

Boysen G, Barbieri CE, Prandi D, Blattner M, Chae SS, Dahija A, Nataraj S, Huang D, Marotz C, Xu L, et al. (2015). SPOP mutation leads to genomic instability in prostate cancer. eLife 4.

Cancer Genome Atlas Research, N. (2015). The Molecular Taxonomy of Primary Prostate Cancer. Cell 163, 1011–1025. [PubMed: 26544944]

Carreira S, Romanel A, Goodall J, Grist E, Ferraldeschi R, Miranda S, Prandi D, Lorente D, Frenel JS, Pezaro C, et al. (2014). Tumor clone dynamics in lethal prostate cancer. Science translational medicine 6, 254ra125.

Carter SL, Cibulskis K, Helman E, McKenna A, Shen H, Zack T, Laird PW, Onofrio RC, Winckler W, Weir BA, et al. (2012). Absolute quantification of somatic DNA alterations in human cancer. Nature biotechnology 30, 413–421.

Chunduri NK, and Storchova Z (2019). The diverse consequences of aneuploidy. Nat Cell Biol 21, 54–62. [PubMed: 30602769]

D'Aurizio R, Pippucci T, Tattini L, Giusti B, Pellegrini M, and Magi A (2016). Enhanced copy number variants detection from whole-exome sequencing data using EXCAVATOR2. Nucleic acids research 44, e154. [PubMed: 27507884]

Danielsen HE, Pradhan M, and Novelli M (2016). Revisiting tumour aneuploidy - the place of ploidy assessment in the molecular era. Nat Rev Clin Oncol 13, 291–304. [PubMed: 26598944]

Davoli T, Uno H, Wooten EC, and Elledge SJ (2017). Tumor aneuploidy correlates with markers of immune evasion and with reduced response to immunotherapy. Science 355.

Degner JF, Marioni JC, Pai AA, Pickrell JK, Nkadori E, Gilad Y, and Pritchard JK (2009). Effect of read-mapping biases on detecting allele-specific expression from RNA-sequencing data. Bioinformatics 25, 3207–3212. [PubMed: 19808877]

Faltas BM, Prandi D, Tagawa ST, Molina AM, Nanus DM, Sternberg C, Rosenberg J, Mosquera JM, Robinson B, Elemento O, et al. (2016). Clonal evolution of chemotherapy-resistant urothelial carcinoma. Nature genetics 48, 1490–1499. [PubMed: 27749842]

Gundem G, Van Loo P, Kremeyer B, Alexandrov LB, Tubio JM, Papaemmanuil E, Brewer DS, Kallio HM, Hognas G, Annala M, et al. (2015). The evolutionary history of lethal metastatic prostate cancer. Nature 520, 353–357. [PubMed: 25830880]

H., W (2009). ggplot2: Elegant Graphics for Data Analysis In Berlin, (Springer).

Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R, and Genome Project Data Processing, S. (2009). The Sequence Alignment/Map format and SAMtools. Bioinformatics 25, 2078–2079. [PubMed: 19505943]

Li Y, and Xie X (2014). Deconvolving tumor purity and ploidy by integrating copy number alterations and loss of heterozygosity. Bioinformatics 30, 2121–2129. [PubMed: 24695406]

Mu P, Zhang Z, Benelli M, Karthaus WR, Hoover E, Chen CC, Wongvipat J, Ku SY, Gao D, Cao Z, et al. (2017). SOX2 promotes lineage plasticity and antiandrogen resistance in TP53- and RB1-deficient prostate cancer. Science 355, 84–88. [PubMed: 28059768]

Olshen AB, Venkatraman ES, Lucito R, and Wigler M (2004). Circular binary segmentation for the analysis of array-based DNA copy number data. Biostatistics 5, 557–572. [PubMed: 15475419]

Prandi D, Baca SC, Romanel A, Barbieri CE, Mosquera JM, Fontugne J, Beltran H, Sboner A, Garraway LA, Rubin MA, and Demichelis F (2014). Unraveling the clonal hierarchy of somatic genomic aberrations. Genome biology 15, 439. [PubMed: 25160065]

Quinlan AR, and Hall IM (2010). BEDTools: a flexible suite of utilities for comparing genomic features. Bioinformatics 26, 841–842. [PubMed: 20110278]

Romanel A, Lago S, Prandi D, Sboner A, and Demichelis F (2015). ASEQ: fast allele-specific studies from next-generation sequencing data. BMC medical genomics 8, 9. [PubMed: 25889339]

Shen R, and Seshan VE (2016). FACETS: allele-specific copy number and clonal heterogeneity analysis tool for high-throughput DNA sequencing. Nucleic acids research 44, e131. [PubMed: 27270079]

Stephens PJ, Greenman CD, Fu B, Yang F, Bignell GR, Mudie LJ, Pleasance ED, Lau KW, Beare D, Stebbings LA, et al. (2011). Massive genomic rearrangement acquired in a single catastrophic event during cancer development. Cell 144, 27–40. [PubMed: 21215367]

Talevich E, Shain AH, Botton T, and Bastian BC (2016). CNVkit: Genome-Wide Copy Number Detection and Visualization from Targeted DNA Sequencing. PLoS Comput Biol 12, e1004873. [PubMed: 27100738]

Taylor AM, Shih J, Ha G, Gao GF, Zhang X, Berger AC, Schumacher SE, Wang C, Hu H, Liu J, et al. (2018). Genomic and Functional Approaches to Understanding Cancer Aneuploidy. Cancer Cell 33, 676–689 e673. [PubMed: 29622463]

Team, R. C. (2017). R: A language and environment for statistical computing R Foundation for Statistical Computing, Vienna, Austria URL https://wwwR-project.org.

Van Loo P, Nordgard SH, Lingjaerde OC, Russnes HG, Rye IH, Sun W, Weigman VJ, Marynen P, Zetterberg A, Naume B, et al. (2010). Allele-specific copy number analysis of tumors. Proceedings of the National Academy of Sciences of the United States of America 107, 16910–16915. [PubMed: 20837533]

Wang Q, Shashikant CS, Jensen M, Altman NS, and Girirajan S (2017). Novel metrics to measure coverage in whole exome sequencing datasets reveal local and global non-uniformity. Scientific reports 7, 885. [PubMed: 28408746]

Zerbino DR, Achuthan P, Akanni W, Amode MR, Barrell D, Bhai J, Billis K, Cummins C, Gall A, Giron CG, et al. (2018). Ensembl 2018. Nucleic acids research 46, D754–D761. [PubMed: 29155950]

Significance Statement

CLONETv2 is an R package that allows for the estimation of DNA purity and ploidy of any tumor sample through the analysis of nucleotide level sequencing data. It includes allele specific copy number characterization as well as clonality estimation of both somatic copy number and single nucleotide variants. CLONETv2 is suited for both tissue and liquid biopsy sequencing data.

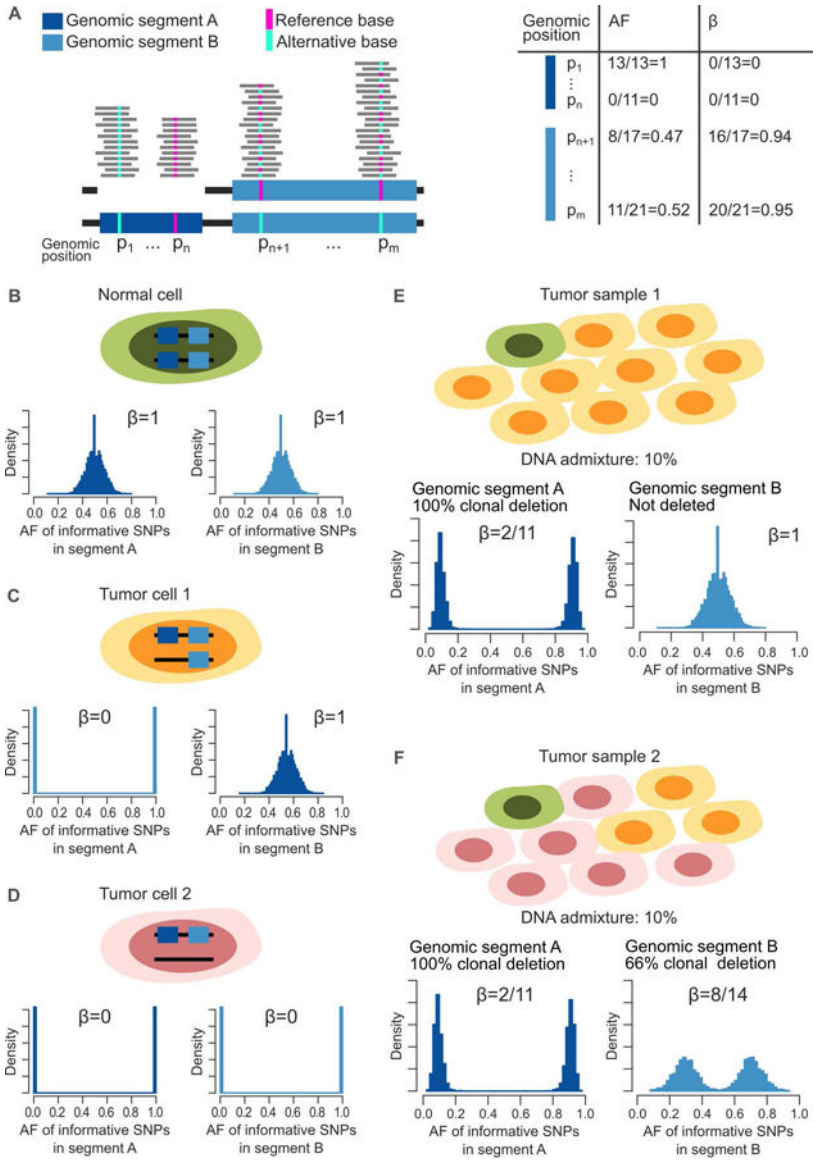

Figure 1. Cartoon of the computation of beta and allelic fraction of informative SNPs.

(**A**) Example of the allelic fraction (AF) and beta (β) values as computed in five genomic positions (p_1 to p_m) corresponding to five informative SNPs. Positions p_1 to p_n are within a hemizygous deleted genomic segment A, while genomic positions p_{n+1} to p_m lie within a wild type genomic segment B. (**B**-**D**) Examples of a normal cell and two different tumor cells. Tumor cells 1 and 2 differ in the status of genomic segment B. Histograms below the cell cartoons report the expected distribution of the AF of SNPs in genomic segments A and B together with the associated beta values. (**E**-**F**) Examples of two different tumor samples. Tumor sample 1 includes one normal cell and nine tumor cells with deleted genomic segment A and wild type genomic segment B. Tumor sample 2 differs from tumor sample 1 in the presence of six tumor cells with a hemizygous deletion of genomic segment B. Expected distribution of the AF of informative SNPs together with estimated beta are depicted below each tumor sample cartoon.

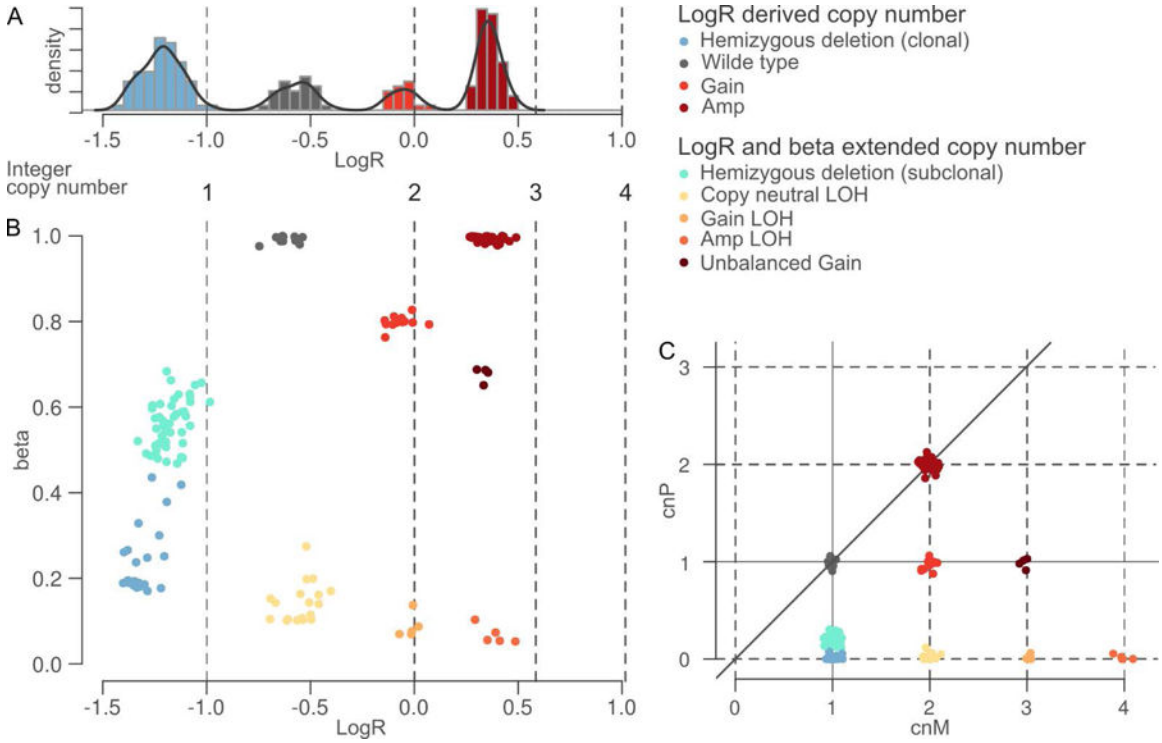

Figure 2. Sketch of CLONETv2 copy number space transformations.

(**A**) Example of histogram and density plots of the distribution of LogR signal in a tumor sample. Expected positions of integer copy number in a diploid 100% pure tumor sample are listed below. (**B**) Expansion of the mono-dimensional LogR signal of panel (**A**) in the 2-dimensional beta vs LogR space. Each dot represents a genomic segment and vertical dashed lines correspond to integer copy number as in panel (**A**). Color code clusters genomic segments with homogenous copy number. (**C**) Allele-specific copy number projection of the beta vs LogR data of panel (**B**). Each dot represents a genomic segment with maternal copy number allele cnM and paternal copy number allele cnP. Maternal and paternal alleles are assigned arbitrarily. The color code is consistent with panel (**B**).

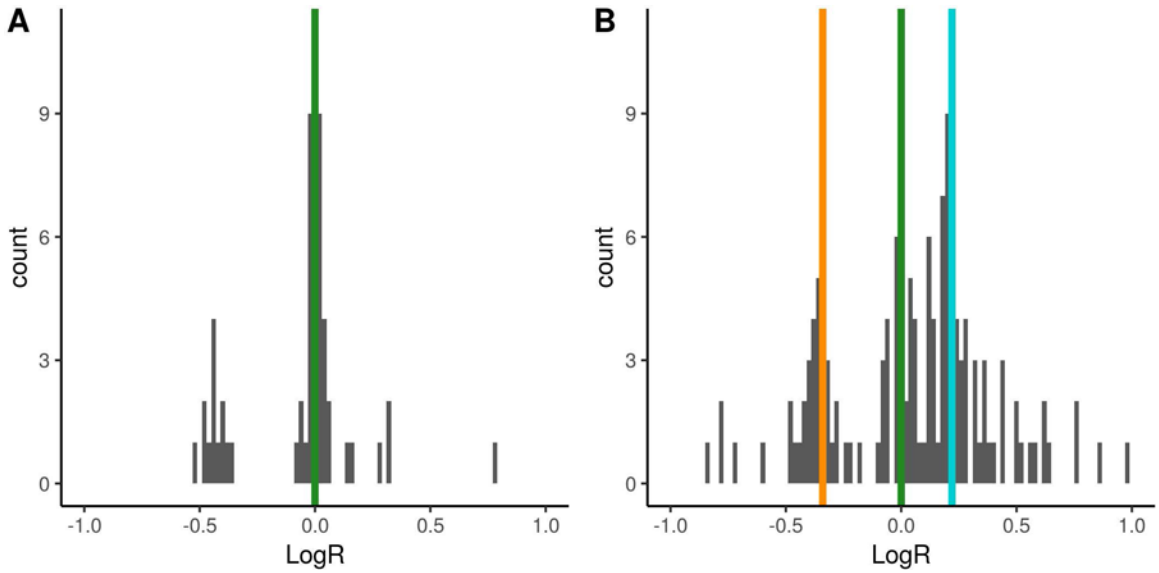

Figure 3. Examples of diploid and aneuploid sample.

Histogram of the LogR of a diploid tumor sample (**A**) and an aneuploid tumor sample (**B**).

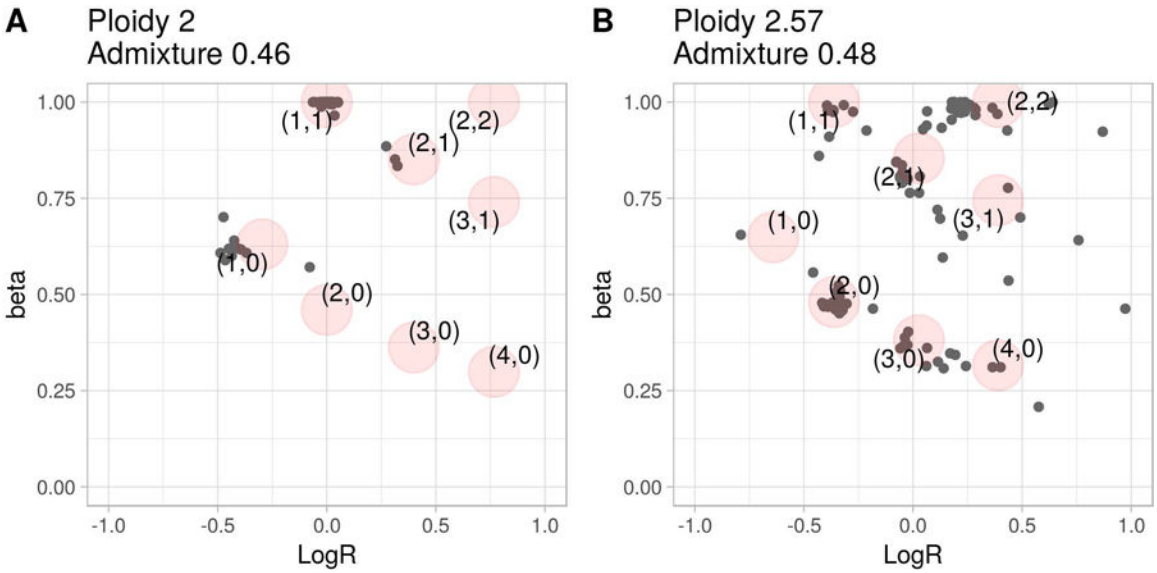

Figure 4. Examples of beta vs LogR space.

Panels (**A**) and (**B**) extend the LogR histograms of Figure 1A and 1B, respectively, to the beta vs LogR space. Each gray dot represents a genomic segment. Large red circles represent expected (beta, LogR) values corresponding to the estimated ploidy and DNA admixture (reported above the corresponding plot). A circle corresponding to clonal homozygous deletions, if represented, would be in $(-\infty, 1)$.

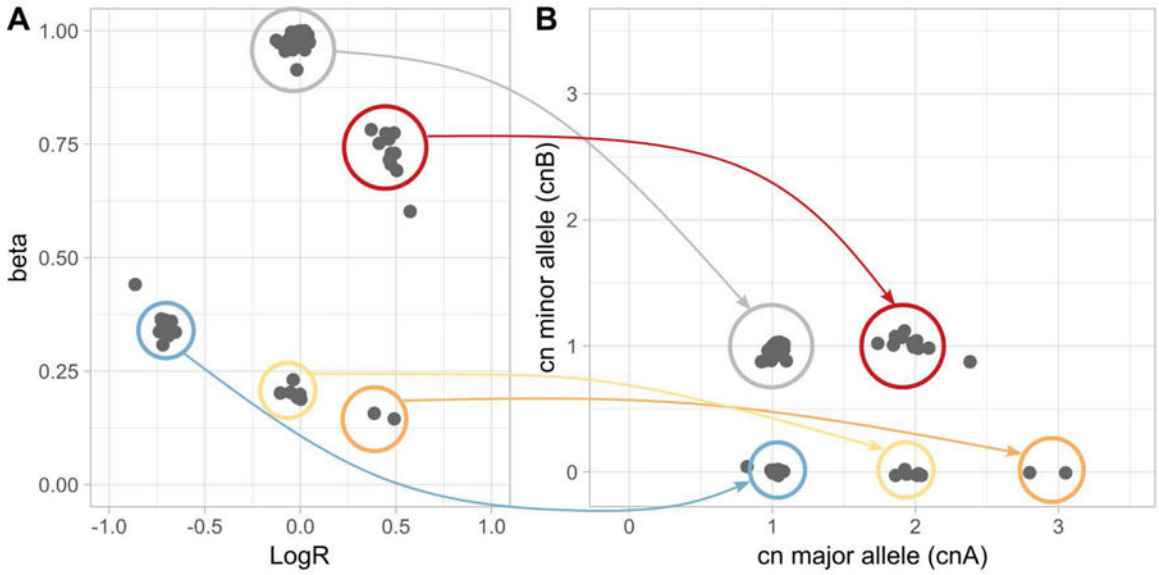

Figure 5. From beta vs LogR to allele specific copy number space.

(**A**) beta vs LogR of a tumor sample (as in Figure 2). (**B**) Allele specific plot obtained transforming data presented in Figure 3A. Each dot corresponds to a genomic segment for which the copy number values of the two alleles are reported (higher copy number values conventionally reported in the x-axis). Colored arrows and circles show how combinations of beta and LogR corresponds to different allele specific copy number values. Color codes: gray Wild Type, light blue hemizygous deletions, red gain, yellow CN-LOH, and orange Gain-LOH. In Gain-LOH, one allele is lost and the LogR indicates gain DNA.

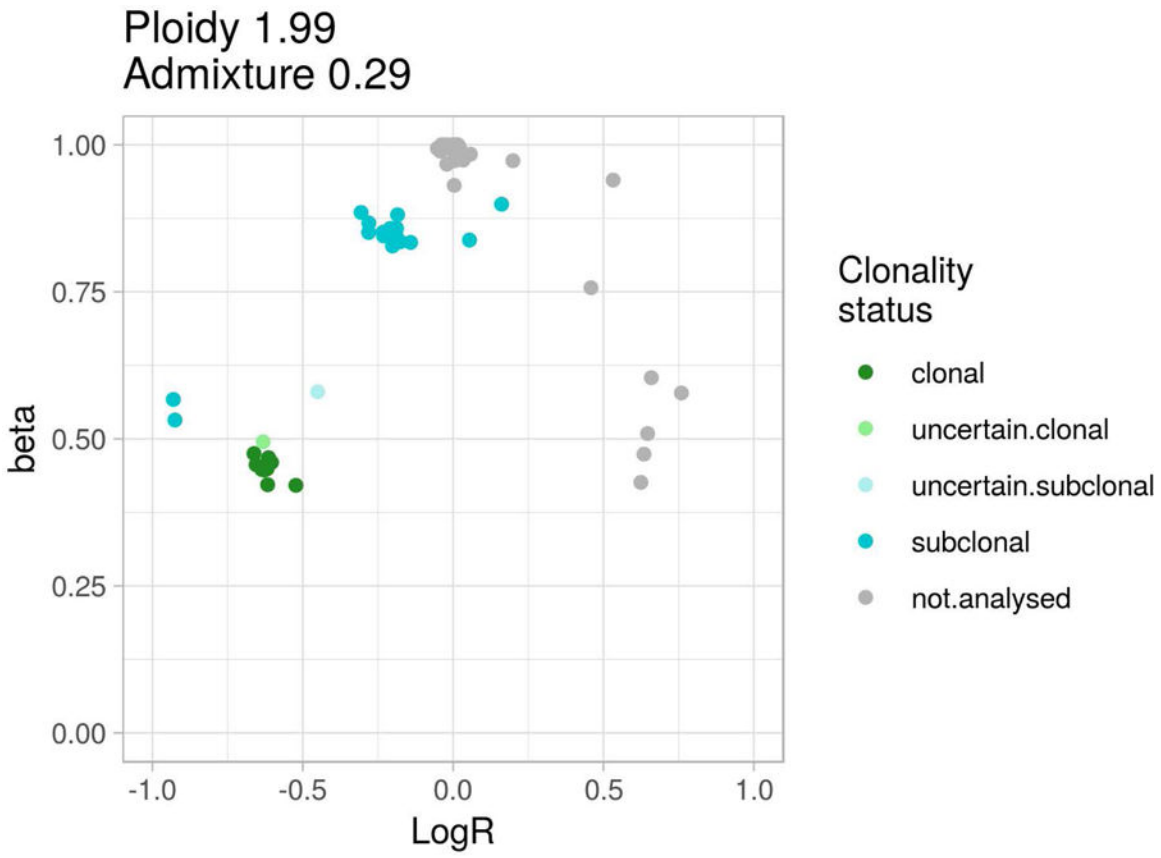

Figure 6. Example of tumor sample with subclonal copy number.

Plot beta vs LogR of a tumor sample with subclonal copy number segments. Each dot represents a genomic segment and color code indicates clonality status as indicated in the color legend.

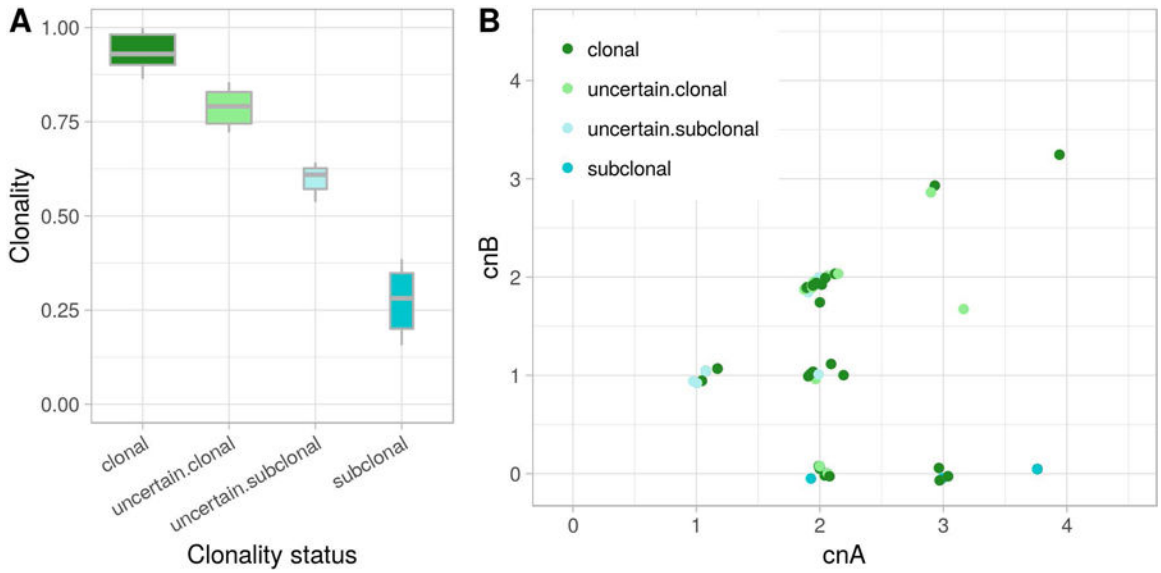

Figure 7. Example of clonality analysis of SNVs.

(**A**) Boxplot reporting the clonality value of the SNVs of a tumor sample. The clonality values (y-axis) distributions are shown including all variants of a tumor sample, stratified by the automatically assigned clonality status class (x-axis). (**B**) For each SNV in panel (**A**), allele specific copy number data of the genomic segment containing the mutations are reported.

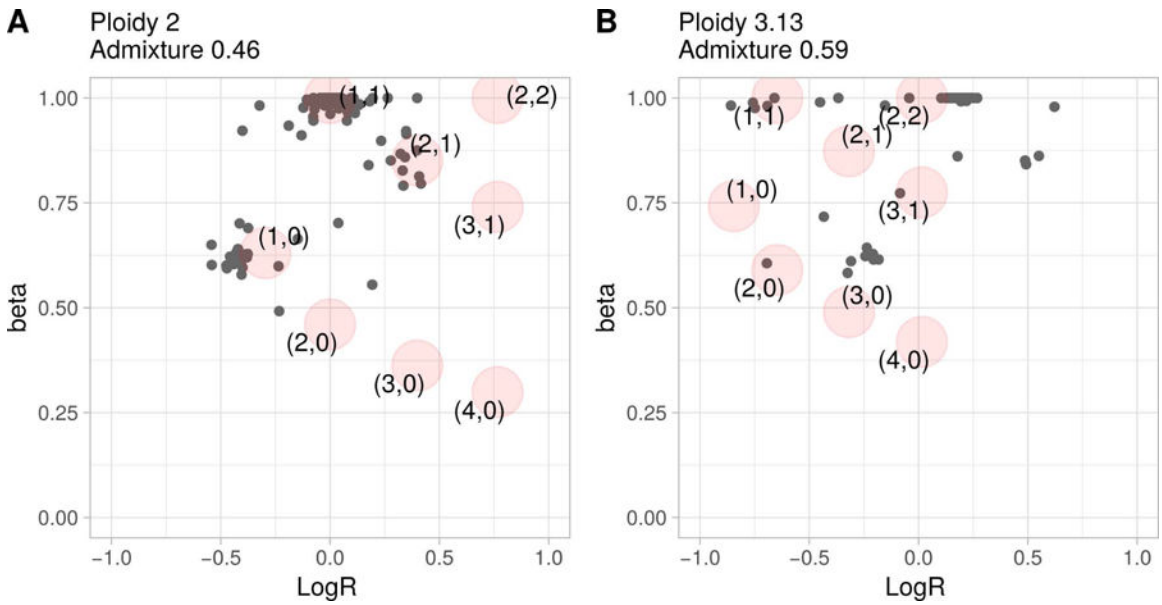

Figure 8. Example of beta vs LogR of segments obtained with different segmentation algorithms.

Plot beta vs LogR for tumor sample from Figure 2A based on the LogR values with EXCAVATOR2 (**A**) or FACETs (**B**).

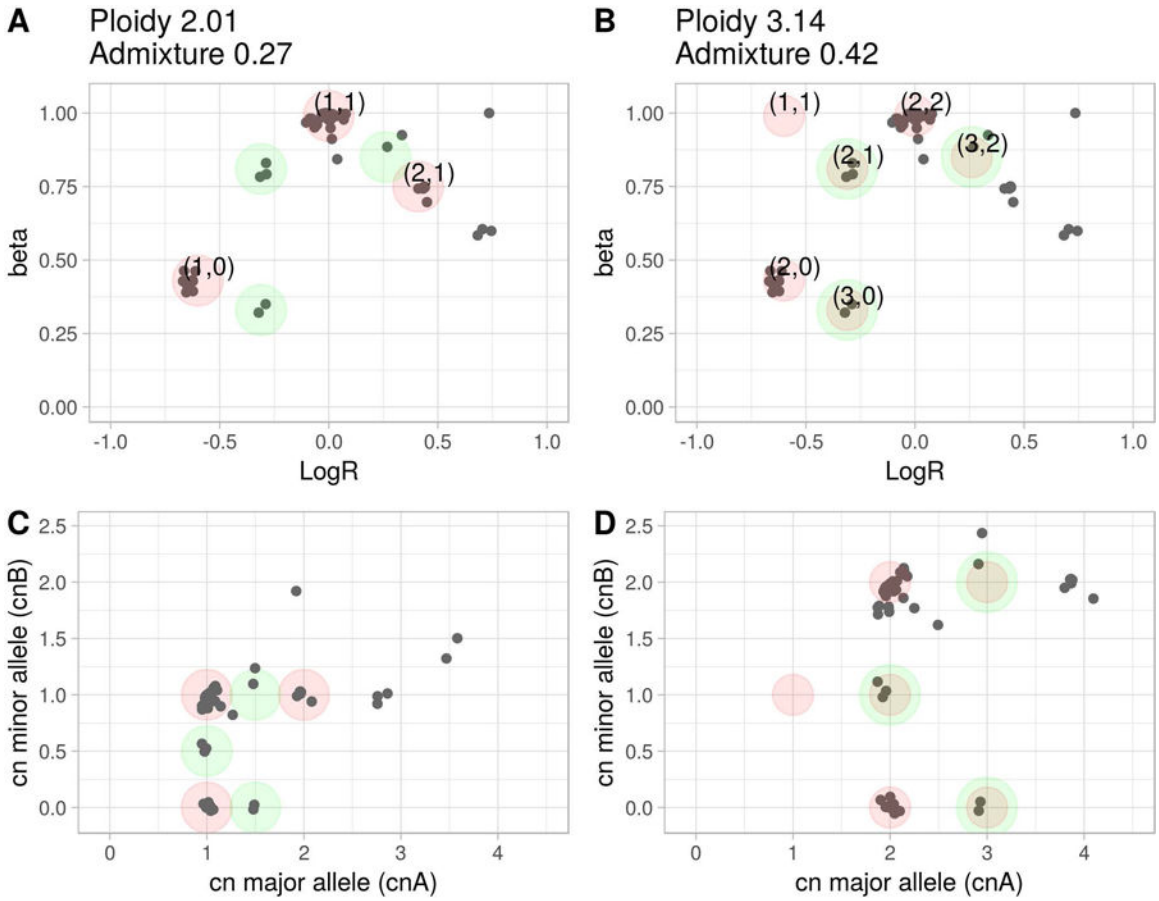

Figure 9. Example of conflicting ploidy estimates.

Beta vs LogR plot of the same tumor sample based on two estimates for ploidy and DNA admixture. Panels (**A**) and (**B**) show expected positions for different allele specific copy number varying ploidy and DNA admixture estimates. Green circles (**A**) highlight genomic segments for which estimates do not fit with observed values. Red circles with green borders (**B**) correspond to green circles in panel (**A**). Panels (**C**) and (**D**) show allele specific copy number plot given the estimates in (**A**) and (**B**), respectively. Circles color code is as for panel (**A**) and (**B**), respectively.

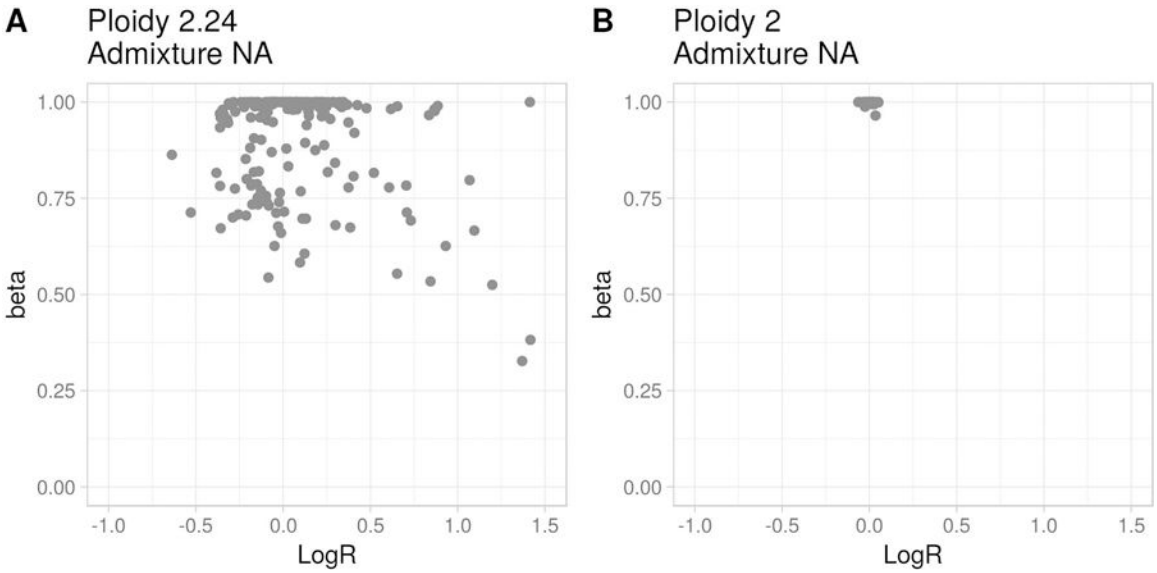

Figure 10. Example of samples with no CLONETv2 DNA admixture estimates.

Examples of tumor samples in the beta vs LogR spaces showing poor segment clusters (**A**) or lack of somatic copy number aberrations (**B**).