# PROBABILISTIC INFERENCE OF CLONE TREES FROM MULTI-REGION SEQUENCING OF TUMORS

by

Lily Zheng

A dissertation submitted to The Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy.

Baltimore, Maryland

March, 2022

# Abstract

Cancer is a complex, adaptive system characterized by constantly evolving subclonal populations of cells as a result of somatic mutation accumulation and selective pressures. Because of this inherent evolutionary nature, modeling tumor subclonal architecture is crucial for understanding disease progression, therapeutic resistance, and relapse. The uncertainty in clonal composition and the multitude of possible ancestral relationships between clones pose statistical and computational challenges to the elucidation of the most probable evolutionary history from bulk tumor sequencing. Evolutionary analyses of tumors can be broken down into two main components: estimation of the proportion of cancer cells in a tumor that harbor a somatic mutation (cancer cell fraction, CCF) and the temporal ordering of somatic mutations. Existing methods have implemented various statistical and computational approaches including Bayesian mixture models, graph enumeration algorithms, and linear algebraic approaches. However, none are able to provide a comprehensive evolutionary analysis that characterizes uncertainty in all three areas: assign-

ment of mutations to clusters, estimation of cancer cellular fractions of mutation clusters, and evolutionary relationships between clones. This thesis develops a new approach, PICTograph, that improves methodology for cancer cell fraction estimation and inference of clone trees from multi-sample data and evaluates uncertainty at each step of evolutionary analyses. PICTograph implements a Bayesian hierarchical model to quantify uncertainty in assigning mutations to subclones and sample posterior distributions of CCFs. Then to identify candidate evolutionary relationships between mutation clusters, PICTograph applies rules based on established evolutionary modeling principles to their estimated CCFs, effectively reducing the space of clone trees for full enumeration. Trees are evaluated using a fitness function, and the highest scoring trees are summarized to visualize the most probable ancestral relationships between mutation clusters. On simulated data, PICTograph performs better than existing methods in both CCF estimation and tree inference. Finally, I apply PICTograph to two multi-region datasets: whole-exome sequencing of intraductal papillary mucinous neoplasms and longitudinal whole-exome sequencing of immunotherapy-treated non-small cell lung cancer.

**Primary Reader and Advisor:** Dr. Rachel Karchin

**Secondary Reader:** Dr. Robert B. Scharpf

# Acknowledgments

I am forever grateful to the numerous people whose support and contributions have made this work possible. First, I would like to thank my thesis advisor, Rachel Karchin. Her guidance, advice, and encouragement of developing my ideas were invaluable to the formation and completion of this dissertation. I would also like to thank my thesis committee members, Rob Scharpf, Laura Wood, and Sarah Wheelan, for their insightful ideas and suggestions. In particular, many thanks to Rob for his contribution to and patience during the development of PICTograph. I am incredibly thankful for the scientific and emotional support from the past and current members of the the Karchin lab: Melody Shao, Rohit Bhattacharya, Ashok Sivakumar, Violeta Beleva Guthrie, Collin Tokheim, and Noushin Niknafs. My thesis was largely inspired from working with and learning from Violeta and Noushin. Finally, I want to thank my friends and family for being there for me on this journey. And special thanks to my dog, Korra, who takes me on walks for my mental health.

# Dedication

To my friends and family

# Contents

# List of Figures

LIST OF FIGURES

# Chapter 1

# Introduction

## 1.1 Cancer as an evolutionary process

According to the somatic mutation theory of cancer, tumors are complex, adaptive systems characterized by subclonal populations of cells that constantly evolve due to accumulation of somatic mutations and selective pressures [1,2]. Cells may gain somatic mutations at random due to mistakes during DNA replication or exposure to DNA damaging agents. Tumorigenesis is an evolutionary process in which cells acquire selectively advantageous mutations (drivers), leading to clonal expansions [1–3]. The remaining, non-driver mutations are known as passenger mutations, and while they do not confer a fitness advantage [3], they are useful for many analyses such as evolutionary inference and mutational signatures. Within a tumor, the different subclones can

be defined by their sets of somatic mutations. Computational modeling of these subclonal architectures has demonstrated utility in understanding disease progression, therapeutic resistance, and relapse [1, 4].

## 1.2 Strategies for modeling cancer evolution

To dissect intratumoral heterogeneity, many bioinformatics tools have been developed to reconstruct tumor evolutionary histories from next-generation sequencing (NGS) data. There are two main classes of cancer evolution models: sample tree phylogenies and clone (or mutation) trees.

### 1.2.1 Sample tree phylogenies

A sample tree phylogeny is a branching diagrams that depict the evolutionary relationships among multi-region samples of a tumor (or multiple tumors) from a single patient. In a sample tree, each sample is a leaf node and internal nodes represent unobserved ancestral states [5–7]. An implicit assumption of sample tree analyses is that the samples are monoclonal or can be meaningfully summarized as the collection of observed mutations, and thus reflect the overall similarity of samples [8]. Sample tree phylogenies are based upon

similarities and differences in the mutation profiles of the samples (e.g. sets of mutations that are absent or present in a given sample) [8]. Graphically, branch lengths can represent the number of mutations accumulated between nodes.

However, tumors typically have high amounts of intratumor genetic heterogeneityand thus bulk tumor samples are often mixtures of multiple cell lineages [8]. Constructing sample trees from heterogeneous samples can lead to the appearance of a somatic variant occurring independently on different branches when the more probable phenomenon is the presence of multiple subclones, some of which are shared between the sequenced regions. From an analysis standpoint, sample trees are useful in representing the overall similarities of the sequenced samples but not necessarily evolutionary history of subclones within the tumor(s) analyzed.

### 1.2.2 Clone trees

The inability to represent intratumor heterogeneity and the evolutionary relationships among different tumor subclones has lead to a rapidly developing focus in the community on clone tree models, which depict the evolutionary relationships among the different genetic cell lineages identified [8]. In a clone tree, each node represents a cluster of mutations, and each extant population of cells comprising a subclone is represented by the aggregation of mutations

from root to leaf [9–13].

Clone tree models rely on an estimate of the proportion of cancer cells in a tumor that harbor a somatic mutation (cancer cell fraction, CCF), as well as the number of mutation clusters. Unlike the variant allele fraction (VAF) where direct estimates are available from standard mutation-calling algorithms [14], the CCF is not observed directly and must be inferred from the VAF, multiplicity, DNA copy number of the tumor genome containing the mutation, and the tumor purity of the bulk sample that was sequenced. As heterogeneity in the mutational composition of subclones gives rise to differences in the observed VAF between mutations, statistical approaches for clustering mutations provide an avenue to inferring the number of subclones and improving CCF estimation by pooling information from all available mutations within a subclone [9, 15, 16]. Bayesian mixture models implemented by Markov Chain Monte Carlo (MCMC) [15] or variational Bayes [16] are attractive for their ability to jointly summarize the uncertainty of CCF estimates and the assignment of mutations to clusters through posterior distributions. These approaches for clustering mutations differ by the types of mutations analyzed (sequence-only or copy number and sequence), the determination of the number of clusters, and how unknown parameters including the VAF, copy number, multiplicity, and purity are modeled to infer CCFs. Limitations of these approaches include assumptions that regions containing the sequenced mutations are diploid and

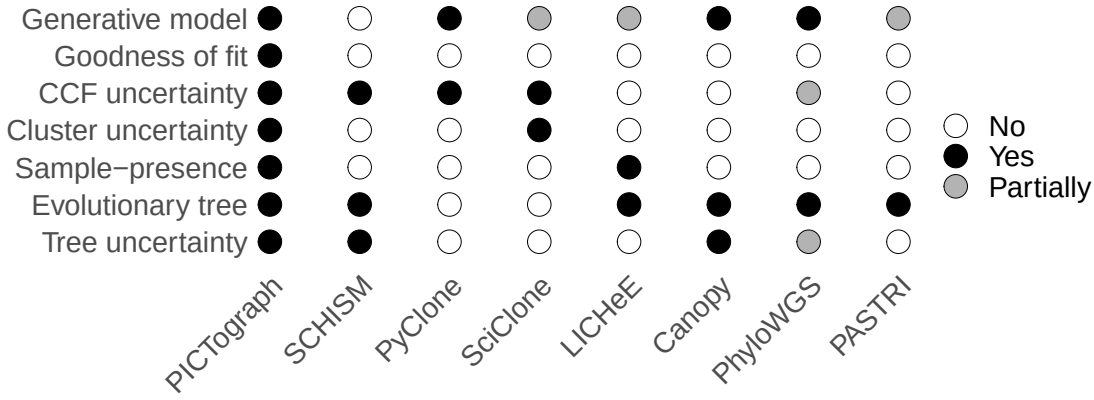



**Figure 1.1:** A comparison of software for evolutionary inference from multi-sample sequencing data. PICTograph develops a generative model for the observed variant allele counts that integrates copy number, multiplicity of the allele, tumor purity, and a latent indicator for the cluster membership of sequenced variants. Posterior predictive distributions available from PICTograph allow a formal assessment of the adequacy of the proposed model for capturing the heterogeneity of variant allele counts within and between samples. As the CCF model is fully Bayesian, posterior distributions of the CCFs and probabilistic estimates of cluster membership are available for all identified variants. Using modal CCF estimates from the posterior of the CCF model, PICTograph identifies all highest scoring trees and depicts tied top-scoring trees as an ensemble tree. Sample-presence, while not novel to PICTograph, focuses computational effort on the most probable trees and enables a relatively fast exhaustive scoring of trees in a more limited search space.

that the variant allele is heterozygous. Additionally, none of the existing approaches provide a comprehensive characterization of the uncertainty in both the assignment of mutations to clusters and the estimation of CCFs (Figure 1.1).

Mutation clusters characterized by their CCFs can be ordered in a branching tree topology [9, 10, 12, 13]. These topologies are generally restricted by the Sum Condition [9, 11, 13, 17], which states that the CCF of an ancestral clone

must be greater than or equal to the sum of CCFs of its descendants. This principle can be applied to pairwise comparisons such that the CCF of any mutation cannot exceed the CCF of its ancestor (lineage precedence, [9]). Existing tree inference methods apply these principles using probabilistic [9, 10, 12] and combinatorial [11, 13, 17] frameworks. Due to the number of somatic mutations and likely subclones in many solid tumors, efficient approaches to explore the most probable trees are needed.

## 1.3 Study design

The way tumors are sampled and sequenced have large effects on the accuracy of evolutionary inference. In particular, the number of samples, sequencing depth and breadth, and tumor purities of samples may influence the resolution and accuracy of subclonal reconstruction.

Tumors can be sequenced either by using a single sample or multiple samples from different regions of the tumor. Given intratumoral heterogeneity and unclear spatial mixing of subclones, single-sample studies can underestimate the number of subclones with tumors. In these studies, populations or mutations that appear to be clonal within the sequenced sample may actually be subclonal in the entire tumor [18]. On the other hand, multi-region sequencing can improve the identification and resolution of all subclones within a tu-

mor. Additional samples can provide more opportunity to sample subclones that may be present at lower frequencies in a tumor and can improve separation of subclones based on differences in CCF patterns across samples and thereby facilitate phylogenetic inference [18].

Evolutionary analysis is also directly impacted by the sequencing technology used. Since the total read counts and variant read counts are direct inputs to estimate CCFs of mutations, the depth of sequencing influences the accuracy and precision of CCF estimation. Typically, increasing sequencing depth improves the precision of CCF estimates akin to having a larger sample size. The breadth of sequencing can range from small gene panels, where tens of genes are targeted, to whole-genome sequencing. Sequencing a larger portion of the genome allows identification of more somatic mutations, which in turn, improves evolutionary inference by providing more data to identify subclones and estimate CCFs (Section 1.2.2).

A third aspect of study design is the tumor purity of the sequenced samples, which refers to the fraction of cancer cells in the tumor sample. Tumor purity is a key variable in copy number analysis and CCF estimation. Higher tumor purity improves evolutionary inference, since a larger portion of the sequencing data comes from the tumor. Laser capture microdissection can be used to increase tumor purity [19–21]. Another challenging aspect of tumor purity is that it must be estimated. There are several strategies for estimating tumor purity:

pathology-based, mutation-based, and copy-number-based. A pathology-based estimate of tumor purity can be obtained by expert pathologists reviewing HE-stained slides of tumor sections. However, estimation by pathologists have been found to be inaccurate [22]. Mutation-based estimates of purity are calculated using the VAFs of either the assumed driver mutation(s) or all mutations in the sample. One issue with mutation-based estimates is their underlying assumption that a one or more mutations in the sample are clonal, which becomes problematic for tumors with poly-clonal evolutionary structure. Copy-number based estimates can be obtained by analysis with several bioinformatics tools for copy number analysis [23, 24]. Since copy-number-based estimates rely on the tumor sample containing copy number alterations, these methods will fail for samples without clonal copy number alterations. Overall, implementing experimental methods to increase tumor purity will improve evolutionary analysis and selection of the approach for purity estimation is important to limit the downstream consequences of inaccurate estimations.

Given the trade-offs of study design, Tarabichi *et al.* recommend that, in general, sequencing more samples is more beneficial than higher-depth sequencing to improve performance of subclonal reconstruction [18].

# Chapter 2

# PICTograph

I have developed a computational method for **P**robabilistic **I**nference of **C**lone **T**rees from mutli-region sequencing data called PICTograph. PICTograph is composed of two primary steps: (1) mutation clustering and cancer cell fraction estimation and (2) tree inference. To cluster mutations an estimate cancer cell fractions (CCF), PICTograph uses a Bayesian hierarchical model and approximates posterior distributions by Markov chain Monte Carlo (MCMC). For tree inference, PICTograph uses the modal CCF estimates for identified mutation clusters, and identifies the most probable trees through filtering, enumeration, and scoring of candidate trees. This chapter is based on material published in Zheng *et al.* 2022 [25].

## 2.1 Estimation of mutation clusters and cancer cell fractions

The first step of evolutionary analyses with PICTograph is estimation of the proportion of cancer cells in a tumor that harbor a somatic mutation (cancer cell fraction, CCF), as well as the number of subclones. This involves estimating the number of mutation clusters that define subclonal populations, inferring the cluster assignment of mutations, and estimating the CCF of each cluster. PICTograph implements a Bayesian hierarchical model and provides multiple visualizations to assess and understand the results.

### 2.1.1 Algorithm and Bayesian hierarchical model

As an initial step for CCF estimation, mutations are separated into sets according to the number of samples for which the mutation was detected (sample presence) (Figure 2.1A). A patient with $S$ sequenced samples could have as many as $2^S - 1$ mutation sets.

Next, each sample-presence set is independently evaluated using a generative model (Figure 2.2) for the observed number of reads with a somatic mutation $y$ for allele $i$ in sample $s$ that permits inference for the latent CCF

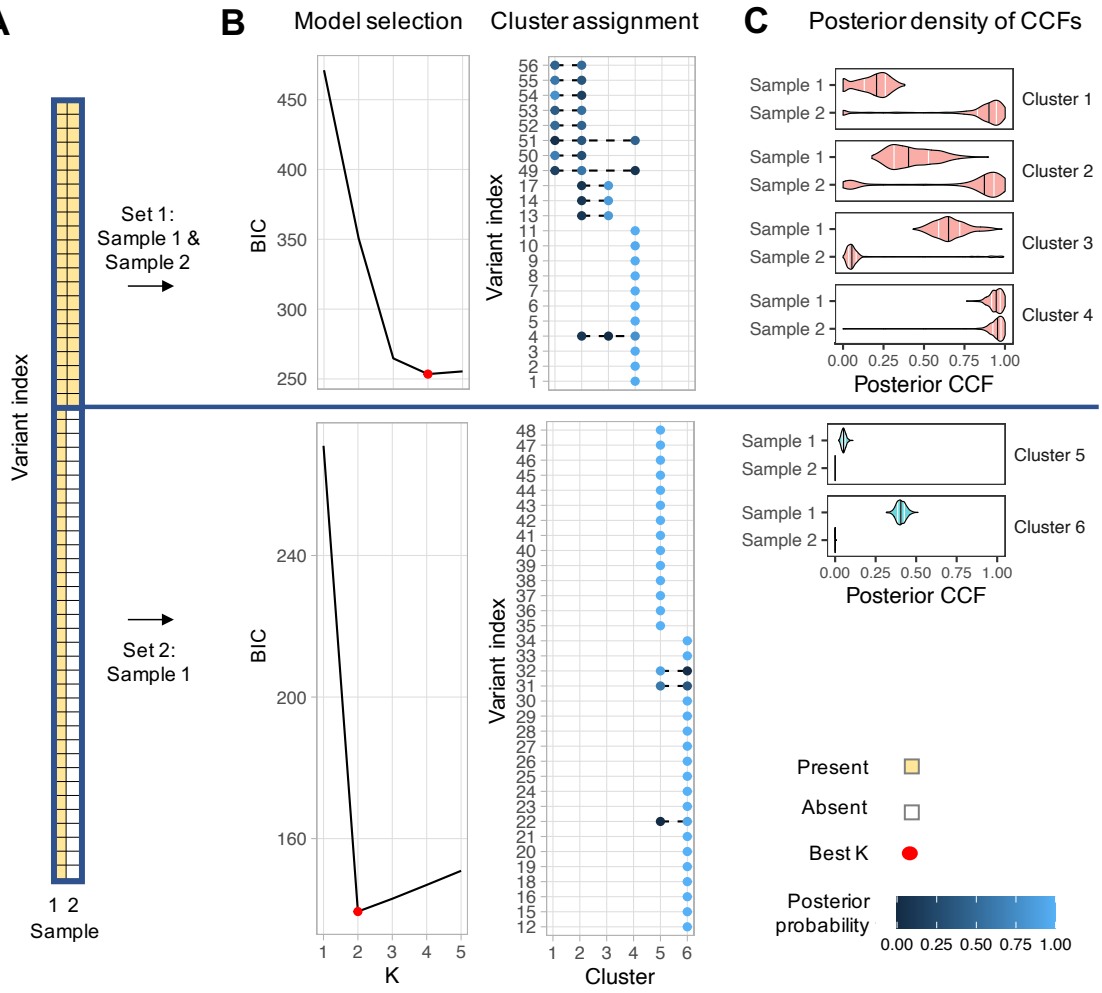



**Figure 2.1:** Overview of approach for mutation clustering and cancer cell fraction (CCF) estimation. (A) To estimate CCFs, PICTograph first separates the mutation data into sets based on sample presence patterns. (B) For each sample-presence set, PICTograph uses MCMC sampling to jointly estimate each mutation's cluster assignment and the CCFs of each cluster for a plausible range of mutation clusters $K$. The Bayesian Information Criterion (BIC) is applied to select the optimal value of $K$. The posterior probabilities of cluster assignments show the level of certainty of membership in each cluster. The total number of clusters obtained for an individual is the sum of the number of clusters identified in each sample-presence set. (C) Posterior distributions of mutation cluster CCFs. The black lines in the violin plots mark the median, the white lines mark the first and third quantiles.

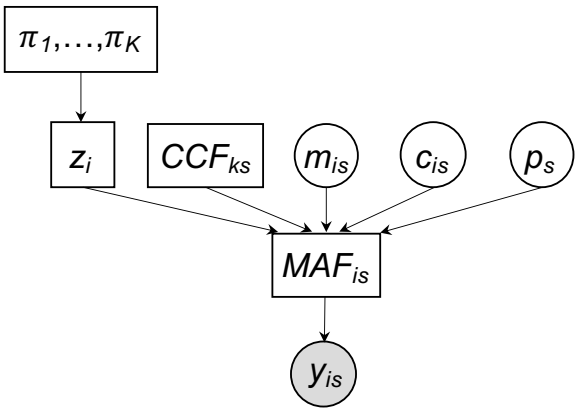



**Figure 2.2:** Bayesian hierarchical model for clonal architecture estimated from multi-sample sequencing.

$$
\begin{aligned}
[y_{is}|\mathbf{VAF}_{is}, n_{is}] &\sim \mathbf{Binomial}(\mathbf{VAF}_{is}, n_{is}) \\
[\mathbf{VAF}_{is}|Z_i = z, m_{is}, c_{is}, CCF_{zs}, p_s] &= \frac{m_{is} \times CCF_{zs} \times p_s}{c_{is} \times p_s + 2 \times (1 - p_s)} \\
[Z_i|\pi_1, \ldots, \pi_K, K] &\sim \mathbf{Multinomial}(\pi_1, \ldots, \pi_K) \\
\pi_1, \ldots, \pi_K &\sim \mathbf{Dirichlet}(1_1, \ldots, 1_K) \\
[\mathbf{CCF}_{zs}|\eta] &\sim \eta\mathbf{Beta}(1,1) + (1-\eta) \times 0 \\
\eta &\sim \mathbf{Beta}(5,2),
\end{aligned}
$$

for $s = 1, \ldots, S$, $z = 1, \ldots, K$ and $i = 1, \ldots, M$. The unobserved parameters $Z_i$ and $CCF_{zs}$ indicate the cluster membership for the $i$th mutation and the cancer cell fraction for cluster $z$ in sample $s$, respectively.

The joint posterior distribution of $\{Z, VAF, \boldsymbol{\pi}, \eta\}$ is approximated by Markov

chain Monte Carlo (MCMC) implemented using JAGS (version 4.3.0). For each sample-presence set, we evaluate a range of possible values for $K$ and select the $K$ that minimizes the Bayesian Information Criterion (BIC) (Figure 2.1B).

Finally, the MCMC chains from the best choice of $K$ for each sample-presence set are merged, resulting in posterior distributions for mutation cluster assignments and cluster CCFs (Figure 2.1 B and C). With $K_t^*$ mutation clusters identified in sample presence set $t$, the total number of mutation clusters for a patient was obtained by $K^*$, $K^* = \sum_t K_t^*$.

### 2.1.2 Assessment and visualization of results

To assist with assessing and interpreting clustering and CCF estimation results, PICTograph is equipped with several visualization functions. The following subsections contain example visualizations of results from the toy dataset available in PICTograph.

#### 2.1.2.1 MCMC chain convergence

Currently, PICTograph is run with a single MCMC chain. Convergence is assessed by visual inspection of the chains (Figure 2.3). As default parameters, MCMC is run for 10,000 iterations, with a burn-in of 1000 and thinning parameter of 10.

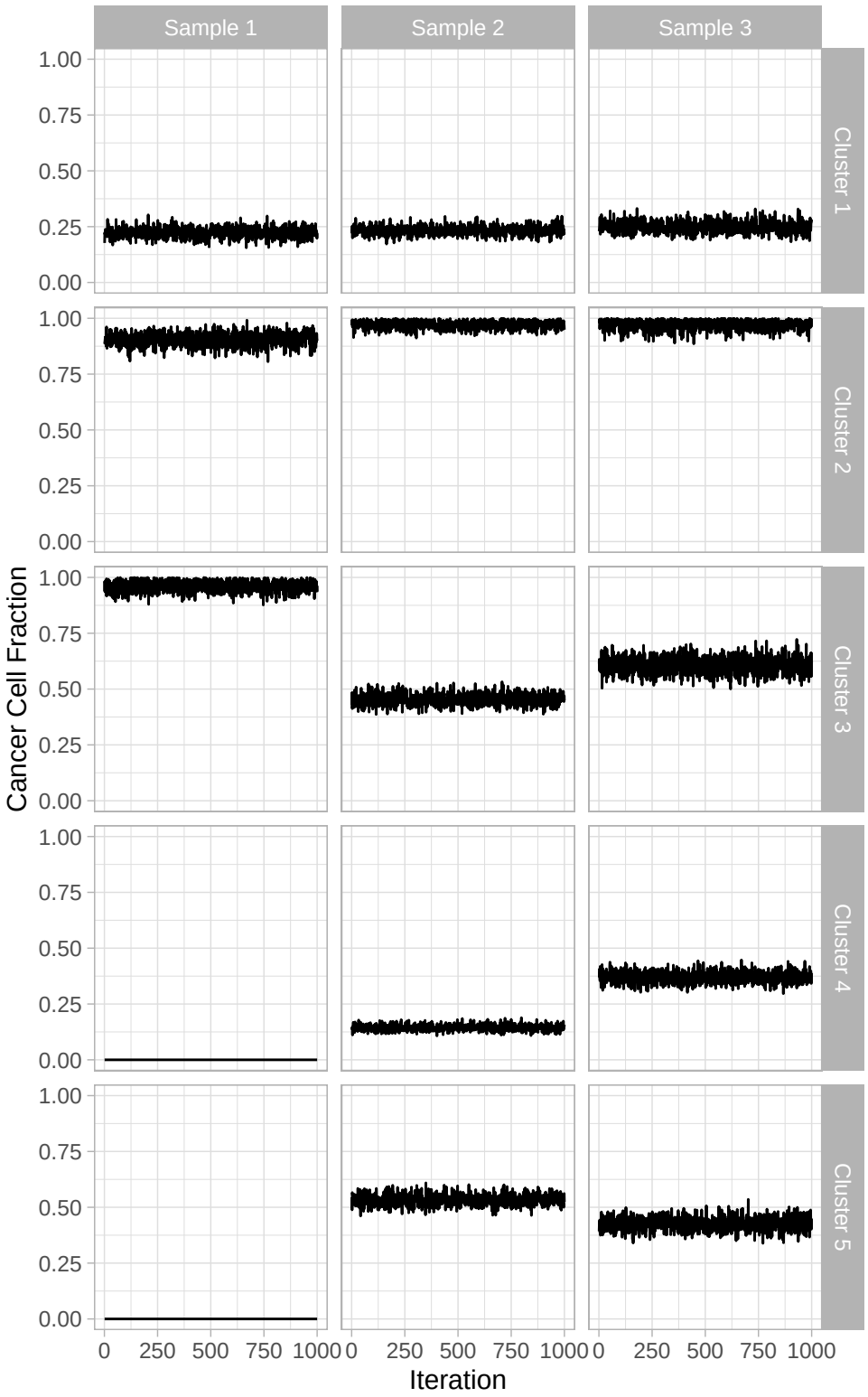



**Figure 2.3:** Example traces for MCMC chains of cluster cancer cell fractions.

#### 2.1.2.2 Model selection

Bayesian information criteria (BIC) [26] is calculated to assist with selection of the number of clusters, $K$. BIC gives an estimate of the model performance such that lower scores are generally preferred. BIC is calculated by the formula:

$$BIC = log(N) \times k - 2 \times loglikelihood \tag{2.1}$$

where $N = I \times S$ is the size of the dataset defined by the total number of observed variants $I$ across all samples $S$, and $k$ is the number of mutation cluster. Loglikelihood is calculated using the model described in Section 2.1.1.

PICTograph automatically stores MCMC chains for all $K$ assessed for all mutation sets, calculates the BIC, and selects the $K$ with minimum BIC as the best model. As a checkpoint, users can visually inspect the BIC plots (Figure 2.4) and confirm the selection of $K$ for each mutation set.

In some cases, several values may be relatively close in BIC or there may be multiple local minima. For these cases, the minimum BIC may not reflect the best choice of $K$. An alternative choice of $K$ is by identification of the elbow or knee point of the BIC plot, which is the point at which adding another cluster does not give much better modeling of the data (diminishing returns). I have implemented two methods for determining the elbow or knee point of a BIC

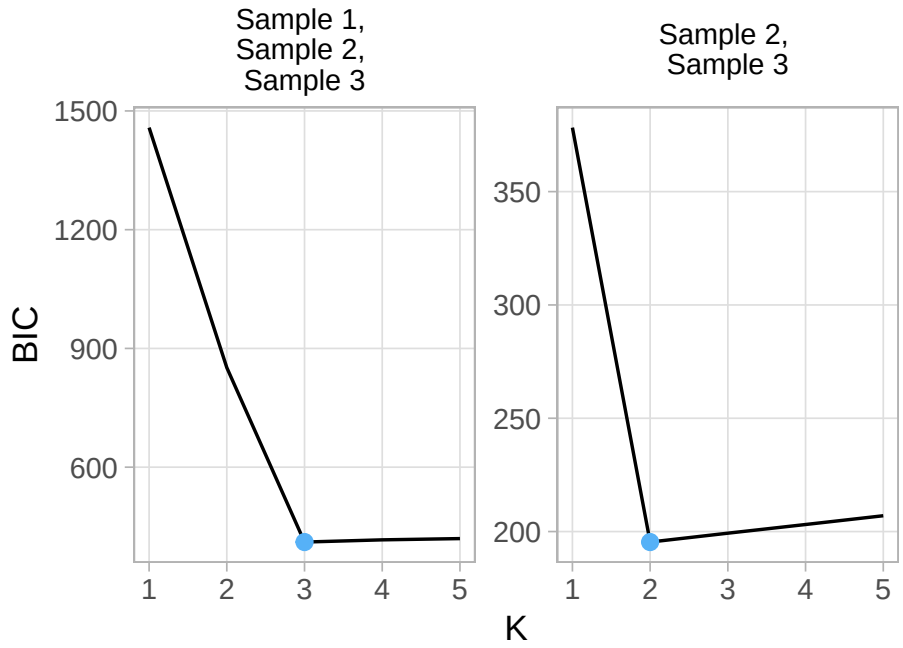



**Figure 2.4:** Example BIC plots for each mutation set. This toy example contains two mutation sets, for which models with 1 through 5 clusters were assessed. The blue dot marks the value with the lowest BIC, which is the default choice for the best value.

plot.

The elbow method originated as a heuristic to determine the number of clusters in K-means cluster models. After running K-means for a range of $K$'s, the sum of squares of the distances from the cluster mean (SSD) is calculated and plotted for all $K$ assessed. The elbow can be visually identified by looking for a "kink" in the SSD plot. One mathematical approach to defining the elbow point is to draw a line $L$ that connects the endpoints of the curve and identify the elbow as the point on the curve that has the greatest perpendicular distance to $L$. The distance from each point on the curve to line $L$ can be calculated as

follows. Let the line $L$ be defined by the endpoints $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$. The distance of a point $(x0, y0)$ from line $L$ is:

$$distance(P_1, P_2, (x_0, y_0)) = \frac{|(x_2 - x_1)(y_1 - y_0) - (x_1 - x_0)(y_2 - y_1)|}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \tag{2.2}$$

This same heuristic can be applied to PICTograph's BIC plots which assess a range of $K$ from 1 to $K_{max}$. Through this formulation, elbow of the BIC plot is defined as the point with the greatest distance to the line defined by its two endpoints $P_1 = (1, BIC_1)$ and $P_2 = (K_{max}, BIC_{K_{max}})$.

Another approach for determining the best choice of $K$ is an angle-based knee point detection method formulated in Zhao *et al.* 2008 [27]. Under this approach, a difference function is first calculated for each $K$:

$$Diff(k) = BIC_{k-1} + BIC_{k+1} - 2 \times BIC_k \tag{2.3}$$

Second, local maxima of the difference values are identified, and the angle of the BIC plot is calculated for each local maxima:

$$angle = atan(\frac{1}{|BIC_k - BIC_{k-1}|} + atan(\frac{1}{|BIC_{k+1} - BIC_k|}) \tag{2.4}$$

Finally, the first $K$ with a minimizing angle is identified as the knee point.

Choice by each method (minimum, elbow, or knee) can be viewed to help determine the best choice of K (Figure 2.5). To assist with model selection, we can inspect the posterior distributions of cluster assignments and CCFs (See section 2.1.2.3).

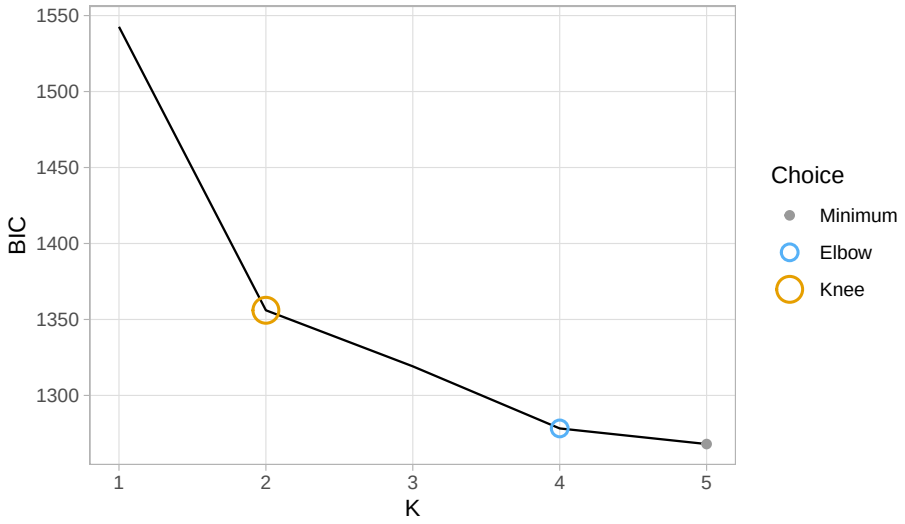



**Figure 2.5:** Example BIC plot for a mutation set with the choices of K marked for three different methods: minimum, elbow, and knee. This example illustrates a case where the minimum BIC, elbow point, and knee point are different values of K. Other cases may have agreement among methods.

#### 2.1.2.3 Posterior distributions of mutation cluster assignments and cluster cancer cell fractions

Since PICTograph's CCF model is fully Bayesian, posterior distributions of the CCFs and probabilistic estimates of cluster membership are available for all identified variants. PICTograph offers two plotting functions to visualize these posterior distributions.

To visualize the posterior distributions of cluster membership for all variants, PICTograph generates a scatter plot, where each point represents a variant and cluster-assignment pair and the point color reflects the magnitude of posterior probability (Figure 2.6). In the toy example, all mutations are assigned to a cluster with 100% certainty. In cases of more uncertain cluster assignment, the plot will show additional points (connected by a dotted line) to mark multiple cluster assignment.

To visualize posterior distributions of cluster CCFs, PICTograph generates violin plots (Figure 2.7). Within each violin, the black lines mark the median, and the white lines mark the first and third quantiles.

Inspecting posterior distributions can be useful for model selection. Particularly, having clusters with very similar CCFs across all samples and many mutations with similar posterior probability of assignment to clusters, is indicative of overestimating in the number of clusters.

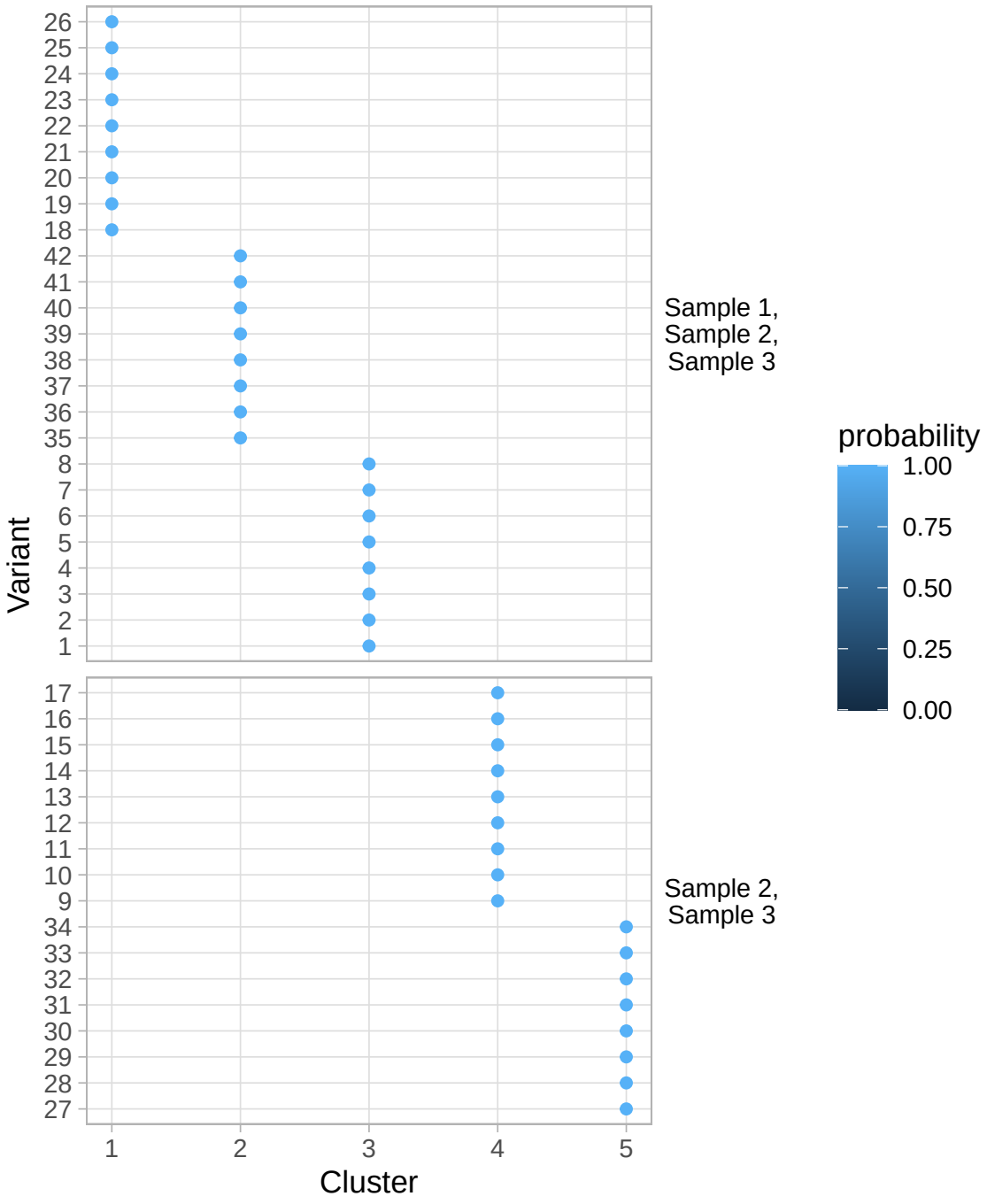



**Figure 2.6:** Example posterior probabilities of mutation cluster assignments.

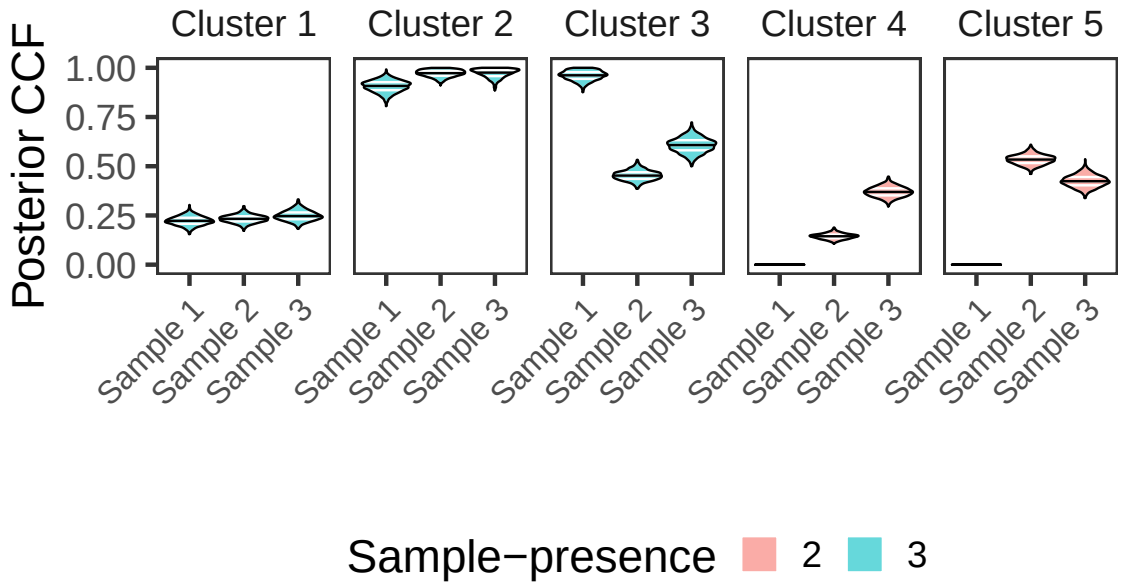



**Figure 2.7:** Example posterior distributions of cluster cancer cell fractions. Medians are marked by black lines and the first and third quantiles are marked by white lines. Colors reflect the number of samples in which the mutation cluster is present.

#### 2.1.2.4 Goodness of fit

Posterior predictive distributions available from PICTograph allow a formal assessment of the adequacy of the proposed model for capturing the heterogeneity of variant allele counts within and between samples. Posterior predictive distributions used to assess goodness-of-fit were obtained by simulating a random ordinate $y^*$ from a Binomial with $VAF_{is}^{(j)}$ for each variant $i$ and sample $s$ at each iteration $j$ of the MCMC.

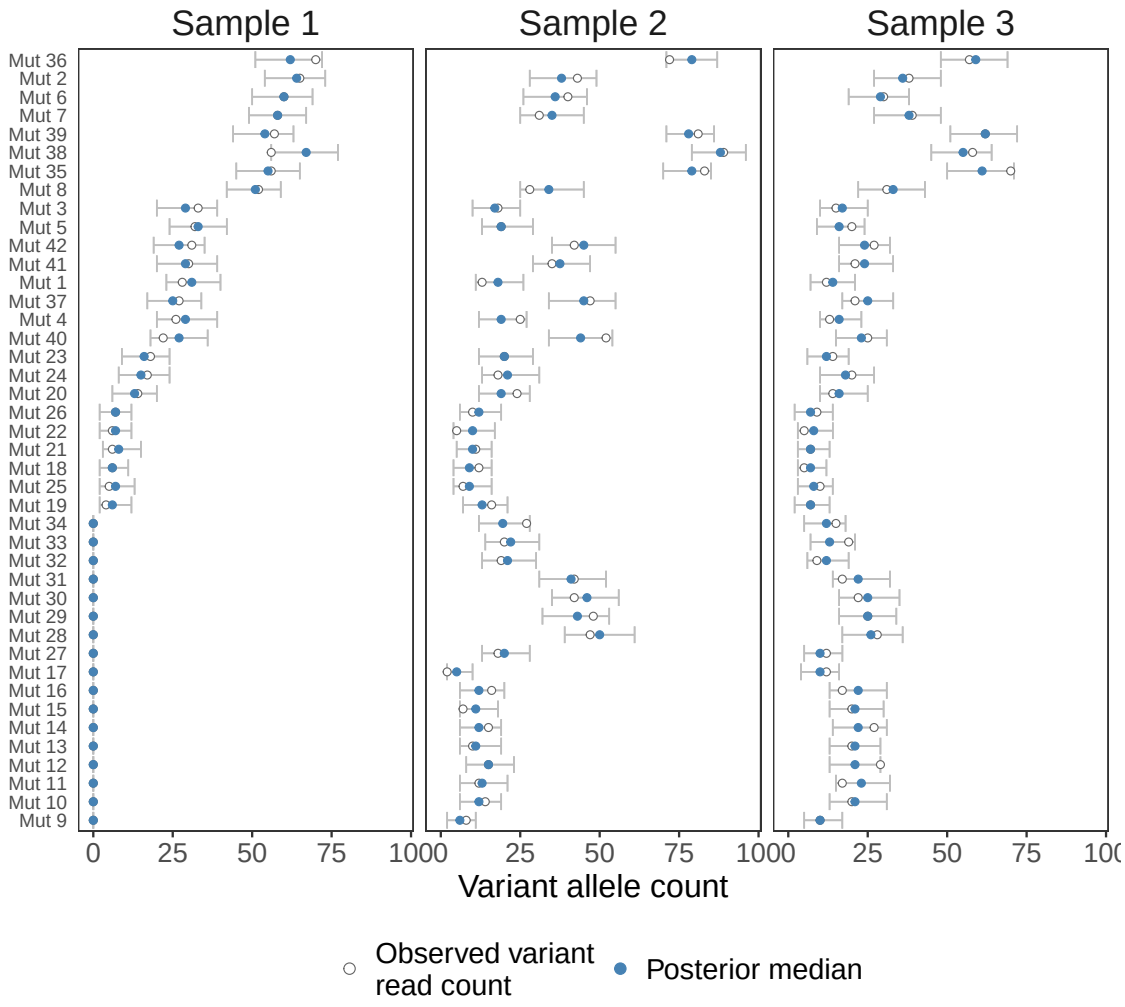



**Figure 2.8:** Example posterior predictive distribution of variant allele counts. The median of the posterior predictive distribution (blue point) and 95% credible intervals (gray segments) suggest that the generative model for allele counts in PICTograph provides a useful approximation of the data.

## 2.2 Tree inference

A tumor clone is a collection of genetically similar cells with a shared evolutionary origin. With cross-sectional whole exome sequencing data, we do not

observe the birth and death of clones as a tumor evolved and only observe indirectly the extant clones. To infer these latent processes, PICTograph constructs a rooted mutation tree that depicts the tumor clonal evolution from normal cells without somatic mutations (the root) to a cluster of mutations with the same CCF profile across samples (a node). With this representation of a rooted mutation tree, a tumor clone is the set of mutations along a path from the root to a node. While our Bayesian model provides a joint posterior distribution for $\{Z, CCF\}$, tree inference is computationally intensive and we limited our analysis to the maximum a posterior estimates of these parameters. Separation of CCF estimation from tree inference has the additional advantage of allowing plug-in estimates for these parameters from other methods.

PICTograph's tree inference algorithm determines the possible edges between mutation clusters, assembles these directed edges into acyclic graphs (evolutionary trees), and evaluates all candidate trees with a scoring function. The following subsections elaborate on the principles used to determine possible edges, the enumeration algorithm used to generate all candidate trees, and various visualization techniques for the results.

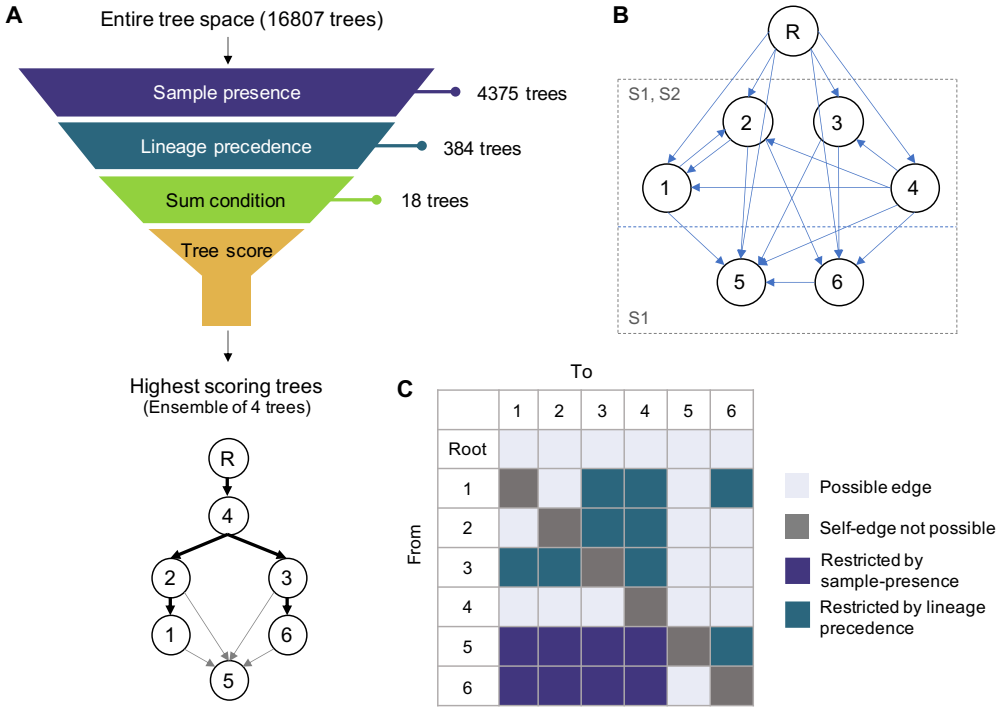



**Figure 2.9:** Overview of approach for tree inference. (A) Sample-presence and lineaage precedence rules are applied to the maximum a posteriori estimates of the cluster CCFs to determine the set of possible edges for evolutionary trees. From this set of edges, PICTograph uses the Gabow-Myers algorithm and the Sum Condition to identify spanning trees that are scored by the fitness function in SCHISM. The highest scoring trees are reported and summarized as an ensemble tree. (B) Mutation clusters (numbered circles)are shown in boxes representing their respective sample-presence sets. The root node is shown as a circle labeled R located above the sample-presence sets. Arrows represent the possible edges determined by applying filtering rules to pair-wise comparisons of CCF estimates of mutation clusters. (C) An adjacency matrix representation of possible edges and those restricted by sample-presence and lineage precedence filters. The restricted adjacency matrix reduces the number of possible trees from 16,807 to 4,375.

### 2.2.1 Restrictions of candidate evolutionary relationships

A key assumption used by many evolutionary methods is the infinite sites assumption [28]. Any cell that lost a mutation during evolution of the cancer by reverting back to the wild-type allele would violate this assumption. The critical role of the infinite sites assumption in determining the likely evolutionary relationships is shared by many of the existing methods, including SCHISM [9], PhyloWGS [10], PyClone [15], Canopy [12], LICHeE [11].

To identify candidate evolutionary relationships between mutation clusters, we identified all directed edges between mutation clusters that were consistent with ideas of sample-presence and lineage precedence. A mutation cluster is considered to be present in a sample if its CCF is at least 0.01. By sample-presence, the descendant cluster must be present in the same sample or a subset of samples in which the ancestral cluster is present. By lineage precedence, the CCF of the descendant cluster are at most $\epsilon_1$ greater than that of the ancestral cluster in each sample. Figure 2.9C shows edges restricted by sample-presence and lineage precedence filters in an adjacency matrix representation. Application of sample-presence and lineage precedence effectively decreases the number of possible trees (Figure 2.9A).

### 2.2.2 A modified enumeration algorithm

Conditional on the set of possible directed edges, we implemented the modified Gabow-Myers algorithm [29] following the pseudo-code of Popic *et al.* [11] to identify the collection of graphs where all the nodes are connected by a minimum number of edges (spanning trees). Trees identified from this algorithm where the sum of the CCFs of the descendants of a parent node exceeded the parent node's CCF by more than $\epsilon_2$ (Sum Condition) were excluded. The $\epsilon$ cutoffs for lineage precedence and Sum Condition were motivated by a desire to avoid eliminating trees near the decision boundaries and current defaults in PICTograph are 0.1 and 0.2, respectively. These thresholds can be lowered to decrease the tree space that is enumerated, or raised if no trees are found.

### 2.2.3 Selection of the most probable solutions

To determine the most probable solution, the set of candidate trees is scored using the scoring function presented in SCHISM [30], which evaluates the tree using the estimated CCF values of the mutation clusters. The highest scoring tree is determined the best solution. In some situations, several trees can share the same score, so all trees sharing the highest score are returned.

The tree fitness function from SCHISM is based on the principles of lineage precedence and lineage divergence [30]. The lineage precedence rule states

that a mutation at a node cannot have cancer cell fraction greater than those of mutations at its parental node [30]. The lineage divergence rule states that the sum of cancer cell fractions of mutations in child nodes cannot exceed the cancer cell fraction of their parent, because these mutations occur in mutually exclusive populations of cells.

Violations of lineage precedence and divergence are summarized by a topology cost and a mass cost, respectively. With $I \rightarrow J$ denoting cluster $I$ as an ancestor of cluster $J$, the topology cost for the edge connecting two mutation clusters $tc(I, J)$ is calculated from a binary *Precedence Order Violation* (*POV* matrix), where non-zero entries mark mutation pairs for which the null hypothesis $I \rightarrow J$ was rejected. The *Cluster Precedence Order Violation* (*CPOV* matrix) is a straightforward extension of the *POV* matrix in which the hypothesis test is applied to pairs of clusters rather than to pairs of mutations.

$$tc[I, J] = CPOV[I, J] = \frac{\sum_{i \in M(I), j \in M(J)} POV[i, j]}{|M(I)| \cdot |M(J)|} \tag{2.5}$$

where $M(X)$ is the set of mutations in cluster $X$ and $|M(X)|$ denotes the number of mutations in cluster $X$. The topology cost of the tree $TC(T)$ is then obtained by summing over all the connected clusters in the tree. Lineage divergence assumes that the sum of CCFs of nodes that are descendants, $D(n)$, of a parent node $p(n)$ can not exceed the CCF of the parent. While parents with descendant relationships that satisfy lineage divergence have no mass cost, the

cost of a violation is given by

$$mc^s(n) = \sum_{q \in D(n)} CCF_q^s - CCF_{p(n)}^s \tag{2.6}$$

The total mass cost of a tree, $MC(T)$, is obtained as the sum over all nodes in a tree $N(T)$,

$$MC(T) = \sum_{n \in N(T)} mc(n) \tag{2.7}$$

where $mc(n)$ is the Euclidean norm of the mass cost across samples, $mc(n) = \sqrt{\sum_{s=1}^{S}(mc^s(n))^2}$. Finally, the fitness of a tree, $F(T)$, is given by

$$F(T) = \exp(-f_x \times [TC(T) + MC(T)]) \tag{2.8}$$

The negative exponent ensures that the highest scoring tree are those with the lowest combined mass and topology costs.

### 2.2.4 Visualization of results

There are three main plots to assist with visualization of the tree inference results. First is a clone tree or ensemble tree in the case of multiple trees that are equally probable. Second is pie charts to visualize the proportion of each subclone in each sample. Third, in the case of longitudinal data, fish plots can be generated to show the proportions of subclones in each sample throughout

evolutionary time.

#### 2.2.4.1 Mutation tree and ensemble tree

Mutation trees are plotted with mutation clusters labeled at the nodes and arrows showing the direct evolutionary relationships between clusters (Figure 2.10A). In cases where multiple trees are tied for the highest score, PICTograph constructs an ensemble tree with edges weighted by their concordance among constituent trees in the ensemble. Visually, concordance was plotted on a gray scale ranging from black (present in all trees) to light gray (present in a subset of trees) (Figure 2.10B).

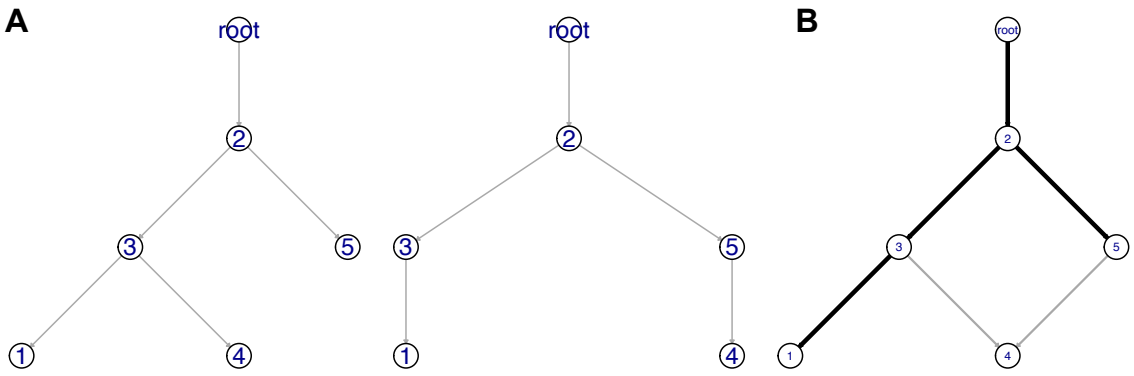



**Figure 2.10:** Example mutation trees (A) and corresponding ensemble tree (B).

#### 2.2.4.2 Subclone proportions

To generate pie charts that display the subclone proportions in each sample, we first must calculate these proportions by using the tree structure and CCF

estimates.

The proportion of tumor cells of the subclone described by node $n$ in a tree in each sample is computed as

$$max(min(CCF_n, CCF_A(n)) - \sum_{q \in D(n)} CCF_q, 0) \tag{2.9}$$

where $CCF_n$ is the CCF of the mutation cluster associated with node $n$, $CCF_A(n)$ is the CCF of the mutation cluster directly upstream of node $n$, and $\sum_{q \in D(n)} CCF_q$ is the sum of CCFs of mutation clusters directly downstream of node $n$. The minimum and maximum arguments are used to account for possible small violations in the sum condition and lineage precedence of the given tree.

Since the subclone proportions depend on the tree structure, when considering ensemble trees, we typically only calculate the proportions for the subclones containing mutation clusters with certain evolutionary relationships (e.g. make up the tree backbone). Alternatively, subclone sample proportions can be calculated for each tree.

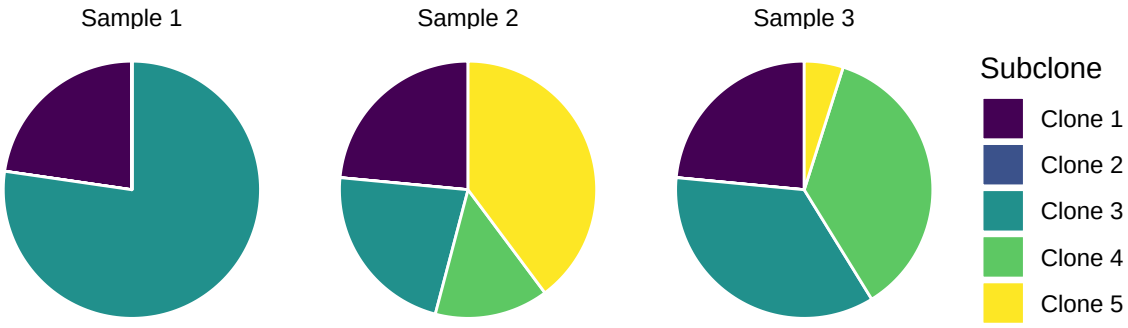



**Figure 2.11:** Example pie chart displaying the proportions of identified subclones present in each each sample.

## 2.3 Evaluation

To evaluate and benchmark PICTograph, we simulated clone trees capturing a broad range of complexity and implemented multiple currently available clone-tree methods. We simulate multi-region datasets across ranges of tumor purity, coverage, number of tumor samples, and subclonal diversity represented as the number of mutation clusters. PICTograph and several existing methods are applied to these simulated data. All methods are assessed by using metrics described in Satas and Raphael (2017) [13]: two metrics for evaluating accuracy of CCF estimation and one for evaluating accuracy of inferred evolutionary relationships.

### 2.3.1 Simulated dataset

Multi-region sequencing data was simulated using an approach implemented in the software SCHISM [9]. First, we generated a random tree with 5, 7, or 10 mutation clusters ($K$). Each mutation cluster was assigned a random number of mutations drawn from a Poisson distribution with a mean of 10. For simplicity, our simulations assume all variants occur on a diploid copy number background. The mutation cluster directly downstream of the root node was assigned a CCF of one, representing clonal mutations present in all tumor cells for a patient. Subsequent mutations representing the emergence of new subclones have CCFs simulated using a modified version of the tree-structured stick-breaking process model [9].

To emulate multi-region sequencing, CCFs were simulated independently for each sample for an individual. Given the tumor purity for sample $s$, $p_s$, a noisy variant allele frequency ($v_{is}$) for a mutation belonging to cluster $z$ was sampled from a beta distribution loosely centered at the true value for the mutation cluster ($VAF_{zs}$). Finally, simulated read counts ($y^*$) for mutation $i$ in sample $s$ were simulated from a binomial distribution parameterized by the coverage and $v$:

$$
\begin{aligned}
[y^*_{is}|Z_i = z] &\sim \text{Binomial}(v_{is}, \text{coverage}) \\
v_{is} &\sim \text{Beta}(\gamma \times VAF_{zs}, \gamma \times (1 - VAF_{zs})) \\
\gamma &\sim \text{Gamma}(10^3, 1) \\
VAF_{zs} &= \frac{CCF_{zs}p_s}{2 \times p_s + 2 \times (1 - p_s)}.
\end{aligned}
$$

We assume that variants with a positive CCF have one or more sequenced reads and that variants with a CCF of zero would have no reads as the probability of a sequencing artifact matching a bona fide variant in another sample would be approximately zero. Each simulated case was parameterized by the number of clusters ($K \in \{5, 7, 10\}$), tumor purity ($p_s \in \{0.5, 0.7, 0.9\}$), and sequencing depth (coverage $\in \{100, 150, 300\}$), and number of sequenced regions ($S \in \{2, 4, 6\}$) per patient). To characterize the stochasticity of model performance, we simulated 50 cases for each set of parameters varying only the seed of the random number generator for a total of 4,050 simulated patients.

### 2.3.2 Metrics

To evaluate method performance, we used the metrics published in Satas and Raphael (2017) [13]. Accuracy of the CCF estimates was measured using two complimentary metrics of divergence. Accuracy of the tree structure

was calculated as the proportion of correctly inferred ancestral relationships from all pairwise mutation comparisons for the top-ranked tree or the average accuracy in the event of multiple trees.

Approaches for estimating CCFs differ according to the mutations assigned to a cluster, the number of mutation clusters identified, and the average CCF estimated for each cluster. For a simulated patient with $L$ sequenced tumors and $K$ clusters, we obtain for each method a $L$ by $K^*$ matrix of estimated CCFs. To compare the CCF estimates to the true CCFs, we used previously described measures of divergence [13]. As these measures are not symmetric (divergence($x$, $y$) $\neq$ divergence($y$, $x$)), we computed divergence in both directions for each method $g$ as

$$\text{Metric}_{g,1} = \sum_{i=1}^{K} {}_{j \in \{1,\ldots,K^*\}} \delta(\boldsymbol{CCF}_i, \boldsymbol{CCF}_j^*) \tag{2.10}$$

$$\text{Metric}_{g,2} = \sum_{j=1}^{K^*} {}_{i \in \{1,\ldots,K\}} \delta(\boldsymbol{CCF}_i, \boldsymbol{CCF}_j^*) \tag{2.11}$$

$\text{Metric}_{g,1}$ provides a distance between estimated and true CCFs that penalizes underfitting ($K^*$ too small) but not overfitting ($K^*$ too large), whereas the opposite is true of $\text{Metric}_{g,2}$.

Each method's ability to recover the true tree is summarized as the proportion of mutations that were correctly placed in the best scoring tree generated

by each method [13]. A mutation pair $m_1, m_2$ may either have $m_1$ and $m_2$ in the same node, or $m_1$ may be ancestral to $m_2$, $m_2$ ancestral to $m_1$, or $m_1$ and $m_2$ may be on distinct branches of the tree. For all pairs of distinct mutations in each sample, we measure whether the reported relationship matched the relationship in the true tree.

### 2.3.3 Comparison to existing methods

Along with PICTograph, we assessed several existing evolutionary methods (Figure 1.1) using the simulated multi-region tumor sequencing data. Methods assessed include SCHISM [9], Canopy [12], PhyloWGS [10], sciClone [16], PyClone [15], and LICHeE [11]. These methods vary in which components of evolutionary analyses they perform. Like PICTograph, three of these methods, SCHISM, Canopy, and PhyloWGS, perform mutation clustering, CCF estimation, and tree inference. sciClone and Pyclone only perform mutation clustering and CCF estimation. LICHeE performs mutation clustering and tree inference, but operates at the level of variant allele frequencies and does not estimate CCFs of its identified clusters. We evaluated each method's ability to correctly identify the true number of mutation clusters, to estimate the true CCF, and to recover true ancestral relationships between clones decreased as the true number of simulated clusters increased or the tumor purity decreased.

For simulated datasets with 100x coverage and 4 samples per case, Canopy

had the lowest level of error for inferring the correct number of mutation clusters (Figure 2.12A), but CCF metric 2 suggests that Canopy's clusters were less consistent with the true values compared to PICTograph (Figure 2.12C). As PhyloWGS consistently under-estimated the number of true clusters in these simulations and performed poorly on subsequent metrics, we excluded PhyloWGS from further analyses. PICTograph, Canopy, and PyClone all exhibited robust performance by CCF metric 1 (Figure 2.12B). sciClone had much higher divergence for CCF metric 1, suggesting that the clusters identified by sciClone were not consistent with the true clusters (Figure 2.12B). We obtained qualitatively similar results at 150x (not shown) and 300x simulated coverage (Figure 2.13). Overall, PICTograph provided more accurate estimates of CCF and $K$ than other methods across a broad range of simulated subclonal complexity.

Next, we evaluated the ability of PICTograph, SCHISM, Canopy, PhyloWGS, and LICHeE to identify subclone trees. Ranking trees by their score (SCHISM, PICTograph, and LICHeE) or posterior probability (Canopy), we calculated accuracy as the proportion of correctly inferred ancestral relationships from all pairwise mutation comparisons Satas2017 for the top-ranked tree or the average accuracy in the event of tied trees. For simulated datasets with 100x coverage and 4 samples per case, we found that PICTograph had the highest average accuracy (mean = 83.2%, IQR = 61.8-99%). Accuracies were lower for SCHISM (mean = 73.4%, IQR = 25.3-98.1%), Canopy (mean = 73.6%, IQR =

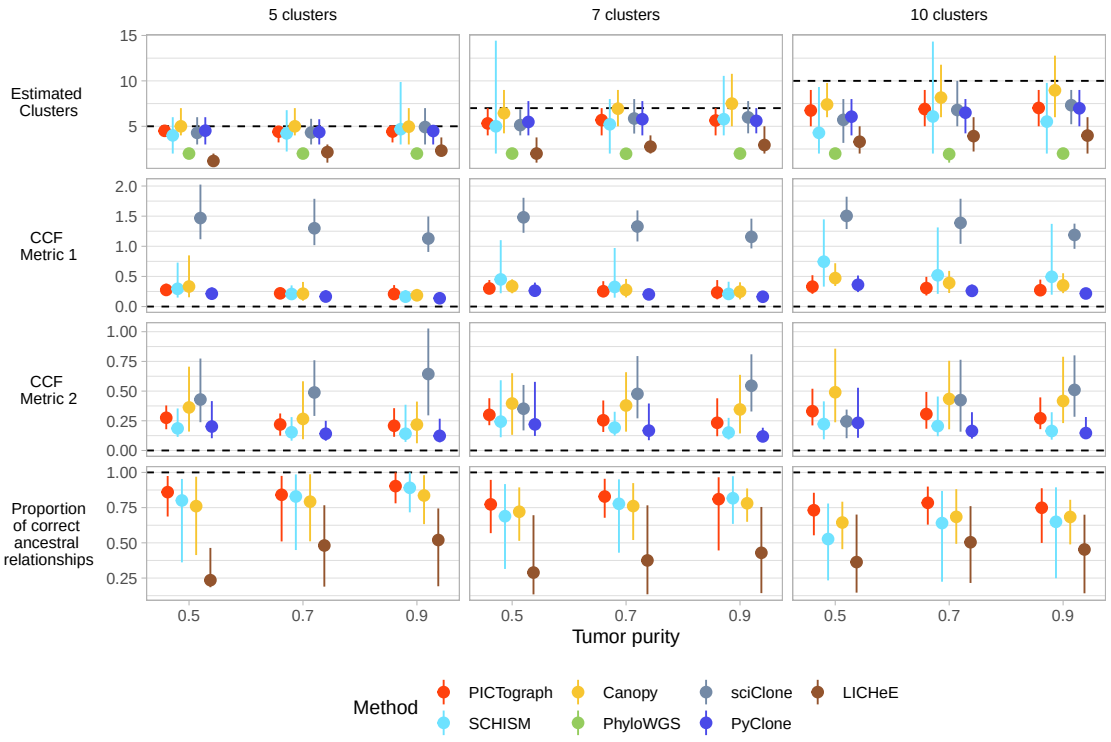



**Figure 2.12:** Simulations of four-region sequence data at 100x coverage. We simulated patients with four sequenced biopsies and an average sequencing depth of 100x for a range of mutation clusters (columns) and tumor purity (range 0.5 - 0.9). For each tumor purity and number of mutation clusters, we simulated 50 patients varying the seed for the random number generator. Performance assessments included estimation of the correct number of clusters, two asymmetric measures of divergence from the true CCF that penalize underfitting (CCF Metric 1) or overfitting (CCF Metric 2), and the proportion of correctly identified ancestral relationships. Dotted lines indicate the best possible score and vertical line segments connect the first and third quantiles (IQR). Collectively, these simulations indicate that PICTograph performs competitively across all the evaluated metrics with more stable performance as indicated by an IQR that is 2-3 fold narrower than competing methods.

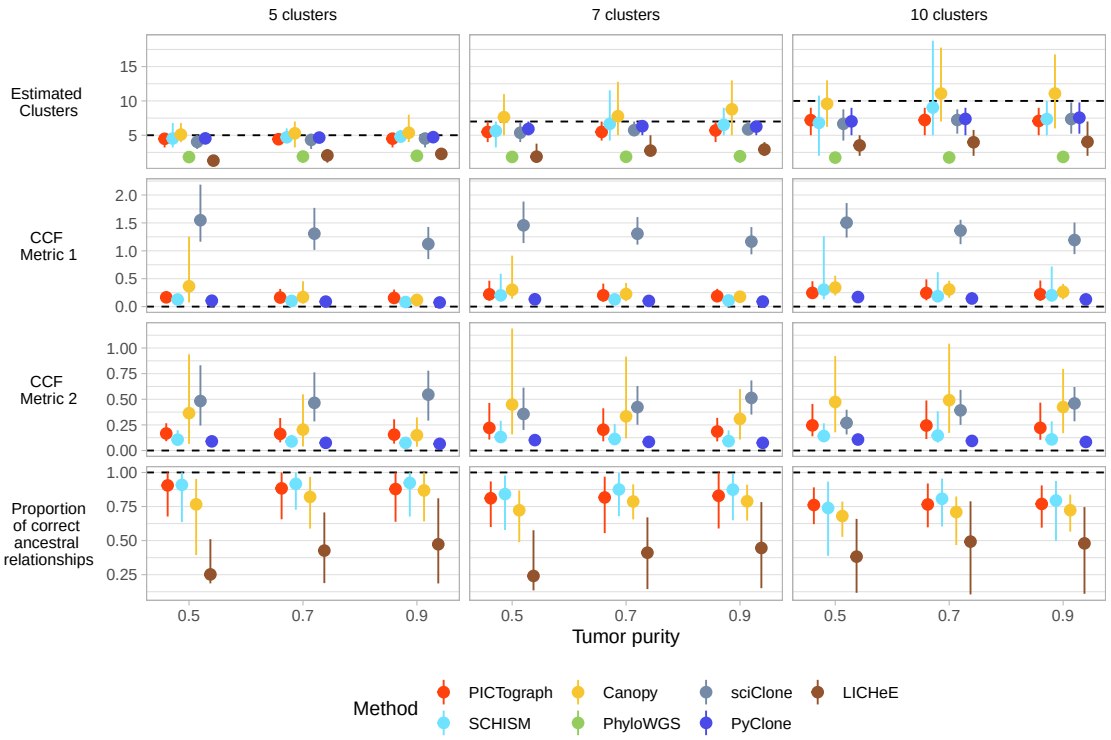



**Figure 2.13:** Simulations of four-region sequencing data at 300x coverage. We simulated patients with four sequenced biopsies and an average sequencing depth of 300x for a range of mutation clusters (columns) and tumor purity (range 0.5 - 0.9). For each tumor purity and number of mutation clusters, we simulated 50 patients varying the seed for the random number generator. Performance assessments included estimation of the correct number of clusters, two asymmetric measures of divergence from the true CCF that penalize underfitting (CCF Metric 1) or overfitting (CCF Metric 2), and the proportion of correctly identified ancestral relationships. Dotted lines indicate the best possible score. Collectively, these simulations indicate that PICTograph performs better than alternative methods particularly as the complexity of the simulated data increases with more mutation clusters.

49.2-95.5%), and LICHeE (mean = 40.3%, IQR = 13.7-74.7%) (Figure 2.12D). Additionally, we noticed more variability in the performance measures for all of the evaluated metrics for Canopy and SCHISM with the IQR often more than 2-fold that of PICTograph. Collectively, these simulations indicate that

recovery of ancestral relationships between subclones by PICTograph is robust to a wide range of mutation clusters and tree complexity.

### 2.3.4 Effects of dataset variables

Various properties of datasets can affect method performance, such as the number of samples, tumor purities of the samples, sequencing coverage, and subclonal diversity (e.g. number of mutation clusters or subclones). For all methods evaluated, the ability to correctly identify the true number of mutation clusters, to estimate the true CCF, and to recover true ancestral relationships between clones decreased as the true number of simulated clusters increased or the tumor purity decreased. As expected, increasing sequencing coverage also improved method performance.

### 2.3.5 Runtime

Wallclock times were recorded on a cluster node with Intel Xeon CPU ES-2470 and 128GB RAM (4 cores for clustering, 1 core for tree inference) for 4050 simulated patients (Section 2.3.1) where the true number of tumor subclones was known. The runtime of the CCF model implemented in PICTograph for these simulations scales linearly with the number of samples and subclones (Figure 2.14). Computational time for the tree inference implementation in

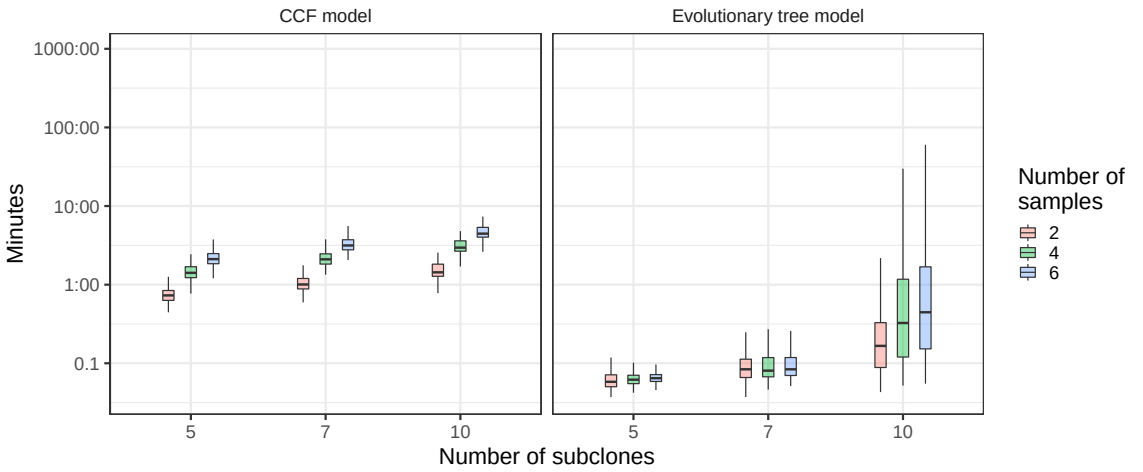



**Figure 2.14:** Computational time for PICTograph depends on the number of samples and subclones. Wallclock times for 4050 simulated patients where the true number of tumor subclones (x-axis) was known. For the CCF model, computational time increases linearly with both the number of subclones and number of samples (left). For tree inference, increasing the number of samples modestly increases computational time, and computational time varies by an order of magnitude depending on the number of tumor subclones (right).

PICTograph can grow exponentially with the number of clusters and depends largely on the extent to which sample-presence could be leveraged to narrow the tree space.

# Chapter 3

# Applications

I have applied PICTograph to two types of datasets: multi-region whole-exome sequencing of tumors and longitudinal whole-exome sequencing of immunotherapy treated cancers. The first application to multi-region whole-exome sequencing of tumors was published in Zheng *et al.* 2022 [25]. The second application is to an on-going study and has not yet been published.

## 3.1 Multi-region whole-exome sequencing of intraductal papillary mucionous neoplasms

Intraductal papillary mucinous neoplasms (IPMNs) are non-invasive precursor lesions that can progress to invasive pancreatic cancer. IPMns can be classified as low-grade or high-grade based on the morphology of the neoplastic epithelium. In order to identify molecular alterations underlying neoplastic progression, Fujikura *et al.* 2020 [20] whole-exome sequenced samples from 17 intraductal papillary mucinous neoplasms (IPMNs) with both low-grade and high-grade dysplasia. Fujikura et al. reconstructed the sample phylogeny for each case using Treeomics [31] to compare genetic alterations in low-grade and high-grade regions of the same IPMN. I have applied PICTograph to a subset of these cases, and will highlight three cases and compare evolutionary patterns found by PICTograph, Treeomics, SCHISM, and Canopy.

### 3.1.1 Analysis pipeline

#### 3.1.1.1 Sequencing data analysis

Fujikura et al. performed whole exome sequencing data analysis for several patients with IPMNs. From each IPMN, 2 to 6 regions were laser capture microdissected. Whole exome sequencing, alignment, and identification of SNVs were obtained as previously described [20]. Mutations were filtered based on coverage and frequency in the tumor and normal samples, and all non-coding and synonymous variants were removed as well as common germline variants found in databases including dbSNP, the 1000 Genomes Project, Exome Sequencing Project (6500), and Exome Aggregation Consortium (ExAC). Mutations were also validated by visual inspection in IGV. Somatic copy number variants (CNVs) were identified with CNVkit, version 0.9.6 [32] using the matched normal samples obtained from the patients as a reference set. To segment the copy ratio profiles, CNVkit's default segmentation method and thresholds were used, and samples with more than 1000 segments were re-segmented using a decreased threshold to reduce the risk of false positive CNV calls. Tumor purity was estimated using somatic mutations in likely copy neutral regions under the assumption that all samples were of at least 40% purity and had multiple clonal somatic mutations. These purity estimates were then used to calculate the integer tumor copy numbers for each segment. Multi-

plicity, $m$, was estimated by applying constraints such that $m \leq c_T$, with $c_T$ representing tumor copy number. $CCF$s were estimated by PICTograph as previously described. For IP22, we adjusted the estimate of allele-specific copy number for the GNAS mutation from 1 to 2 (out of a total copy number of 3) as a value of 1 for the allele-specific copy number resulted in an extra mutation cluster with CCF values that were incompatible with valid trees.

#### 3.1.1.2 Evolutionary analysis with PICTograph

Following the alignment and identification of somatic mutations in each sequenced lesion of a patient, PICTograph stratified the identified mutations into sets by sample-presence. Mutation clustering and CCF estimation was performed independently for each sample-presence set for a range of number of clusters $K$ from 1 to 10. Next, BIC was calculated for each $K$ assessed, and the $K$ with minimum BIC was chosen for each sample-presence set. Results for each sample-presence set were then merged. Best mutation cluster assignments were recorded by querying the cluster assignment for each mutation with the highest posterior probability. Point estimates for CCFs were calculated using the modes of the posterior distributions.

For tree inference, PICTograph used the point estimations for cluster CCFs calculated in the clustering step to determine possible edges. Lineage precedence threshold of 0.1 and sum condition threshold of 0.2 were used as a start-

ing point. Spanning trees were enumerated with the modified Gabow-Meyers algorithm. If no trees are found, the lineage precedence and sum condition thresholds were increased by 0.1, possible edges were re-determined, and enumeration was run again. Once one or more trees were found, the scoring function was applied and the highest scoring trees were returned. Subclone proportions were calculated for each sample in a case for either one specified tree or an ensemble tree's backbone (edges agreed upon by all highest-scoring trees).

#### 3.1.1.3 Evolutionary analysis with other clone tree methods

SCHISM (v1.1.3) was run using the $k$-means algorithm for clustering with 10 random initializations each having a minimum and maximum cluster count of 1 and 20, respectively. We used the default genetic algorithm parameters supplied in the SCHISM's usage example. Canopy was run under settings that do not incorporate copy number alterations as our simulations were limited to single nucleotide variants. For each simulated patient, we ran Canopy with 10 Markov chains using random starts with possible $K^*$ ranges of 2-10, 2-12, or 5-15, for true $K$ of 5, 7, and 10, respectively.

### 3.1.2 Results

To illustrate the range of tumor complexity found in real-world applications, we analyzed the clonal evolution of intraductal papillary mucinous neoplasms for three patients: IP29, IP22, and IP09. Along with PICTograph, we also applied SCHISM and Canopy for comparison.

#### 3.1.2.1 IP29

Patient IP29 contained two samples comprised of one low-grade and one high-grade dysplasia with a total of 49 somatic mutations (Figure 3.1A). PICTograph identified 6 mutation clusters (Figure 3.1A,B) and two equally probable clone trees (Figure 3.2B). Interestingly, three of the clusters (2, 3, and 5) contain no known drivers. While the majority of variants were classified into a single mutation cluster with posterior probability near 1, several variants could be assigned to two clusters with positive posterior probability (Figure 3.1C).

Patient IP29 had early mutations in *KRAS* followed by mutations in genes including *SF3B1* and *BSN* leading to clonal expansion of one branch, and mutations in *GNAS* and *RNF43* leading to clonal expansion in a different branch (Figure 3.2B-E). We note that there was some uncertainty as to the timing of the *BSN* mutation (cluster 4 or cluster 3), and similarly whether *RNF43* may have been acquired later (cluster 5 instead of cluster 6) (Figure 3.1C and Fig-

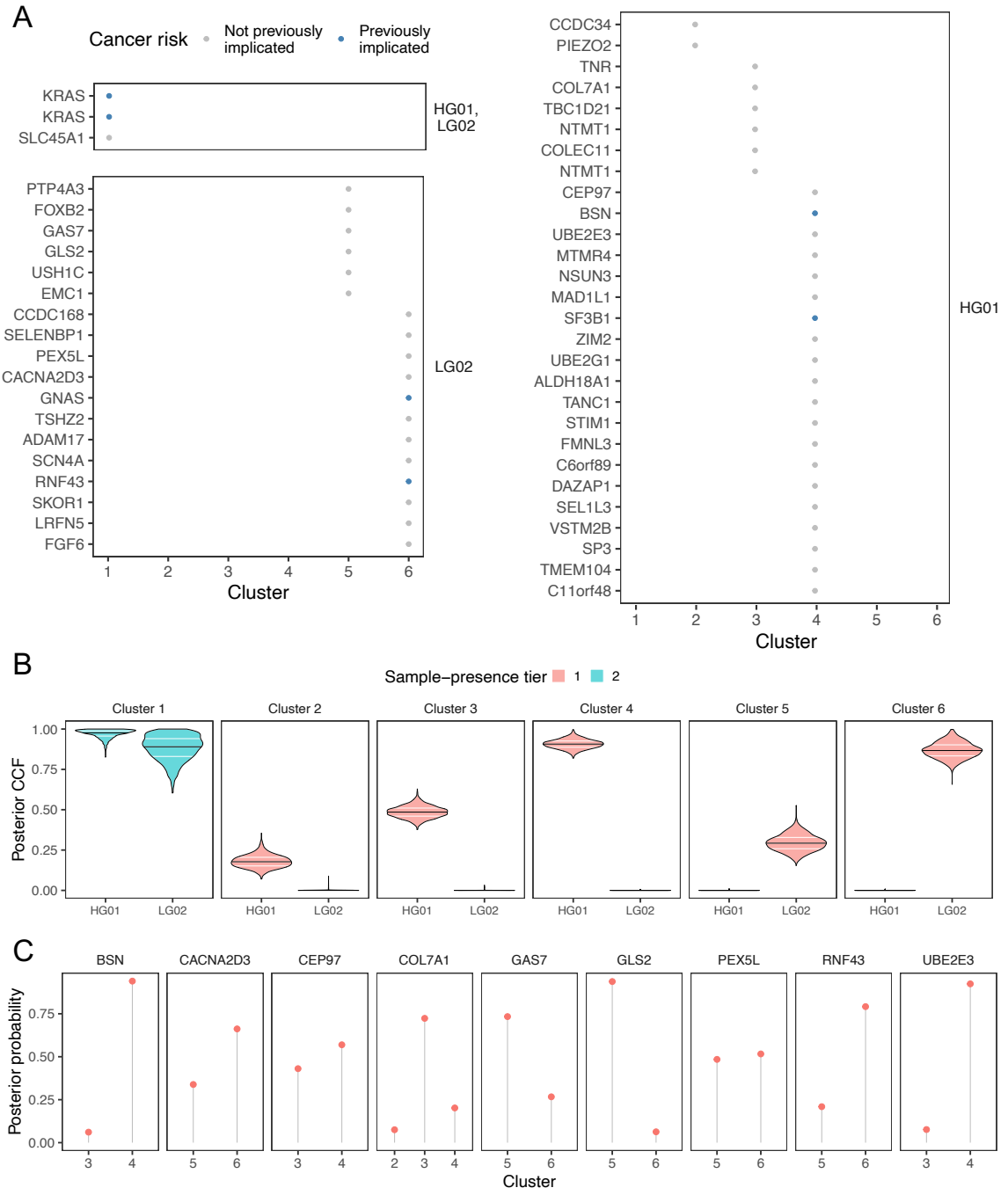



**Figure 3.1:** Mutation clustering and CCF estimation by PICTograph for patient IP29. (A) Cluster assignments of mutations with the highest probability. Mutations are labeled by the gene in which they occur. Blue marks known driver genes. (B) Posterior distributions of cancer cell fractions. (C) Probability of cluster assignments for mutations with higher uncertainty.

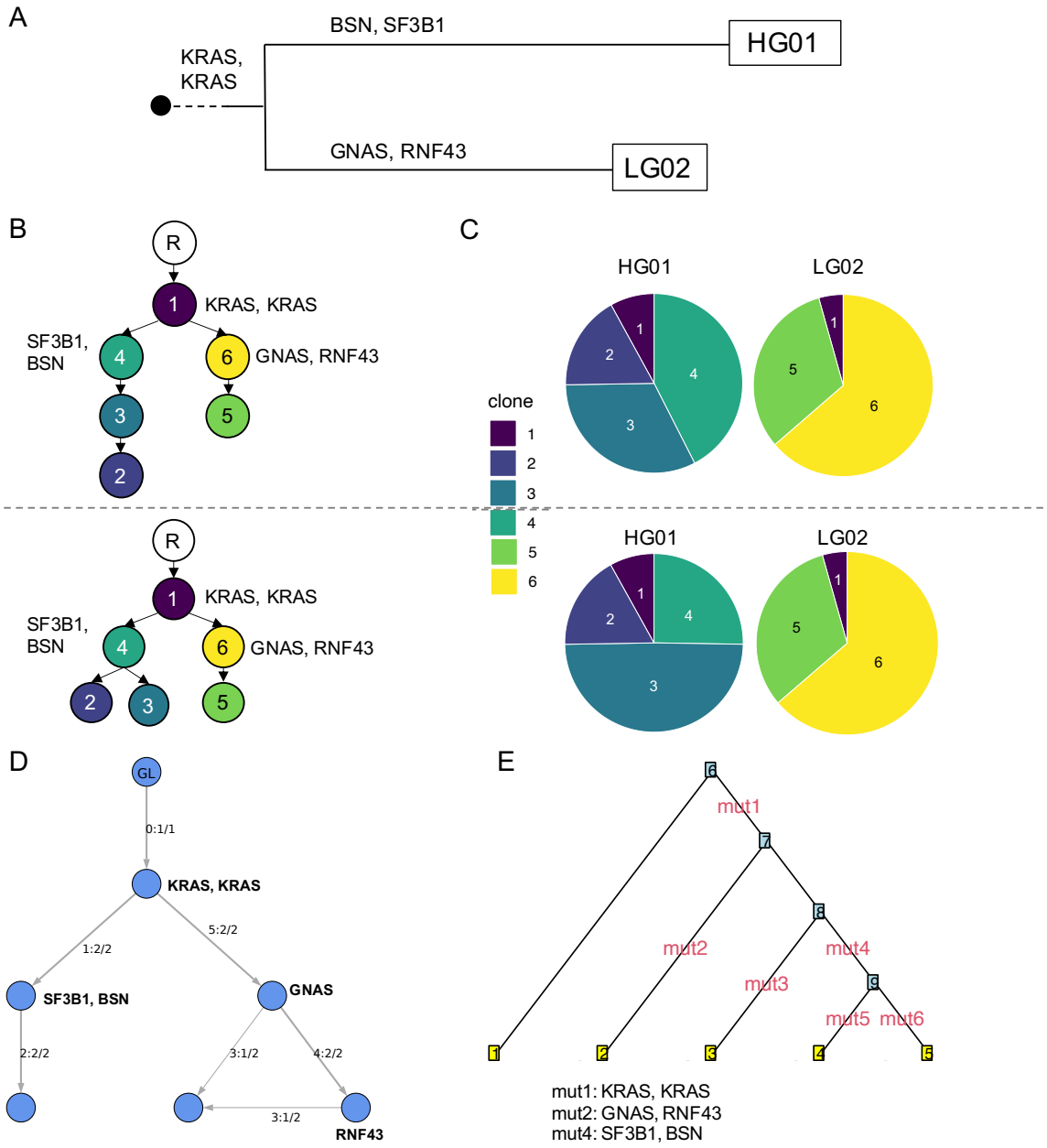



**Figure 3.2:** Evolutionary models for patient IP29. (A) Sample phylogeny inferred by Treeomics. (B) Mutation trees inferred by PICTograph. (C) Pie charts showing proportion of subclones in each sample by PICTograph. (D) Mutation tree inferred by SCHISM. (E) Clone tree inferred by Canopy.

ure 3.2B). PICTograph has a much higher posterior probability that *BSN* and *SF3B1* belong to cluster 4 versus cluster 3, and this ordering is consistent with our a priori belief that the emergence of this branch would be attributable to these known drivers. From this point, whether subclone 2 evolved from a single cell in subclone 3 (Figure 3.2B, top) or subclones 2 and 3 evolved from two distinct cells in subclone 4 (Figure 3.2B, bottom) can not be determined from the available data and are displayed graphically by PICTograph as two equally probable clone trees. The tree produced by SCHISM is structurally more similar to clone tree 1, showing linear evolution following the cluster with *SF3B1* an *BSN* (Figure 3.2D). In contrast, the tree produced by Canopy is most similar to clone tree 2, showing branching evolution following the cluster with *SF3B1* an *BSN* (Figure 3.2E). The cluster assignments of the mutations occurring in driver genes (*KRAS*, *SF3B1*, *BSN*, *GNAS*, *RNF43*) are identical for PICTograph and canopy, with SCHISM only differing in mutations in *GNAS* an *RNF43* being assigned to separate clusters. Overall, these three methods agree on the ordering of these mutations, with the two *KRAS* mutations occuring in the originating clone, one branch containing *SF3B1* and *BSN*, and another branch containing *GNAS* and *RNF43*.

#### 3.1.2.2 IP22

For Patient IP22, two low-grade and one high-grade dysplasia lesions were sequenced and a total of 83 somatic mutations were detected (Figure 3.3), including two distinct mutations of *CDKN2A*. PICTograph identified 10 mutation clusters (Figure 3.3 and Figure 3.4A). While the majority of variants were assigned to a single mutation cluster with posterior probability near 1, several variants could be assigned to two clusters with positive posterior probability including *SLC9A4* that had nearly equal posterior probabilities for clusters 8 and 9 (Figure 3.4B).

PICTograph identified 6 equally probable clone trees summarized by an ensemble tree (Figure 3.5B). All 6 trees have the same ordering for 8 of the 10 mutation clusters and have consensus that the *KRAS* and *GNAS* variants occurred very early (consistent with existing models for pancreatic cancer progression [33]) and that the two *CDKN2A* mutations (a single base C>T substitution at position 172 in cluster 2 and a deletion of 10 bp at position 300 in cluster 6) occurred on separate branches. The 10 bp *CDKN2A* deletion co-occurred with a variant in *KLF4*, a gene known to regulate *TP53* [20], while the single base substitution in *CDKN2A* occurred in a lineage that subsequently acquired a mutation in *RNF43*, a gene known to regulate Wnt-signalling [21]. As further corroboration that separate lineages acquired the *CDKN2A* mutations, the DNA copy number for the *CDKN2A* locus was one for this patient

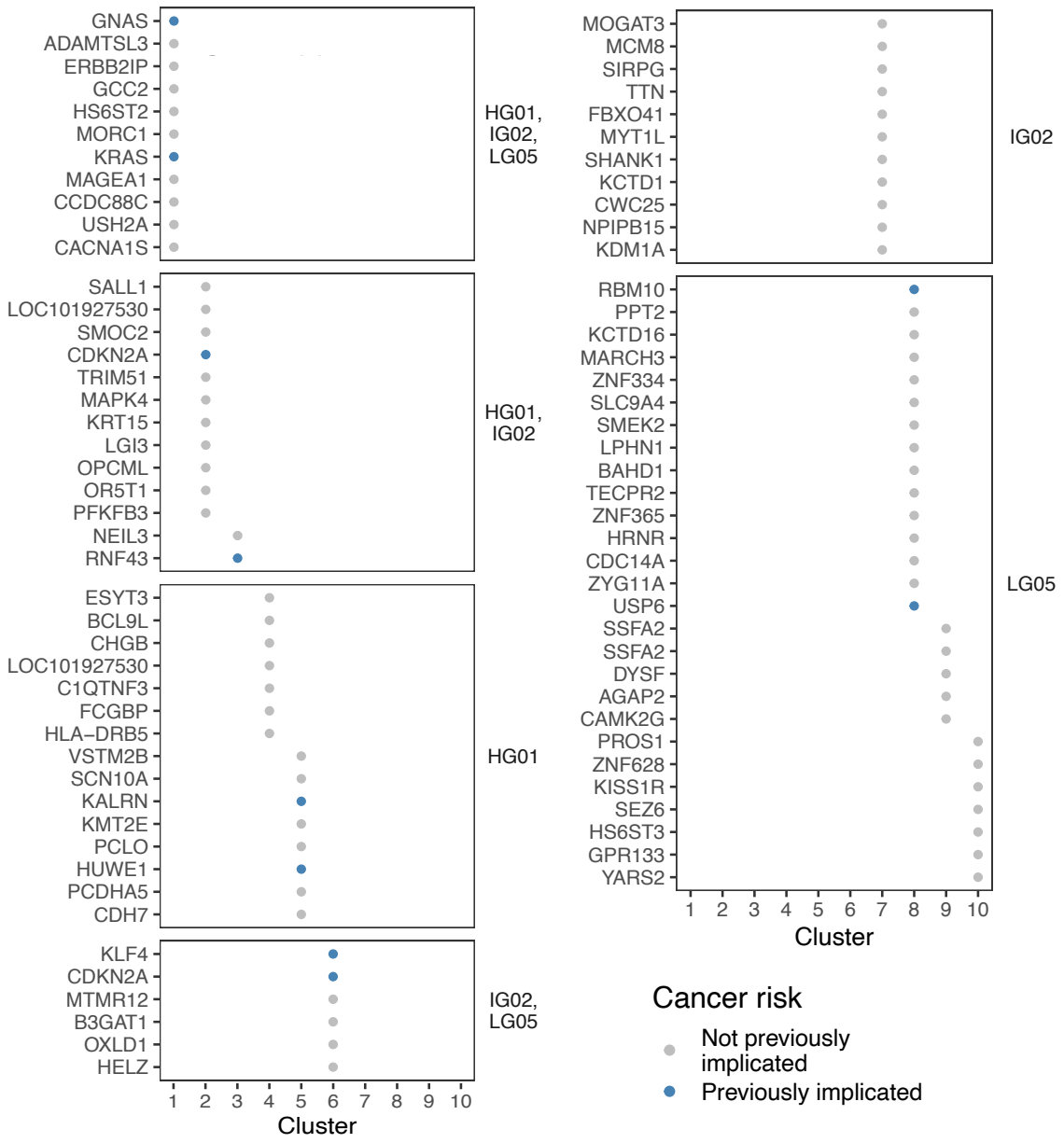



**Figure 3.3:** Most likely mutation cluster assignments for patient IP22. PICTograph identified ten distinct CCF clusters across the three samples available for patient IP22.

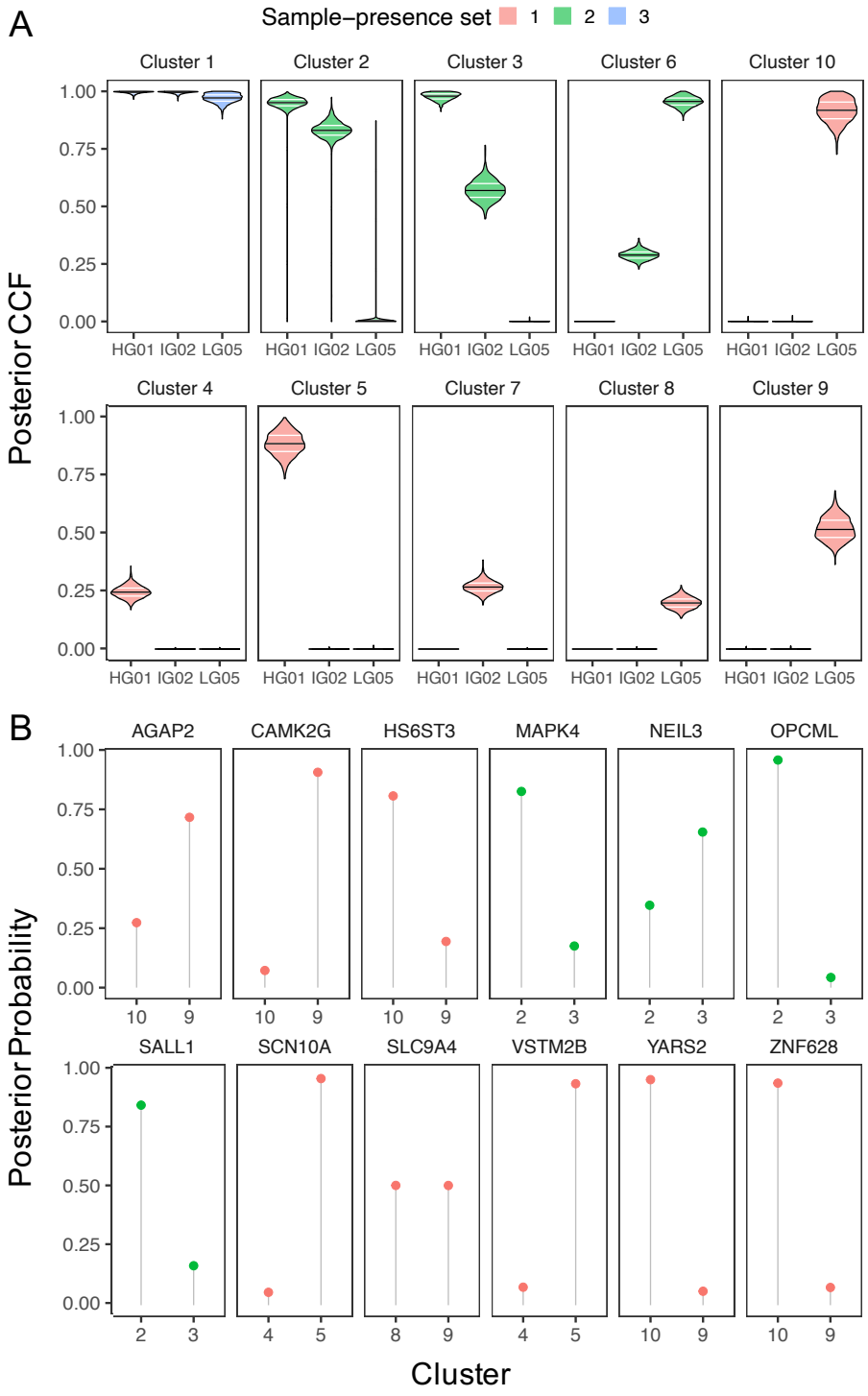



**Figure 3.4:** Mutation clustering and CCF estimation for patient IP22. (A) Posterior distributions of CCFs for 10 identified clusters across the three samples available for patient IP22. (B) 12 variants could be assigned to two clusters with positive posterior probability.

and there were no reads containing both the 10 bp deletion and the single base substitution [20]. Sample LG02 contains a mixture of both of the *CDKN2A* lineages: 75% of one lineage (subclones 2 and 3) and 25% of the other (subclone 6) (Figure 3.5C).

The tree found by PICTograph differed from the sample-tree found by Treeomics and was similar to the trees found by other clone tree methods. As sample-tree methods assume that the samples capture a single subclone, the sample tree analysis of this data incorrectly suggests a linear evolutionary relationship between the two *CDKN2A* mutations (Figure 3.5A). PICTograph was more similar to the other clone-tree methods, SCHISM and Canopy (Figure 3.5D-E), that were evaluated for this patient, but PICTograph highlights the uncertainty of the cluster assignments. While 71 of the 83 somatic mutations were assigned to a single cluster with probability near 1, 12 variants had appreciable posterior probabilities for a second mutation cluster implying that the relative timing of these mutations and their ancestral relationships were also uncertain.

#### 3.1.2.3 IP09

Two high-grade and two low-grade dysplasia lesions were microdissected from patient IP09 and 155 somatic mutations were identified through whole exome sequencing. PICTograph identified 11 distinct mutation clusters across the 4 available samples (Figure 3.6). Using the modal CCF to determine presence-

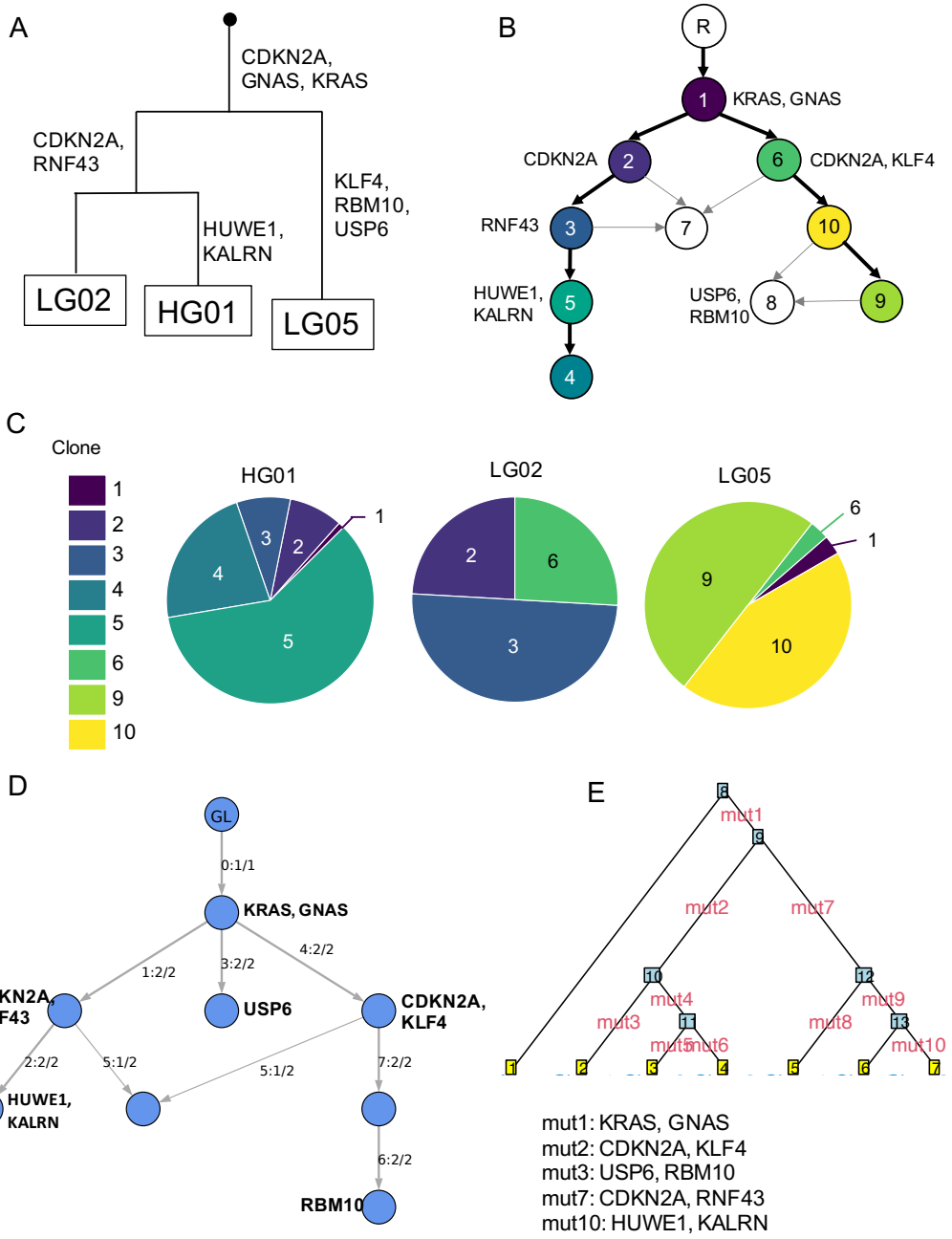



**Figure 3.5:** Sample tree and clone trees for patient IP22. (A) Sample-tree obtained from Treeomics. (B) Ensemble tree for the six top-scoring trees identified by PICTograph. (C) Pie charts displaying the relative proportions of each subclone found by PICTograph in the available lesions. (D) Consensus tree for SCHISM. Edges are labeled by x:y/z, where x is the mutation cluster number, y is the number of trees with the edge, and z is the total number of top-scoring trees. (E) Single tree with the highest posterior probability found by Canopy.

absence sets resulted in one cluster of mutations with variant alleles detected in all four samples and one cluster with mutations present in two samples. The remaining 9 clusters contain mutations present in only a single sample. The cluster assignment was uncertain for many of the variants (Figure 3.7), particularly for variants in clusters 5-7 involving genes such as *ASTN2*, *ALOX5*, and *PCDHA3*.

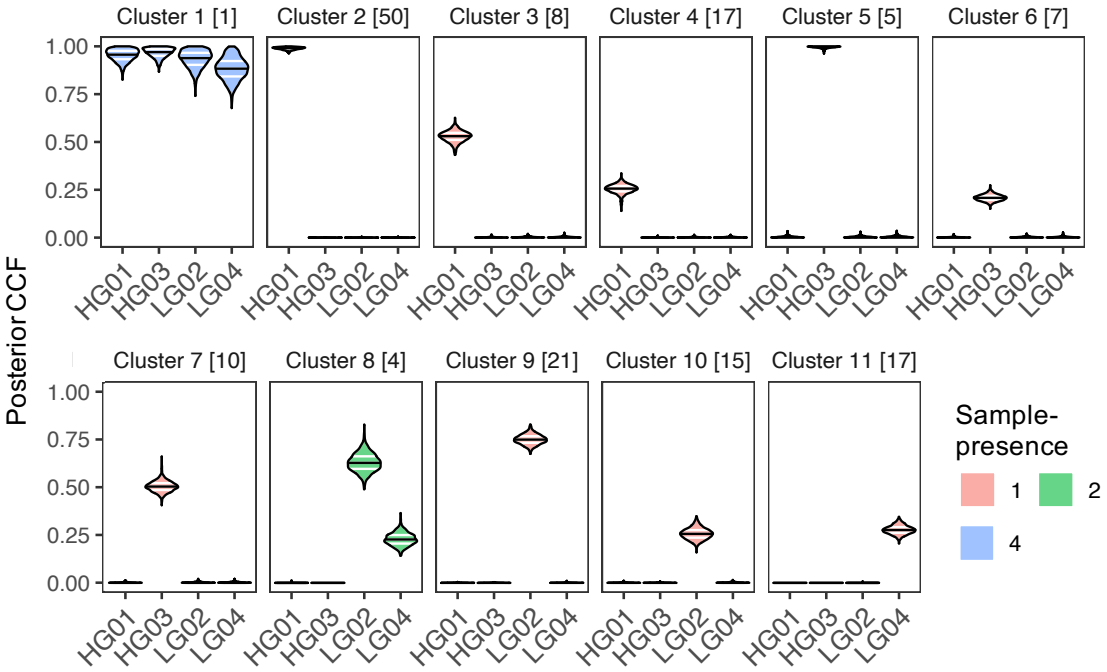



**Figure 3.6:** Posterior distributions of CCFs found by PICTograph for IP09. The number of mutations assigned to each cluster is listed in brackets.

PICTograph identified four equally probable clone trees (Figure 3.8B) that all identified subclone 1 containing the driver mutation *KRAS* G12D as the truncal clone, which then led to clonal expansion branching into 4 main subclones. Samples HG01, HG03, and LG02 are each comprised of a single lin-

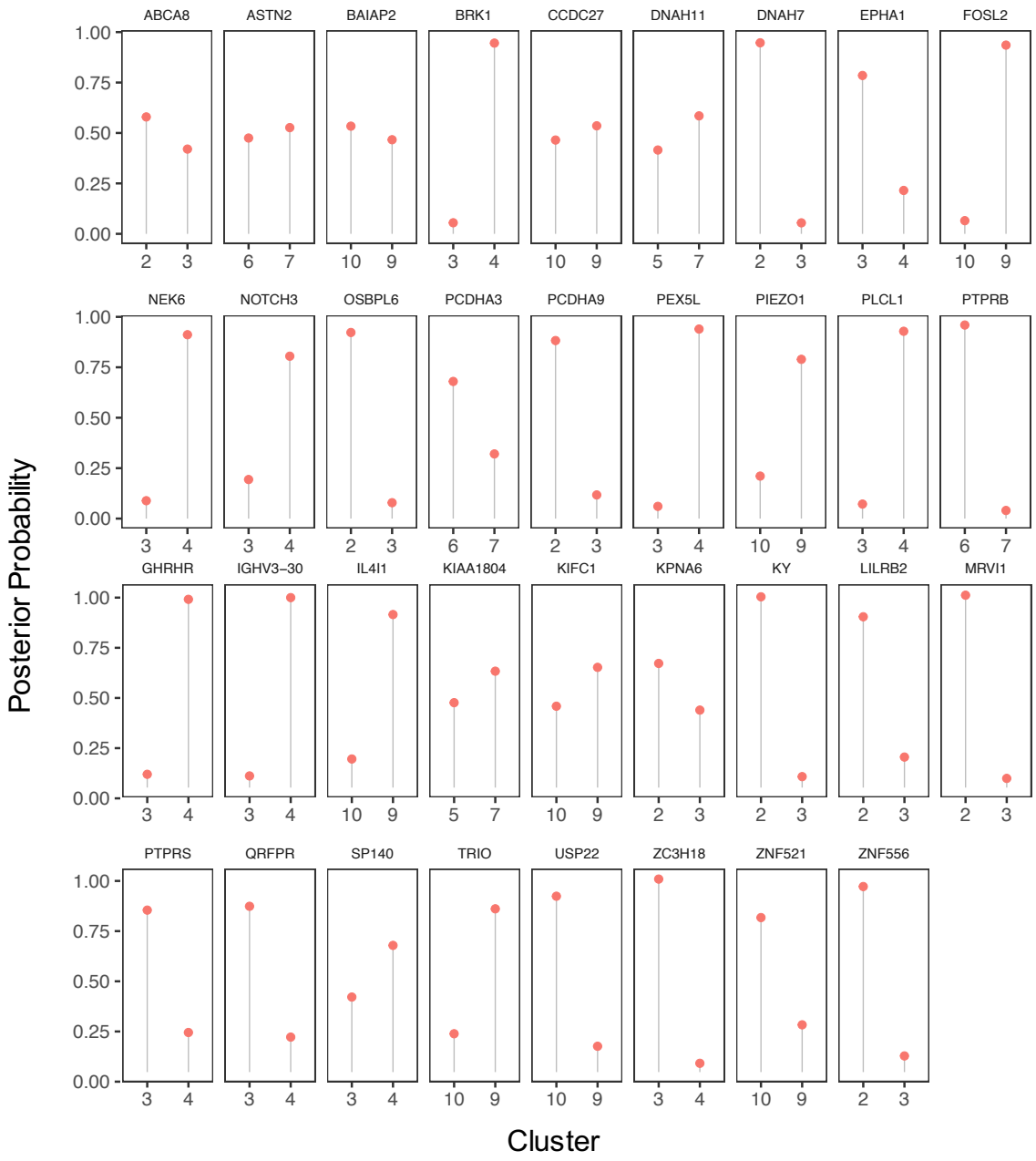



**Figure 3.7:** Mutations with uncertain cluster assignments found by PICTograph for IP09.

eage, while LG04 is a mix of two lineages (Figure 3.8C). The diversity of driver mutations in these separate lineages highlight the mutational complexity of

pancreatic cancer precursor lesions.

Additional subclones originated from cells acquiring mutations in *TP53* and *CDKN2A*, *ARID1A*, and *GNAS*/*KMT2C* and are broadly supported by all three evolutionary models (D-F). *ASTN2* is inferred to be an early driver event by SCHISM and is clustered with *KRAS* G12. Canopy places *ASTN2* downstream of *KRAS* and on a separate branch from *ARID1A*. While *ASTN2* had similar posterior probability for membership in cluster 6 and 7, there was strong support for *ASTN2* belonging to a subclone with *ARID1A* as the parent in PICTograph. This ordering is also consistent with the involvement of the tumor suppressor *ARID1A* in chromatin remodeling, while the role of *ASTN2* as a driver of cancer is less clear. The large number of distinct clonal populations emerging from the initial *KRAS* subclone underscore the difficulty in treating pancreatic cancer as the molecular diversity could be beneficial for mounting resistance to treatment.

### 3.1.3 Discussion

Overall, our analyses captured known early driver mutations in IPMN tumorigenesis and placed known oncogenic driver *KRAS* as the earliest truncal mutation, followed by *GNAS* and tumor suppressor gene alterations in *CDKN2A*, *RNF43*, *TP53*, and *ARID1A*. Our analyses also showed mixing of different clonal lineages in 2 of the 3 analyzed cases, highlighting the utility

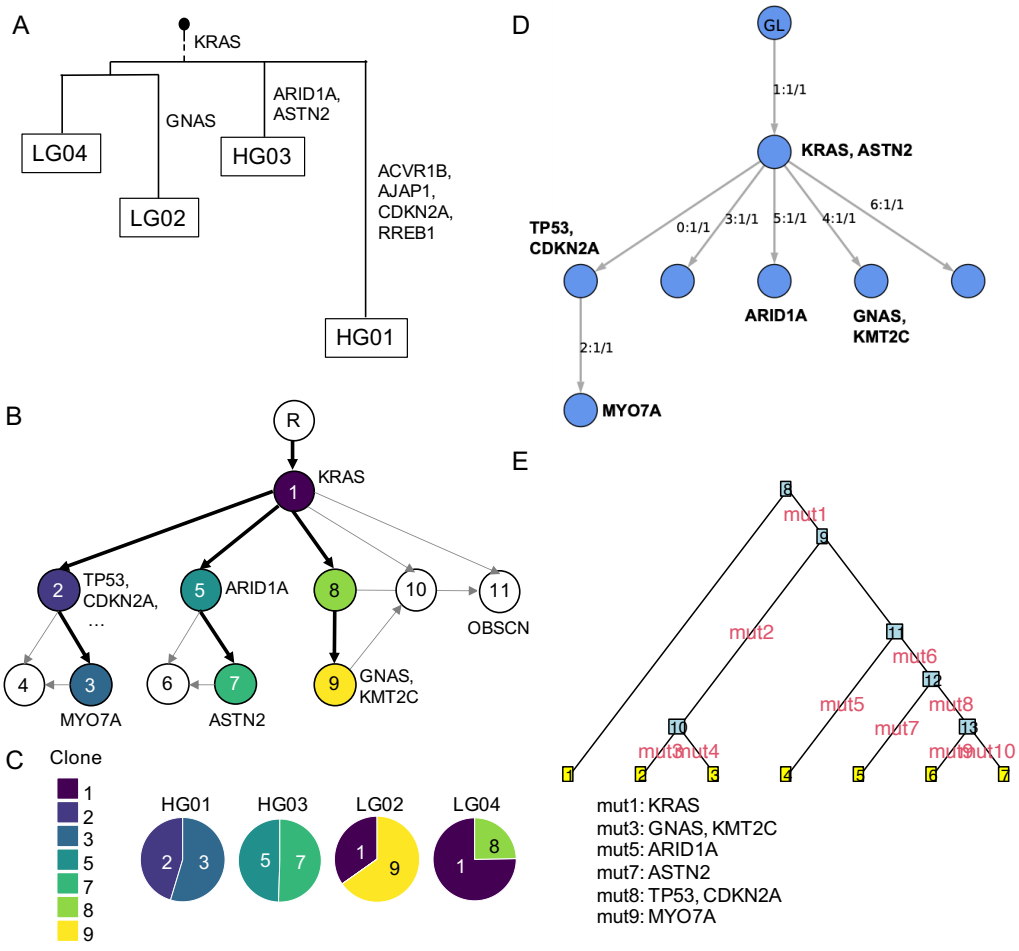



**Figure 3.8:** Evolutionary models for patient IP09. (A) Sample phylogeny inferred by Treeomics. (B) Ensemble tree inferred by PICTograph. (C) Pie charts showing proportion of subclones in each sample by PICTograph. (D) Mutation tree inferred by SCHISM. (E) Clone tree inferred by Canopy.

of a clone-tree based approach. Our ensemble-based visualizations of the top-scoring trees revealed consistent evolutionary relationships and different possible evolutionary paths to subclones that are equally likely given the observed data.

Our application of PICTograph to pancreatic cancer precursor lesions is motivated by previous studies indicating that pancreatic cancers are heteroge-

neous at earlier stages [21, 31, 34, 35]. Consistent with these studies, our analyses highlight two patients where different clonal lineages were identified by PICTograph from multi-region sequencing. The absence of mixing of subclonal lineages for the sequenced samples in the third patient may reflect the careful microdissection of the samples sequenced or a more homogeneous composition of the tumor. We anticipate that the application of clone-tree methods such as PICTograph for evolutionary inference will be especially critical for studies employing bulk tissue sequencing where clonal diversity of the bulk sample is likely to be substantial.

## 3.2 Longitudinal whole-exome sequencing of immunotherapy treated non-small cell lung cancer

As part of its normal function, the immune system detects and destroys abnormal cells to help the body fight infections and other diseases. This most likely prevents or curbs the growth of many cancers. Checkpoints such as the PD-1 Checkpoint are a normal part of the immune system and keep immune responses from being too strong. However, when tumors exploit the PD-1/PD-L1–mediated negative feedback mechanism, this results in evasion of

immune detection and elimination. Immunotherapies that block checkpoints essentially allow immune cells to respond more strongly to cancer.

The clinical trial J1414 studies the safety, feasibility, and tumor response when giving nivolumab to patients with resectable high-risk non-small cell lung cancer (NSCLC) in the pre-operative setting. Nivolumab is a anti PD-1 monoclonal antibody that blocks the PD-1/PD-L1–mediated negative feedback mechanism, thus activating an antitumor immune response. Ultimately, it is highly desirable to discover prospective biomarkers of response and toxicity to allow patients with NSCLC who are most likely to derive benefit to receive anti-PD-1 treatment, and conversely to minimize the risk of toxicity and ineffective treatment for patients who are unlikely to benefit.

To illuminate the subclone dynamics influenced by immunotherapy treatment, I have applied PICTograph to this longitudinal dataset. Whole-exome sequencing data of the cancer is available for multiple time-points: pre-treatment, post-treatment, and occasionally recurrence.

### 3.2.1 Extension to longitudinal data

Longitudinal data provides extra time-point information for each sample. However, as we currently are using the Infinite Sites Assumption [28] for modeling evolution, the time-point information does not provide any additional restrictions over the existing ones from sample-presence (Section 2.2.1). Future

work is need to relax the Infinite Sites Assumption to allow modeling of mutation loss and parallel evolution.

## 3.2.2 Analysis pipeline

### 3.2.2.1 Sequencing data analysis

Whole-exome sequencing, data processing, and mutation calling were performed by PGDx. The reference genome used was hg19. PGDx mutation data provided us with read count information for a set of filtered mutations. Because this might exclude mutations present at lower frequency in some samples, we queried reads for case mutations not present in samples (”missing” mutations). For each case, we collected a list of all mutations across all samples. Then for each sample, we used mpileup to query read counts for any mutations not present in that sample.

Copy number analysis was performed using FACETS [32] using standard parameters for whole-exome sequencing. Copy number was queried for each mutation locus. Mutations that fall in regions with unavailable copy number estimates and mutations that occurred in regions with deletions were excluded from downstream analysis.

Multiplicity was estimated using a confidence interval approach in the following steps:

1. A 95% confidence interval (CI) for the variable $V_{exp}$ is constructed by using the distinct mutant read counts and distinct coverage at of the mutated locus.

2. Substitution of the other known variable yields a confidence interval for the product $mC$

3. m is estimated based on the following rules:

   (a) If the CI for mC overlaps an integer value, that value is estimated to indicate the multiplicity of the mutation and the mutation is clonal (C=1)

   (b) If the upper bound of the CI for $mC$ is below 1, the multiplicity is set to 1, and the mutation is subclonal, unless the resulting estimate for C is within a tolerance threshold (0.25) of 1

   (c) If the CI for $mC$ is above 1 and does not overlap any integer values, multiplicity is greater than 1 and m is set such that the confidence interval for $C$ falls within the expected intervals of [0,1]

Finally, visual inspection was performed on mutations that are present in only one sample to confirm gains/losses.

#### 3.2.2.2 Evolutionary analysis with PICTograph

Mutation clustering and CCF estimation was performed independently for each sample-presence set for a range of number of clusters $K$ from 1 to 5. Next, BIC was calculated for each $K$ assessed. Model selection was performed manually for each case by visually inspecting the BIC plots and when necessary, the cluster assignment and CCF violin plots of each model. When the minimum BIC, elbow of the BIC plot, and the knee of the BIC plot agreed on the choice of $K$, that $K$ was selected as the best model and further inspection was not performed for that mutation set. Results for each sample-presence set were then merged. Best mutation cluster assignments were recorded by querying the cluster assignment for each mutation with the highest posterior probability. Point estimates for CCFs were calculated using the modes of the posterior distributions.

For tree inference, PICTograph used the point estimations for cluster CCFs calculated in the clustering step to determine possible edges. Lineage precedence threshold of 0.1 and sum condition threshold of 0.2 were used as a starting point. Spanning trees were enumerated with the modified Gabow-Meyers algorithm. If no trees are found, the lineage precedence and sum condition thresholds were increased by 0.1, possible edges were re-determined, and enumeration was run again. Once one or more trees were found, the scoring function was applied and the highest scoring trees were returned. Subclone propor-

tions were calculated for cases that have a single highest-scoring tree.

### 3.2.3 Results

We analyzed the clonal evolution of non-squamous cell lung cancers for patients that underwent immunotherapy treatment. Here, I show the evolutionary patterns found by PICTograph for two cases: CGLU215 and CGLU220.

#### 3.2.3.1 CGLU215

For Patient CGLU215, one pre-treatment sample and one post-treatment sample were sequenced and a total of 321 somatic mutations were detected. After mapping mutations to copy number results, estimating mulitplicity, and visual inspection, 22 mutations were removed from evolutionary analysis.

From the remaining 299 somatic mutations, PICTograph identified 4 mutation clusters (Figure 3.9). Of these, 2 clusters were present in both pre and post-treatment samples, 1 cluster was present in only the pre-treatment sample, and 1 cluster was present in only the post-treatment sample (Figure 3.9A). Most mutations occured in both pre and post-treatment samples and were assigned to clusters 1 and 2, with 109 and 184 mutations, respectively. The private clusters 3 and 4 have much fewer mutation, 4 and 2, respectively. A total of 9 mutations were located in genes previously implicated in cancer. The majority of these mutations were assigned to the clonal cluster 2, including muta-

tions in AKT2, AR, CCNE1, MYCN, NOTCH1, PHOX2B, RB1, and CREBBP, which had the highest CCFs in both samples. The mutation in ARID1A was assigned to cluster 1, which is also present in samples from both time-points.

PICTograph identified a single highest-scoring tree (Figure 3.9B). The originating clone is defined by Cluster 2, followed by Cluster 1, and then branched into clusters 3 and 4. Analysis of subclone proportions (Figure 3.9C) revealed that the pre-treatment sample (CGLU215T1) was predominantly subclone 3 (60%) and also contains subclones 1 and 2 at lower proportions (14 and 17%, respectively). The post-treatment sample (CGLU215T2) consisted of mostly subclone 4 (48%) and a smaller proportion of subclone 2 (35%). This suggests that immnotherapy treatment negatively selected for subclone 3 and may have facilitated the emergence of subclone 4.

Subclone dynamics can be visualized across time using fishplots [36] (Figure 3.10). Tracking changes in clonal architecture may provide additional insight into therapeutic response and resistance. Although current analyses provide insight to the subclone proportions at the pre and post-treatment time-points, the dynamics between these two time-points is unknown. For Patient CGLU215, subclone 3 is lost and subclone 4 emerges between pre and post-treatment, but it is unclear at what times during immunotherapy treatment these events occur (Figure 3.10B). Further analysis of the liquid biopsy data collected during treatment for this cohort may provide additional insights to

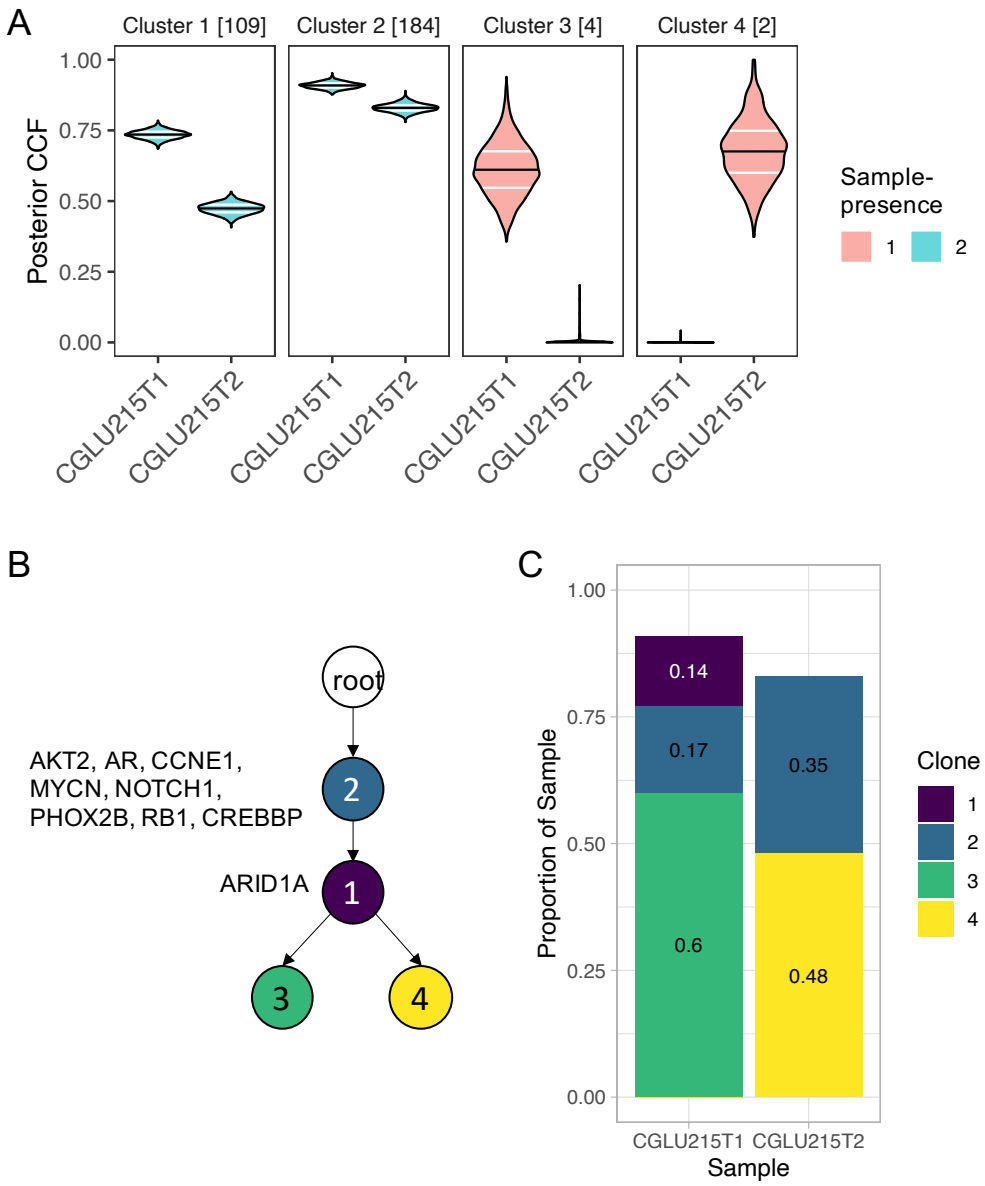



**Figure 3.9:** Mutation clustering, CCF estimation, and tree inference for CGLU215. CGLU215T1 is the pre-treatment sample and CGLU215T2 is the post-treatment sample. (A) Posterior distributions of CCFs found by PICTograph for IP09. The number of mutations assigned to each cluster is listed in brackets. (B) Highest-scoring tree inferred by PICTograph. (C) Bar charts showing proportion of subclones in each sample by PICTograph.

the subclone dynamics between time-points for which we have whole-exome sequencing data.

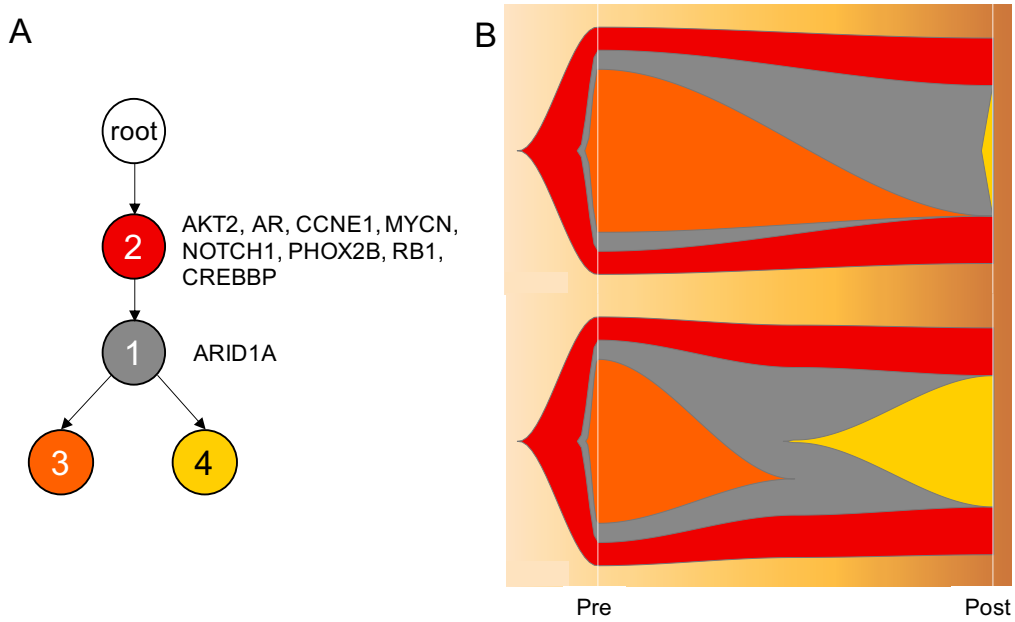



**Figure 3.10:** Example visualization of subclone dynamics across time for Patient CGLU215. (A) Highest-scoring tree inferred by PICTograph. (B) Two fishplots showing subclone proportions at pre and post-treatment time points. Top fishplot was generated by providing data at the two time points. Bottom fishplot was generated by supplementing additional dummy data in between the pre and post timepoints to manipulate the shapes of the subclones.

#### 3.2.3.2 CGLU220

For Patient CGLU220, one pre-treatment sample and one post-treatment sample were whole-exome sequenced. After mapping the 27 identified mutations to copy number results, estimating mulitplicity, and visual inspection, 5 mutations were removed, leaving 22 mutations for evolutionary analysis. Only one mutation was located in a gene with clinical significance, CDKN2A.

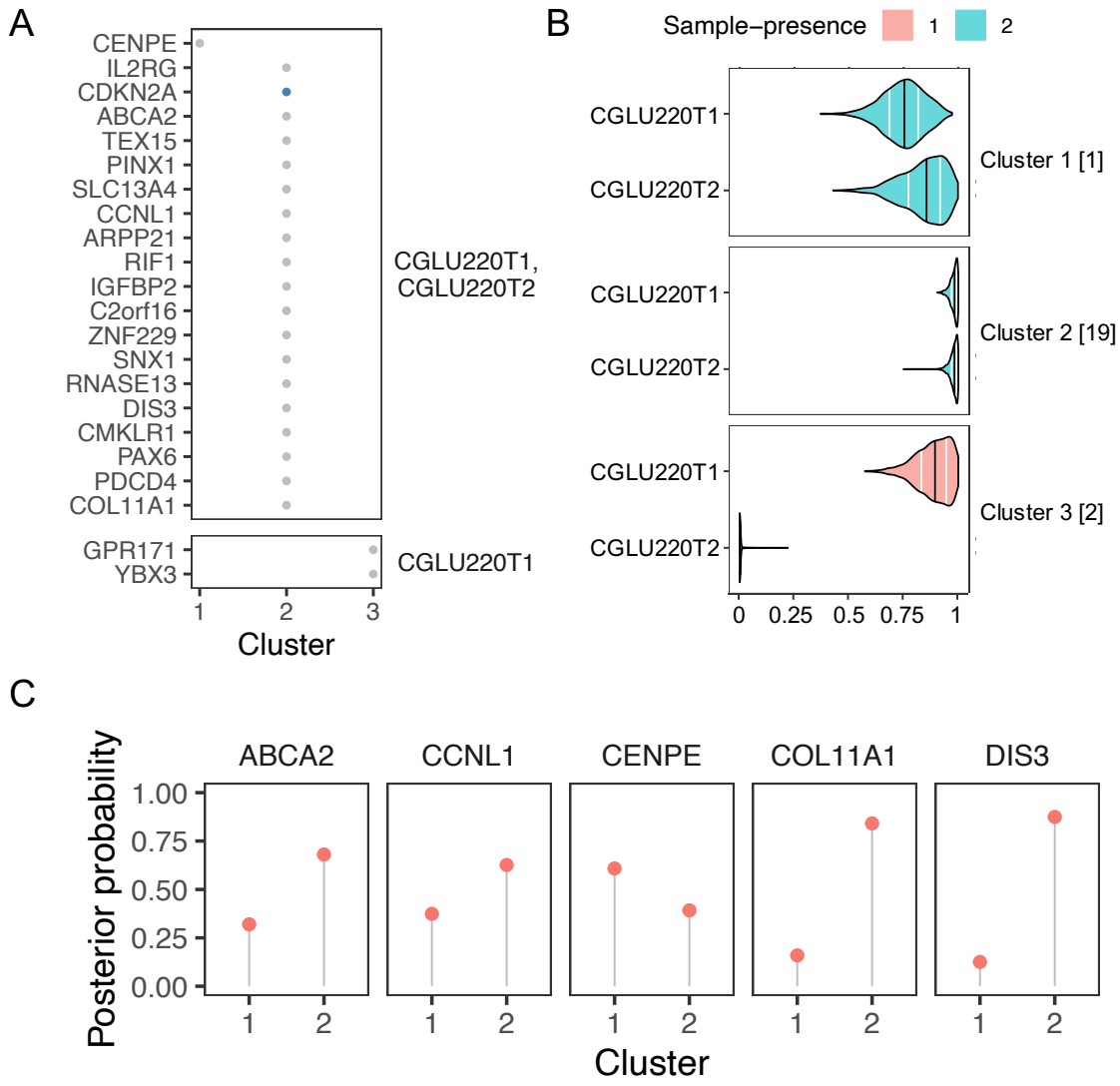



**Figure 3.11:** Mutation clustering and CCF estimation for Patient CGLU220. CGLU220T1 is the pre-treatment sample and CGLU220T2 is the post-treatment sample. (A) Most likely mutation cluster assignments. (B) Posterior distributions of CCFs found by PICTograph. The number of mutations assigned to each cluster is listed in brackets. (C) Posterior probabilities of cluster assignments for mutations with uncertain cluster assignment. Mutations shown are those with cluster assignment probabilities between 10 and 90%.

PICTograph identified 3 mutation clusters (Figure 3.11A-B). Two clusters are present in both pre and post-treatment samples and one cluster is present in only the pre-treatment sample. PICTograph identified a single, linear tree (Figure 3.12A). Notably, the CDKN2A mutation was assigned to cluster 2, which is the originating clone. Calculation of subclone proportions in each sample revealed that the pre-treatment sample (CGLU220T1) is 77% subclone 3 and 22% subclone 2, while the post-treatment sample (CGLU220T2) is 92% subclone 1 and 7% subclone 2 (Figure 3.12B). Following immunotherapy treatment, subclone 3 is lost, and subclone 1 becomes the dominating clone.

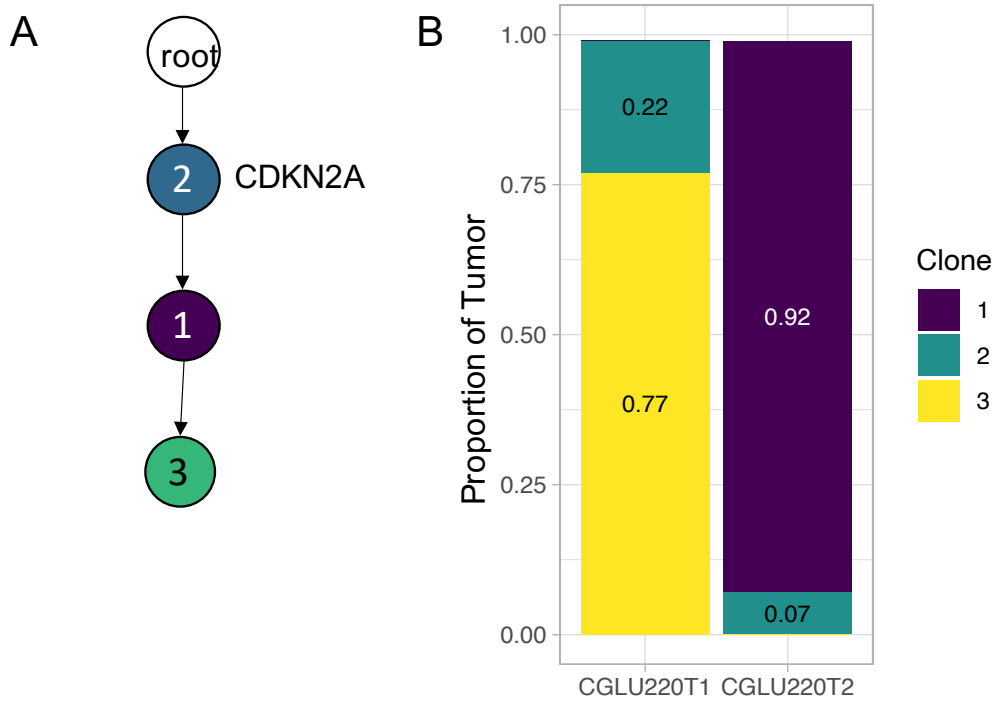



**Figure 3.12:** Clone tree and subclone proportions for Patient CGLU220. CGLU220T1 is the pre-treatment sample and CGLU220T2 is the post-treatment sample. (A) Highest-scoring tree inferred by PICTograph. (B) Bar charts showing proportion of subclones in each sample by PICTograph.

### 3.2.4 Discussion

Overall, evolutionary analyses with PICTograph revealed different tree structures of the two cancers presented, highlighting the utility of a clone-tree based approach. While the evolution of the cancer of patient CGLU215 exhibited a branching pattern, the cancer of patient CGLU220 exhibited a linear pattern of evolution. PICTograph also placed known oncogenic drivers as truncal mutations, consistent with known evolutionary patterns of cancers.

A key motivation of our application of PICTograph to immunotherapy-treated lung cancers is to understand the impact of immunotherapy treatment on evolution and subclone dynamics. From clone-tree analyses, subclone proportions can be calculated for each available sample. This allows comparisons of subclone proportions of pre and post-treatment samples, and can reveal subclones that are lost and gained following immunotherapy treatment. Further analyses of the mutations defining subclones that are either lost or gained and combination with analyses of other data types such as gene expression or HLA data may lead to biological insights to which patients are most likely to benefit from immunotherapy treatment.

# Chapter 4

# Discussion

## 4.1 Utility in cancer studies

I have developed a new algorithm PICTograph that estimates the cancer cell fraction (CCF) of each somatic mutation and infers the most likely evolutionary tree topology from multi-region bulk sequencing data. PICTograph leverages the sample-presence patterns of mutations to inform mutation clustering and to constrain the space of possible trees. By modeling the joint distribution of allelic frequencies across samples, PICTograph resolves cluster memberships and reveals subclonal populations that would otherwise be indistinguishable with single sample analyses. PICTograph performs well over a wide range of simulated multi-sample tumor complexity encountered in experimental applications, and is therefore likely to be of broad utility for a number of cancer

types and stages of cancer progression.

## 4.2 Future development

While this approach outperforms existing state-of-the-art methods for inferring correct ancestral relationships in a comprehensive series of simulations, PICTograph has several limitations. Expanding upon the existing method would improve modeling capabilities.

### 4.2.1 Expanding the clustering model

PICTograph's clustering algorithm assumes that the purity, copy number, and multiplicity of a mutation have already been correctly estimated and, implicitly, that these characteristics are measured without error. Noise of these estimates are not currently reflected in the posterior for the CCFs or in the posterior probability of mutation membership to clusters. Expanding the Bayesian model to include uncertainty in these variables would allow capture of potential noise of these estimates.

### 4.2.2 Modeling of copy number alterations

PICTograph identifies the most probable trees as those that satisfy the lineage precedence and Sum Condition principles, criteria based on the infinite sites assumption [28]. Any cell that lost a mutation during evolution of the cancer by reverting back to the wild-type allele would violate this assumption. The critical role of the infinite sites assumption in determining the likely evolutionary relationships is shared by many of the existing methods, including SCHISM, PhyloWGS, PyClone, Canopy, LICHeE. However, many cancers acquire copy number alterations, and loss of heterozygosity is fairly common [37]. The challenge of modeling CNAs lies in resolving the temporal ordering of mutations that overlap CNAs. Copy number is a key variable in estimating the CCFs of mutatitons. Thus, when a mutation occurs in the same region as a copy gain or loss, the CCF of the mutation may differ depending on the relative timing of the mutations and whether the allele with the mutation is duplicated or lost. A few existing methods that model CNAs with single nucleotide variants and insertions/deletions assess all possible scenarios of ordering these smaller mutations with overlapping CNAs [10, 12].

### 4.2.3 Joint inference of mutation clustering, cancer cell fraction estimation, and clone trees

Currently, PICTograph infers ancestral relationships for the latent tumor subclones based on maximum a posteriori (MAP) estimates of the CCFs and uncertainty of these estimates, while available from the posterior, is not reflected in the set of possible evolutionary trees. While Bayesian structural models such as MC3 [38] would enable updates to the tree given the VAFs and CCFs, these models are well known to be slow mixing [39, 40]. As the CCFs, cluster assignments, and number of clusters are also unknown, obtaining a useful approximation to the joint distribution of these parameters and the structural model will require efficient approaches to sampling. Starting values for such models will be important for efficient mixing, and MAP estimates from PICTograph could provide a useful initialization.

# Appendix A

# Installing and using PICTograph

PICTograph is available as an R package and can be installed from GitHub.

## A.1 Installation

PICTograph is available from the Karchin Lab GitHub.

Dependencies must first be installed. Version of R should be $\geq$ 3.5.0. PICTograph uses the JAGS library for Bayesian data analysis, which is installed outside of R. JAGS can be downloaded and installed for your OS here.

To install the main branch of PICTograph from GitHub, you will need the 'devtools' package. Start R and enter:

```
library(devtools)

install_github("KarchinLab/pictograph", build_vignettes = TRUE)
```

Some newer features (for model selection and additional plotting functions) are currently only available on the development branch. To install the development branch, clone the repository, switch to the 'dev' branch, and install from local source in R.

From command line:

```
git clone https://github.com/KarchinLab/pictograph.git

git checkout dev
```

From R:

```
library(devtools)

install_local("/path/to/pictograph")
```

## A.2 Usage example

The following usage example can be found in the vignette of the development branch of PICTograph and may be moved to the main version of PICTograph in the future.

# PICTograph

## 1. Introduction

This tutorial walks through how to run PICTograph on a toy example. PICTograph infers the clonal evolution of tumors from multi-region sequencing data. It models uncertainty in assigning mutations to subclones using a Bayesian hierarchical model and reduces the space of possible evolutionary trees by using constraints based on principles of sample-presence, lineage precedence, and Sum Condition. The inputs to PICTograph are variant read counts of single nucleotide variants (SNVs), sequencing depth of mutation loci, number of mutant alleles (multiplicity), DNA copy number of the tumor genome containing the mutation, and the tumor purity of each sample. PICTograph summarizes the posterior distributions of the mutation cluster assignments and the cancer cell fractions (CCF) for each cluster by the mode. The estimates of cluster CCFs are then used to determine the most probable trees. Multiple trees that share the same score can be summarized as an ensemble tree, where edges are weighted by their concordance among constituent trees in the ensemble.

## 2. Input data

The input data are organized as a list of 7 objects. Variant read count (y), depth (n), total copy number (tcn), and multiplicity (m) are stored in matrices where the columns are samples, and rows are variants. Purity is supplied as a vector. I and S are integers representing the number of variants and number of samples, respectively.

Here we use a toy example with 3 tumor samples and 42 mutations. This data was simulated from a true tree containing 5 mutation clusters.

```
data("sim_data_1")
input_data <- list(y = sim_data_1$y,
                   n = sim_data_1$n,
                   purity = sim_data_1$purity,
                   tcn = sim_data_1$tcn,
                   m = sim_data_1$m,
                   I = sim_data_1$I,
                   S = sim_data_1$S)
```

## 3. Clustering mutations and estimating CCFs

The first step of evolutionary analysis is clustering mutations and estimating their CCFs. This is comprised of three steps:

a. Splitting the mutation data into sets by sample-presence and approximating the joint posterior by Markov chain Monte Carlo (MCMC) across a range of possible values for the number of clusters, $K$
b. Selecting the best $K$ for each mutation set
c. Merging the best chains of each mutation set

### 3a. Run clustering and CCF estimation separately for each mutation set

In toy example, we run a short MCMC chain with only 1000 iterations (`n.iter`), burn-in of 100 (`n.burn`), and no thinning (`thin`). In practice, we recommend running the MCMC for longer e.g. 10,000 iterations, burn-in of 1000, and thinning by 10. PICTograph separates mutations into sets by sample-presence patterns, and clusters mutations separately within each set. We can set the maximum number of clusters to evaluate with `max_K`.

```
all_set_results <- clusterSep(input_data,
                              n.iter = 1000, n.burn = 100, thin = 1,
                              max_K = 5)
```

This gives us a list with results for each mutation set, which contains `all_chains`, `BIC`, `best_chains`, and `best_K`.

`all_chains` is a list of MCMC chains for each value of $K$ tested. For each $K$, there are chains for cluster CCF (`w_chain`), mutation cluster assignments (`z_chain`), and simulated variant read counts (`ystar_chain`) for posterior predictive distributions.

`BIC` is a table of the BIC for each $K$ assessed.

As default, PICTograph chooses the $K$ with the lowest BIC. The MCMC chains for this chosen $K$ are under `best_chains` and the $K$ chosen is listed under `best_K`.

```
str(all_set_results, give.attr = F, max.level = 4)
#> List of 2
#>  $ 111:List of 4
#>   ..$ all_chains :List of 5
#>   .. ..$ K1:List of 3
#>   .. .. ..$ w_chain    : tibble [3,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ z_chain    : tibble [25,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ ystar_chain: tibble [75,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ K2:List of 3
#>   .. .. ..$ w_chain    : tibble [6,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ z_chain    : tibble [25,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ ystar_chain: tibble [75,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ K3:List of 3
#>   .. .. ..$ w_chain    : tibble [9,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ z_chain    : tibble [25,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ ystar_chain: tibble [75,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ K4:List of 3
#>   .. .. ..$ w_chain    : tibble [12,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ z_chain    : tibble [25,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ ystar_chain: tibble [75,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ K5:List of 3
#>   .. .. ..$ w_chain    : tibble [15,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ z_chain    : tibble [25,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ ystar_chain: tibble [75,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   ..$ BIC        : tibble [5 × 2] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ K_tested: int [1:5] 1 2 3 4 5
#>   .. ..$ BIC     : num [1:5] 1458 851 411 417 419
#>   ..$ best_chains:List of 3
#>   .. ..$ w_chain    : tibble [9,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ z_chain    : tibble [25,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ ystar_chain: tibble [75,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   ..$ best_K     : int 3
#>  $ 011:List of 4
#>   ..$ all_chains :List of 5
#>   .. ..$ K1:List of 3
#>   .. .. ..$ w_chain    : tibble [3,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ z_chain    : tibble [17,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ ystar_chain: tibble [51,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ K2:List of 3
#>   .. .. ..$ w_chain    : tibble [6,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ z_chain    : tibble [17,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ ystar_chain: tibble [51,000 × 4] (S3: tbl_df/tbl/data.frame)
```

```
#>   .. ..$ K3:List of 3
#>   .. .. ..$ w_chain    : tibble [9,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ z_chain    : tibble [17,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ ystar_chain: tibble [51,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ K4:List of 3
#>   .. .. ..$ w_chain    : tibble [12,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ z_chain    : tibble [17,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ ystar_chain: tibble [51,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ K5:List of 3
#>   .. .. ..$ w_chain    : tibble [15,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ z_chain    : tibble [17,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. .. ..$ ystar_chain: tibble [51,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   ..$ BIC        : tibble [5 × 2] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ K_tested: int [1:5] 1 2 3 4 5
#>   .. ..$ BIC     : num [1:5] 378 195 199 203 207
#>   ..$ best_chains:List of 3
#>   .. ..$ w_chain    : tibble [6,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ z_chain    : tibble [17,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   .. ..$ ystar_chain: tibble [51,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   ..$ best_K     : int 2
```

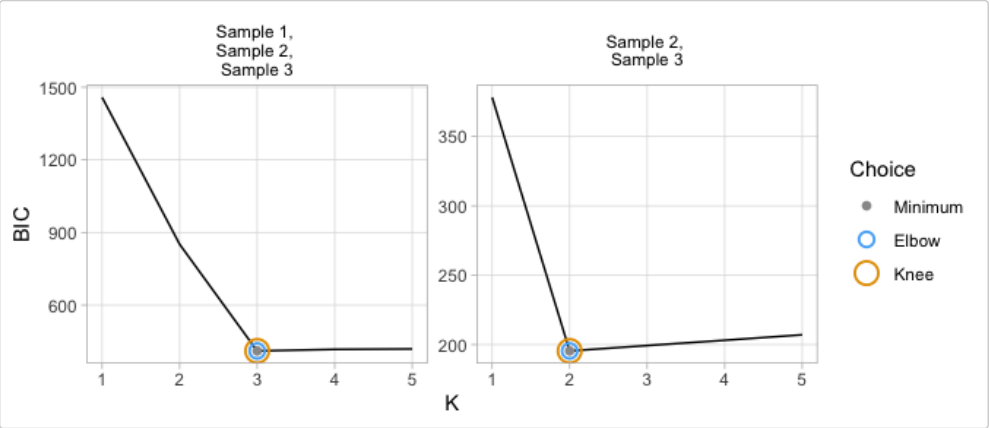
### 3b. Select the best number of clusters, $K$, for each mutation set

From `all_set_results`, we can extract the MCMC chains for the best $K$ of each mutation set using `collectBestKChains`. As default, PICTograph chooses the $K$ with the lowest BIC, and these chains will be automatically extracted. Users also have the option to specify the $K$ to choose for each mutation set by supplying a vector of integers to the parameter `chosen_K` in the same order as the listed sets in `all_set_results`.

```
best_set_chains <- collectBestKChains(all_set_results)
str(best_set_chains, give.attr = F, max.level = 2)
#> List of 2
#>  $ 111:List of 3
#>   ..$ w_chain    : tibble [9,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   ..$ z_chain    : tibble [25,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   ..$ ystar_chain: tibble [75,000 × 4] (S3: tbl_df/tbl/data.frame)
#>  $ 011:List of 3
#>   ..$ w_chain    : tibble [6,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   ..$ z_chain    : tibble [17,000 × 4] (S3: tbl_df/tbl/data.frame)
#>   ..$ ystar_chain: tibble [51,000 × 4] (S3: tbl_df/tbl/data.frame)
```

BIC values for all $K$ assessed in each mutation set can be visualized using `plotBIC`. The minimum, elbow, and knee points are marked.

```
plotBIC(all_set_results)
```




### 3c. Merge results for all mutation sets

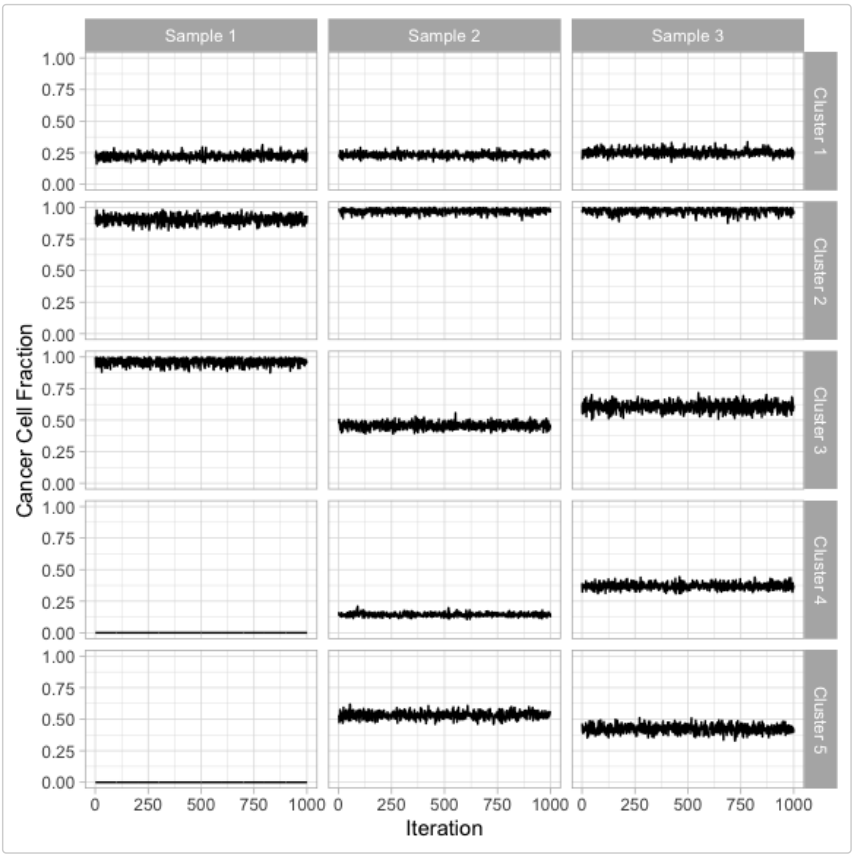
Finally, we can merge `best_set_chains` to obtain chains with the final mutation cluster numbering and correct mutation indices (original order provided in input data).

```
chains <- mergeSetChains(best_set_chains, input_data)
```

### Visualizing clustering and CCF estimation results

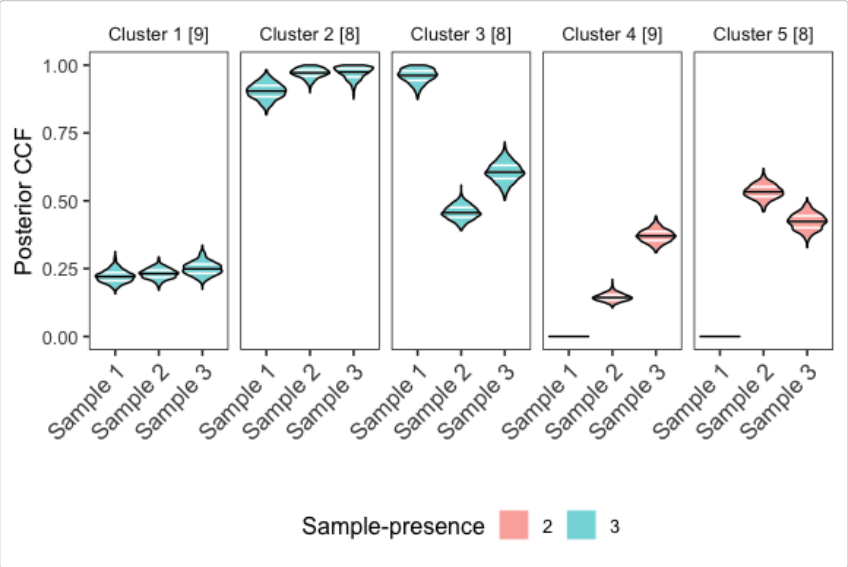
Traces for CCF chains can be visualized to check for convergence.

```
plotChainsCCF(chains$w_chain)
```



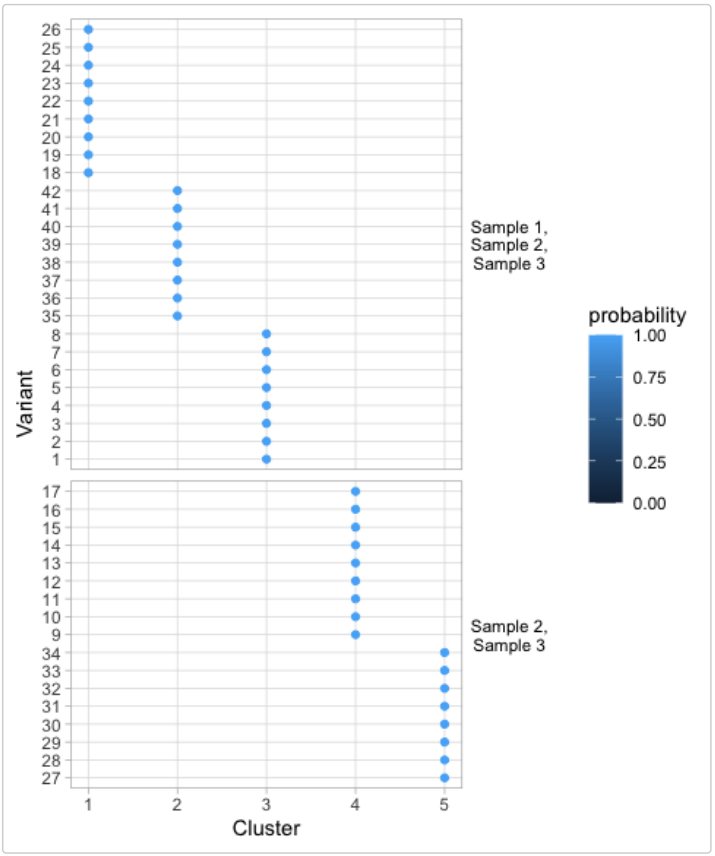

The posterior distribtuion of cluster CCFs can be visualized as violin plots. The number of mutations assigned to each cluster is listed in brackets after the cluster name.

```
plotCCFViolin(chains$w_chain, chains$z_chain, indata = input_data)
```




We can also visualize the posterior probabilities of mutation cluster assignments and determine the most probable cluster assignments. In this toy example, there is high concordance of the cluster assignments of mutations across the MCMC chain.

```
plotClusterAssignmentProbVertical(chains$z_chain, chains$w_chain)
```




We can write tables for estimated cluster CCFs and mutation cluster assignments.

```
writeClusterCCFsTable(chains$w_chain)
#> # A tibble: 5 × 4
#>   Cluster `Sample 1` `Sample 2` `Sample 3`
#>     <int>      <dbl>      <dbl>      <dbl>
#> 1       1       0.22       0.23       0.24
#> 2       2       0.91       0.97       0.99
#> 3       3       0.97       0.45       0.6
#> 4       4       0          0.14       0.37
#> 5       5       0          0.53       0.42
writeClusterAssignmentsTable(chains$z_chain)
#> # A tibble: 42 × 2
#>    Mut_ID Cluster
#>    <chr>    <dbl>
#>  1 Mut18        1
#>  2 Mut19        1
#>  3 Mut20        1
#>  4 Mut21        1
#>  5 Mut22        1
#>  6 Mut23        1
#>  7 Mut24        1
#>  8 Mut25        1
#>  9 Mut26        1
```

```
#> 10 Mut35        2
#> # … with 32 more rows
```

## 4. Tree inference

We can then use the mutation cluster CCF estimates for tree inference. We first determine the possible edges by applying sample presence and lineage precedence filters. Note: the filtered edges must be in an object named `graph_G`.

```
w_mat <- estimateCCFs(chains$w_chain)
graph_G_pre <- prepareGraphForGabowMyers(w_mat, zero.thresh = 0.01)
graph_G <- filterEdgesBasedOnCCFs(graph_G_pre, w_mat, thresh = 0.1)
```

Next, we enumerate this constrained tree space, and apply a filtered based on the Sum Condition. Here we use a threshold of 0.2 for the maximum allowed violation of the Sum Condition.

```
enumerateSpanningTreesModified(graph_G, w_mat, sum_filter_thresh=0.2)
```

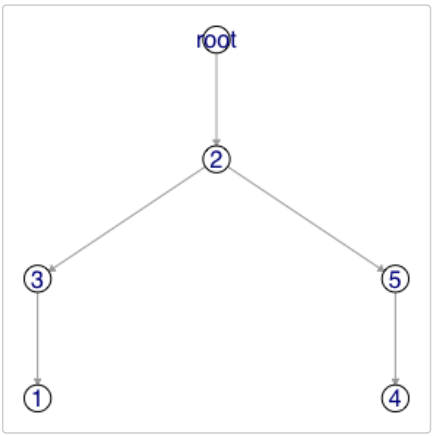
All spanning trees given by the possible edges that pass the Sum Condition filter are stored in `all_spanning_trees`.

```
length(all_spanning_trees)
#> [1] 2
all_spanning_trees
#> [[1]]
#> # A tibble: 5 × 3
#>   edge    parent child
#>   <chr>   <chr>  <chr>
#> 1 root->2 root   2
#> 2 2->3    2      3
#> 3 3->1    3      1
#> 4 3->4    3      4
#> 5 2->5    2      5
#>
#> [[2]]
#> # A tibble: 5 × 3
#>   edge    parent child
#>   <chr>   <chr>  <chr>
#> 1 root->2 root   2
#> 2 2->3    2      3
#> 3 3->1    3      1
#> 4 2->5    2      5
#> 5 5->4    5      4
```

We then calculate a fitness score for all the trees that have passed our filtering and identify the highest scoring tree.

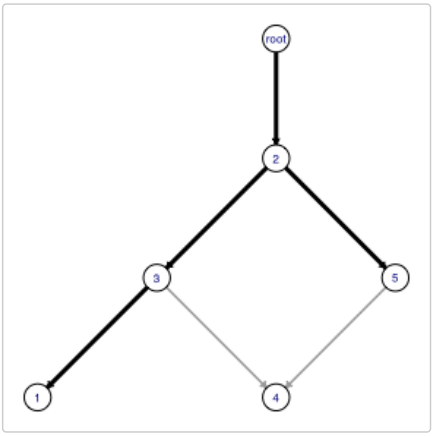
```
# calculate SCHISM fitness score for all trees
scores <- calcTreeScores(chains$w_chain, all_spanning_trees)
scores
#> [1] 0.7046881 0.7408182
# highest scoring tree
best_tree <- all_spanning_trees[[which.max(scores)]]
```

```
# plot tree
plotTree(best_tree)
```



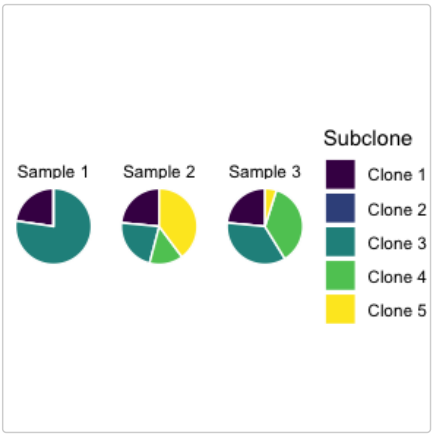

In this toy example, there is only one tree with the maximum score. In some cases, multiple trees will share the maximum score. We can plot an ensemble tree to visualize the evolutionary relationships (edges) that are shared among multiple trees. In the ensemble tree, edges are weighted by the number of trees in which they are represented. To illustrate this plotting function, here we plot an ensemble of the two trees that were enumerated. The solid black edges represent those supported in both trees.

```
plotEnsembleTree(all_spanning_trees)
```




```
subclone_props <- calcSubcloneProportions(w_mat, best_tree)
plotSubclonePie(subclone_props)
```

# APPENDIX A. INSTALLING AND USING PICTOGRAPH



# Bibliography

[1] M. Greaves and C. C. Maley, "Clonal evolution in cancer," *Nature*, vol. 481, pp. 306–313, 2012.

[2] N. Beerenwinkel, R. F. Schwarz, M. Gerstung, and F. Markowetz, "Cancer Evolution: Mathematical Models and Computational Inference," *Systematic Biology*, vol. 64, no. 1, pp. e1–e25, 10 2014. [Online]. Available: https://doi.org/10.1093/sysbio/syu081

[3] B. Vogelstein, N. Papadopoulos, V. E. Velculescu, S. Zhou, L. A. Diaz, and K. W. Kinzler, "Cancer genome landscapes," *Science*, vol. 339, no. 6127, pp. 1546–1558, 2013. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.1235122

[4] N. McGranahan and C. Swanton, "Clonal Heterogeneity and Tumor Evolution: Past, Present, and the Future," *CELL*, vol. 168, pp. 613–628, 2017.

[5] M. Gerlinger, S. Horswell, J. Larkin, A. J. Rowan, M. P. Salm, I. Varela, R. Fisher, N. McGranahan, N. Matthews, C. R. Santos, P. Martinez,

B. Phillimore, S. Begum, A. Rabinowitz, B. Spencer-Dene, S. Gulati, P. A. Bates, G. Stamp, L. Pickering, M. Gore, D. L. Nicol, S. Hazell, P. A. Futreal, A. Stewart, and C. Swanton, "Genomic architecture and evolution of clear cell renal cell carcinomas defined by multiregion sequencing," *Nature Genetics*, vol. 46, no. 3, pp. 225–233, feb 2014.

[6] Z.-M. Zhao, B. Zhao, Y. Bai, A. Iamarino, S. G. Gaffney, J. Schlessinger, R. P. Lifton, D. L. Rimm, and J. P. Townsend, "Early and multiple origins of metastatic lineages within primary tumors," *Proceedings of the National Academy of Sciences*, vol. 113, no. 8, pp. 2140–2145, feb 2016.

[7] W. Zhai, T. K.-H. Lim, T. Zhang, S.-T. Phang, Z. Tiang, P. Guan, M.-H. Ng, J. Q. Lim, F. Yao, Z. Li, P. Y. Ng, J. Yan, B. K. Goh, A. Y.-F. Chung, S.-P. Choo, C. C. Khor, W. W.-J. Soon, K. W.-K. Sung, R. S.-Y. Foo, and P. K.-H. Chow, "The spatial organization of intra-tumour heterogeneity and evolutionary trajectories of metastases in hepatocellular carcinoma," *Nature Communications*, vol. 8, no. 1, feb 2017.

[8] J. M. Alves, T. Prieto, and D. Posada, "Multiregional tumor trees are not phylogenies." *Trends in cancer*, vol. 3, pp. 546–550, Aug. 2017.

[9] N. Niknafs, V. Beleva-Guthrie, D. Q. Naiman, and R. Karchin, "Subclonal hierarchy inference from somatic mutations: Automatic reconstruction of

cancer evolutionary trees from multi-region next generation sequencing." *PLoS computational biology*, vol. 11, p. e1004416, Oct. 2015.

[10] A. G. Deshwar, S. Vembu, C. K. Yung, G. H. Jang, L. Stein, and Q. Morris, "Phylowgs: Reconstructing subclonal composition and evolution from whole-genome sequencing of tumors," *Genome Biology*, vol. 16, 2015.

[11] V. Popic, R. Salari, I. Hajirasouliha, D. Kashef-Haghighi, R. B. West, and S. Batzoglou, "Fast and scalable inference of multi-sample cancer lineages." *Genome biology*, vol. 16, p. 91, May 2015.

[12] Y. Jiang, Y. Qiu, A. J. Minn, and N. R. Zhang, "Assessing intratumor heterogeneity and tracking longitudinal and spatial clonal evolutionary history by next-generation sequencing," *Proceedings of the National Academy of Sciences*, vol. 113, no. 37, pp. E5528–E5537, aug 2016.

[13] G. Satas and B. J. Raphael, "Tumor phylogeny inference using tree-constrained importance sampling," *Bioinformatics*, vol. 33, pp. i152–i160, 2017.

[14] D. Benjamin, T. Sato, K. Cibulskis, G. Getz, C. Stewart, and L. Lichtenstein, "Calling somatic SNVs and indels with mutect2," *bioRxiv preprint*, dec 2019.

[15] A. Roth, J. Khattra, D. Yap, A. Wan, E. Laks, J. Biele, G. Ha, S. Apari-

cio, A. Bouchard-Côté, and S. P. Shah, "PyClone: statistical inference of clonal population structure in cancer," *Nature Methods*, vol. 11, pp. 396–398, 2014.

[16] C. A. Miller, B. S. White, N. D. Dees, M. Griffith, J. S. Welch, O. L. Griffith, R. Vij, M. H. Tomasson, T. A. Graubert, M. J. Walter, M. J. Ellis, W. Schierding, J. F. DiPersio, T. J. Ley, E. R. Mardis, R. K. Wilson, and L. Ding, "SciClone: Inferring clonal architecture and tracking the spatial and temporal patterns of tumor evolution," *PLoS Computational Biology*, vol. 10, no. 8, p. e1003665, Aug. 2014. [Online]. Available: https://ui.adsabs.harvard.edu/abs/2014PLSCB..10E3665M

[17] M. El-Kebir, L. Oesper, H. Acheson-Field, and B. J. Raphael, "Reconstruction of clonal trees and tumor composition from multi-sample sequencing data." *Bioinformatics (Oxford, England)*, vol. 31, pp. i62–i70, Jun. 2015.

[18] M. Tarabichi, A. Salcedo, A. G. Deshwar, M. Ni Leathlobhair, J. Wintersinger, D. C. Wedge, P. Van Loo, Q. D. Morris, and P. C. Boutros, "A practical guide to cancer subclonal reconstruction from DNA sequencing," *Nature Methods*, vol. 18, no. 2, pp. 144–155, 2021.

[19] B. Domazet, G. T. Maclennan, A. Lopez-Beltran, R. Montironi, and L. Cheng, "Laser capture microdissection in the genomic and proteomic

era: targeting the genetic basis of cancer," *International journal of clinical and experimental pathology*, vol. 1, no. 6, pp. 475–488, Mar. 2008.

[20] K. Fujikura, W. Hosoda, M. Felsenstein, Q. Song, J. G. Reiter, L. Zheng, V. B. Guthrie, N. Rincon, M. D. Molin, J. Dudley, J. D. Cohen, P. Wang, C. G. Fischer, A. M. Braxton, M. Noë, M. Jongepier, C. F. del Castillo, M. Mino-Kenudson, C. M. Schmidt, M. T. Yip-Schneider, R. T. Lawlor, R. Salvia, N. J. Roberts, E. D. Thompson, R. Karchin, A. M. Lennon, Y. Jiao, and L. D. Wood, "Multiregion whole-exome sequencing of intraductal papillary mucinous neoplasms reveals frequent somatic *KLF4* mutations predominantly in low-grade regions," *Gut*, vol. 70, pp. 928–939, 10 2021.

[21] C. G. Fischer, V. B. Guthrie, A. M. Braxton, L. Zheng, P. Wang, Q. Song, J. F. Griffin, P. E. Chianchiano, W. Hosoda, N. Niknafs, S. Springer, M. D. Molin, D. Masica, R. B. Scharpf, E. D. Thompson, J. He, C. L. Wolfgang, R. H. Hruban, N. J. Roberts, A. M. Lennon, Y. Jiao, R. Karchin, and L. D. Wood, "Intraductal papillary mucinous neoplasms arise from multiple independent clones, each with distinct mutations," *Gastroenterology*, vol. 157, no. 4, pp. 1123–1137.e22, oct 2019.

[22] A. J. J. Smits, J. A. Kummer, P. C. de Bruin, M. Bol, J. G. van den Tweel, K. A. Seldenrijk, S. M. Willems, G. J. A. Offerhaus, R. A. de Weger, P. J. van Diest, and A. Vink, "The estimation of tumor cell percentage for molecular

testing by pathologists is not accurate," *Modern Pathology*, vol. 27, no. 2, pp. 168–174, 2014.

[23] R. Shen and V. E. Seshan, "Facets: allele-specific copy number and clonal heterogeneity analysis tool for high-throughput dna sequencing." *Nucleic acids research*, vol. 44, p. e131, Sep. 2016.

[24] M. Riester, A. P. Singh, A. R. Brannon, K. Yu, C. D. Campbell, D. Y. Chiang, and M. P. Morrissey, "PureCN: copy number calling and SNV classification using targeted short read sequencing," *Source Code for Biology and Medicine*, vol. 11, no. 1, p. 13, 2016.

[25] L. Zheng, N. Niknafs, L. D. Wood, R. Karchin, and R. B. Scharpf, "Estimation of cancer cell fractions and clone treesfrom multi-region sequencing of tumors," *(Under Revision) Bioinformatics*, 2022.

[26] G. Schwarz, "Estimating the Dimension of a Model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461 – 464, 1978. [Online]. Available: https://doi.org/10.1214/aos/1176344136

[27] Q. Zhao, V. Hautamaki, and P. Fränti, "Knee point detection in bic for detecting the number of clusters," in *Advanced Concepts for Intelligent Vision Systems*, J. Blanc-Talon, S. Bourennane, W. Philips, D. Popescu, and P. Scheunders, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 664–673.

[28] J. Ma, A. Ratan, B. J. Raney, B. B. Suh, W. Miller, and D. Haussler, "The infinite sites model of genome evolution." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, pp. 14 254–14 261, Sep. 2008.

[29] H. N. Gabow and E. W. Myers, "Finding all spanning trees of directed and undirected graphs," *SIAM Journal on Computing*, vol. 7, no. 3, pp. 280–287, aug 1978.

[30] N. Niknafs, V. Beleva-Guthrie, D. Q. Naiman, and R. Karchin, "Subclonal hierarchy inference from somatic mutations: Automatic reconstruction of cancer evolutionary trees from multi-region next generation sequencing." *PLoS computational biology*, vol. 11, p. e1004416, Oct. 2015.

[31] J. G. Reiter, A. P. Makohon-Moore, J. M. Gerold, I. Bozic, K. Chatterjee, C. A. Iacobuzio-Donahue, B. Vogelstein, and M. A. Nowak, "Reconstructing metastatic seeding patterns of human cancers." *Nature communications*, vol. 8, p. 14114, Jan. 2017.

[32] E. Talevich, A. H. Shain, T. Botton, and B. C. Bastian, "CNVkit: Genome-wide copy number detection and visualization from targeted DNA sequencing," *PLOS Computational Biology*, vol. 12, no. 4, p. e1004873, apr 2016.

[33] R. H. Hruban, M. Goggins, J. Parsons, and S. E. Kern, "Progression model

for pancreatic cancer." *Clinical Cancer Research*, vol. 6, pp. 2969–2972, Aug. 2000.

[34] M. Felsenstein, M. Noë, D. L. Masica, W. Hosoda, P. Chianchiano, C. G. Fischer, G. Lionheart, L. A. A. Brosens, A. Pea, J. Yu, G. Gemenetzis, V. P. Groot, M. A. Makary, J. He, M. J. Weiss, J. L. Cameron, C. L. Wolfgang, R. H. Hruban, N. J. Roberts, R. Karchin, M. G. Goggins, and L. D. Wood, "IPMNs with co-occurring invasive cancers: neighbours but not always relatives," *Gut*, vol. 67, no. 9, pp. 1652–1662, mar 2018.

[35] J. Wu, H. Matthaei, A. Maitra, M. D. Molin, L. D. Wood, J. R. Eshleman, M. Goggins, M. I. Canto, R. D. Schulick, B. H. Edil, C. L. Wolfgang, A. P. Klein, L. A. Diaz, P. J. Allen, C. M. Schmidt, K. W. Kinzler, N. Papadopoulos, R. H. Hruban, and B. Vogelstein, "Recurrent GNAS mutations define an unexpected pathway for pancreatic cyst development," *Science Translational Medicine*, vol. 3, no. 92, pp. 92ra66–92ra66, jul 2011.

[36] C. A. Miller, J. McMichael, H. X. Dang, C. A. Maher, L. Ding, T. J. Ley, E. R. Mardis, and R. K. Wilson, "Visualizing tumor evolution with the fishplot package for R," *BMC Genomics*, vol. 17, no. 1, p. 880, 2016.

[37] J. Kuipers, K. Jahn, B. J. Raphael, and N. Beerenwinkel, "Single-cell sequencing data reveal widespread recurrence and loss of mutational hits in

the life histories of tumors," *Genome Research*, vol. 27, no. 11, pp. 1885–1894, Nov. 2017.

[38] D. Madigan, J. York, and D. Allard, "Bayesian graphical models for discrete data," *International Statistical Review / Revue Internationale de Statistique*, vol. 63, no. 2, pp. 215–232, 1995. [Online]. Available: http://www.jstor.org/stable/1403615

[39] R. Goudie and S. Mukherjee, "A gibbs sampler for learning DAGs." *Journal of Machine Learning Research*, vol. 17, pp. 1–39, 2016.

[40] J. Kuipers, P. Suter, and G. Moffa, "Efficient sampling and structure learning of bayesian networks," 2021.