Trabajo Fin de Máster

# Sensitivity analysis of the DeCiFer algorithm to plausible variations in the inferred tumor purity.

Emilia Balboa Beltrán

**Máster en Técnicas Estadísticas**

**Curso 2023-2024**

# Propuesta de Trabajo Fin de Máster

| |
|---|
| **Título en galego:** Análisis de sensibilidade do algoritmo DeCiFer a plausibles variacións na pureza tumoral inferida. |
| **Título en español:** Análisis de sensibilidad del algoritmo DeCiFer a plausibles variaciones en la pureza tumoral inferida. |
| **English title:** Sensitivity analysis of the DeCiFer algorithm to plausible variations in the inferred tumor purity |
| **Modalidad:** Modalidad A |
| **Autor/a:** Emilia Balboa Beltrán, Universidad de A Coruña |
| **Director/a:** José Antonio Vilar Fernández, Universidad de A Coruña |
| **Breve resumen del trabajo:**<br>The DeCiFer algorithm states to infer the evolutionary history shared by SNVs based in different constraints. One such constraint is the plausibility between VAF values and tumor purity estimates. The aim of this master's thesis is to analyze the sensitivity of the DeCiFer algorithm to perturbations in the input purity value. |
| **Recomendaciones:** |
| **Otras observaciones:** |

*El director propuesto para este TFM, Don José Antonio Vilar Fernández, no se responsabiliza del contenido de este Trabajo Fin de Máster por no haber tenido acceso a la versión que figura en esta memoria con la antelación suficiente, impidiendo que realizara sus observaciones antes del depósito de la misma.*

# Contents

**Resumen** ix

**Prefacio** xi

**1 Introduction** 1

**2 Variant calling** 3

**3 Characterization of tumor clones** 7

3.1 Inference of the cellular prevalence or cancer cell fraction. . . . . 9

3.1.1 Variant allele frequency, tumor purity and ploidy and allele-specific copy number inference . . . . . 9

3.1.2 Estimation of CP/CCF . . . . . 13

3.1.3 Multiplicity inference . . . . . 14

3.1.4 DeCiFer: Inference of CCF and DCF under the single-split copy number assumption . . . . . 15

3.2 Clustering . . . . . 25

3.2.1 SNV-based subclonal reconstruction - A Bayesian Dirichlet process . . . . . 25

3.2.2 CNA based subclonal reconstruction. The Battenberg algorithm . . . . . 28

3.2.3 DeCiFer: simultaneous clustering and genotype selection using the DCF . . . . . 31

3.2.4 The DeCiFer algorithm . . . . . 32

**4 Sensitivity analysis to purity changes of the DeCiFer algorithm** 35

**5 Simulation and data analysis** 39

**6 Conclusions** 61

**Bibliografía** 63

# Resumen

## Resumen en español

Se han formulado varios algoritmos que intentan reconstruir la arquitectura subclonal del tumor resolviendo problemas de optimización; sin embargo, la mayoría de ellos utilizan heurísticas o son altamente exigentes computacionalmente. Además, no suelen considerar la posible historia evolutiva común de las mutaciones ni considerar todas las posibles fuentes de ruido aleatorio que se transmitirán a lo largo del proceso de reconstrucción. Por lo tanto, después de una breve búsqueda, dado que este trabajo no pretende realizar un análisis sistemático de todos los algoritmos que cubren el área, decidí centrarlo en un algoritmo publicado recientemente y desarrollado con la intención de corregir fallos en algoritmos anteriores. Este algoritmo se llama "DeCiFer" y se explica en el artículo "DeCiFering the elusive cancer cell fraccion in tumor heterogeneity and Evolution". El algoritmo DeCiFer pretende inferir la historia evolutiva compartida por los SNVs en función de diferentes restricciones. Una de esas limitaciones es la verosimilitud entre los valores de VAF y las estimaciones de pureza del tumor. Consecuentemente, en este estudio se realiza una evaluación de la robustez del algoritmo ante variaciones en los datos de pureza de entrada del algoritmo, debido a errores más que plausibles que pueden ocurrir en los pasos anteriores (posibles fuentes de ruido aleatorio), tanto a nivel de laboratorio como a nivel del ámbito de la bioinformática/estadística utilizada para su estimación. El objetivo de este trabajo de fin de máster es analizar la sensibilidad del algoritmo DeCiFer a perturbaciones en el valor de pureza de entrada.

## English abstract

Several algorithms have been formulated that attempt to reconstruct the tumor subclonal architecture by solving optimization problems, however, most of them use heuristics or are highly computationally demanding. Furthermore, they do not usually consider the possible common evolutionary history of the mutations or consider all the possible sources of random noise that will be carried throughout the reconstruction process. Therefore, after a brief search, since this work does not intend to perform a systematic analysis of all the algorithms that cover this area, I decided to focus it on a recently published algorithm developed with the intention of amending flaws in previous algorithms. This algorithm is called "DeCiFer" and it is explained in the paper "DeCiFering the elusive cancer cell fraction in tumor heterogeneity and evolution". The DeCiFer algorithm states to infer the evolutionary history shared by SNVs based in different constraints. One such constraint is the plausibility between VAF values and tumor purity estimates. Consequently, in this study an evaluation of the robustness of this algorithm to variations in the input purity data due to more than plausible errors that can occur in the previous steps (possible sources of random noise), both at the laboratory and bioinformatics/statistic level used for its estimation. The aim of this master's thesis is to analyze the sensitivity of the DeCiFer algorithm to perturbations in the input purity value.

# Prefacio

Several algorithms have been formulated that attempt to reconstruct the tumor subclonal architecture by solving optimization problems, however, most of them use heuristics or are highly computationally demanding. Furthermore, they do not usually consider the possible evolution of the mutations or consider all the possible sources of random noise that will be carried throughout the reconstruction process. Therefore, after a brief search, since this work does not intend to perform a systematic analysis of all the algorithms that cover this area, I decided to focus it on a recently published algorithm developed with the intention of amending flaws in previous algorithms, even though I also refer to other algorithms that the authors mention or are based on. This algorithm is called "DeCiFer" and it is explained in the paper "DeCiFering the elusive cancer cell fraction in tumor heterogeneity and evolution".

An evaluation of the accuracy of this and other algorithms, had been compared in simulated data produced by an algorithm called "clevRsim" whose results are summarized in the recently published paper called "Reconstructing Clonal Evolution—A Systematic Evaluation of Current Bioinformatics Approaches" with not very good results for the DeCiFer algorithm. One of the plausible causes that could account for these differences could be the tested method the authors used to evaluate these algorithms, since as point out a recent pre-print "Evaluation of simulation methods for tumor subclonal reconstruction" most of the reconstruction algorithms test their accuracy using their own custom-made simulation data, which does not allow a comparison among them. Simulation algorithms, on the other hand, also have their own limitations such as not considering all the possible alterations that a tumor can suffer.

Therefore, several reasons can account for the differences in the evaluation and comparison of the reconstruction algorithms, from the type of data used (real or simulated) for their evaluation, to the flaws when considering all the possible alteration that can be produced in the tumor or, as it will be explained in the following section, different sources of variability and uncertainty in the input data, that are not always considered into the reconstruction algorithms.

How consistent these algorithms are to small variations in the input data due to more than plausible errors that can occur in the previous steps (this is the previously mention possible sources of random noise that will be carried throughout the reconstruction process), both at the laboratory and bioinformatics/statistic level, is another important point of consideration.

The aim of this master's thesis is to analyze how the DeCiFer algorithm behaves due to these disturbances in the input data, this is the sensitivity of the algorithm to changes, specifically, in the tumor purity.

Some of the steps of the DeCiFer algorithm are common to previous algorithms so I will explain them as a general process while I will refer those steps that are particular to the DeCiFer algorithm.

# Chapter 1

# Introduction

Cancer is a disease of genetic origin that initiates in a cell or group of cells that undergo alterations that cause a higher than normal rate of mutation and confers them different characteristics, including a growth advantage over normal cells. This produces a mass of tumor cells that inherits the previous genetic background to which new mutations are added, giving rise to a heterogeneous group of cells (Figure 1.1).

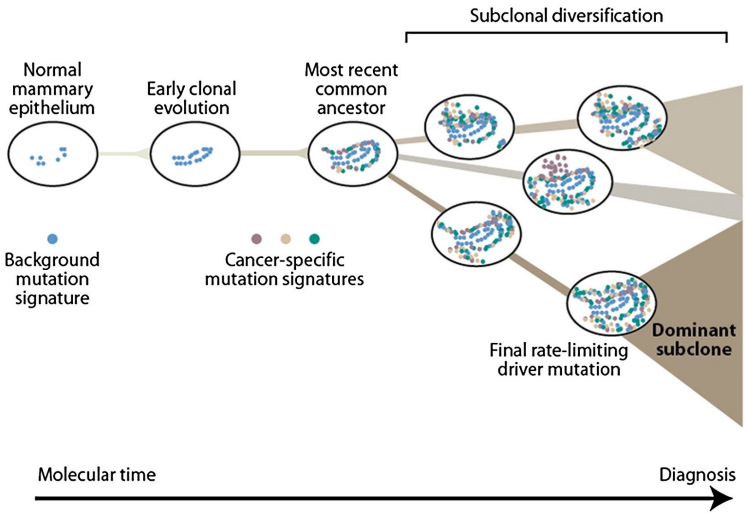



Figure 1.1: Cancer evolution. Taken from Nik-Zainal, S., et al., 2012

Therefore, the evolution of the tumor cells from their origin is characterized by these mutational changes which allows to identify different ***cellular clones***, defined as a group of tumor cells with a common evolutionary history derived from the ***ancestral cell***, and ***subclones***, defined as a subgroup of tumor cells with a common evolutionary history derived from a cellular clone. However, as it is currently not possible to observe this process directly, the data acquired are snapshots of a moment in this evolution, whether in one or several samples, but always in a fraction of the total tumor, several models of tumor evolution have been proposed trying to explain the observed data, which are portrayed in phylogenetic trees once the subclonal architecture has been reconstructed (Figure 1.2).

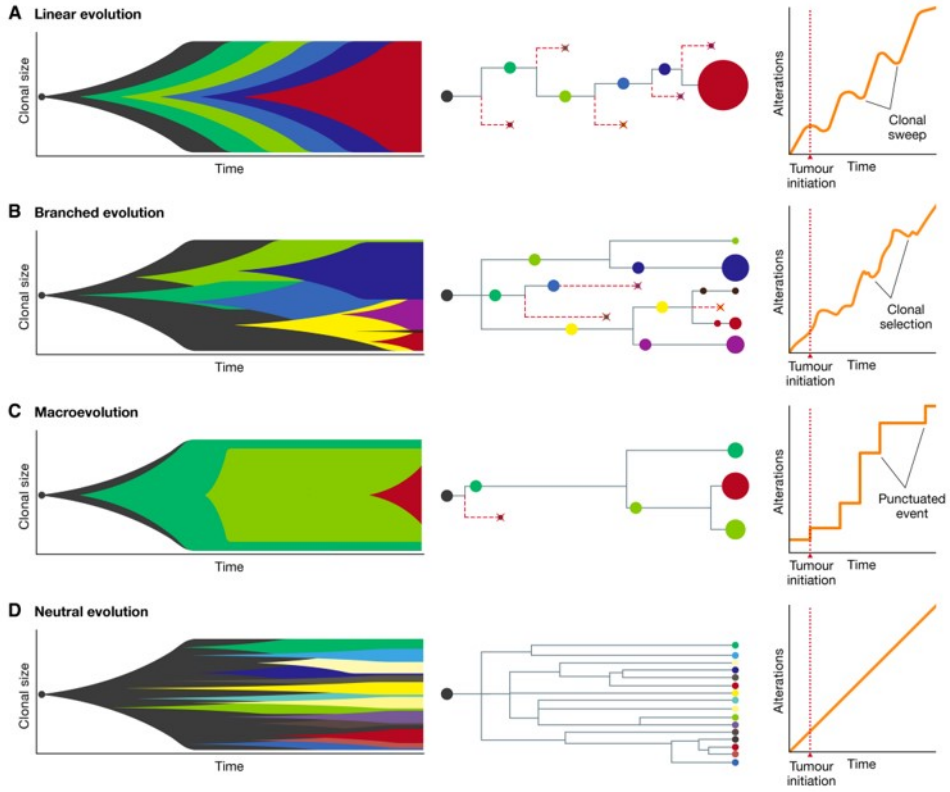



Figure 1.2: Models of tumour evolution. Taken from Vendramin et al., 2021

The subclonal architecture of the tumor is reconstructed using the information provided by the genetic sequences of the analyzed biopsies. From these sequences, an attempt is made to infer the succession of genetic alterations produced in the tumor through subclonal reconstruction. These methods not only help to identify the veracity and precision of the proposed models, but also to identify which model is acting in that tumor and also formulate new hypotheses about its evolution. These methods provide key information to identify subclonal driver mutations, patterns of parallel evolution, and differences in mutational signatures between cell populations. The information derived from knowledge of these evolutionary models is essential in the clinical setting, as it reveals mechanisms of resistance to therapy, tumor dissemination and metastasis. (Nik-Zainal, S., et al., 2012)

Subclonal reconstruction methods follow these steps:

1. **Variant calling process**
2. **Characterization of tumor clones**
3. **Phylogenetic reconstruction**

# Chapter 2

# Variant calling

The variant calling process is intended to identify the variants present in the tumor samples, and implies a series of successive steps, "sequencing, read mapping or de novo assembly, variant calling, filtering of false positives, and sometimes phasing." (Olson, N.D., et al., 2023)

### Sequencing

The first step in subclonal tumor reconstruction is to obtain the DNA sequences of the tumor cells. Once the patient sample is extracted, it is processed to isolate and label the DNA so that it is readable by the sequencing platforms. This involves several chemical reactions whose efficiency is high but not without technical artifacts. To reduce random noise, all laboratory procedures are performed in parallel with reference samples. This process is usually referred as wetlab. (Figure 2.1.1)

Subsequent steps involves bioinformatic analysis to interpret the labeled DNA as the nucleotide sequence using algorithms (outside of the scope of this work) that call the sequence, that is, interpret a fluorescent signal and assign the corresponding nucleotide call (Figure 2.1.2). Different parameters influence in the quality of the sequences obtained, from the type (blood, tissue...) and ***purity*** of the sample (this is, the percentage of the tumor cells relative to normal cell in the sample), DNA extraction method and sequencing protocol, to the sequenced region itself, since there are regions that are more difficult to analyze and more prone to introduce artifacts. (Olson, N.D., et al., 2023)

### Read mapping or de novo assembly

Once the raw sequences are obtained, it is necessary to assemble them, this is mapping their position in the genome, using a reference sequence (Figure 2.1.3). This is another source of possible errors since base-calling accuracy, length of the sequence and the sequenced region itself influence in the correct assembly. (Olson, N.D., et al., 2023)

### Variant calling

Variant calling algorithms (outside of the scope of this work) identify the variants present (alleles) on each chromosomal copy of the genome in the sample compared to a reference genome. (Figure 2.1.3). (Bagger, F.O., et al., 2024)

### Filtering of false positives

Once the variants have been identified, false positives are filtered due to poor reading quality or the appearance of possible artifacts. Filtering of false positives were initially based on features specified by experts, nowadays the algorithms make use of machine learning or deep learning to filter

them, currently, the most used architecture is convolutional neural networks, which considers the characteristics of the sequences of the patient and the reference sample (Figure 2.1.4).

In the best scenario, ***somatic variants*** (variants specific of the tumor) are identify by their comparison with ***germline samples*** (samples taken from normal cells) from the same patient using algorithms based on bayesian statistics adjusted by different factors (outside of the scope of this work) that estimate the genotype of both samples, normal and tumor, and determine whether the observed differences are real or a consequence of technical artifacts. In a not so optimal scenario, in cases where a matched normal sample is not available, the tumor sequences are compared to a database of genetic variation, with the further problem derived from not being processed at the same time so random noise cannot be controlled, or a panel of normal samples, with the inconvenient of calling somatic variants when, in fact, they are germline variants specific to each individual, which might not be a problem depending on the number of normal samples in the panel. (Cortes-Ciriano, I., et al., 2022)

Whatever the case may be, the expected frequency when analyzing germline samples versus tumor samples differs, the ***variant allele frequency (VAF)*** of a ***heterozygous*** (different alleles in a specific location) variant in a germline sample of a ***diploid genome*** (the complete set of chromosomes in a human, one chromosome proceeds from the father, the other from the mother) is around 50%, however, the variant allele frequency of a variant in a tumor sample can theoretically vary due to tumor ***aneuploidy*** (variations from diploidy in a human due to a missing chromosomes or one or more extra chromosomes) and purity, over a continuous range $\in (0, 100\%)$. Therefore, sequence variants from tumor samples undergo additional filtering steps based on their:

- ***Read depth.*** The *read depth* is the number of replicates obtained from the same amplified region, the greater the sequencing depth, the greater the reliability of the results, since the results must be redundant. However, the sequencing platforms have limitations, so a balance must be struck between the coverage of the genome to be analyzed and the sequencing depth.
- ***Variant frequency.*** Very low frequency mutations are usually difficult to estimate and their accuracy depends on the read depth of the sequence. Algorithms usually set a threshold value to consider them a true variant.
- ***Relevance.*** Depending on the goal of the study, additional filtering steps can be included that consider the known relevance of a specific mutation.

Hence, from the initial 500000 somatic variants that can be found in a tumor, after the filtering steps, between 20 and 1500 variants may remain for further evaluation. (Bagger, F.O., et al., 2024)

## Haplotype phasing

A haplotype is a DNA fragment that tend to be inherited in bloc since there is a low probability of recombination (an exchange between the mother's and father's chromosomes after the fusion of the sperm and the egg) within the fragment, hence, single nucleotide polymorphisms (SNPs), (a DNA variation at a specific location present in at least 1% of the population), inside of the fragment are inherited together. Each haplotype can then proceed from the mother or the father. Phasing refers to ordering heterozygous SNPs into haplotypes considering the chromosome to which they belong, that of the mother or the father. Phasing can be accomplished by comparing reads to samples from the mother or father or by using large population panels to impute the most likely order of alleles. (Olson, N.D., et al., 2023)

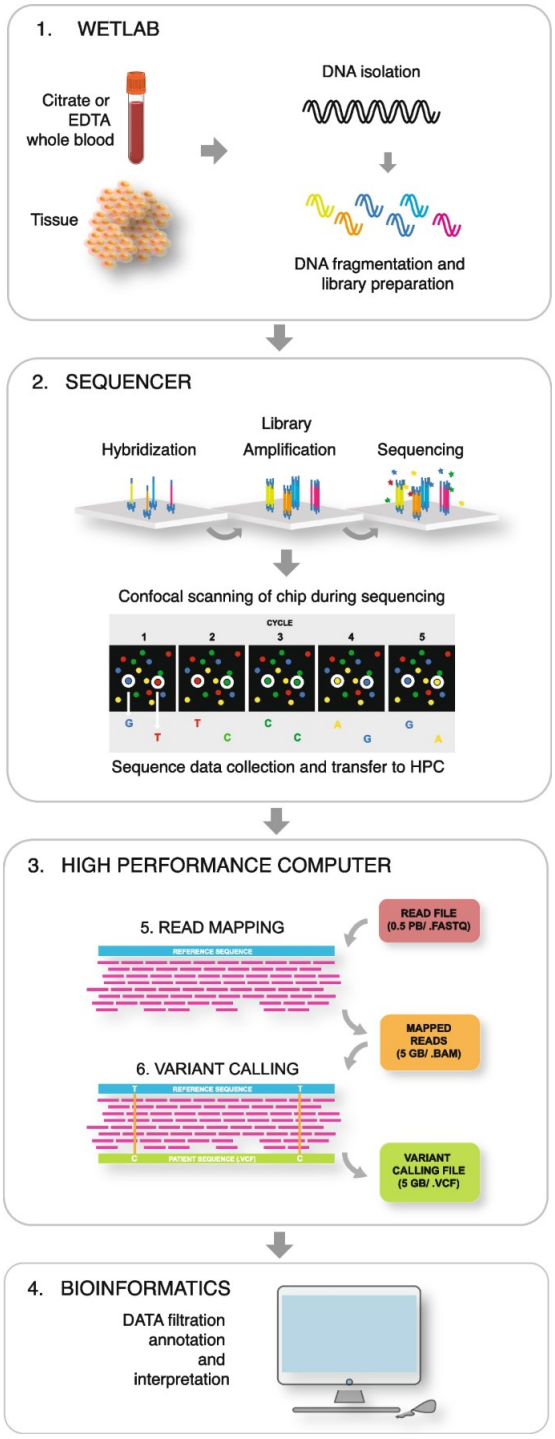



Figure 2.1: Schematic representation of the WGS laboratory and bioinformatics flow. Taken from FO Bagger et al. 2024

# Chapter 3

# Characterization of tumor clones

Characterization of tumor clones is intended to determine the tumor composition in terms of purity, nº of subclones and their prevalence, and ***mutational load*** (this is, the total number of mutations), by inferring the cellular prevalence or fraction of cancer cells, and, subsequently, determining the mutational profiles of those subclones using clustering algorithms. (Salcedo, A., et al., 2020) (Figure 3.1)

Tumor subclonal reconstruction algorithms based on cell genetics usually employ the same approach. One the distinctive features the different clones are distinguished by are the mutations that harbor, therefore, the first step is the identification and analysis of the mutations present in the tumor from the sequencing data. These mutations can be ***single nucleotide variants (SNV)*** (variations in only one nucleotide), ***copy number alterations*** (alterations in the number of copies of a DNA segment), or small ***insertions***, ***deletions*** (***indels***). Once the frequency of these alterations is calculated, the ***cellular prevalence (CP)*** is estimated, this is, the fraction of cells that harbor a mutation in the sample, and/or, the ***cancer cell fraction (CCF)***, this is, the fraction of cancer cells that harbor at least one copy of the mutation. Subsequently, it is performed the clustering of SNVs in cellular subclones based on the similarity of CP/CCF values. Once the clusters are defined it can be deduced the phylogenetics trees.

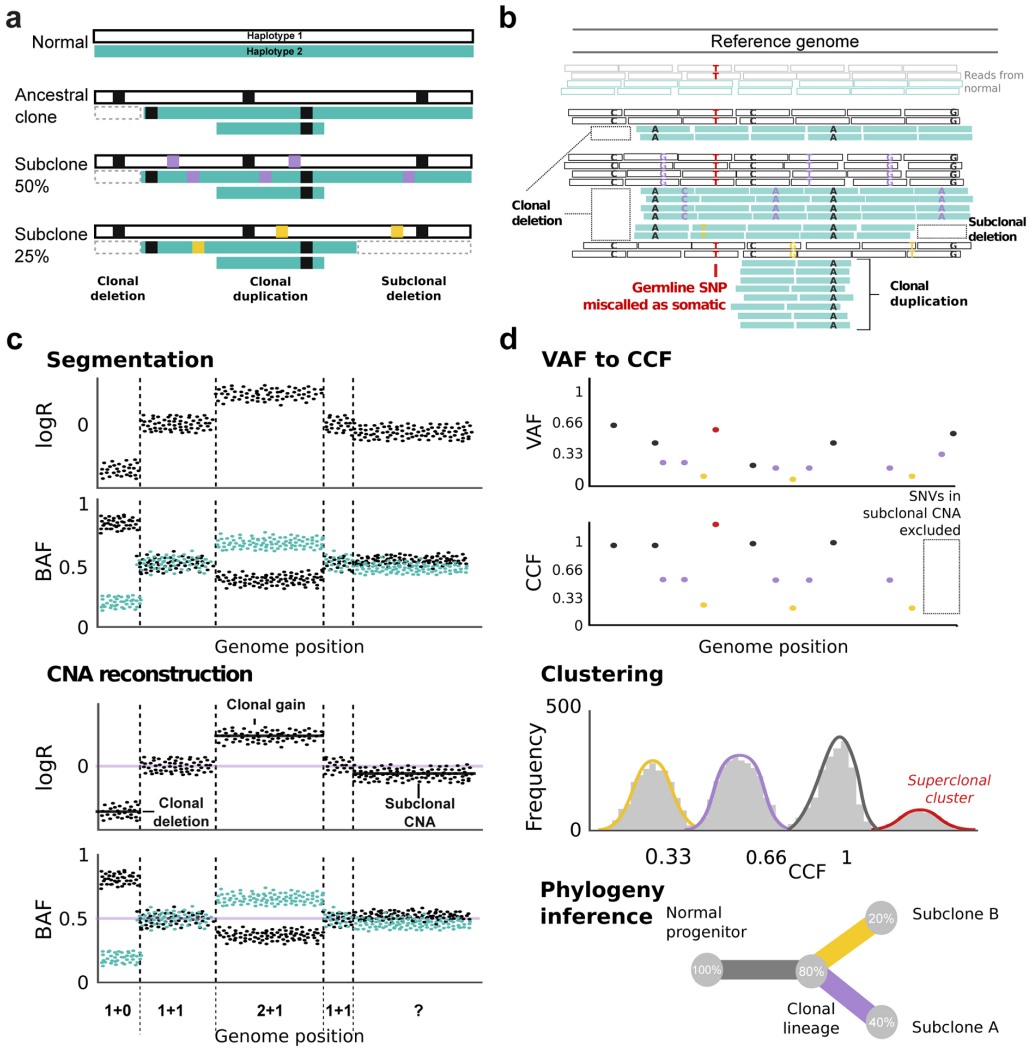
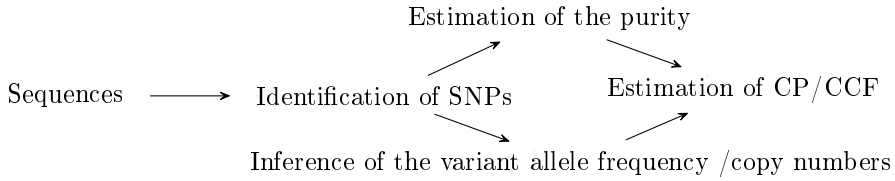



Figure 3.1: Standard Workflow and Input Data for Subclonal Reconstruction. (a) A simplified example of tumor clonal genotypes. We illustrate a tumor containing two subclones at 50% (purple) and 25% (yellow) CCF, both descended from a common ancestral clone (100% CCF, black). The remaining 25% of tumor cells are indistinguishable from the ancestor. (b) First, somatic mutations are called from aligned reads. Read depth must be much higher (coverage >60x) than illustrated for mutation calling and subclonal reconstruction. Similarly, an elevated local mutation burden is illustrated. A somatic variant caller identifies somatic SNVs by comparing to a matched normal, although germline SNP contamination may occur. (c) Second, CNA reconstruction is performed. It typically uses read depth and B-allele frequency (BAF) data for heterozygous SNPs. (d) Third, CNAs are used to translate the measured SNV VAF to a CCF/CP estimate. This procedure relies on an accurate SNV multiplicity estimates which are typically inaccurate in subclonal CNAs so we exclude these regions from the analysis. SNV CCFs are then clustered to identify (sub)clonal lineages in the sample. False positive SNVs or inaccurate CNAs can cause spurious superclonal clusters (i.e. with CCF>1). Finally, phylogenetic reconstruction infers the ancestral relationships among lineages. Taken from Tarabichi et al. (Tarabichi, M., et al., 2021) Reproduced with permission from Springer Nature

## 3.1 Inference of the cellular prevalence or cancer cell fraction.

The cellular prevalence of a tumor, or the cancer cell fraction, is inferred from the relation between the frequency of mutations detected in the analyzed sample and its purity.




### 3.1.1 Variant allele frequency, tumor purity and ploidy and allele-specific copy number inference

The ***variant allele frequency (VAF)*** is the ratio of the variant allele to the germline allele, therefore it is calculated based on the ratio of paired sequences; being $v_i$ the estimated VAF, for a biallelic $SNV_i$ in a position i, it goes as follows:

$$v_i = \frac{r_{mut,i}}{r_{mut,i} + r_{ref,i}} \tag{3.1}$$

where $r_{mut,i}$ and $r_{ref,i}$ are the number of reads of the mutant variant and the reference variant, respectively. (Dentro, S.C., et al., 2017)

Tumor purity and ***ploidy***, this is, the number of sets of chromosomes, as well as allele-specific copy number can be inferred by optimization algorithms from the information provided by the SNPs.
Different algorithms have been developed that estimate these values from the sequencing data but they all have an error in the precision of any of the inference. In this section I choose to explain two of them. The first one is one of the most used algorithms to this aim is the ***ASCAT algorithm*** that uses the **B-allele frequency (BAF)** and **Log R** estimates. (VanLoo et al., 2010))
The second, the HATCHet (Holistic Allele-specific Tumor Copy-number Heterogeneity) algorithm, was developed by the same group that developed the DeCiFer algorithm and uses the **read-depth ratio (RDR)** and also the BAF estimates. (Zaccaria, S. and B.J. Raphael, 2020)

***B-allele frequency (BAF)*** is the normalized frequency of the two possible alleles in heterozygous SNPs (allele A and B). Being $n_{A,i}$ y $n_{B,i}$ the copy number of alleles A and B at a location i, that are inferred by the number of reads at that location $r_{A,i}$ and $r_{B,i}$, respectively, its estimation follows:

$$b_i = \frac{n_{B,i}}{n_{A,i} + n_{B,i}} \tag{3.2}$$

In the absence of chromosomal alterations, the BAF value is 0.5, deviations from this value indicate somatic alterations. Considering the tumor and normal cells in the sample and knowing the purity of the tumor $\rho$, the estimation of BAF, $b_i$, can be expressed as:

$$b_i = \frac{\rho\ n_{B,t} + (1-\rho) n_{B,n}}{\rho\ (n_{A,t} + n_{B,t}) + (1-\rho)(n_{A,n} + n_{B,n})} \tag{3.3}$$

The ***logR*** is another parameter that can be extracted from the sequenced samples, this value indicates the total signal intensity. For a diploid sample in a 100% tumor sample it goes as:

$$r_i = \gamma \; log2 \left( \frac{n_{A,i} + n_{B,i}}{\psi_n} \right) \tag{3.4}$$

where $\gamma$ is a technology-specific constant $\leq 1$, $\psi_n$ indicates ploidy in normal cells which are assume to be diploid, therefore, $\psi_n = 2$ and $r_i = 0$.

While aneuploidy does not affect BAF values, Log R values are affected not only by the tumor purity but ploidy, therefore:

$$r_i = \gamma \; log2 \left( \frac{2 \; (1 - \rho) + \rho \; (n_{A,i} + n_{B,i})}{\psi_n (1 - \rho) + \rho \psi_t} \right) \tag{3.5}$$

being $\psi_t$ the ***tumor ploidy*** and $\rho \in (0, 1)$.

### *The ASCAT algorithm*

The ASCAT algorithm attempts to infer allele-specific copy number and optimal values for purity and ploidy of a tumor by calculating the local minima of the genome-wide error between observed and expected allele-specific copy numbers values starting from a combination of a predetermined pairs of values of $\rho$ and $\psi_t$. (Van Loo, P., et al., 2010, Tarabichi, M., et al., 2021)

This algorithm includes a previous step with a segmentation and filtering algorithm (the ***Allele-Specific Piecewise Constant Fitting (ASPCF)*** algorithm) that reduces the noise from the input LogR and BAF data. (Figure 3.1.C) The ASPCF algorithm simultaneously fits piecewise constant regression functions to the data coercing segmentation to the same location in both, LogR and BAF functions.

Let be a data set of different locations, where $x_1 < x_2 < ... < x_n$ denotes the probes locations with an associated LogR $(r_1, ..., r_n)$ and BAF $(b_1, ..., b_n)$ values, the ASPCF algorithm aims to infer how the region has been segmented by grouping the probes into subsets $I_1, ..., I_Q$, through a segmentation that minimizes the penalized optimization criterion with respect to the number of segments Q and the assignment of the probes to segments.

$$\sum_{j=1}^{Q} \sum_{i \in I_j} [w(r_i - (ave \{r_s\}_{s \in I_j}))^2 + (1 - w)(b_i - ave(\{b_s\}_{s \in I_j}))^2] + \lambda Q \tag{3.6}$$

The terms in the brackets are the goodness of fit to the LogR and BAF data, respectively, where $ave \{r_s\}_{s \in I_j}$ and $ave \{b_s\}_{s \in I_j}$ indicates the average of $r_s$ and $b_s$, concordantly, for probes s in the segment I, while the last one is a penalty for discontinuities, being $\lambda > 0$ a constant, which must be provided as well as the minimal length of the segment, while $w = 0.5$ by default. This implies a heterozygous SNP is needed to start a new segment. (VanLoo et al., 2010, Dentro, Stefan Christiaan, 2020)

The next step it to compute the mean deviation from 0.5 (named **d**) and the SD (called **s**) for each previously determined segment. If there is no allelic bias, this is, the alleles are balanced ($A_s \approx B_s$), for a given constant $\tau > 0$, if $d < \tau s$ the single BAF value returned is 0.5, otherwise, if allelic bias exists, this is, $d \geq \tau s$, two symmetric values around 0.5 are obtained.

The input for the ASCAT algorithm is the ASPCF-smoothed data.
Considering the number of copies is a non-negative integer (NNI), the algorithm assumes the expected value of the allele-specific copy numbers would be the NNI value closest to the germline, therefore, rounds the observed value to its nearest NNI. The algorithm, then, minimizes the distance between

this value and the observed allele-specific copy number starting from a previously set grid of values of purity, $\rho \in (0,\ 1)$ and ploidy $\psi_t$. (VanLoo et al., 2010, Tarabichi, M., et al., 2021)

$$d(\rho, \psi_t) = \sum_i w_i((\widehat{n_{A,i}}(\rho, \psi_t) - round(\widehat{n_{A,i}}(\rho, \psi_t)))^2 + (\widehat{n_{B,i}}(\rho, \psi_t) - round(\widehat{n_{B,i}}(\rho, \psi_t)))^2) \tag{3.7}$$

for segments with allelic bias the weight is $w_i = 1$, while for those without allelic bias $w_i = 0.05$.

A ***goodness-of-fit score (g)*** can be calculated for each local minima determined, rescaling the previous result to a percentage of $g = 100\%$ when d = 0, and g = 0 when d = the distance obtained when the allele-specific copy numbers for each SNP differ 0.25 from nonnegative integer ($d = \sum_1 w_i(2{\cdot}0.25^2)$, value chosen as a reasonable maximum distance (averaged over all probes).

After excluding improbable inferences, the algorithm gives one or multiple plausible results (Figure 3.2).

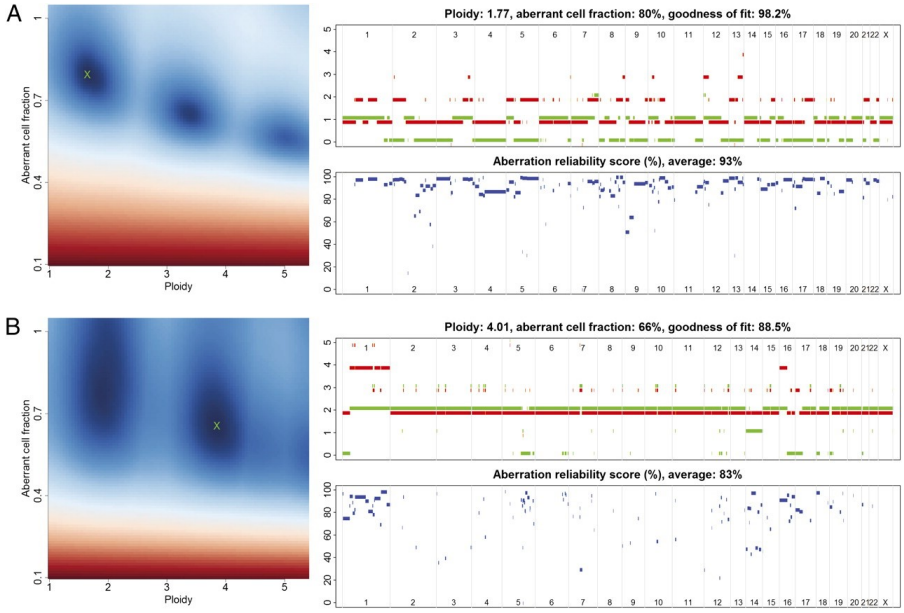



Figure 3.2: ASCAT profiles and their calculation. Two examples are given: (A) a tumor with ploidy close to 2n and (B) a tumor with ploidy close to 4n. (Left) ASCAT first determines the ploidy of the tumor cells $\psi_t$ and the fraction of aberrant cells $\rho$. This procedure evaluates the goodness of fit for a grid of possible values for both parameters (blue, good solution; red, bad solution). On the basis of this goodness of fit, the optimal solution is selected (green cross). Using the resulting tumor ploidy and aberrant cell fraction, an ASCAT profile is calculated (Upper Right), containing the allele-specific copy number of all assayed loci [copy number on the y axis vs. the genomic location on the x axis; green, allele with lowest copy number; red, allele with highest copy number; for illustrative purposes only, both lines are slightly shifted (red, down; green, up) such that they do not overlap; only probes heterozygous in the germline are shown]. Finally, for all aberrations found, an aberration reliability score is calculated (Lower Right). Taken from VanLoo et al., 2010

For this algorithm to be able to infer purity, it requires that aneuploidy exists (which is expected in most tumors), there is no power to infer purity in a mixture with tumor cells in a diploid state. (Tarabichi, M., et al., 2021)

Solving the optimization problems of the ASCAT algorithm gives inferences about tumor purity and ploidy and subsequently assigns a copy number status to each segment using the following equations:

$$\widehat{n_{A,s}} = \frac{\rho - 1 - (1 - b_s)\ 2^{l_s}\ (2(1-\rho) + \rho\psi_t)}{\rho} \tag{3.8}$$

$$\widehat{n_{B,s}} = \frac{\rho - 1 + b_s \; 2^{l_s} \; (2(1-\rho) + \rho\psi_t)}{\rho} \tag{3.9}$$

where $\widehat{n_{A,s}}$ and $\widehat{n_{B,s}}$ are the copy number inferred for allele A and B, respectively, of the segment s, $b_s$ and $l_s$ the BAF and logR value for that same segment and $\psi_t$ is the average ploidy for the tumor cells in the analyzed sample. (Van Loo, P., et al., 2010)

### *The HATCHet algorithm*

HATCHet approaches the identification of allele and clone specific CNAs clones and their proportion by clustering the ***read-depth ratio (RDR)*** (the ratio between observed and expected reads in a chromosomal position) and BAFs from several samples, accounting for the presence of CNAs and whole-genome duplications (WGDs). Therefore, HATCHet algorithm differentiates from previous algorithm in that it directly infers fractional copy numbers instead of using tumor ploidy and purity to model the RDRs and BAFs and inferred the CNAs, since the authors do not consider these estimators appropriate to analyze tumor heterogeneity.

The HATCHet algorithm returns for each segment $s$ in each clone $i$ the copy-number states $(a_{s,i}, b_{s,i})$, in its matrix form $A = a_{s,i}$ and $B = b_{s,i}$ and for each sample, $p$ the clone proportions $u_{i,p}$, in its matrix form $U = u_{i,p}$.

The algorithm inferences these matrix by two modules. The first one infers for each segment $s$ in each sample $p$ the allele-specific fractional copy numbers

$$f^A_{s,p} = \sum_i a_{s,i} u_{i,p} \; --> matrix \; F^A \tag{3.10}$$

$$f^B_{s,p} = \sum_i b_{s,i} u_{i,p} \; --> matrix \; F^B \tag{3.11}$$

Neither $F^A$ nor $F^B$ are identifiable from the DNA sequences, but they can be inferred under some assumptions knowing the state of the WGDs.

The second module solves the matrix factorization problem

$$F^A = AU$$
$$F^B = BU$$

using further constraints to infer allele and clone-specific copy numbers and clone proportions.

The ensuing optimization problem is solved by a coordinate-descent algorithm and, to avoid over-fitting, uses a model selection criteria to obtain the optimal in a balance between solutions with many subclonal CNAs and solutions with WGD.

The tumor purity is calculated as the sum of the proportion of all tumor clones present in a sample,

$$\rho_p = \sum_{i=2}^{n} u_{i,p} \tag{3.12}$$

being the clone proportions represented as a matrix $U = [u_{i,p}]$

### 3.1.2 Estimation of CP/CCF

Knowing the estimated VAF, $v_i$, and tumor purity $\rho$ in a tumor without chromosomal aberrations, $\psi = 2$, in a region with a heterozygous SNV, the estimated value of CCF, $c_i$, can be computed as: (Satas, G., et al., 2021)

$$c_i \approx \frac{\psi\ v_i}{\rho} \tag{3.13}$$

Nonetheless, tumor cells ploidy can differ from ploidy in normal cells, therefore, it is required to infer the copy number state of the SNV and identify which is the altered allele, whether it harbors the SNV or not. This copy number state in a cell or cellular clon is called ***multiplicity of the mutation***. In other words, multiplicity of a mutation refers to the number of copies of that SNV that a cell harbor.

Satas, G., et al., 2021, exposes this concept with a formal notation defining the genotype of a cell as a function of 3 parameters considering the number of copies is a non-negative integer (NNI), the nº of maternal copies, $x$, the nº of paternal copies, $y$, and the multiplicity of a mutation, $m$, the genotype of a cell is defined by $(x, y, m)$, being $m \leq x + y$ and, CCF the fraction of cells with $m \geq 1$.

Considering the possibility of aneuploidy in the tumor cells and the multiplicity of the mutation, the CCF, $c_i$, calculations are generalized as:

$$c_i \approx \frac{F}{M}\frac{v_i}{\rho} \tag{3.14}$$

being F the estimation of the ***fractional copy number***, this is, the mean of the number of copies in all the cells, normal and tumor cells included.

$$F = \rho n_{tot,t,i} + (1 - \rho) n_{tot,n,i} \tag{3.15}$$

where $n_{tot,t,i}$ and $n_{tot,n,i}$ is the tumor and normal mean, respectively, of the number of chromosomal copies.

Some methods group SNVs as a fraction of all cells in the sample, normal and tumor cells included, this is the estimated CP, $cp_i$,:

$$cp_i = c_i \rho \tag{3.16}$$

### 3.1.3 Multiplicity inference

Most of the methods infer multiplicity based on the assumption that all cells harboring a SNP have the same multiplicity value, which is known as the ***"constant mutation multiplicity"*** assumption. (Satas, G., et al., 2021)

The equation 3.14 implies that $m = M$ is fixed for all the cells that harbors the SNV, Satas, G., et al., 2021, formulate it as "At every SNV locus, there exists an integer $M \geq 1$ such that all genotypes at the locus have the form (x, y, m) where either $m = 0$ or $m = M$." (Satas, G., et al., 2021) Therefore, there is a CCF value for each SNP. To infer this M value, there are methods that round its multiplicity value to its nearest NNI, using an heuristic approach, as follows:

being $u_i$

$$u_i = v_i \frac{1}{\rho} \left[\rho n_{tot,t,i} + (1-\rho) n_{tot,n,i}\right] \tag{3.17}$$

the percentage of alleles that harbor a mutation, the $CFF_i$ is,

$$c_i = \frac{u_i}{m_i} \tag{3.18}$$

so, knowing the $u_i$ it could be inferred the mutation multiplicity $m_i$ and the CCF that harbor that mutation, under the following reasoning (Dentro, S.C., et al., 2017):

- the mutation is clonal, $CCF = 1$, since all the cells harbor the SNV and in each cell the number of chromosomal copies, $m_i$, is an integer
- the mutation is subclonal, $CCF < 1$, and, if it is only harbored by a chromosomal copy, its $m_i = 1$, therefore $u_i$ will be $<1$

In these conditions, $m_i$ can be deduce from $u_i$

$$m_i \begin{cases} |u_i|, & u_i \geq 1 \\ 1, & u_i < 1 \end{cases} \tag{3.19}$$

However, multiplicity is not a constant parameter among cells in a tumor due to the possibility of alterations in the subclonal copy number, which could lead to non-identifiability issues, this is a parameter inferred by modeling that is not reliable.(Satas, G., et al., 2021)

Some methods apply different approaches based in evolutionary models that evaluate both, the SNVs and copy number alterations, but to a highly computational cost. (Satas, G., et al., 2021)

### 3.1.4 DeCiFer: Inference of CCF and DCF under the single-split copy number assumption

Satas, G., et al., 2021, highlighted the "overly simplistic assumptions" that algorithms until then made when estimating the CCF based on the "constant mutation multiplicity" assumption. To make more genetically realistic inferences about the prevalence of the subclones in a tumor they construct a novel statistic, the ***descendant cell fraction (DCF)*** that considered (more complex evolution) chromosomal deletion after a mutation occurs. The DCF consider the mutation prevalence at the time of sample analysis but also allows reconstructing its history in order to avoid erroneous inferences in the evolutionary history of the mutations and generate more parsimonious tumor phylogenetic trees.

Besides (to solve previous problems) Satas, G., et al., 2021 developed an algorithm **DeCiFer** that "estimates DCFs and clusters mutations using a phylogenetic model".

Previously explained, tumor samples, processed and then analyzed by the sequencing platform, are presented in the form of nucleotide reads (overlapping and repeated) that, collate against a reference sample, allow to identify the mutations the tumor harbors. The DeCiFer algorithm, subject to the restrictions imposed by a plausible evolutionary process, deciphers the parameters that can explain the observed data with the aim of segregating these mutations into clusters that allow determining the fraction of cancer cells with similar genetic background, since they come from the same evolutionary line. Like previous methods they do not assume multiplicity as a constant, and introduce the ***"single-split copy number assumption"*** for CCF calculation based on standard evolution models and copy number alterations. (Satas, G., et al., 2021)

#### Inference of CCF under DeCiFer notation

Being $\Gamma$ the ***"genotype set"*** at the SNV locus, for each genotype $(x, y, m)$ its sample prevalence is specified as $g_{(x,y,m)}$. The proportion of genotypes in $\Gamma$ is defined by:

$$\mathbf{g} = [g_{(x,y,m)}]_{(x,y,m)\in\Gamma} \quad \text{that satisfy} \begin{cases} g_{(x,y,m)} \geq 0 \\ \sum_{(x,y,m)\in\Gamma} g_{(x,y,m)} = 1 \end{cases} \tag{3.20}$$

Therefore, for a given purity, $\rho$, knowing the pair $(\Gamma, \mathbf{g})$, the CCF is calculated "uniquely" by:

$$c = \frac{1}{\rho} \sum_{(x,y,m)\in\Gamma_{CCF}} g_{(x,y,m)} \tag{3.21}$$

being $\Gamma_{CCF} = (x, y, m) \in \Gamma | m \geq 1 \subseteq \Gamma$ the genotype set that harbor the SNV 3.1.4 .

$\Gamma$ and $\mathbf{g}$ cannot be inferred from the data, but it can the proportion of cells with a given number of copies $\boldsymbol{\mu} = \mu_{(x,y)}$ at a locus (using e.g. Battenberg, explained in section 3.2.2),

$$\mu_{(x,y)} = \sum_{(x,y,m)\in\Gamma} g_{(x,y,m)} \tag{3.22}$$

as well as the VAF, for all copy numbers (x, y).

$$v = \frac{1}{F} \sum_{(x,y,m)\in\Gamma} m \cdot g_{(x,y,m)} \tag{3.23}$$

where $F$ is the fractional copy number defined as $\sum_{(x,y)}(x + y) \cdot \mu_{(x,y)}$.

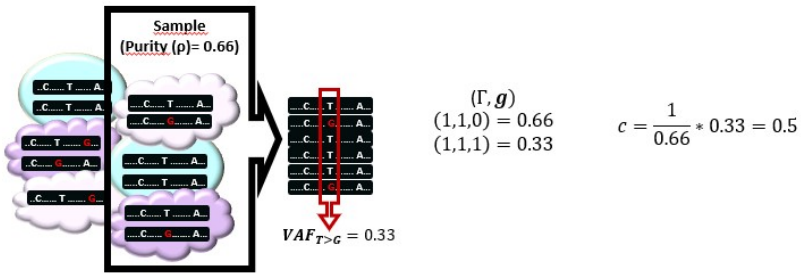



Figure 3.3: The figure shows the calculated CCF knowing the genotype set and the prevalence of each genotype for a given sample whose purity value is also known. A tumor sample is illustrated (normal cells are rounded and colored in blue, and the tumor cells are colored in a range of violet). Each cell contains its DNA chain with the mutations indicated in red. The VAF values for the reads obtained are also indicated.

### Single-split copy number assumption

Since different genotypes $(\Gamma, \mathbf{g})$ can explain the data, their method is based in establishing restrictions to the possible values that can fit with the VAF, $v$, and $\boldsymbol{\mu}$ statistics.

With that aim, they establish the "single-split copy-number assumption" which implies that at each locus of a mutation there are exactly a number of copies $(x^*, y^*)$ with two different genotypes $(x^*, y^*, 0)$ and $(x^*, y^*, m^*)$, which basically means assuming the heterozygosity of the locus subject to the evolutionary rules 3.1.4.

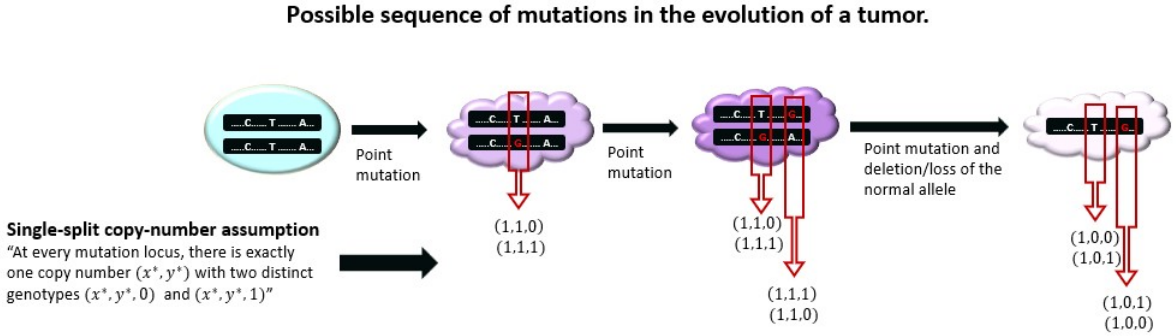



Figure 3.4: Representation of the Single-split copy number assumption. A tumor sample is illustrated (normal cells are rounded and colored in blue, and the tumor cells are colored in a range of violet). Each cell contains its DNA chain with the mutations indicated in red.

If a set of genotypes, $\Gamma$, adheres to this assumption, it is denoted as $\Gamma^*$.
These sets, $\Gamma^*$, have two desirable properties:

1. "They arise from standard evolutionary models for SNV and copy number aberrations".
   Evolutionary events with low probability, such as homoplasy, this is, the occurrence of the same mutation that arises independently in different subclones or chromosomes, that were allowed under constant mutation multiplicity, are restricted under this assumption.

2. "If genotype proportions $\mathbf{g}$ satisfying equations 3.22 and 3.23, then they are unique."
   If this condition is met, $\Gamma$ is consistent with $\mu$ and VAF, but the CCF remains non-identifiable since $\boldsymbol{\mu}$ and VAF are not enough to determine $\Gamma^*$.

To prove the 2º statement Satas, G., et al., 2021, developed the reasoning to state the following lemmas:

Considering the following possible cases:

- if $(x,y) = (x^*,y^*) \rightarrow (x^*,y^*) = g_{(x^*,y^*,0)} + g_{(x^*,y^*,m)} \begin{cases} g_{(x^*,y^*,0)} = (1-\lambda)\mu_{(x^*,y^*)} \\ g_{(x^*,y^*,m)} = \lambda\ \mu_{(x^*,y^*)} \end{cases}$ for $\lambda \in [0,1]$

  where $\lambda$ is the proportion of cells with copy-number state $(x,y)$ that have the mutation.

- if $(x,y) \neq (x^*,y^*)$
  $\Big\downarrow$ under the single-split copy number assumption $(x',y',m) \in \Gamma^*$
  $(x,y) = (x',y') \xrightarrow[Eq.3.22]{satisfy} g_{(x,y,m)} = \mu_{(x,y)}$

Therefore,

**Lemma 1** Given $v$, $\mu$, $\Gamma^*$, if $\exists\ \mathbf{g} \mid Con(\{\Gamma^*, \mathbf{g}, v, \boldsymbol{\mu}\})$, $\mathbf{g}$ are "uniquely determined as":

$$g(x,y,m) = \begin{cases} \mu(x,y), & if (x,y) \neq (x^*,y^*), \\ (1-\lambda)\mu(x^\star,y^*), & if (x,y) = (x^*,y^*) \quad and \quad m = 0, \\ \lambda\mu(x^*,y^*), & if (x,y) = (x^*,y^*) \quad and \quad m = m^*, \end{cases} \tag{3.24}$$

Substituting in equation 3.23

$$v = \frac{1}{F}\left[0 \cdot (1-\lambda)\mu_{(x^*,y^*)} + m^* \cdot \lambda\mu_{(x^*,y^*)} + \sum_{\substack{(x,y,m)\in\Gamma \\ (x,y)\neq(x^*,y^*)}} m \cdot \mu_{(x,y)}\right] \tag{3.25}$$

and solving for $\lambda$

$$\lambda = \frac{1}{m\mu_{(x^*,y^*)}}\left[vF - \sum_{\substack{(x,y,m)\in\Gamma \\ (x,y)\neq(x^*,y^*)}} m \cdot \mu_{(x,y)}\right] \tag{3.26}$$

**Lemma 2** Given $v, \boldsymbol{\mu}, \Gamma^*$,

$$\exists\ \mathbf{g} \mid Con(\{\Gamma^*, \mathbf{g}, v, \boldsymbol{\mu}\}) \iff \begin{cases} \forall (x,y) \text{ such that } \mu_{(x,y)} > 0, \exists \text{ state } (x,m,y) \in \Gamma \text{ for some } m \\ \lambda \in [0,1] \end{cases} \tag{3.27}$$

**Lemma 3** If $v \in [v^-, v^+]$, $v$ is possible for $\boldsymbol{\mu}$ and a $\Gamma^*$ being $Con(\{\Gamma^*, \boldsymbol{\mu}\})$.

From equation 3.25 being $Con(\{\mathbf{g}, \boldsymbol{\mu}\})$. Since $g_{(x^*,y^*,m^*)} = \lambda\mu_{(x^*,y^*)}$ and $\lambda \in [0,1]$; when $\lambda = 0$, $\mathbf{g}$ is minimized, $g_{(x^*,y^*,m^*)} = 0$.

$$v^- = \min_{\lambda} \frac{1}{F} \left[ 0 \cdot (1-\lambda)\mu_{(x^*,y^*)} + m^* \cdot \lambda\mu_{(x^*,y^*)} + \sum_{\substack{(x,y,m)\in\Gamma \\ (x,y)\neq(x^*,y^*)}} m \cdot \mu_{(x,y)} \right] \tag{3.28}$$

Thus,

$$v^- = \frac{1}{F} \left[ \sum_{\substack{(x,y,m)\in\Gamma^* \\ (x,y)\neq(x^*,y^*)}} m \cdot \mu_{(x,y)} \right] \tag{3.29}$$

And when $\lambda = 1$, $\mathbf{g}$ is maximized, $g_{(x^*,y^*,m^*)} = 1$.

$$v^+ = \frac{1}{F} \left[ m^*\mu_{(x^*,y^*)} + \sum_{\substack{(x,y,m)\in\Gamma^* \\ (x,y)\neq(x^*,y^*)}} m \cdot \mu_{(x,y)} \right] = v^- + \frac{m^*\mu_{(x^*,y^*)}}{F} \tag{3.30}$$

**Lemma 4** Given $\boldsymbol{\mu}$, $\Gamma$ and $\rho$, the CCFs c resulting from $\mathbf{g}$ such that $Con(\{\Gamma, \mathbf{g}, \boldsymbol{\mu}\})$ are "uniquely determined as":

$$c = \frac{1}{\rho} \left[ \lambda \cdot \mu_{(x^*,y^*)} + \sum_{\substack{(x,y,m)\in\Gamma_{CCF} \\ (x,y)\neq(x^*,y^*)}} \mu_{(x,y)} \right] \tag{3.31}$$

Deduced from equation 3.21 having into account that $\Gamma_{CCF}$ refers to the genotype set $\Gamma$ where $m \geq 1$, and substituting the $g_{(x,y,m)}$ by the split state $\mu_{(x^*,y^*)}$ from 3.24.

**Theorem 1** : Given $v$, $\rho$, $\boldsymbol{\mu}$, and $\Gamma^*$ being $Con(\{\Gamma^*, v, \boldsymbol{\mu}\})$, "the CCF c is uniquely determined by"

$$c = \frac{1}{\rho m^*} \left[ vF - \sum_{\substack{(x,y,m)\in\Gamma_{CCF} \\ (x,y)\neq(x^*,y^*)}} (m + m^*) \cdot \mu_{(x,y)} \right] \tag{3.32}$$

where $\Gamma_{CCF} = (x,y,m) \in \Gamma | m \geq 1 \subseteq \Gamma$.

This expression is achieved by solving for $\lambda \cdot \mu_{(x^*,y^*)}$ from equation 3.25

$$\lambda \cdot \mu_{(x^*,y^*)} = \frac{Fv}{m^*} - \frac{1}{m^*} \sum_{\substack{(x,y,m)\in\Gamma \\ (x,y)\neq(x^*,y^*)}} m \cdot \mu_{(x,y)} \tag{3.33}$$

Substituting $\lambda \cdot \mu_{(x^*,y^*)}$ in 3.21

$$c = \frac{1}{\rho}\left[\frac{Fv}{m^*} - \frac{1}{m^*}\sum_{\substack{(x,y,m)\in\Gamma \\ (x,y)\neq(x^*,y^*)}} m\cdot\mu_{(x,y)} + \sum_{\substack{(x,y,m)\in\Gamma_{CCF} \\ (x,y)\neq(x^*,y^*)}} \mu_{(x,y)}\right] \tag{3.34}$$

For $m \geq 1, (x,y,m) \in \Gamma_{CCF}$

$$\sum_{\substack{(x,y,m)\in\Gamma \\ (x,y)\neq(x^*,y^*)}} m\cdot\mu_{(x,y)} = \sum_{\substack{(x,y,m)\in\Gamma_{CCF} \\ (x,y)\neq(x^*,y^*)}} m\cdot\mu_{(x,y)} \tag{3.35}$$

$$c = \frac{1}{\rho}\left[\frac{Fv}{m^*} - \frac{1}{m^*}\sum_{\substack{(x,y,m)\in\Gamma_{CCF} \\ (x,y)\neq(x^*,y^*)}} m\cdot\mu_{(x,y)} + \sum_{\substack{(x,y,m)\in\Gamma_{CCF} \\ (x,y)\neq(x^*,y^*)}} \mu_{(x,y)}\right] \tag{3.36}$$

***Probabilistic model for CCF***

Satas, G., et al., 2021 also present a probabilistic model that includes the uncertainty due to sequencing errors and coverage in the VAF estimation in the DeCiFer algorithm.

The probabilistic model for CCF for an individual SNV calculates the posterior probability $Pr(c|a,t,\boldsymbol{\mu},\Gamma)$ of the CCF considering the observed data and a single split copy number set. Solving VAF from equation 3.14 and using a change of variable, CCF by VAF, it can be derived the "probability distribution Pr(c) for the CCF from any probability distribution Pr(v) on the VAF" as

$$Pr(c|a,t,\rho,\boldsymbol{\mu},\Gamma) = \frac{\rho m^*}{F} Pr(V(c)|a,t,\boldsymbol{\mu},\Gamma) \tag{3.37}$$

where V(c) is obtained solving the equation 3.32 for $v$

$$V(c) = \frac{c\rho m^*}{F} + \frac{1}{F}\sum_{\substack{(x,y,m)\in\Gamma_{CCF} \\ (x,y)\neq(x^*,y^*)}} (m+m^*)\cdot\mu_{(x,y)} \tag{3.38}$$

To resolve $Pr(c|a,t,\boldsymbol{\mu},\Gamma)$ the authors apply the Bayes´ Theorem

$$Pr(V(c)|a,t,\boldsymbol{\mu},\Gamma) \propto Pr(V(c)|t,\boldsymbol{\mu},\Gamma) Pr(a|V(c),t,\boldsymbol{\mu},\Gamma) \tag{3.39}$$

Considering equation 3.22 and 3.23, the authors assume that given $\mu$ and $\Gamma$ VAF is conditionally independent of the total read count $t$, while, given VAF $V(c)$ and $t$, the variant read count $a$ is conditionally independent of $\mu$ and $\Gamma$ (is the $\hat{v} = a/t$) which gives the posterior probability for VAF V(c)

$$Pr(V(c)|a,t,\boldsymbol{\mu},\Gamma) \propto Pr(V(c)|\boldsymbol{\mu},\Gamma) Pr(a|V(c),t) \tag{3.40}$$

For a given VAF value the likelihood of the observed variant read counts is $Pr(a|V(c),t)$ while the prior probability of the VAF given copy-number proportion and a genotype set is $Pr(V(c)|\boldsymbol{\mu},\Gamma)$

The prior $Pr(V(c)|\boldsymbol{\mu},\Gamma)$ has support only on the range $[v^-, v^+]$

For the likelihood DeCiFer uses a binomial or beta-binomial distribution and a uniform prior for the "feasible range" of VAFs, $Pr(v(c)|\boldsymbol{\mu},\Gamma) \propto 1_{V(c)\in[v^-,v^+]}$

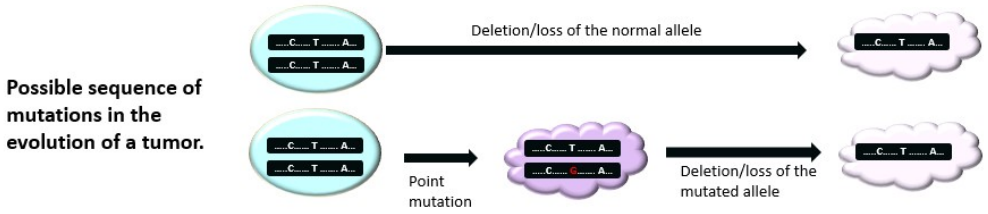
"The following posterior distribution over the CCF c"

$$Pr(V(c)|a,t,\boldsymbol{\mu},\Gamma) = \frac{1}{Z} 1_{V(c)\in[v^-,v^+]} B(a|V(c),t,...) \tag{3.41}$$

being B the binomial or beta-binomial distribution and Z a normalization constant.

### Descendant cell fraction

The statistic DCF consider potential SNV deletion, therefore, generalizes the CCF statistic. "The DCF of a mutation is the proportion of cells in a sample that are descendants of the ancestral cell where the mutation was first introduced." 3.1.4



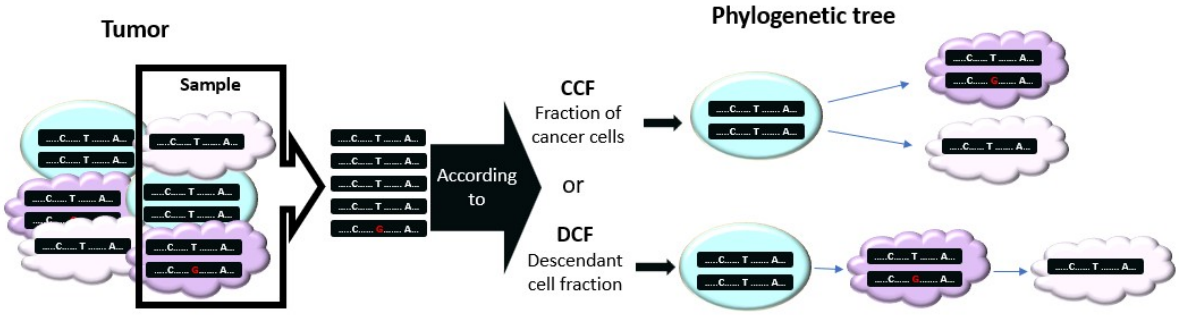




Figure 3.5: Representation of the difference between CCF and DCF. A tumor sample is illustrated (normal cells are rounded and colored in blue, and the tumor cells are colored in a range of violet). Each cell contains its DNA chain with the mutations indicated in red.

The authors formally define DCF using a term coined as ***"genotype tree"*** $T_\Gamma = (\Gamma, E)$, whose roots are the genotype set $\Gamma$ and the branches $E$ indicates the relations between the genotypes. A "genotype tree" describes the evolutionary history of a single SNV.

Since the DCF synopsizes a genotype tree $T_\Gamma$ and genotype proportions $\mathbf{g}$, a genotype tree is assigned to each SNV considering the parsimony contraint imposed by the possible number of distict DCF values.

Analogous to the CCF (equation 3.21), the DCF, $d$, of an SNV is defined as,

$$DCF = \frac{1}{\rho} \sum_{(x,y,m)\in\Gamma_{(DFC)}} g_{(x,y,m)} \tag{3.42}$$

where $\Gamma_{(DCF)} \subseteq \Gamma$ indicates the genotypes that descent from the genotype $(x^*, y^*, m^*)$.

In an analogous way to the theorem 1, the following theorem is defined.

**Theorem 2** : Given $v$, $\rho$, $\boldsymbol{\mu}$, and $\Gamma^*$ being $Con(\{\Gamma^*, v, \boldsymbol{\mu}\})$, "the DCF c is uniquely determined by"

$$DCF = \frac{1}{\rho m^*} \left[ vF - \sum_{\substack{(x,y,m)\in\Gamma_{DCF} \\ (x,y)\neq(x^*,y^*)}} (m + m^*) \cdot \mu_{(x,y)} \right] \tag{3.43}$$

where $\Gamma_{DCF}$ is the set of genotypes in genotype tree $T_{\Gamma^*}$ that are descendants of the state $(x^*, y^*, m^*)$.

***Probabilistic model for DCF***

The probabilistic model for DCF is analogue to the one for CCF.

where V(d) is obtained solving the equation 3.43

$$V(d) = \frac{c\rho m^*}{F} + \frac{1}{F} \sum_{\substack{(x,y,m)\in\Gamma_{CCF} \\ (x,y)\neq(x^*,y^*)}} (m + m^*) \cdot \mu_{(x,y)} \tag{3.44}$$

In this case V(d) also depends on $T_\Gamma$, therefore

$$Pr(d|a, t, \rho, \boldsymbol{\mu}, \Gamma, T_\Gamma) = \frac{\rho m^*}{F} Pr(V(d)|a, t, \boldsymbol{\mu}, \Gamma) \tag{3.45}$$

and the posterior distribution over the DCF c for an individual SNV in one sample

$$Pr(d|a, t, \boldsymbol{\mu}, \Gamma, T_\Gamma) = \frac{1}{Z} 1_{V(d)\in[v^-,v^+]} B(a|V(d), t, ...) \tag{3.46}$$

In the next graphs (Figure 3.1.4 and Figure 3.1.4) are shown a summary of the previously explained statistics in a situation when all the parameters are known.

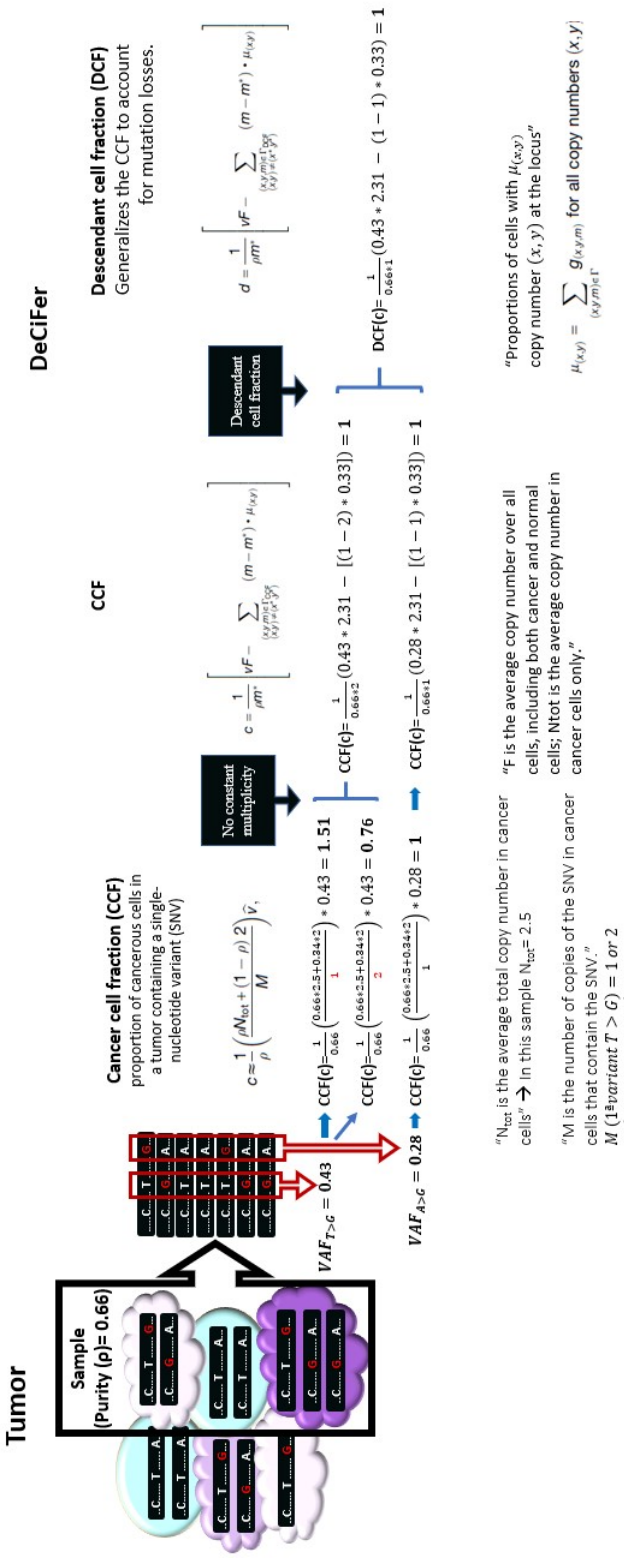



Figure 3.6: Summary of the calculations performed to infer the different statistics, CCF under the assumption of constant multiplicity, and CCF under the assumption of non-constant multiplicity and DCF implemented in the DeCiFer algorithm.

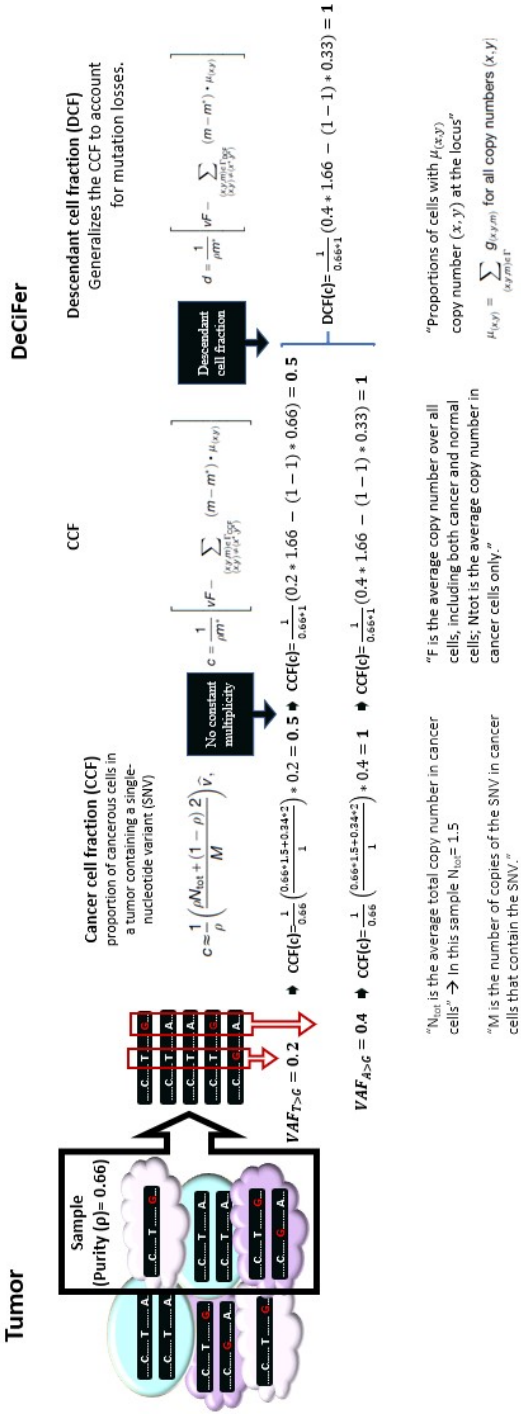



Figure 3.7: Summary of the calculations performed to infer the different statistics, CCF under the assumption of constant multiplicity, and CCF under the assumption of non-constant multiplicity and DCF implemented in the DeCiFer algorithm.

## 3.2 Clustering

The next step is clustering mutations with similar CCFs.

### 3.2.1 SNV-based subclonal reconstruction - A Bayesian Dirichlet process

Nik-Zainal et al. used a ***Bayesian Dirichlet process*** to simultaneously infer the "unknown fraction of tumor cells" in each "unknown number of subclones" that harbor an "unknown proportion of all somatic mutations".

SNV-based subclonal reconstruction is based on clustering the SNVs with similar CCF or CP. However, VAF reckoning, and therefore the CCP or CP, is affected by methodological variables such as sequencing depth (i.e., the number of reads in each region), since precision of inference increases with the number of readings. Although the initial depth is a chosen parameter, the distribution of sequences throughout the genome depends on the physical-chemical properties of the region in question, there are regions of the genome that are more difficult to sequence than others, therefore, a different distribution of values can be observed in the CCF estimates from different mutations, for a given clone or subclone. This variability, namely, the observed variability in CCF estimation from a tumor can be modeled by a binomial distribution (Dentro, S.C.et al., 2017, Nik-Zainal, S., et al., 2012):

$$r_i \sim Bin(r_{tot,i},\ p_i) \tag{3.47}$$

being $r_i$ the nº of reads with the variant allele in the i location, $r_{tot,i}$ the total depth reads in the i location (known from data) and $p_i$ the probability of observe the variant allele, that could be define as:

$$p_i = \zeta_i \pi_i \tag{3.48}$$

where $\zeta_i$ is the proportion of expected reads if mutation is clonal (since is unknown if it is clonal or subclonal, probability is maximized) given purity and copy number for that locus, and $\pi_i \in (0,\ 1)$ the true fraction of tumor cells that harbor the mutation.

The inference of $\pi_i$, this is, the CCF, is the base of subclonal reconstruction. There are different methods to this endeavor, but since there is not information about the distribution of $\pi_i$, one of the most prevalent is to use a Dirichlet process (DP) as a prior distribution that models subclonal fractions,

$$\pi_i \sim DP(\alpha P_0) \tag{3.49}$$

being $DP(\alpha P_0)$ a DP with a base probability distribution $P_0$ and a dispersion parameter $\alpha > 0$. (Dentro, S.C., et al., 2017, Nik-Zainal, S., et al., 2012, Dunson, D., 2010)

The ***Dirichlet distribution*** can represents a sampling of an unknown number of distributions, which allows to coestimate both the nº of distributions (nº of subclones or clusters), $h$, and their properties (the CCF and the somatic mutation they harbor), being $P_0$ the observed sampling, it can be used to infer each cluster, $h$, and its probability weight $\omega_h$, through the stick-breaking representation, where the true probability distribution P is discrete, and its probability mass function is formulated as the sum of the product of the probability weight of the each mutation cluster, $\omega_h$, and, $\delta_{\pi_h}$ represents a point mass (indicator function) of the location in the CCF space, $\pi$, while the locations $\pi_h$ are independent and identical distributed according to $P_0$, and $\delta_{\pi_h}$ value is equal to 0 for all locations except $\delta_{\pi_h}(\pi_h) = 1$ (Dentro, S.C., et al., 2017, Nik-Zainal, S., et al., 2012):

$$P(\pi) = \sum_{h=1}^{\infty} \omega_h \delta_{\pi_h}(\pi), \qquad \pi_h \sim P_0 \tag{3.50}$$

being

$$\omega h = V_h \prod_{l<h}(1 - V_l), \qquad V_h \sim \beta(1, \alpha) \tag{3.51}$$

where $V$ represents the proportion of the remaining stick that is broken off.

Therefore, in order to capture low frequency subclones, the weight was corrected by considering the probability of identifying a mutation in a fraction of tumor cells, this is its sensitivity, $\mathbf{S}_{\pi_h}$, whose model parameters can be estimated by bootstrap using simulated data.

$$\omega_{h.corrected} = \frac{k_h \mathbf{S}_{\pi_h}}{\sum_i k_i \mathbf{S}_{\pi_i}} \tag{3.52}$$

Once set the prior distribution for $P_0$ and $\alpha$, the posterior distribution is usually inferred using Markov chain Monte Carlo (MCMC) algorithms. The accuracy of this approach was evaluated using simulated data (Figure 3.8).

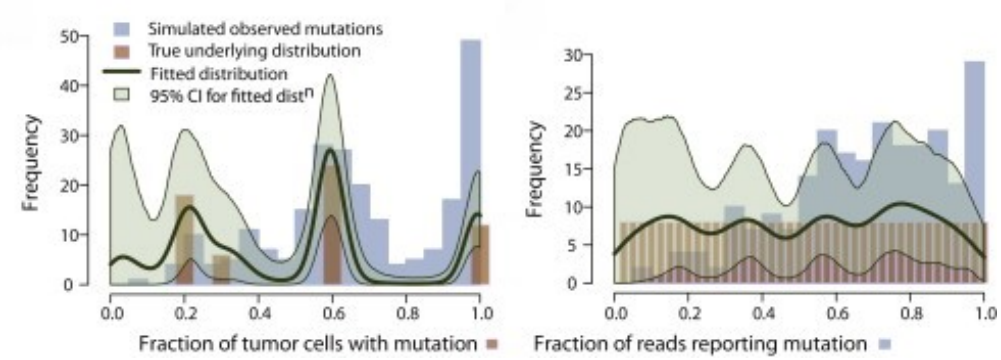



Figure 3.8: (Left) Mutations (blue histogram) from an "in silico" simulation of a tumor in which fully clonal mutations account for 20% of mutations, 40% mutations are found in a subclone representing 60% of tumor cells, 10% mutations in a subclone at 30% and 20% mutations in a subclone at 20% of tumor cells (pink bars). The simulated mutations have also been subject to correction for the sensitivity of detection at different fractions of tumor cells, hence there are fewer "observed" mutations at 20% of tumor cells than at 100% despite there being more "true" mutations at this level. Statistical modeling by a Bayesian Dirichlet process of the simulated mutations is shown as a dark green line. Also shown are the 95% posterior confidence intervals for the fitted distribution (pale green area). (Right) Mutations (blue histogram) from an "in silico" simulation of a tumor in which there are 40 subclones, evenly spread from 0%–100% of tumor cells and each contributing 2.5% of mutations (pink bars). The simulated mutations have been subject to correction for the sensitivity of detection at different fractions of tumor cells, hence there are fewer "observed" mutations at 20% of tumor cells than at 100% despite there being the same number of “true” mutations at this level. Statistical modeling by a Bayesian Dirichlet process of the simulated mutations is shown as a dark green line. Also shown are the 95% posterior confidence intervals for the fitted distribution (pale green area). Taken from Nik-Zainal et al., 2012

And then test in real data (Figure 3.9).

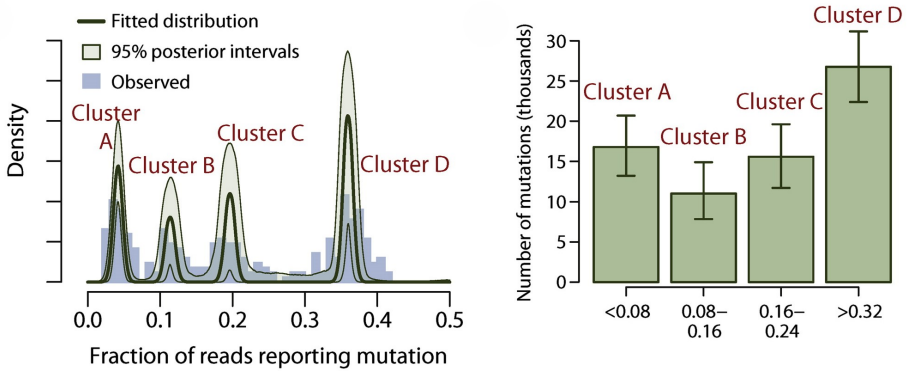



Figure 3.9: (Left) Statistical modeling of the distribution of clonal and subclonal mutations by a Bayesian Dirichlet process. The empiric histogram of mutations is shown in pale blue, with the fitted distribution as a dark green line. Also shown are the 95% posterior confidence intervals for the fitted distribution (pale green area). Four separate clusters of mutations, named A–D, are identified. (Right) Estimated number of mutations found in clusters A–D, with the error bars representing the 95% posterior confidence intervals. Taken from Nik-Zainal et al., 2012

The power to detect clonal or subclonal mutations depends on the coverage of the sequenced region, $c_s$, the purity of the sample, $\rho$, and the tumor $\psi_t$ and normal $\psi_n$ ploidy, and can be aproximate by the following metric (Dentro, S.C., et al., 2017):

$$p_s = c_s \frac{\rho}{\rho\psi_t + (1-\rho)\psi_n} \tag{3.53}$$

### 3.2.2 CNA based subclonal reconstruction. The Battenberg algorithm

Nik-Zainal et al. developed the ***Battenberg algorithm*** that tries to improve the statistical power by analyzing haplotypes, instead of individual SNPs, in order to detect subclonal population. This algorithm is based on the ASCAT algorithm.

Inferring haplotypes allows more accurate estimates of BAF, which in turn, allows subclonal populations to be detected more accurately. Once germline haplotypes are inferred, the next step is to phase a somatic mutation to a nearby heterozygous SNP to cluster mutations into the haplotypes, which is accomplished by analyzing whether there are reads that contain both and, if so, count the number of read. If a somatic mutation is affected by alteration alterations in the copy number and/or belong to a subclone, the percentage of the reads would be above or below 50%, if this is significant, the mutation will be assigned into the deleted or retained allele. (Figure 3.10).

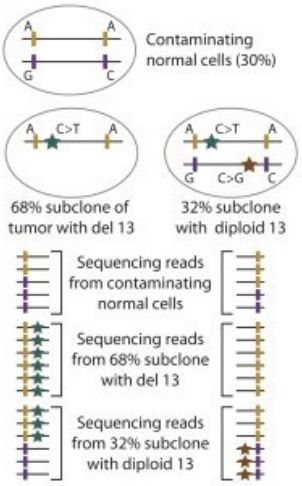



Figure 3.10: Phasing of mutations (stars) with adjacent germline heterozygous SNPs (vertical lines) allows determination of whether a mutation is on the retained or subclonally deleted parental copy of a chromosome. Taken from Nik-Zainal et al., 2012

To identify whether two close somatic mutations are mutually exclusive or are part of the same subclonal population (subclonal evolution), similarly, it is necessary to find reads that decipher the right combination of alleles (Figure 3.11). ((Dentro, S.C., et al., 2017, Nik-Zainal et al., 2012)

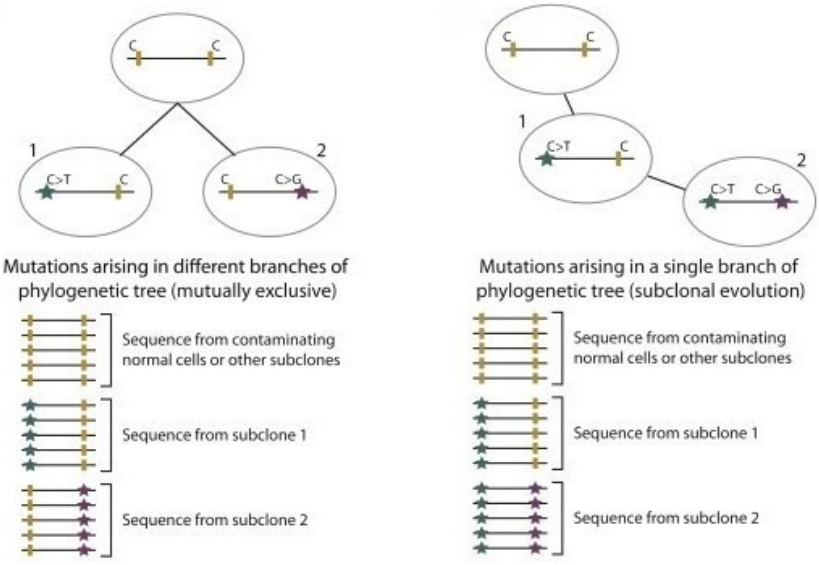



Figure 3.11: (Left) Phasing of subclonal mutations (stars) with other nearby subclonal mutations allows determination of whether they are in separate phylogenetic lineages, in which case no sequencing reads will report both variants together (mutually exclusive pair of mutations). (Rigth) Similar phasing analysis can identify cases where the later subclonal mutation has arisen on an allele linked with a previous subclonal mutation. Taken from Nik-Zainal et al., 2012

The first steps of the Battenberg algorithm are similar to the ones previously explained to infer tumor purity and ploidy and allele-specific copy number, in section 3.1.1, using the ASCAT algorithm.

Once inferred the copy number values using the germline heterozygous SNPs for both alleles in the haplotype, (equations 3.8 and 3.9), BAF values can be calculated. The criteria to decide if a fragment belongs to a tumor clon or subclone is based on a t-test, that evaluates how far the ***observed BAF*** value is from the ***expected BAF*** value if the haplotype was clonal.

The previous equations (3.8 and 3.9) can render non-integer values that must be rounded to obtain clonal copy number states used to calculate the expected BAF, $\widehat{b_h}$:

$$\widehat{b_h} = \frac{1-\rho+\rho\ n_B}{2(1-\rho)+\rho\ (n_A+n_B)} \tag{3.54}$$

where $n_A$ and $n_B$ are the integer allele-specific copy number; depending on the rounding up ($\lceil n_A \rceil$, $\lceil n_B \rceil$) or down ($\lfloor n_A \rfloor$, $\lfloor n_B \rfloor$), four possibilities can be obtained that the authors of the Battenberg algorithm presume must be within a maximum error of $\pm 1$ for most segments.

### *Clonal inference*

For clonal inference the criterion used to choose between the four values is the one that minimizes the distance of the observed BAF value $b_h$ from the expected BAF value $\widehat{b_h}$.
Once the most likely combination of alleles is determined a t-test is performed against the observed BAF to accept the haplotype as clonal if the test is not significant ($\alpha = 0.05$). If the BAF value obtained does not explain a clonal state then the hypothesis to test is if the haplotype is subclonal.

### *Subclonal inference*

If the segment is subclonal, under the assumption that during tumor evolution mutations in a specific location occur once, three types of cell populations coexist: normal cells $(1-\rho)$, tumor cells with $(\rho\tau)$ and without $(\rho(1-\tau))$ a ***subclonal mutation*** (being $\tau \in (0,\ 1)$), which determine the haplotype frequency and coverage depth/LogR data of the genomic segment.

$$h_f = \frac{1-\rho+\rho\tau n_{B,1}+\rho(1-\tau)n_{B,2}}{2(1-\rho)+\rho\tau(n_{A,1}+n_{B,1})+\rho(1-\tau)(n_{A,2}+n_{B,2})} \tag{3.55}$$

This haplotype can be contained in two or more subclones populations, therefore, the most parsimonious approach to estimate $\tau$ is to assume that subclones differ by less than 1 copy, which restrict the copy number to four combinations, although for a given $h_f$ only two of the combinations are possible (figure 3.12) that can be inferred using the total copy number $n_A+n_B$ or $l_s$ since $n_A+n_B = \frac{2\rho-2+2^{l_s}\psi}{\rho}$

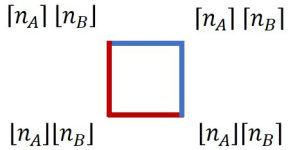



Figure 3.12: The two plausible combinations are $\lfloor n_A \rfloor + \lfloor n_B \rfloor$ and $\lfloor n_A \rfloor + \lfloor n_B \rfloor + 1$ (in red) or $\lfloor n_A \rfloor + \lfloor n_B \rfloor + 1$ and $\lfloor n_A \rfloor + \lfloor n_B \rfloor + 2$ (in blue)

For subclonal inference, the criterion is accepting the segment as subclonal if the observed value $\hat{h_f}$ is significantly different from the theoretical value $h_f$ in the four scenarios.

Therefore, if a subclone $\tau$ harbors a copy number state $n_{A,1}+n_{B,1}$ and another subclone $1-\tau$ harbors $n_{A,2}+n_{B,2}$, $\tau$ can be inferred as

$$\tau = \frac{1-\rho+\rho n_{B,2}-2h_f(1-\rho)-h_f\rho(n_{A,2}+n_{B,2})}{h_f\rho(n_{A,1}+n_{B,1})-h_f\rho(n_{A,2}+n_{B,2})-\rho n_{B,1}+\rho n_{B,2}} \tag{3.56}$$

where standard deviation and intervals of confidence can be calculated by bootstrapping. (Dentro, S.C., et al., 2017)

### 3.2.3 DeCiFer: simultaneous clustering and genotype selection using the DCF

To perform clustering of the DCFs it is necessary to infer the pair $(\Gamma^*, T_{\Gamma^*})$ values, but individual SNVs cannot provide this type of information and multiple values can fit the observed data. An approach to this problem is analyze the SNVs together, which allows, assuming a limited number of possible DCF values, to obtain constraints to reduce ambiguity by making the following assumption:

***"There exist DCF values $d_1, ..., d_k$ such that for every SNV in a tumor sample at least one $d_j$ is a valid DCF for the SNV."*** (equation 3.43)

Although this assumption allows a partition of the SNVs into k groups according to their DCF, the difficulty lies in simultaneously select the genotype tree for each SNV and clustering them since SNVs may have more that one possible DCF value, which leads to set the following problem:

#### Probabilistic mutation clustering and genotype selection problem

The simultaneous selection and clustering problem in a general setting with $p$ bulk sequencing samples from the same patient, being $\boldsymbol{a}_i = [a_{i,\ell}]_{\ell=i,...,p}$ and $\boldsymbol{t}_i = [t_{i,\ell}]_{\ell=i,...,p}$ variant and total read counts, respectively, $M_i = [\boldsymbol{\mu}_{i,\ell}]_{\ell=i,...,p}$ copy-number proportions for each SNV at the location i of each sample $\ell$, $\mathcal{G}_i$ the pair of $(\Gamma^*, T_{\Gamma^*})$ that are consistent with $\boldsymbol{\mu}$ , $\boldsymbol{s} = s_i \in 1, ...|\mathcal{G}_i|$, the set of $\mathcal{G}_i$, and $\boldsymbol{z} = z_i \in 1, ...k$ the cluster assignment for the SNV i, is depicted by:

For each SNV at location i, find $D$, a set of DCF values of size $k$ (an integer $>0$), such as $D^* = \boldsymbol{d_1, ...d_k}$ for each SNV i, and select $s_i^*$, that determines the DCF $d_i$ and $z_i^*$, so that the probability distribution of the DCF of cluster $j$ in sample $\ell$, $d^*_{z_i^*,\ell}$, knowing the variant, $\boldsymbol{a_i}$, and total, $\boldsymbol{t_i}$, read counts, copy-number proportions $\boldsymbol{\mu}_i$ and a set $\mathcal{G}_i$ of pairs of genotype sets and trees $(\Gamma_i^*, T_i)$, is maximized.

$$D^*, \boldsymbol{s}^*, \boldsymbol{z}^* = \underset{D,\boldsymbol{s},\boldsymbol{z}}{argmax} \prod_{i=1}^{n} \prod_{\ell=1}^{p} Pr(d^*_{z_i^*,\ell} | a_{i,\ell}, t_{i,\ell}, \boldsymbol{\mu}_{i,\ell}, \Gamma_{i,\ell}, T_{i,s_\ell^*}) \tag{3.57}$$

The product is calculated since the variant read counts are conditionally independent for a given $z$ and DCF.

The ***Mutation Clustering and Genotype Selection*** problem is an instance of the ***Hitting Set*** problem, which is known to be equivalent to the ***Set Cover*** problem and is NP-complete."

The DeCiFer algorithm is intended to solve this maximization.

### 3.2.4 The DeCiFer algorithm

Constraints imposed by DeCiFer are stronger than by single-split copy-number assumption since genotype trees $T_\Gamma$ must obey the follow evolutionary model

1. Each mutation occurs once, but subsequent copy-number alterations may result in its loss or amplification.
2. For a SNV locus, each specific copy number of the allele $(x, y)$ is reached exactly once.

   Therefore the ***Dollo´s law of irreversibility*** applies for SNVs and refers to the fact that a mutation produced will not revert to the previous state, and the infinite alleles assumption applies for copy-number alterations, that specifies that there is countless number of states for a locus. (Mallory, X.F., et al., 2020)
3. Mutation multiplicity is due to a change in copy-number alteration.

Under these constraints, the mutation multiplicity for the split copy number $(x^*, y^*)$ is $m^* = 1$

#### Model selection

Satas, G., et al., 2021 used two approaches to estimate the number of clusters $k$, as schematized below:

$$k \begin{cases} p+2 \text{ fixed clusters} \begin{cases} \text{truncal cluster} -> d_T = [\rho_\ell]_{\ell \in [1,\dots p]} \\ \text{absent cluster} -> d_0 = [0]_{\ell \in [1,\dots p]} \\ p\text{-clusters} \end{cases} \\ \text{Variable clusters} \\ \text{(model selection criterion)} -> min > p+2 \begin{cases} \text{truncal cluster} -> d_T = [\rho_\ell]_{\ell \in [1,\dots p]} \\ \text{absent cluster} -> d_0 = [0]_{\ell \in [1,\dots p]} \\ p\text{-clusters} \end{cases} \\ \text{Standard elbow method} \end{cases}$$

The first approach considers the existence of three groups, first, a cluster whose SNVs are present in all cells, therefore, fixed to the purity of the sample, second, a cluster whose mutations are not feasible in none of the fraction of cells so their posterior probability would be 0, required for the continuity of the optimization and third, a p-clusters related to the SNVs distinctive of each sample.

The second approach is select the clusters using a model-selection criterion based on the standard elbow method, the number of clusters are defined by the user and range between a minimal value of $p + 2$ clusters, where $p$ is the number of samples to the maximal set. The algorithm then computed the objective function over that predetermined number of clusters and chooses the minimal number $k$ that significantly improves the optimal value of the objective function over the previous values, this is the elbow of the function.

#### DeCiFer algorithm

The DeCiFer algorithm tries to find the values of $(D^*, \boldsymbol{s}^*, \boldsymbol{z}^*)$ that maximizes the probability of the value of the fraction of descendant cells that contain the SNVs assigned to cluster $\boldsymbol{z}^*$ in sample $\ell$, according to the "single-split copy number" assumption.

To achieve that aim, to optimize the equation 3.57, the DeCiFer algorithm uses a coordinate ascent approach to first searching for the optimal $D$ value and then search the $d_{\ell,j}$ that maximize the posterior probability. The algorithm alternatively optimizes the DCF clusters, given the possible plausible genotypic trees and the assignment of the clusters, which are basically the clustering of SNVs by their frequency, by the VAFs, and then given the DCFs, optimizes the SNV clusters and the

assignment of genotypic trees using a coordinante ascent algorithm. To start the algorithm the authors set a $k$ value and extract as many values from a symmetric Dirichlet distribution as $k$, the associated number obtained is used to initiate the algorithm $D(0)$.

But knowing the descendant cell fraction of a given SNVs, $D$, recalling that this value refers to the fraction of cells in the same phylogenetic branch, its genotype tree $s_i$ and the cluster to which this SNP is assigned $z_i$ is conditionally independent (conditional to not be in the same descendant cell fraction $D$) of the possible cluster and set of genotype assignments of other SNVs that are not in the same phylogenetic branch, so each $SNV_i$ can be optimized individually,

$$s_i^{(q)}, z_i^{(q)} | D^{(q-1)} = \underset{s \in 1,..,|\mathcal{G}_i|, z \in 1,..,k}{argmax} \prod_{\ell=1}^{p} Pr(d_{z,\ell}^{(q-1)} | a_{i,\ell}, t_{i,\ell}, \boldsymbol{\mu}_{i,\ell}, \Gamma_{i,s}, T_{i,s}) \tag{3.58}$$

by evaluating of the full range of plausible cluster and genotype set assignments. The next step is to find the optimal value for the cluster cell fraction contained in a matrix of dimensions $[0,1]^{(k*p)}$ being $k$ the number of possible clusters and $p$ the number of samples, therefore the optimal value $D^{(q)} \in [0,1]^{(k*p)}$.

In an analogous way, knowing the genotype tree $s^{(q)}$ and the cluster to which that SNP is assigned $z^{(q)}$ the DCF value $d_{\ell,j}$ in sample $\ell$ for cluster $j$ is conditionally (conditional to not be in the same genotype tree $s^{(q)}$ and same cluster $z^{(q)}$) independent of all other DCF values

Therefore, the algorithm searches its optimal value

$$d_{\ell,j}^{(q)} | s^{(q)}, z^{(q)} = \underset{s \in d \in [0,1]}{argmax} \prod_{i; z_i^{(q)} = j} Pr(d | a_{i,\ell}, t_{i,\ell}, \boldsymbol{\mu}_{i,\ell}, \Gamma_{i,s}, T_{i,s}) \tag{3.59}$$

that is accomplished by finding the minimum value in the range $[0,1]$ using Brent's algorithm. When the cluster and genotype set assignments reach their optimal value, the following iterations do not change that value, the algorithm terminates.

# Chapter 4

# Sensitivity analysis to purity changes of the DeCiFer algorithm

Subclonal reconstruction of tumors is based on the inference of various parameters from sequencing reads. The behavior of the algorithm to perturbations of these parameters is important to establish the dependence of the precision of the inferred parameters and the robustness of the results and its reliability. One of the most important parameters that influence the estimation of the CCF or DCF, as explained above, is the purity of the analyzed sample.

One of the approaches to analyse sensitivity is the one-at-a-time method. Although this method has its disadvantages since it does not take into account the possible interaction between the different factors considered in the algorithm, in the case of the present study it pinpoints the importance of considering its uncertainty in the development of clustering algorithms, since the DeCiFer algorithm includes the uncertainty derived from sequencing errors and coverage for VAF estimations in its probabilistic model for the CCF, but it does not mention any appraisal for uncertainty due to the possible inaccuracy in purity estimation.

The DeCiFer algorithm does not infer purity; this value is provided as input data. To infer the purity value of the sample analyzed different algorithms have different approaches, as previously explained, the ASCAT/Battenberg algorithm infer the optimal value of this parameter simultaneously to the ploidy inference by minimizing the distance between the expected and observed allele-specific copy numbers, while the HATCHet algorithm reckons the purity value as the sum of the proportion of all tumor clones present in the sample once this value has been previously obtained. Nevertheless, any of the methods used entails an error as it can be observed from the ASCAT profile grid (Figure 3.2) or the comparison among several algorithms that are shown in the following graphs (Figure 4) performed by Zaccaria, S. and B.J. Raphael, 2020, where, as can be seen, the relative error varies considerably between different analyzed algorithms when simulation data were used.

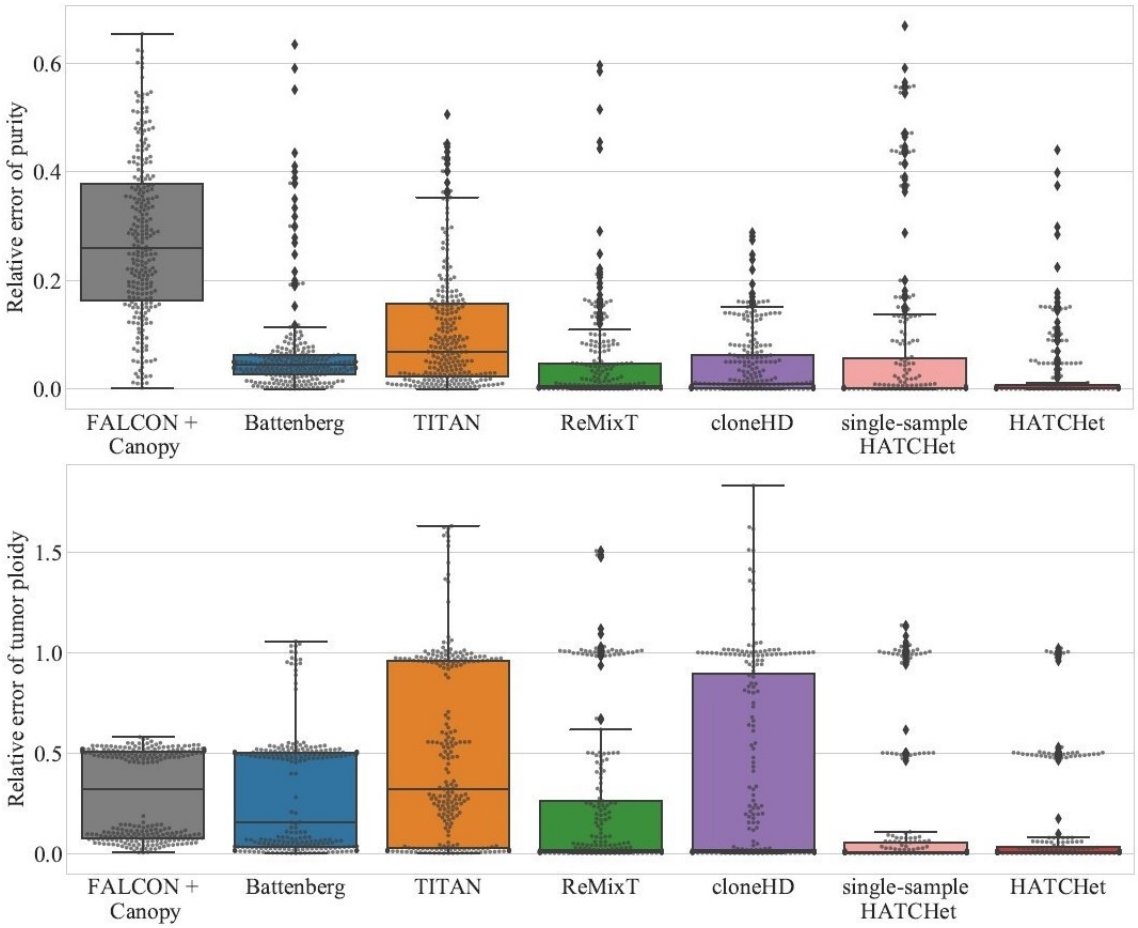



Figure 4.1: Relative error of tumor purity and ploidy. The relative error of tumor purity and ploidy is computed when running seven methods on all the 256 simulated samples, 128 without a WGD and 128 with a WGD, by considering free values of all parameters. The considered methods are five current state-of-the-art methods (Battenberg, TITAN, cloneHD, Canopy with FALCON, and ReMixT) and HATCHet, which has been applied both separately on single samples (single-sample HATCHet) and jointly on multiple samples from the same patient (HATCHet). Box plots show the median and the interquartile range (IQR), and the whiskers denote the lowest and highest values within 1.5 times the IQR from the first and third quartiles, respectively. Adapted from Zaccaria, S. and B.J. Raphael, 2020

Theoretically, following the equation 3.21 for a given purity, $\rho$, knowing the pair $(\Gamma, \mathbf{g})$, the CCF is "uniquely" calculated. But considering the variability in purity inference for given sequences, a lower purity estimate would increase the estimated CCF, while a higher purity estimate would decrease the estimated CCF, as it can be seen in the following figure 4. Analogous results are for the DCF (equation 3.43).

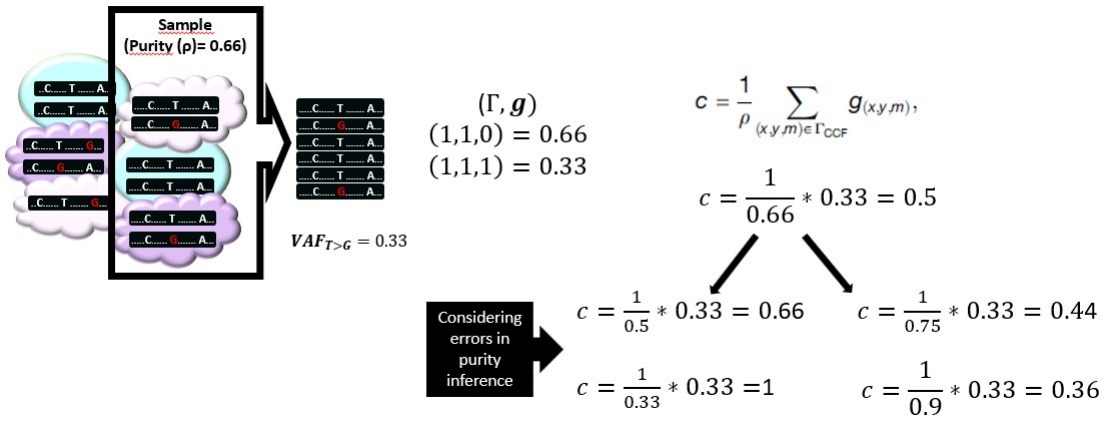



Figure 4.2:

Moreover, the DeCiFer algorithm also takes the purity values to initialize the coordinate ascent algorithm used to solve the probabilistic mutation clustering and genotype selection problem and anchor the truncal cluster to the purity value of the sample. Thus, this value determines the centroid of the main clusted infered. Therefore, the simulation performed is intended to analyze the influence of purity perturbations in the clustering of cells performed by the DeCiFer algorithm.

# Chapter 5

# Simulation and data analysis

To analyze the clustering differences when perturbations in the purity values are introduced the data from the DeCiFer demo were used (https://github.com/raphael-group/decifer), which provides the files for several prostate patients that contain the mutations and the sample´s purity.

The DeCiFer algorithm is written in Phyton and runs in linux. The basic input data, required by this algorithm to perform clustering, are two files, a .tsv with the number of the variant and reference reads, the copy number and their proportion and a .tsv with the tumor purity of each sample analyzed. As an additional input to fit beta-binomial distributions a .tsv with the read counts of the germline variants and the number of allele-specific copy number per segment is required, but these archives are not provided in the demo.

The output returns a file with the selected number of SNVs assigned to clusters and provides the inferred cluster and state tree for each mutation as well as the point estimate of the mutation DCF and the inferred CCF under the constant mutation multiplicity assumption.

The DeCiFer algorithm can be running using different input parameters established by the user. One such parameter is the VAF standard deviation value chosen to establish that an SNV belongs to a group with respect to the value of the center of the cluster. The default value is 1.5 SD.

Simulations are running introducing perturbations in purity from 1% to 50% with the following parameters:

1. Using the same parameters given in the DeCiFer algorithm, which are a fixed number of clusters between 5 and 8, a number of restarts of 20 and for deterministic reproducibility a seed of 17.

2. Modifying the default value of standard deviations set to select the SNVs assigned to a cluster.

For illustrative and comparative purposes, an attempt has been made to reproduce the data and graphs using the same patient´s data (patient 12 and patient 17) and chromosomal region indicated in the original paper, taking into account that the article uses a beta-binomial generative model but the demo runs in the default binomial model mode, and this parameter cannot be changed without the appropriate files that the demo does not provide. The original graphics shown in (Satas, G., et al., 2021) are not reproduced here due to copyright restrictions and reproduction permissions.

### *Patient 12*

As mentioned above, the DeCiFer algorithm is provided with a .tsv file with the number of variant and reference reads. The file for patient 12 contains information from 3554 SNVs. The following figure 5.1 shows the VAF data calculated from that input file with the aim of depicting the dispersion of data that the algorithm intends to cluster.

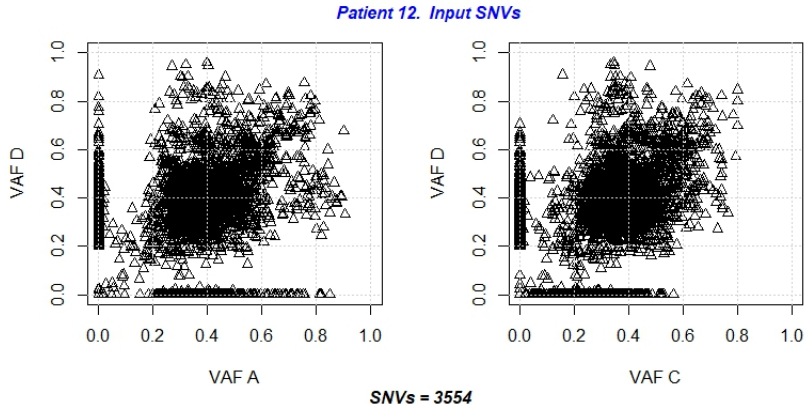



Figure 5.1: The figure depicts the VAF data dispersion, reckoned from the variant and reference readings for each of the SNVs in the file provided in the DeCifer demo of the patient numbered as 12, which is used as input data of the algorithm. The file contains 3554 SNV readings for each sample. In both graphs, the VAF values of the SNVs contained in sample D are represented on the $y$ axis, in the graph on the left, the VAF values of the SNVs contained in sample A are represented on the $x$ axis, while in the graph on the right the VAF values of the SNVs contained in sample C are represented on the $x$ axis.

Simulations performed with the DeCiFer algorithm using different values of purity and values of standard deviation of the patient 12 are exposed in subsequent graphs.

Figure 5.2 represents the results of the DeCiFer algorithm for patient number 12 resulting from running the algorithm with the initial purity, provided in the demo file, and subsequent simulations in which the algorithm is executed with files with different percentages of the initial purity values, indicated in each row of the graph. The first row of graphs attempts to reproduce Figure 6 from the study by Satas, G., et al., 2021, in which data from patient 12 on chromosome 6q are selected and grouped into the main cluster, considering that the article uses the beta-binomial model instead of the default binomial model used in this analysis. As shown in the graph, a 1% reduction in the input purity data does not alter the results but reductions equal or greater than 1.5% have a radical impact causing the disappearance of the groups of SNVs with the higher VAF values. As consequence, the inference of CCF and DCF changes drastically. When the input purity data is the initial one, two cluster of CCF are observed that correspond to the two clusters of VAF data, with different values for each sample, while by clustering both groups together (as the algorithm infers that both groups of SNVs belong to the same phylogenetic branch) the inference of the DCF value is equal to 1 for both groups in the three samples.

The effect on the CCF and DCF estimators of a decrease in the input purity value greater than 1.5%, which causes the group with the highest VAF values to be lost, is that both estimators have similar results. In comparison with the previous result, in the case of the CCF only the cluster with the lower value remains, but in the case of the DCF, the information that corresponds to the evolution of the cells is lost, so it is inferred that this group of SNVs belong to a subclone instead of a clone, just as the CCF estimator does.

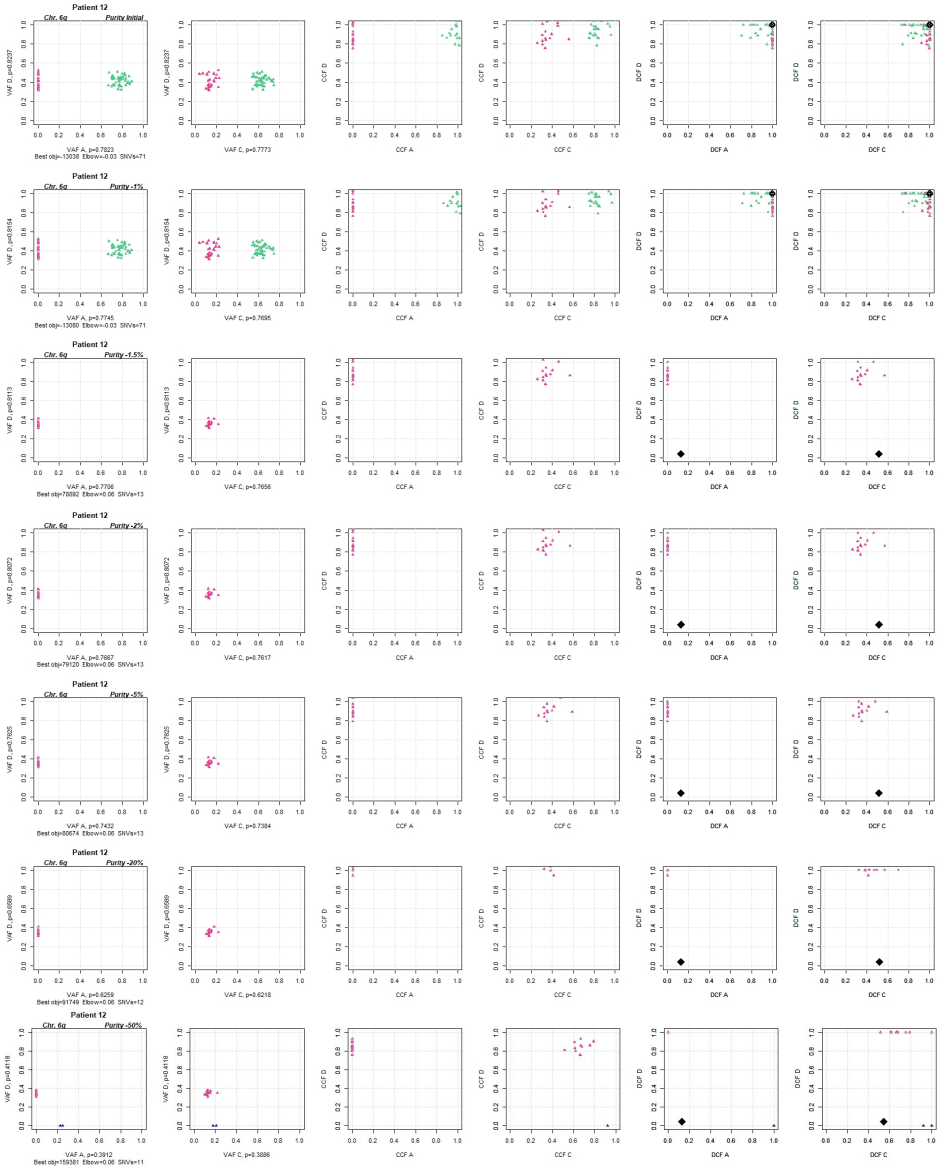


Figure 5.2: The graphs show the results of simulations of the DeCiFer algorithm when different purity values are used as input data. For illustrative and comparative purposes, an attempt has been made to reproduce figure 6 from the study by Satas, G., et al., 2021, in which data from patient 12 on chromosome 6q are selected in the main group (as noted in the main text the article uses the beta-binomial model instead of the default binomial model used in this analysis). Similar to the original figure, the first row of the graphs shows the results obtained using the purity values provided by the demo (named "initials"). The first two graphs represents the VAF values of the SNVs selected for clustering, the next two show the CCF calculated under the assumption of constant multiplicity and the last ones show the results of the DCF estimator, implemented by the DeCiFer algorithm. The following rows, with decreasing percentages of the purity values with respect to the initial one, show the SNVs that remain compared to those selected when the input data is the initial purity, for the same estimators. Additionally, the axes indicate the purity values for each sample, as well as the best objective and elbow values returned by the algorithm, and the number of SNVs selected for the clusters with the default standard deviation of 1.5. Data colored in pink and green belong to the same cluster and only differs by their VAF value. Data colored blue indicates additional values included in the main group when modifying input purity values that were not selected when running the algorithm with the initial purity value.

The following tables (Figure 5.3) shows the SNVs selected for clustering when the input purity values are -5%, -20% and -50% in comparison to the initial value, respectively. In the last two tables it is seen that the SNVs with the highest VAF value are not selected when the purity value decreases from the previous value.

Patient 12. Purity -5%

| | Chr | Pos | Ref | Var | VAR_D | VAR_C | VAR_A | TOT_D | TOT_C | TOT_A | VAF_D | VAF_C | VAF_A | cluster |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 6 | 78584733 | G | T | 32 | 8 | 0 | 77 | 67 | 57 | 0.4155844 | 0.1194030 | 0 | 5 |
| 2 | 6 | 85453204 | C | A | 33 | 12 | 0 | 81 | 68 | 63 | 0.4074074 | 0.1764706 | 0 | 5 |
| 3 | 6 | 157372830 | A | G | 26 | 10 | 0 | 68 | 77 | 59 | 0.3823529 | 0.1298701 | 0 | 5 |
| 4 | 6 | 166120335 | A | T | 23 | 12 | 0 | 62 | 78 | 71 | 0.3709677 | 0.1538462 | 0 | 5 |
| 5 | 6 | 87405219 | T | G | 25 | 9 | 0 | 68 | 76 | 46 | 0.3676471 | 0.1184211 | 0 | 5 |
| 6 | 6 | 124060534 | A | T | 23 | 11 | 0 | 65 | 75 | 60 | 0.3538462 | 0.1466667 | 0 | 5 |
| 7 | 6 | 144291198 | A | G | 24 | 10 | 0 | 68 | 77 | 52 | 0.3529412 | 0.1298701 | 0 | 5 |
| 8 | 6 | 120272872 | A | G | 22 | 15 | 0 | 63 | 69 | 49 | 0.3492063 | 0.2173913 | 0 | 5 |
| 9 | 6 | 82385222 | C | T | 24 | 10 | 0 | 69 | 76 | 64 | 0.3478261 | 0.1315789 | 0 | 5 |
| 10 | 6 | 70386109 | C | A | 26 | 8 | 0 | 76 | 67 | 69 | 0.3421053 | 0.1194030 | 0 | 5 |
| 11 | 6 | 68871922 | T | A | 21 | 7 | 0 | 63 | 70 | 48 | 0.3333333 | 0.1000000 | 0 | 5 |
| 12 | 6 | 110421314 | T | C | 21 | 9 | 0 | 64 | 75 | 77 | 0.3281250 | 0.1200000 | 0 | 5 |
| 13 | 6 | 79872713 | T | G | 19 | 8 | 0 | 61 | 62 | 55 | 0.3114754 | 0.1290323 | 0 | 5 |

Patient 12. Purity -20%

| | Chr | Pos | Ref | Var | VAR_D | VAR_C | VAR_A | TOT_D | TOT_C | TOT_A | VAF_D | VAF_C | VAF_A | cluster |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 6 | 85453204 | C | A | 33 | 12 | 0 | 81 | 68 | 63 | 0.4074074 | 0.1764706 | 0 | 5 |
| 2 | 6 | 157372830 | A | G | 26 | 10 | 0 | 68 | 77 | 59 | 0.3823529 | 0.1298701 | 0 | 5 |
| 3 | 6 | 166120335 | A | T | 23 | 12 | 0 | 62 | 78 | 71 | 0.3709677 | 0.1538462 | 0 | 5 |
| 4 | 6 | 87405219 | T | G | 25 | 9 | 0 | 68 | 76 | 46 | 0.3676471 | 0.1184211 | 0 | 5 |
| 5 | 6 | 124060534 | A | T | 23 | 11 | 0 | 65 | 75 | 60 | 0.3538462 | 0.1466667 | 0 | 5 |
| 6 | 6 | 144291198 | A | G | 24 | 10 | 0 | 68 | 77 | 52 | 0.3529412 | 0.1298701 | 0 | 5 |
| 7 | 6 | 120272872 | A | G | 22 | 15 | 0 | 63 | 69 | 49 | 0.3492063 | 0.2173913 | 0 | 5 |
| 8 | 6 | 82385222 | C | T | 24 | 10 | 0 | 69 | 76 | 64 | 0.3478261 | 0.1315789 | 0 | 5 |
| 9 | 6 | 70386109 | C | A | 26 | 8 | 0 | 76 | 67 | 69 | 0.3421053 | 0.1194030 | 0 | 5 |
| 10 | 6 | 68871922 | T | A | 21 | 7 | 0 | 63 | 70 | 48 | 0.3333333 | 0.1000000 | 0 | 5 |
| 11 | 6 | 110421314 | T | C | 21 | 9 | 0 | 64 | 75 | 77 | 0.3281250 | 0.1200000 | 0 | 5 |
| 12 | 6 | 79872713 | T | G | 19 | 8 | 0 | 61 | 62 | 55 | 0.3114754 | 0.1290323 | 0 | 5 |

Patient 12. Purity -50%

| | Chr | Pos | Ref | Var | VAR_D | VAR_C | VAR_A | TOT_D | TOT_C | TOT_A | VAF_D | VAF_C | VAF_A | cluster |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 6 | 157372830 | A | G | 26 | 10 | 0 | 68 | 77 | 59 | 0.3823529 | 0.1298701 | 0 | 5 |
| 2 | 6 | 166120335 | A | T | 23 | 12 | 0 | 62 | 78 | 71 | 0.3709677 | 0.1538462 | 0 | 5 |
| 3 | 6 | 87405219 | T | G | 25 | 9 | 0 | 68 | 76 | 46 | 0.3676471 | 0.1184211 | 0 | 5 |
| 4 | 6 | 124060534 | A | T | 23 | 11 | 0 | 65 | 75 | 60 | 0.3538462 | 0.1466667 | 0 | 5 |
| 5 | 6 | 144291198 | A | G | 24 | 10 | 0 | 68 | 77 | 52 | 0.3529412 | 0.1298701 | 0 | 5 |
| 6 | 6 | 120272872 | A | G | 22 | 15 | 0 | 63 | 69 | 49 | 0.3492063 | 0.2173913 | 0 | 5 |
| 7 | 6 | 82385222 | C | T | 24 | 10 | 0 | 69 | 76 | 64 | 0.3478261 | 0.1315789 | 0 | 5 |
| 8 | 6 | 70386109 | C | A | 26 | 8 | 0 | 76 | 67 | 69 | 0.3421053 | 0.1194030 | 0 | 5 |
| 9 | 6 | 68871922 | T | A | 21 | 7 | 0 | 63 | 70 | 48 | 0.3333333 | 0.1000000 | 0 | 5 |
| 10 | 6 | 110421314 | T | C | 21 | 9 | 0 | 64 | 75 | 77 | 0.3281250 | 0.1200000 | 0 | 5 |
| 11 | 6 | 79872713 | T | G | 19 | 8 | 0 | 61 | 62 | 55 | 0.3114754 | 0.1290323 | 0 | 5 |

Figure 5.3: The tables show part of the output file returned by the DeCiFer algorithm and the VAF calculated from the variant and reference readings for each SNV.

When the input purity value increases, the group of SNVs with higher VAF values remains, although the selection of SNVs differ somehow in comparison to those selected with the initial purity values. As expected, as the centroid of the clusters changes, the algorithm permute the SNVs included in the cluster until it reaches its optimal value for the given values. In blue are indicated the additional selected SNVs that are not included with the initial purity values. In this case the CCF and DCF estimates agree with theory; an increase in the inferred purity value decreases the CCF and DCF estimate (Figure 5.4).

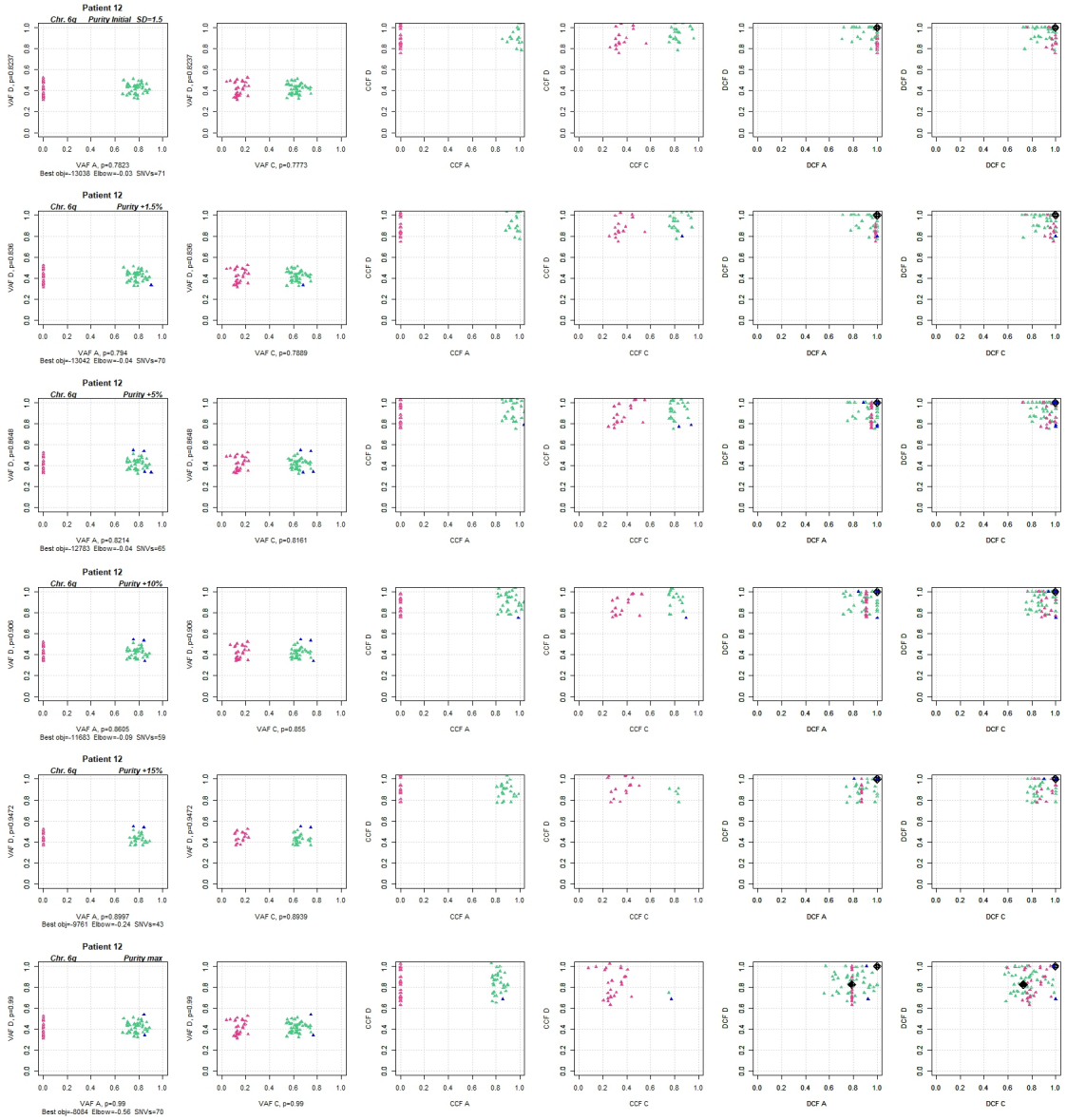



Figure 5.4: The graphs show the results of simulations of the DeCiFer algorithm when different purity values are used as input data. For illustrative and comparative purposes, an attempt has been made to reproduce figure 6 from the study by Satas, G., et al., 2021 in which data from patient 12 on chromosome 6q are selected in the main group (as noted in the main text the article uses the beta-binomial model instead of the default binomial model used in this analysis). Similar to the original figure, the first row of the graphs shows the results obtained using the purity values provided by the demo (named "initials"). The first two graphs represents the VAF values of the SNVs selected for clustering, the next two show the CCF calculated under the assumption of constant multiplicity and the last ones show the results of the DCF estimator, implemented by the DeCiFer algorithm. The following rows, with increasing percentages of the purity values with respect to the initial one, show the SNVs that remain compared to those selected when the input data is the initial purity, for the same estimators. Additionally, the axes indicate the purity values for each sample, as well as the best objective and elbow values returned by the algorithm, and the number of SNVs selected for the clusters with the default standard deviation of 1.5. Data colored in pink and green belong to the same cluster and only differs by their VAF value. Data colored blue indicates additional values included in the main group when modifying input purity values that were not selected when running the algorithm with the initial purity value.

The effect of increasing input purity values over the selection of SNVs is seen in the figure (Figure 5.5) which reflects that inferring a higher purity value has a slight impact on the SNVs clustering, difference that increases as the inferred purity value does.

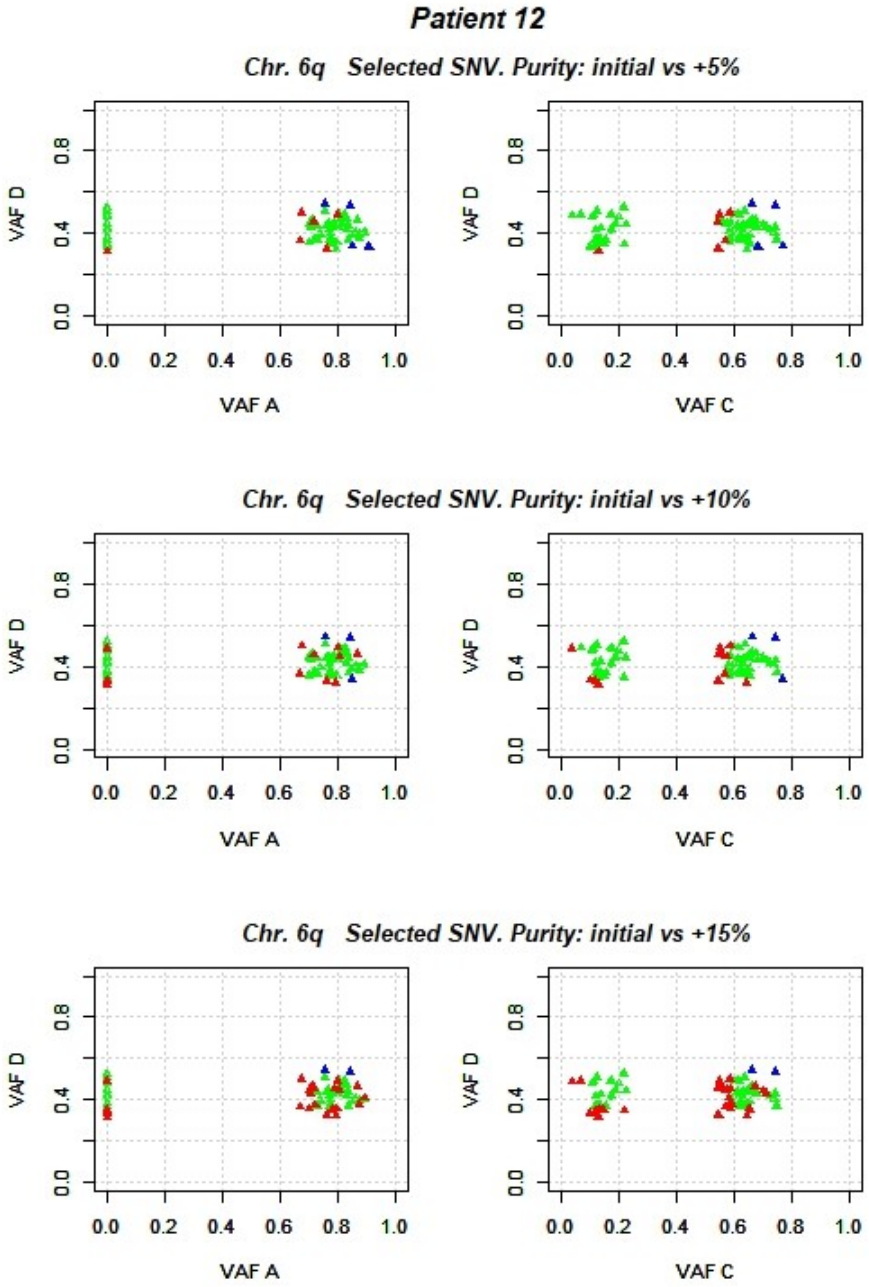



Figure 5.5: Comparison between the results of the selected SNVs when the input data is the initial purity value versus increasing proportions of 5%, 10% and 15% in the input purity values. VAF data values that are common between the results of running the algorithm with the two indicated purity values are colored green, in red are those unique to the file run with the initial purity, and blue are those unique to the file run with the other purity value indicated.

One of the main constraints over the selected SNVs are their distance to the center of the cluster. Increasing the input parameter referred to the standard deviation from a $SD = 1.5$ to $SD = 20$ causes the selection of all the SNV in that chromosomal region, and barely any differences are observed in reference to the input purity value (Figure 5.6).

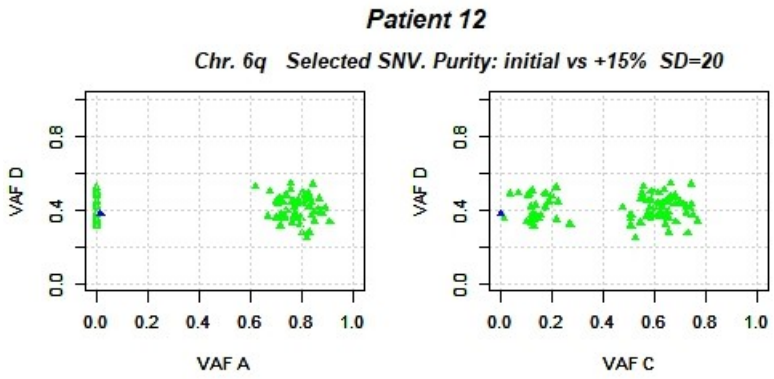



Figure 5.6: Comparison between the results of the selected SNVs, when the input data is the initial purity value versus an increment in the 15% in its inferred value, modifying the input parameter referred to the standard deviation from $SD = 1.5$ to $SD = 20$. VAF data values that are common between the results of running the algorithm with the two indicated purity values are colored green, in red are those unique to the file run with the initial purity, and blue are those unique to the file run with the other indicated purity value.

The authors states in the DeCiFer web page of the algorithm (https://github.com/raphael-group/decifer) that "This default behavior filters out noisy data or germline contamination that manifests as e.g. SNVs being assigned to the truncal cluster yet having very low DCF values in the *point estimate DCF* column of the output file."
This effect is not seen in the data; when the input standard deviation parameters are increased the included SNVs do not have distributed point estimate DCF values different from the initial one as seen in figure 5.7 however, an alteration is observed in the DCF inference when modifying this parameter when the input purity value is lower than the initial one (Figure 5.8).

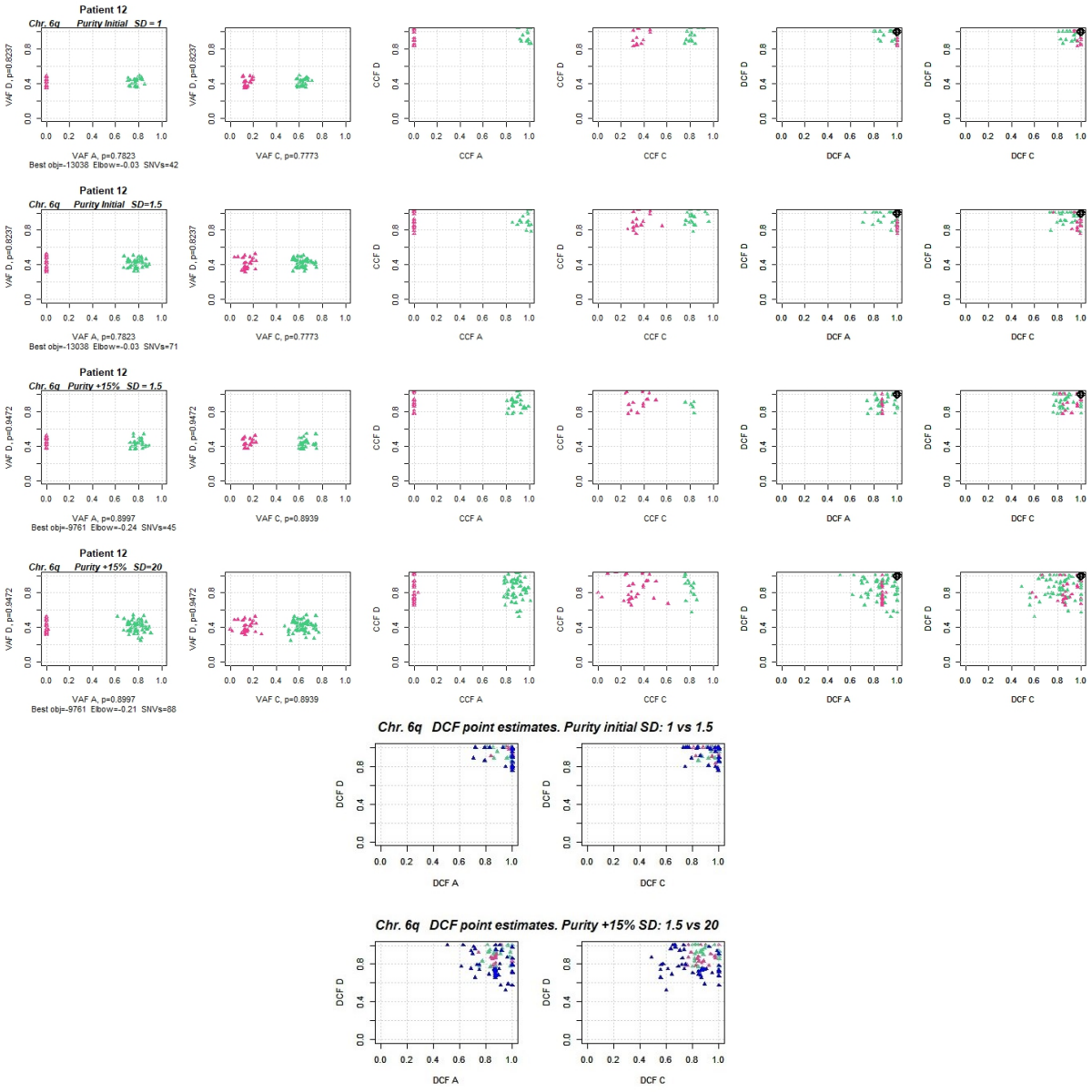



Figure 5.7: The figure show in the first two rows a comparison of the results obtained when running the algorithm with the initial values of purity but modifying the input standard deviation parameter (SD=1% and SD=1.5%), in the next two rows a comparison when the input value of purity is increased in a 15% and the input standard deviation parameters is also modified (SD=1.5% and SD=20%). The last two rows show in detail a comparison of the distribution of the point DCF values when the standard deviation is increased (blue dots).

By increasing the input standard deviation parameter, it can be observed that the SNVs that had disappeared when decreasing the input purity value, reappear. This does not affect the CCF estimate much but has a profound impact on the DCF estimate due to the inference this estimator realizes about the evolutionary history of SNVs. Unlike the previous analysis, under these circumstances a common evolution is not inferred, so the DCF estimator assigns the SNVs to different fraction of tumor cells (Figure 5.8).

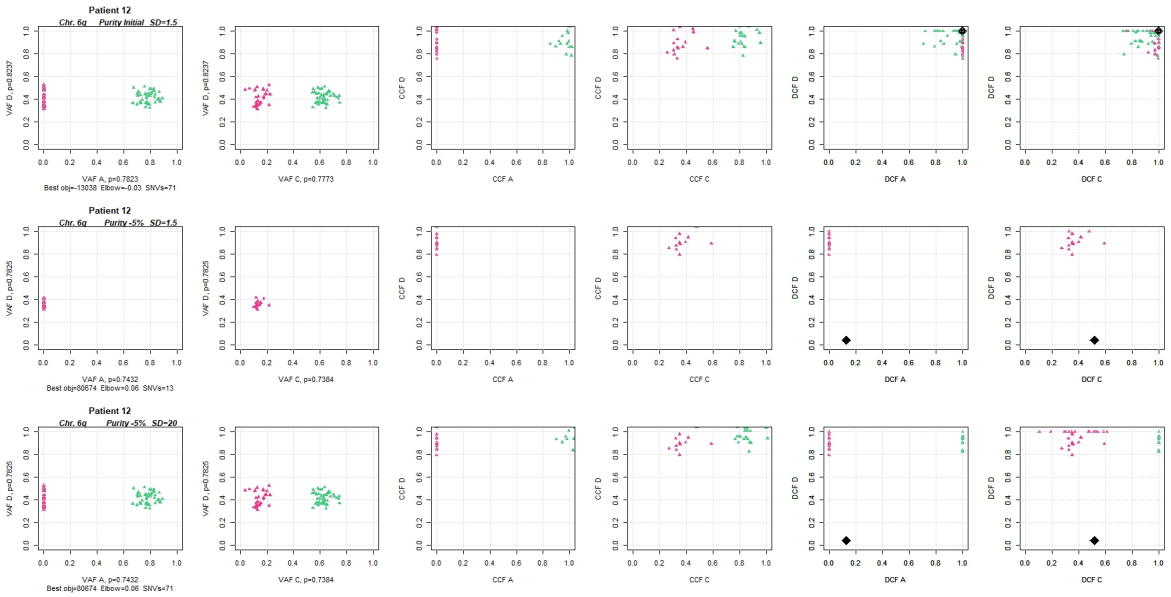



Figure 5.8: Comparison between the results of the selected SNVs when the input data is the initial purity value versus a 5% decrease in its inferred value and comparison between the results of increasing the standard deviation from $SD = 1.5$ to $SD = 20$ when the input purity data assumes a 5% decrease in its inferred value.

The previous results are also observed when all the genome is analyzed (Figure 5.9 and Figure 5.10). When input purity values decrease, there is a noticeable tendency to not select SNVs with the highest VAF values, which can denote a border effect in the algorithm due to link the truncal cluster to the purity value. The border effect observed when the input purity value decreases in not seen when the input purity value increases (Figure 5.10).

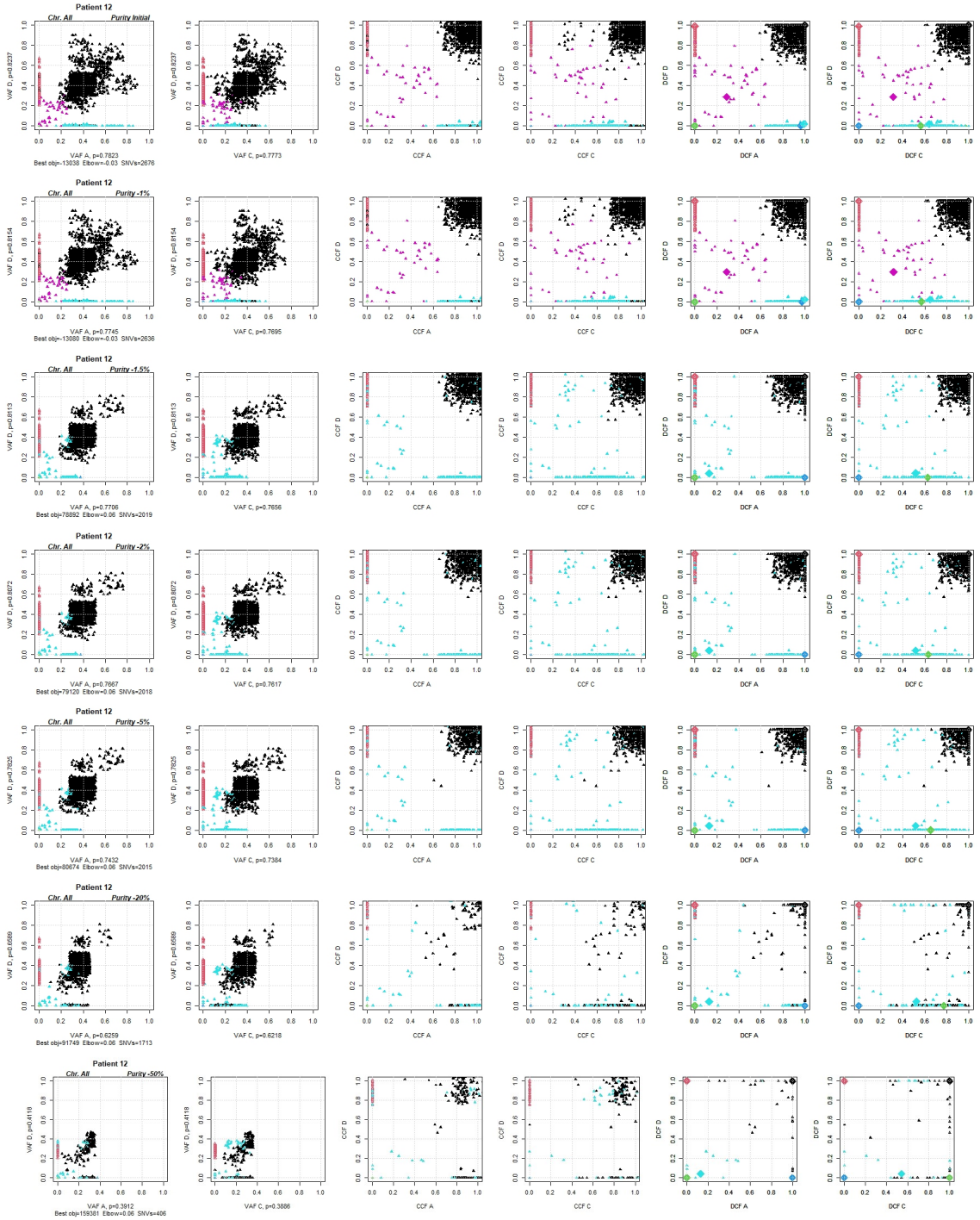



Figure 5.9: The graphs show the results of simulations of the DeCiFer algorithm when different purity values are used as input data for all the SNVs analyzed of the patient 12. The first row of the graphs shows the results obtained using the purity values provided by the demo (named "initials"). The first two graphs represents the VAF values of the SNVs selected for clustering, the next two show the CCF calculated under the assumption of constant multiplicity and the last ones show the results of the DCF estimator, implemented by the DeCiFer algorithm. The following rows, with decreasing percentages of the purity values with respect to the initial one, show the SNVs that remain compared to those selected when the input data is the initial purity, for the same estimators. Additionally, the axes indicate the purity values for each sample, as well as the best objective and elbow values returned by the algorithm, and the number of SNVs selected for the clusters with the default standard deviation of 1.5. Data are colored according to each cluster.

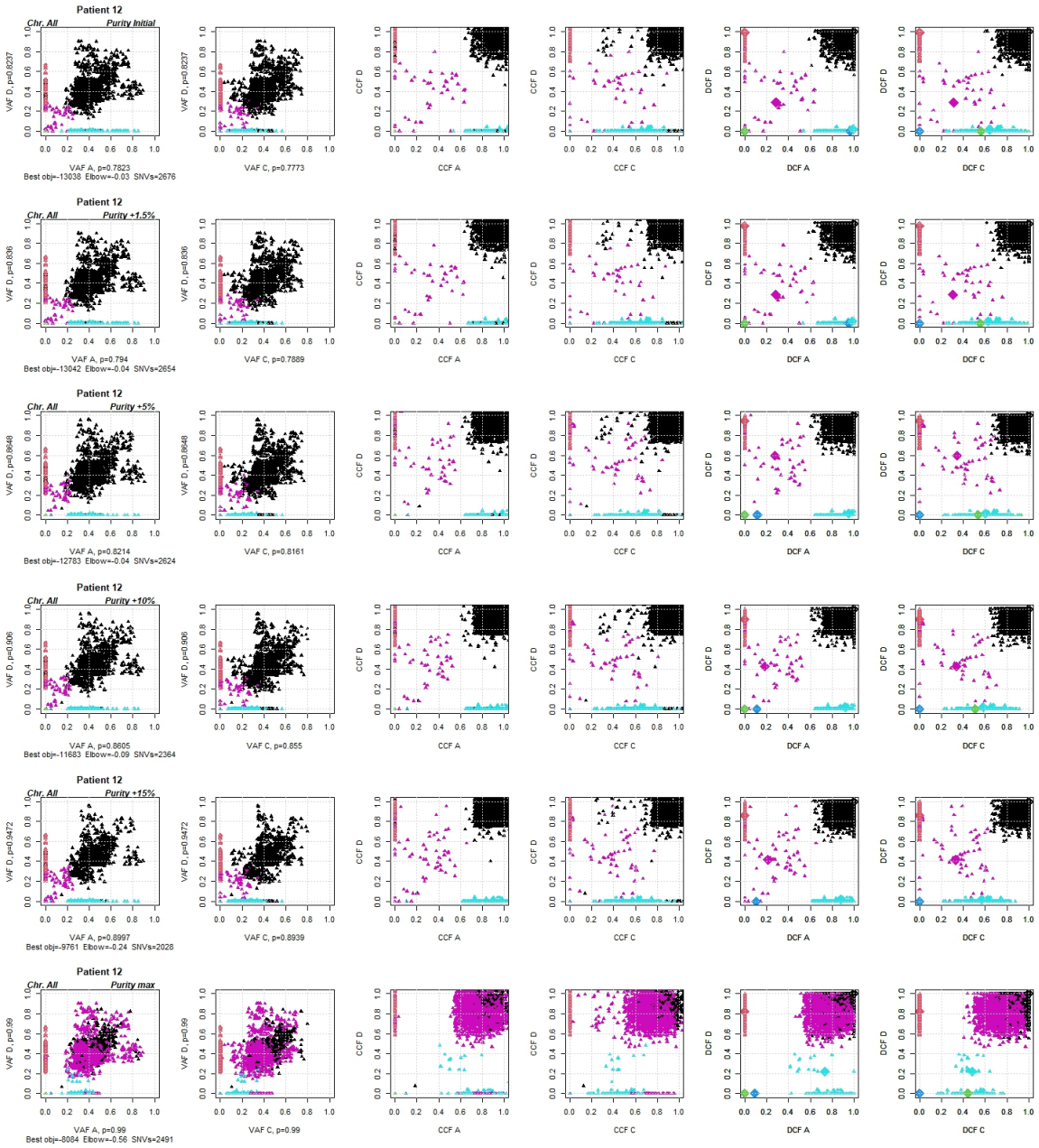



Figure 5.10: The graphs show the results of simulations of the DeCiFer algorithm when different purity values are used as input data for all the SNVs analyzed of the patient 12. The first row of the graphs shows the results obtained using the purity values provided by the demo (named "initials"). The first two graphs represents the VAF values of the SNVs selected for clustering, the next two show the CCF calculated under the assumption of constant multiplicity and the last ones show the results of the DCF estimator, implemented by the DeCiFer algorithm. The following rows, with increasing percentages of the purity values with respect to the initial one, show the SNVs that remain compared to those selected when the input data is the initial purity, for the same estimators. Additionally, the axes indicate the purity values for each sample, as well as the best objective and elbow values returned by the algorithm, and the number of SNVs selected for the clusters with the default standard deviation of 1.5. Data are colored according to each cluster.

As can be seen in the following table 5.11, the optimization improves when the input purity value increases (the best objective value decreases), while it worsens when the input purity value decreases (the best objective value increases).

| Input purity value | +15% | +10% | +5% | +1.5% | I | -1% | -1.5% | -2% | -5% | -20% | -50% |
|---|---|---|---|---|---|---|---|---|---|---|---|
| BS | -9761 | -11683 | -12783 | -13042 | -13038 | -13080 | 78892 | 79120 | 80674 | 91749 | 159381 |
| E | -0.2418 | -0.0915 | -0.0431 | -0.0352 | -0.0261 | -0.0288 | 0.06 | 0.06 | 0.06 | 0.06 | 0.06 |
| Best objective (BS) ; Elbow or silhouette score (E) | | | | | | | | | | | |

Figure 5.11: Table summarizing the values of best objective and elbow or silhouette score returned by the DeCiFer algorithm when analyzing the data of the patient 12 for each input purity value

In the following graphs (Figure 5.12 is shown a comparison between the limits of each cluster, which indicates the location of the centroid, for each sample. Note that the y axis is adjusted to the lower and upper limit of the cluster, therefore only the tendency can be compared.

For the main clusters, in the three samples, there is a tendency to lower the cluster limits as the input purity value increases, which agrees with theory, for a given VAF value, if the inferred purity increases, the reckoned CCF and DCF decrease.

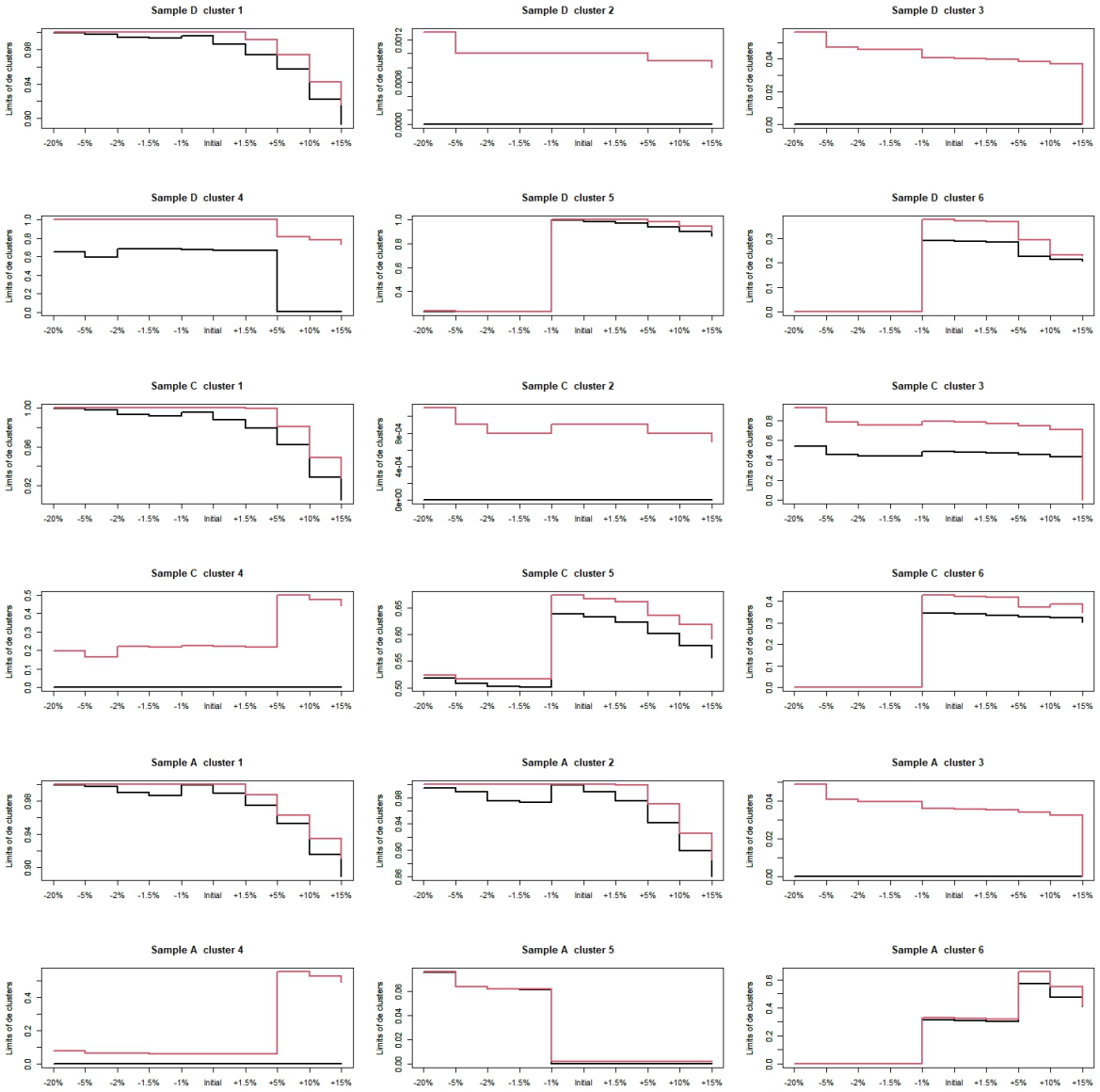



Figure 5.12: Comparison of the limits of each cluster as the percentage of the input purity values varies for each sample of the patient 12. The upper limit is indicated with a red line and the lower limit with a black line. Note the y axis values varies between the clusters.

The following table 5.13 shows the clustering analysis comparison for the results of variations in the input purity values compared to the initial input purity value.

| Input purity value | RI | ARI | Chi² | p-value | NID | NVI | NMI |
|---|---|---|---|---|---|---|---|
| +1.5% | 0.9993 | 0.9986 | 13072.1229 | 1 | 0.0038 | 0.0069 | 0.9962 |
| +5% | 0.9973 | 0.9945 | 9756.6715 | 1 | 0.0127 | 0.0249 | 0.9873 |
| +10% | 0.9892 | 0.9785 | 8550.5872 | 1 | 0.0395 | 0.0698 | 0.9605 |
| +15% | 0.9868 | 0.9733 | 5383.8138 | 1 | 0.0438 | 0.0804 | 0.9562 |
| -1% | 1 | 1 | 13180 | 1 | 0 | 0 | 1 |
| -1.5% | 0.9643 | 0.924 | 7608.0631 | 1 | 0.1302 | 0.2199 | 0.8698 |
| -2% | 0.9643 | 0.9239 | 7602.1424 | 1 | 0.1304 | 0.22 | 0.8696 |
| -5% | 0.9612 | 0.9172 | 7539.7341 | 1 | 0.141 | 0.2369 | 0.859 |
| -20% | 0.8906 | 0.7698 | 5683.3179 | 1 | 0.3427 | 0.4651 | 0.6573 |

Rand Index (RI), Adjusted Rand Index (ARI), Chi-square statistics (Chi²), Normalized information distance (NID), Normalized variation of information (NVI), Normalized mutual information (NMI)

Figure 5.13: Table summarizing different metrics for clustering analysis

The Rand index (RI) and its adjusted value (ARI) measure the similarity between the clusters. Variations in the input purity value barely affects the clustering results but the input purity value of -20%. The analysis shows a linear but slight decrease in the similitude of the clustering when the difference with respect to the initial purity value increases. The results of the $Chi^2$ and its $p-value$ associated also supports that the null hypothesis of no differences between the clusters cannot be rejected.

The normalized information distance (NID) value is a measure of the distance between the elements of a cluster, "the minimal information distance between x and y is the length of the shortest program for a universal computer to transform x into y and y into x" Vitányi, P., et al., 2008. A similar trend to the previous values is observed.

The normalized variation of information (NVI) is a measure of the distance of two clusters while the normalized mutual information (NMI) is a measure of the mutual dependence between the clusters. These measures also support the similarity of the clustering obtained by increasing the input purity values but a higher difference when decreasing the input purity values which may be related to the loss of information between the clustering when DCF estimator is inferred. As observed in the Figure 5.8 increasing the standard deviation parameter and decreasing the input purity value decreases the similarity with the results obtained with the initial purity values, as expected, since SNVs are included in the same group that should be segregated into different clusters, if the algorithm did not induce this border effect due to the initial purity value.

| Input purity value | SD | RI | ARI | Chi² | p-value | NID | NVI | NMI |
|---|---|---|---|---|---|---|---|---|
| -5% | 1.5 | 0.9612 | 0.9172 | 7539.7341 | 1 | 0.141 | 0.2369 | 0.859 |
| -5% | 20 | 0.7106 | 0.4152 | 10564.3878 | 1 | 0.5636 | 0.7173 | 0.4364 |

Rand Index (RI), Adjusted Rand Index (ARI), Chi-square statistics (Chi²), Normalized information distance (NID), Normalized variation of information (NVI), Normalized mutual information (NMI)

Figure 5.14: Comparison of the metrics for clustering analysis (in compararison to the results obtained with the input initial purity value) when the input parameter of standard deviation is increased from 1.5 to 20 when the input purity data assumes a 5% decrease in its inferred value.

### *Patient 17*

The file for patient 17 contains information from 12658 SNVs. The following figure 5.15 shows the VAF data calculated from that input file with the aim of depict the dispersion of data that the algorithm intends to cluster.

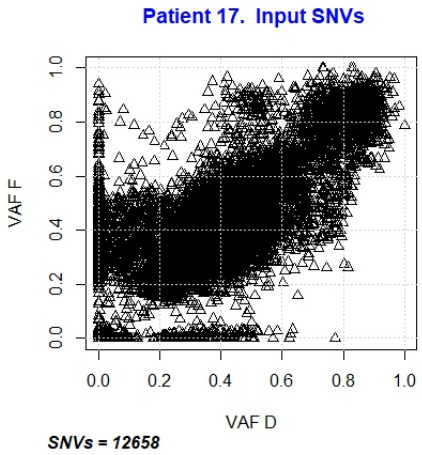



Figure 5.15: The figure depicts the VAF data dispersion, reckoned from the variant and reference readings for each of the SNVs in the file provided in the DeCifer demo of the patient numbered as 17, which is used as input data of the algorithm. The file contains 12658 SNV readings for each sample. The VAF values of the SNVs contained in sample F are represented on the $y$ axis while the VAF values of the SNVs contained in sample D are represented on the $x$ axis.

Figure 5.16 represents the results of the DeCiFer algorithm for patient number 17 resulting from running the algorithm with the input data with the initial purity, provided in the demo file, and subsequent simulations in which the algorithm is executed with files with different percentages of the initial purity values, indicated in each row of the graph. The first row of graphs attempts to reproduce Figure 5 from the study by Satas, G., et al., 2021, in which data from patient 17 on chromosome 6q are selected and grouped into the main cluster, considering that the article uses the beta-binomial model instead of the default binomial model used in this analysis.

As shown in the graph, and unlike what was observed with patient 12, the VAF graph does not change until the input purity values decrease significantly up to 20%, as do the inferred CCF and DCF values. In this case, from this value of -20%, no similarity is observed between the inferred CCF and DCF values.

However, when all the patient's SNVs are analyzed, areas in which SNVs are not selected are observed with an input purity value of -1%, and as observed with patient 12 data, as the input value of purity decreases, these differences increase, and the SNVs with higher VAF also disappear (Figure 5.17).

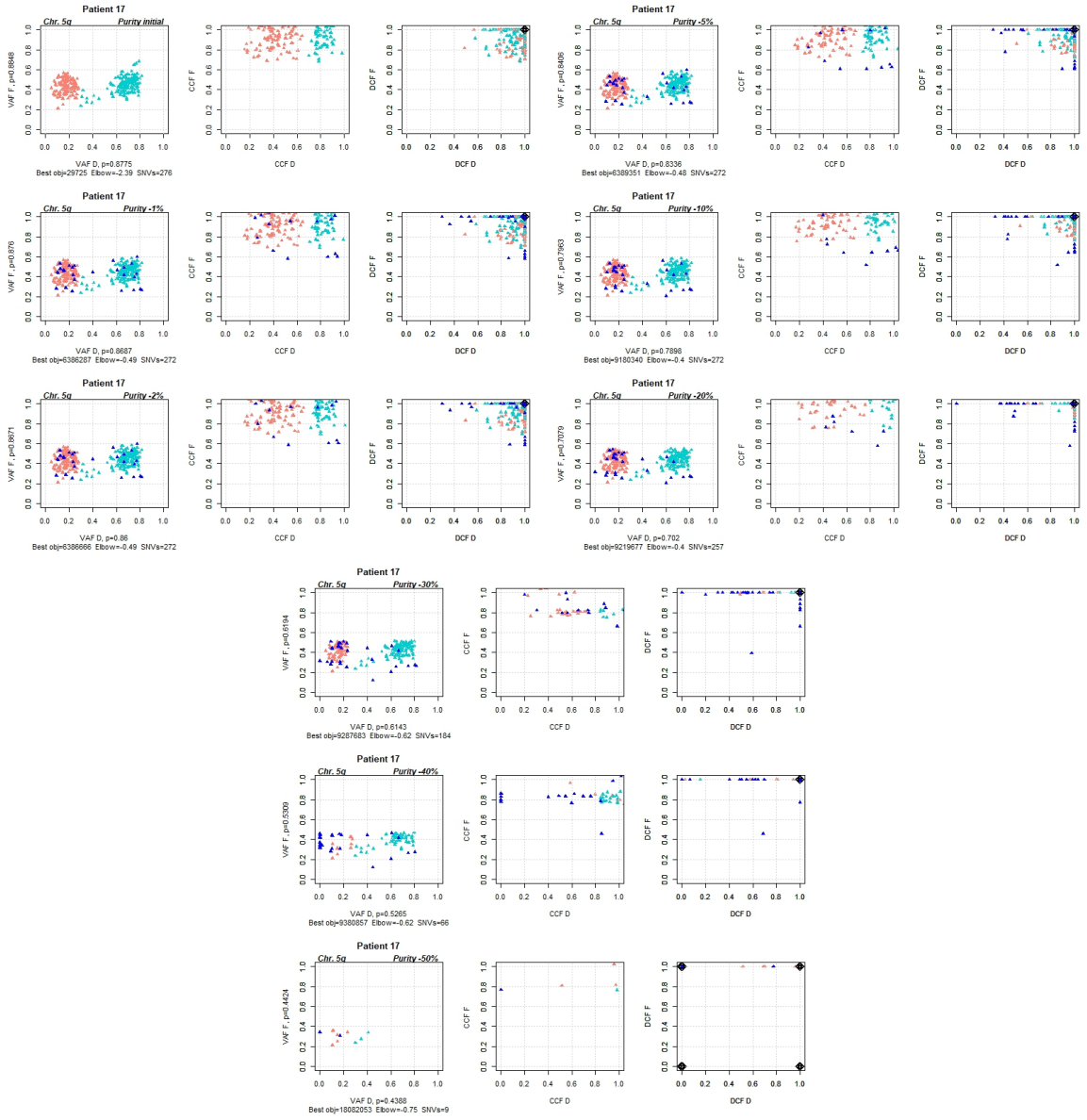



Figure 5.16: The graphs show the results of simulations of the DeCiFer algorithm when different purity values are used as input data. For illustrative and comparative purposes, an attempt has been made to reproduce figure 5 from the study by Satas, G., et al., 2021, in which data from patient 17 on chromosome 5q are selected in the main group (as noted in the main text the article uses the beta-binomial model instead of the default binomial model used in this analysis). Similar to the original figure, the first row of the graphs shows the results obtained using the purity values provided by the demo (named "initials"). The first graph represents the VAF values of the SNVs selected for clustering, the next show the CCF calculated under the assumption of constant multiplicity and the last one shows the results of the DCF estimator, implemented by the DeCiFer algorithm. The following rows, with decreasing percentages of the purity values with respect to the initial one, show the SNVs that remain compared to those selected when the input data is the initial purity, for the same estimators. Additionally, the axes indicate the purity values for each sample, as well as the best objective and elbow values returned by the algorithm, and the number of SNVs selected for the clusters with the default standard deviation of 1.5. Data colored in orange and blue belong to the same cluster and only differs by their VAF value. Data colored blue indicates additional values included in the main group when modifying input purity values that were not selected when running the algorithm with the initial purity value.

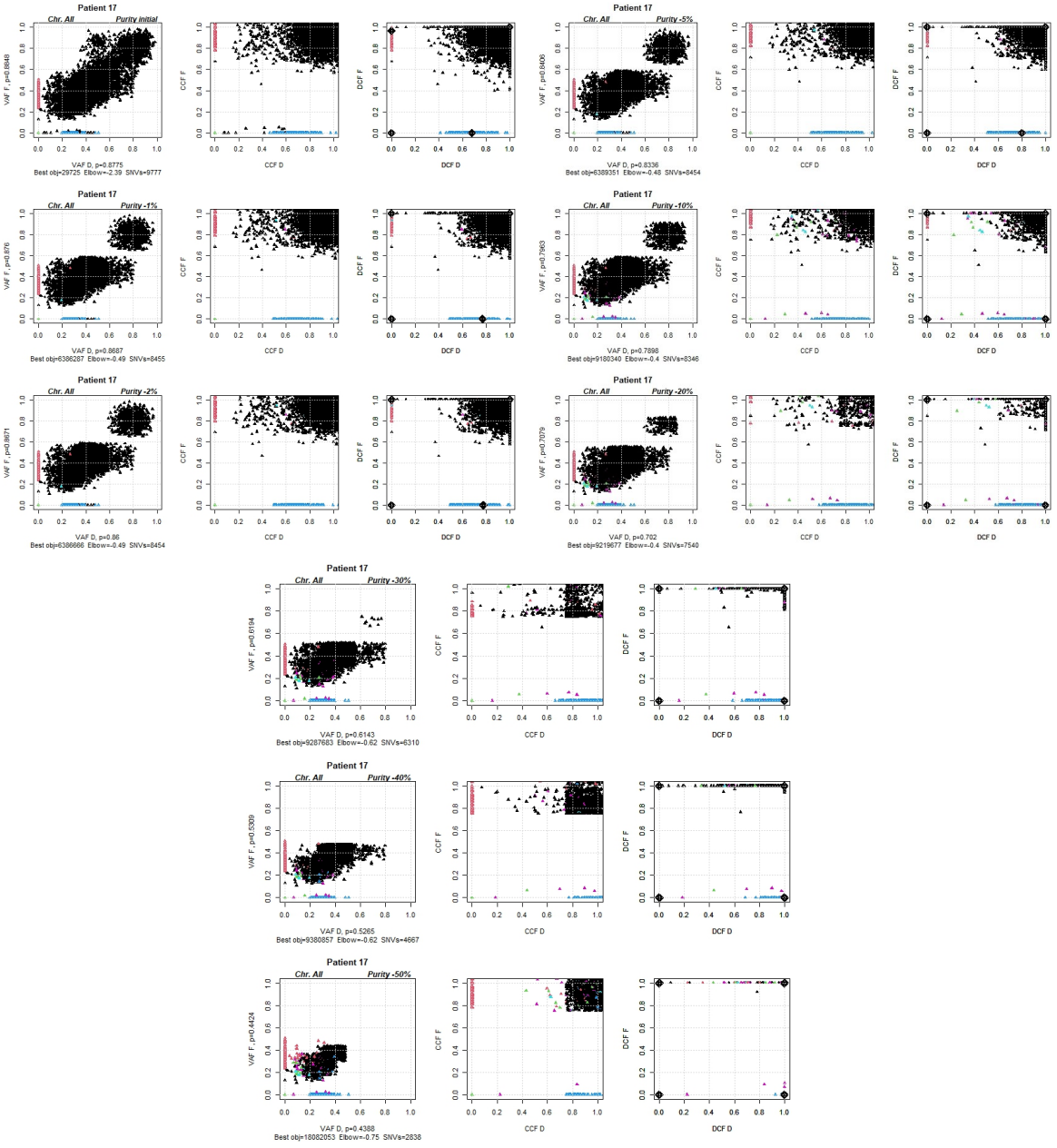



Figure 5.17: The graphs show the results of simulations of the DeCiFer algorithm when different purity values are used as input data for all the SNVs analyzed of the patient 17. The first row of the graphs shows the results obtained using the purity values provided by the demo (named "initials"). The first graph represents the VAF values of the SNVs selected for clustering, the next show the CCF calculated under the assumption of constant multiplicity and the last one show the results of the DCF estimator, implemented by the DeCiFer algorithm. The following rows, with decreasing percentages of the purity values with respect to the initial one, show the SNVs that remain compared to those selected when the input data is the initial purity, for the same estimators. Additionally, the axes indicate the purity values for each sample, as well as the best objective and elbow values returned by the algorithm, and the number of SNVs selected for the clusters with the default standard deviation of 1.5. Data colored blue indicates additional values included in the main group when modifying input purity values that were not selected when running the algorithm with the initial purity value.

And, similar to patient 12, the border effect observed when the input purity value decrease is not seen when the input purity value increases (Figure 5.18).

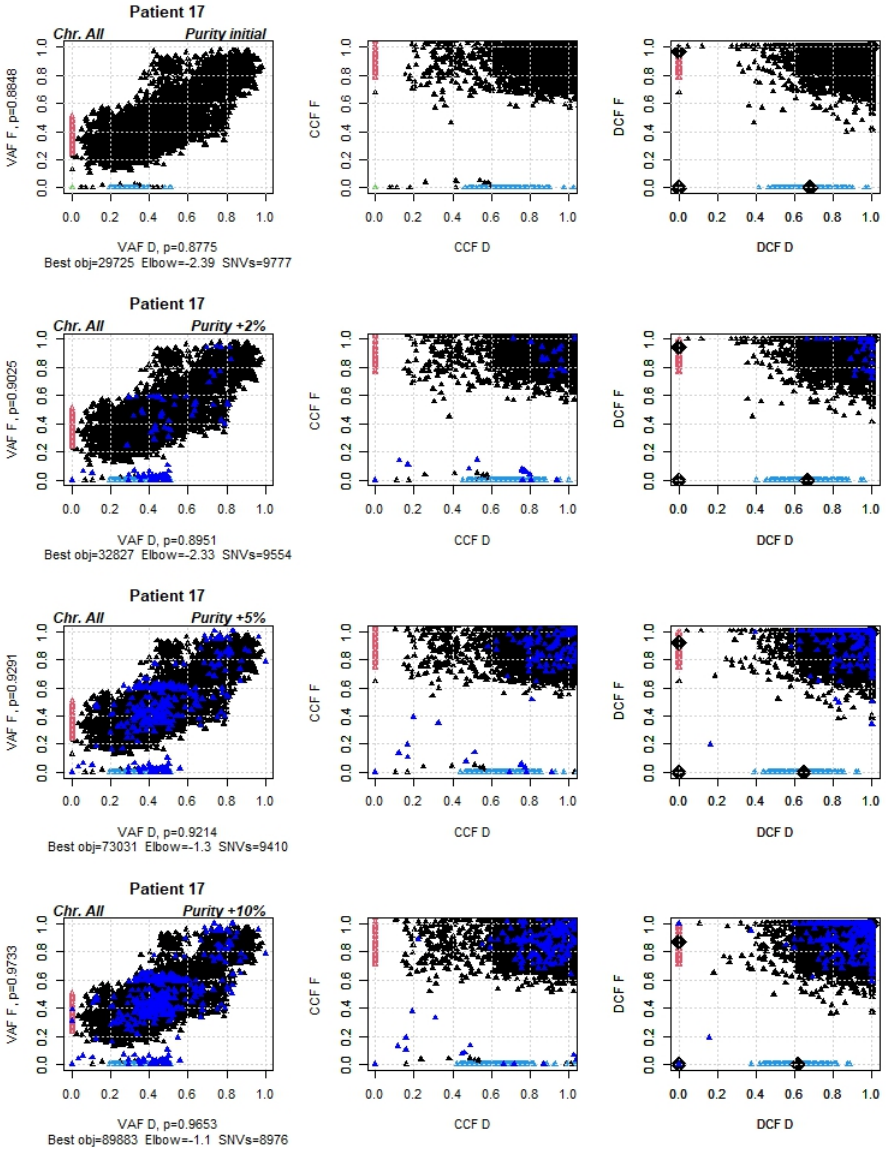



Figure 5.18: The graphs show the results of simulations of the DeCiFer algorithm when different purity values are used as input data for all the SNVs analyzed of the patient 17. The first row of the graphs shows the results obtained using the purity values provided by the demo (named "initials"). The first graph represents the VAF values of the SNVs selected for clustering, the next show the CCF calculated under the assumption of constant multiplicity and the last one show the results of the DCF estimator, implemented by the DeCiFer algorithm. The following rows, with increasing percentages of the purity values with respect to the initial one, show the SNVs that remain compared to those selected when the input data is the initial purity, for the same estimators. Additionally, the axes indicate the purity values for each sample, as well as the best objective and elbow values returned by the algorithm, and the number of SNVs selected for the clusters with the default standard deviation of 1.5. Data colored blue indicates additional values included in the main group when modifying input purity values that were not selected when running the algorithm with the initial purity value.

The following table 5.19 shows the best objective value corresponds to the initial purity value, the increase or decrease of this value worsens the optimization, although it is more noticeable when the input purity value decreases.

| Input purity value | +10% | +5% | +2% | I | -2% | -5% | -10% | -20% | -30% | -40% | -50% |
|---|---|---|---|---|---|---|---|---|---|---|---|
| BS | 89883 | 73031 | 32827 | 29725 | 6386666 | 6389351 | 9180340 | 9219677 | 9287683 | 9380857 | 18082053 |
| E | -1.098 | -1.3014 | -2.3336 | -2.3938 | -0.4866 | -0.4821 | -0.3999 | -0.3974 | -0.6221 | -0.617 | -0.7548 |
| Best objective (BS) ; Elbow or silhouette score (E) | | | | | | | | | | | |

Figure 5.19: Table summarizing the values of best objective and elbow or silhouette score returned by the DeCiFer algorithm when analyzing the data of the patient 17 for each input purity value

A comparison between the limits of each cluster, which indicates the location of the centroid, for each sample is shown in the following graphs. Note that the y axis is adjusted to the lower and upper limit of the cluster, therefore only the tendency can be compared.

For all clusters in both samples there is a clear tendency to lower the cluster limits as the input purity value increases, which agrees with theory, for a given VAF value, if the inferred purity increases, the reckoned CCF and DCF decrease (Figure 5.20).

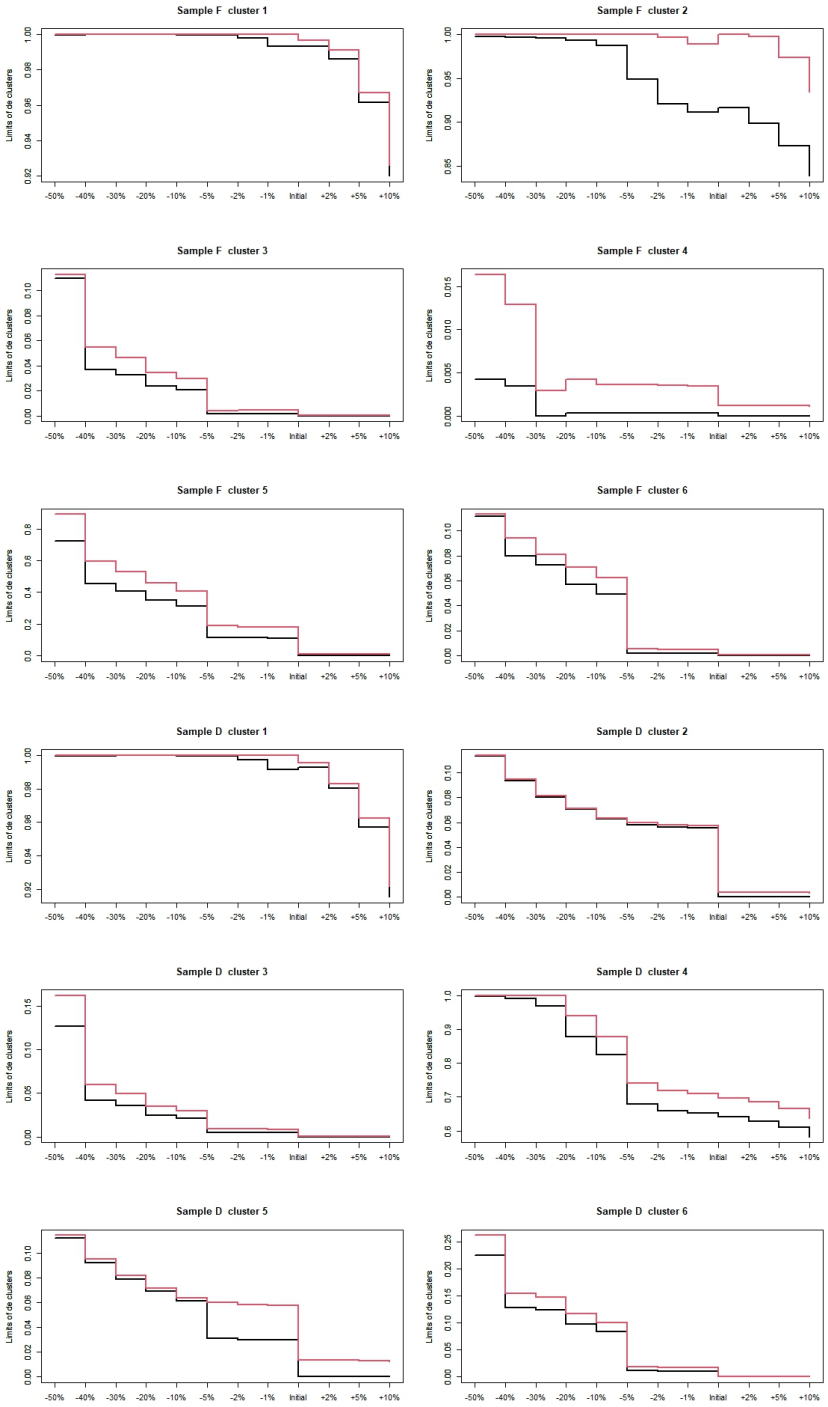



Figure 5.20: Comparison of the limits of each cluster as the percentage of the input purity values varies for each sample of the patient 17. The upper limit is indicated with a red line and the lower limit with a black line. Note the y axis values varies between the clusters.

The following table 5.21 shows the clustering analysis comparison for the results of variations in the input purity values compared to the initial input purity value. All metrics show similarity between the clustering performed with the initial purity value and the different variations of this value, although it gets slightly worse as the input value of purity decreases.

| Input purity value | RI | ARI | $Chi^2$ | p-value | NID | NVI | NMI |
|---|---|---|---|---|---|---|---|
| +2% | 1 | 1 | 47770 | 1 | 0 | 0 | 1 |
| +5% | 1 | 1 | 47050 | 1 | 0 | 0 | 1 |
| +10% | 1 | 1 | 44880 | 1 | 0 | 0 | 1 |
| -2% | 0.9971 | 0.9921 | 40596.9156 | 1 | 0.023 | 0.0326 | 0.977 |
| -5% | 0.9971 | 0.9921 | 40596.9156 | 1 | 0.023 | 0.0326 | 0.977 |
| -10% | 0.9789 | 0.9443 | 35379.2301 | 1 | 0.1297 | 0.1783 | 0.8703 |
| -20% | 0.9772 | 0.9431 | 32014.78 | 1 | 0.1289 | 0.1779 | 0.8711 |
| -30% | 0.9725 | 0.9365 | 26733.6563 | 1 | 0.1359 | 0.1883 | 0.8641 |
| -40% | 0.9655 | 0.9272 | 19804.3474 | 1 | 0.1465 | 0.2061 | 0.8535 |

Rand Index (RI), Adjusted Rand Index (ARI), Chi-square statistics ($Chi^2$), Normalized information distance (NID), Normalized variation of information (NVI), Normalized mutual information (NMI)

Figure 5.21: Table summarizing different metrics for clustering analysis

# Chapter 6

# Conclusions

Since the information gathered from the sequencing reads is limited and suffer from non-identifiability problems, the methods used for genotype clustering are maximizing algorithms and need to use constraints to limits the possible solution that can explain the observed data.

One such constraint is the plausibility between VAF values and tumor purity estimates. Given that high values of VAF are no plausible with low tumor purity estimations, the DeCiFer algorithm establish a limit for the selection of feasible VAF values to performed the clustering, which is observed as a border effect when the input purity value decreases under a certain threshold. The DeCiFer algorithm achives this aim by initializing the coordinate ascent algorithm fixing the cell fraction values to the sample purity to perform clustering and infer the DCF values. This also allows for filtering out possible artefacts that could happen during the sequencing or the variant calling process.

The difference between the robustness of the results observed for the selected region analyzed for each patient may actually be explained by an underestimation of the purity of sample 12 (the purity value provided in the demonstration), very close to the lower threshold for plausible values of VAF, as the algorithm returns a better optimization value with higher input purity values. Likewise, the robustness of the results observed in the region analyzed for patient 17 to greater decreases in the input purity value can be explained for that same reason, reinforced by the fact that the algorithm also uses information from the other samples to perform clustering, which is highlighted by the authors as an improvement of this algorithm compared to others. Thus, in the case of patient 17, the clustering is performed using information from 5 samples, with the value of the sample with the highest purity being $\rho = 0.91$, while the clustering performed for patient 12 uses information from 3 samples being the value of the sample with the highest purity of $\rho = 0.83$, these differences appear to allow for higher plausible VAF values for clustering of SNVs belonging to patient 17.

Furthermore, the variation in the selection of SNVs for close VAF values reflects the interaction between the constraints imposed by the DeCiFer algorithm, as SNVs with similar VAF values do not have to belong to the same phylogenetic branch.

Although the imposed constrains for clustering allows to avoid implausible results and the robustness of the results of the DeCiFer algorithm increases with the number of samples, in a similar way to the probabilistic model implemented for the uncertainty due to sequencing errors and coverage for VAF estimates, the uncertainty in the estimation of tumor purity should also be modeled to avoid drastic interpretation of the data due to small errors in its inference.

# Bibliography


[1] Van Loo, P., et al., Allele-specific copy number analysis of tumors. Proc Natl Acad Sci U S A, 2010. 107(39): p. 16910-5.

[2] Tarabichi, M., et al., A practical guide to cancer subclonal reconstruction from DNA sequencing. Nat Methods, 2021. 18(2): p. 144-155.

[3] Satas, G., et al., DeCiFering the elusive cancer cell fraction in tumor heterogeneity and evolution. Cell Syst, 2021. 12(10): p. 1004-1018 e10.n

[4] Dentro, S.C., D.C. Wedge, and P. Van Loo, Principles of Reconstructing the Subclonal Architecture of Cancers. Cold Spring Harb Perspect Med, 2017. 7(8).

[5] Sandmann, S., et al., Reconstructing Clonal Evolution-A Systematic Evaluation of Current Bioinformatics Approaches. Int J Environ Res Public Health, 2023. 20(6).

[6] Dentro, Stefan Christiaan. (2020). The Intra-Tumour Heterogeneity Landscape of Human Cancers. Tesis. University of Cambridge.

[7] Nik-Zainal, S., et al., The life history of 21 breast cancers. Cell, 2012. 149(5): p. 994-1007.

[8] Kader, T., M. Zethoven, and K. Gorringe, Evaluating statistical approaches to define clonal origin of tumours using bulk DNA sequencing: context is everything. Genome Biology, 2022. 23.

[9] Barnell, E.K., et al., Standard operating procedure for somatic variant refinement of sequencing data with paired tumor and normal samples. Genet Med, 2019. 21(4): p. 972-981.

[10] Salcedo, A., et al., A community effort to create standards for evaluating tumor subclonal reconstruction. Nat Biotechnol, 2020. 38(1): p. 97-107.

[11] Olson, N.D., et al., Variant calling and benchmarking in an era of complete human genome sequences. Nat Rev Genet, 2023. 24(7): p. 464-483.

[12] Cortes-Ciriano, I., et al., Computational analysis of cancer genome sequencing data. Nat Rev Genet, 2022. 23(5): p. 298-314.

[13] Bagger, F.O., et al., Whole genome sequencing in clinical practice. BMC Med Genomics, 2024. 17(1): p. 39.

[14] Vendramin, R., K. Litchfield, and C. Swanton, Cancer evolution: Darwin and beyond. EMBO J, 2021. 40(18): p. e108389.

[15] Dunson, D., Nonparametric Bayes Applications to Biostatistics. Bayesian Nonparametrics, 2010.

[16] Zaccaria, S. and B.J. Raphael, Accurate quantification of copy-number aberrations and whole-genome duplications in multi-sample tumor sequencing data. Nat Commun, 2020. 11(1): p. 4301.

[17] Vitányi, P., et al., Normalized Information Distance. Information Theory and Statistical Learning, 2008

[] Mallory, X.F., et al., Methods for copy number aberration detection from single-cell DNA-sequencing data. Genome Biology, 2020. 21(1): p. 208.