Bioinformatics, 31(9), 2015, 1349–1356
doi: 10.1093/bioinformatics/btv003
Advance Access Publication Date: 6 January 2015
Original Paper

OXFORD

Genome analysis

Clonality inference in multiple tumor samples using phylogeny

Salem Malikic[1,†], Andrew W. McPherson[2,†], Nilgun Donmez[3,†] and Cenk S. Sahinalp[1,4,*]

[1]School of Computing Science, Simon Fraser University, Burnaby, BC, Canada, [2]BC Cancer Agency, Vancouver, BC, Canada, [3]Vancouver Prostate Center, Vancouver, BC, Canada and [4]School of Informatics and Computing, Indiana University, Bloomington, IN, USA

*To whom correspondence should be addressed.
†The authors wish it to be known that, in their opinion, the first three authors should be regarded as Joint First Authors.
Associate Editor: John Hancock

Received on August 26, 2014; revised on December 29, 2014; accepted on December 30, 2014

Abstract

Motivation: Intra-tumor heterogeneity presents itself through the evolution of subclones during cancer progression. Although recent research suggests that this heterogeneity has clinical implications, *in silico* determination of the clonal subpopulations remains a challenge.
Results: We address this problem through a novel combinatorial method, named clonality inference in tumors using phylogeny (CITUP), that infers clonal populations and their frequencies while satisfying phylogenetic constraints and is able to exploit data from multiple samples. Using simulated datasets and deep sequencing data from two cancer studies, we show that CITUP predicts clonal frequencies and the underlying phylogeny with high accuracy.
Availability and implementation: CITUP is freely available at: http://sourceforge.net/projects/citup/.
Contact: cenk@sfu.ca
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Most human tumors exhibit a large degree of heterogeneity. This heterogeneity is not only apparent in histology, but also presents itself in various features such as gene expression changes, genomic copy number alterations and somatic mutations as well as other aberrations. Although the origins of the intra-tumor heterogeneity are still debated, research suggests that this diversity is likely to have clinical implications. For instance, Merlo *et al.* (2010) have reported a correlation between clonal diversity and progression to esophageal adenocarcinoma in Barrett's esophagus.

The implications of tumor heterogeneity are not limited to diagnostics. It has been suggested that clonal diversity may also be linked to metastatic potential and drug response. Looking at biopsies from pancreas and prostate adenocarcinomas, Ruiz *et al.* (2011) found that metastatic tumors were derived from certain clonal populations. In colorectal cancer, Kreso *et al.* (2013) reported that clonal diversity affects chemotherapy tolerance. By tracking 150 lentivirus-marked lineages from 10 human colorectal cancers, they found that previously minor or dormant clones were promoted by chemotherapy, thus, reducing the effectiveness of the treatment.

Although the multi-clonal nature is virtually common to most tumor samples, determining the clonal subpopulations is a challenging process. This problem could potentially be alleviated by single-cell sequencing; however, the current cost of these methods are prohibitive in the scales that would be necessary to representatively sample a tumor tissue. Methods such as fluorescence *in situ* hybridization (FISH) or silver *in situ* hybridization (SISH) can also assess a small number of probes in individual cells of a tumor sample. On the other hand, these methods are quite limited in scope and can not offer the same genome wide perspective as high-throughput sequencing methods.

In silico, separation of the clonal subpopulations may provide a viable alternative to these methods. In a pioneering article, Schwartz and Shackney (2010) developed an unmixing method based on a

geometric model to distinguish a small number of cancer subtypes in gene expression data. After determining cell types and their relative frequencies in different tumor samples, this method infers a phylogenetic tree that best fits the cell types identified. An alternative method, named TrAp (Strino *et al.*, 2013), generates possible phylogenetic trees following certain parsimony and sparsity conditions using a greedy approach. More recently, PhyloSub (Jiao *et al.*, 2014), which is based on Bayesian inference is developed. This method relies on the well-known Monte Carlo Markov Chain (MCMC) sampling paradigm to infer a distribution over all possible phylogenies. Another statistical method, named PyClone (Roth *et al.*, 2014)—also based on MCMC sampling—leverages copy number genotypes to estimate subclonal frequencies. Unlike PhyloSub, however, this method does not infer phylogenies.

In this article, we present a combinatorial algorithm, named clonality inference in tumors using phylogeny (CITUP), that can exploit data obtained from multiple samples from a single patient to infer the tumor phylogeny more accurately. Our framework also involves generating possible phylogenetic trees; unlike the previous approaches mentioned earlier; however, CITUP has the ability to find optimal solutions based on an exact Quadratic Integer Programming (QIP) formulation.

Another tree-based method, named Rec-BTP (Hajirasouliha *et al.*, 2014), is closely related to our framework. In this approach, mutations are subjected to a binary-tree partition, where a binary tree with the least number of conflicting triplets is sought using an approximation algorithm. In contrast to our framework, however, this method can not handle multiple samples.

Our work is also related to other studies with slightly different goals (Oesper *et al.*, 2013; Salari *et al.*, 2013). Although THetA (Oesper *et al.*, 2013) predicts subclonal populations and their proportions given a sample from high-throughput sequencing data, it does not aim to infer any pyhlogenetic relationship between the subclones. Although the method proposed by Salari *et al.* (2013) infers tumor phylogenies from multiple samples like CITUP, the goal of that study is to improve somatic single nucleotide variant (SNV) calls. Moreover, their model places the samples as leaves in a single phylogenetic tree (thus leaves do not represent clonal subpopulations but rather samples, each of which is a mixture of subclones) whereas our model assumes a shared tree between samples with different clonal frequencies.

2 Background

Similar to Jiao *et al.* (2014), we make the infinite sites assumption about tumor evolution: somatic mutations are gained at most once per individual and cannot be lost via a subsequent reversion mutation. Assuming mutations cannot be lost or reverted, a mutation gained in a tumor cell will be present in all of the descendants of that tumor cell. Trivially, a mutation that occurred in the single common ancestor of a tumor will be present in 100% of the tumor cells, while a mutation that occurred in a specific lineage of the tumor phylogeny will be present in a smaller proportion. We refer to the proportion of tumor cells harboring a mutation as the *frequency* of the mutation.

Frequencies of single nucleotide mutations and small indels are measurable with a high degree of confidence using targeted deep sequencing. In brief, the region encompassing the variant is polymerase chain reaction amplified from a bulk tumor sample, and sequenced to high depth ($>1000\times$ coverage). Technological advances now allow many variants to be amplified and sequenced in parallel. For a variant in a diploid heterozygous region of the tumor genome, the allelic ratio of the variant is approximately half the frequency of the mutation. Allele-specific copy number measurements, obtained using sequencing or arrays, can be used to exclude genomic regions that are not diploid heterozygous throughout the population of tumor cells.

Targeted deep sequencing can thus be used to measure the frequency of a mutation in the subpopulation of tumor cells at the specific time point or anatomical site sampled for the experiment. Three problems arise with the phylogenetic interpretation of deep sequencing data: determination of the genome (variant presence/absence) of the major populations of tumor cells, inference of the phylogeny relating those populations, and estimation of the proportion of each population that exists in each sample.

In the following section, we show how to formulate these problems simultaneously as a single combinatorial optimization task and introduce two approaches to solve this task. Then in Section 4, we report the performance of CITUP on simulated datasets followed by two real datasets on chronic lymphocytic leukemia (CLL) and acute myeloid leukemia (AML). We conclude with a discussion on the limitations and utility of our approach in Section 5.

3 Methods

3.1 Combinatorial formulation

Let F be an $|M|\times|S|$ matrix of frequencies measured on the set M of mutations for the set S of samples. For each mutation i and sample s, the corresponding element f_{is} in F is calculated as $(2\cdot q_{\text{var}})/(q_{\text{var}}+q_{\text{ref}})$, where q_{var} is the number of reads with the variant allele (i.e. with the mutation) and q_{ref} is the number of reads with the reference allele (i.e without the mutation) in that sample. Given F, our objective is to simultaneously identify the genotype (i.e. mutational composition) of each subpopulation, the proportion of each subpopulation in each sample and the global phylogenetic relationship relating subpopulations. We impose the same phylogenetic tree structure on all samples.

Let $\mathbb{T}$ represent the space of all rooted trees (see Supplementary Methods Section 1.1 for an explanation of how we derive these trees), and let $T\in\mathbb{T}$ be a hypothetical phylogenetic tree relating $N=|V(T)|$ genetically distinct subpopulations. Let $D(v)$ be the set of descendants of node v. In our formulation, genotypes are represented with nodes (also referred to as subclones in the text) and subtrees rooted at a specific node are named clones. A mutation occurring at a node in the tree is inherited by its descendants. Thus, an assignment of the set of mutations to their node of origin is sufficient to describe the genotypes of all nodes.

Define A to be an $N\times|S|$ matrix which denote the *subclonal proportions* in each sample. An element of A, denoted by α_{vs}, represents the proportion of genotype v in sample s. Subclonal proportions add up to 1 in each sample (Equation 1). Similarly, we define the *clone proportion* β_{vs} as the proportion of the clone rooted at node v in sample s. Clone proportions are related to subclone proportions via the sum rule given in Equation (2).

$$\forall s\in S:\sum_{v\in V}\alpha_{vs}=1 \tag{1}$$

$$\beta_{vs}=\alpha_{vs}+\sum_{u\in D(v)}\alpha_{us} \tag{2}$$

The expected value of the frequency of a mutation is equal to the clone proportion of the node to which the mutation was assigned.

Thus, the squared error incurred by assigning a single mutation i to a node v in sample s is given by:

$$e_{ivs} = (f_{is} - \beta_{vs})^2 \tag{3}$$

Let Δ be an $|M| \times N$ binary matrix such that $\delta_{iv} = 1$ iff mutation i originated at node v, otherwise $\delta_{iv} = 0$. Given $T \in \mathbb{T}$, Δ and A, the total squared error can be written as:

$$E(T, \Delta, A) = \sum_{i \in M} \sum_{s \in S} \sum_{v \in V} \delta_{iv} e_{ivs} \tag{4}$$

Minimization of squared error may result in overfitting, assigning each mutation to a unique node in a large tree. Instead, we minimize the Bayesian information criterion (BIC) under the assumption that the noise is normally distributed with known variance σ^2 (see Supplementary Methods Section 1.2 for details). The negative log likelihood can be expressed (within an additive factor) as:

$$L(F|T, \Delta, A) = \frac{E(T, \Delta, A)}{2\sigma^2} \tag{5}$$

Finally, BIC can be expressed as follows:

$$\mathrm{BIC}(T, \Delta, A) = 2 \cdot L(F|T, \Delta, A) + |S| \cdot (N - 1) \cdot \log |M| \tag{6}$$

We propose to identify the optimal genotypes Δ_{opt}, the subclone proportions A_{opt} and the phylogenetic relationship T_{opt} as given by Equation (7).

$$\Delta_{\mathrm{opt}}, A_{\mathrm{opt}}, T_{\mathrm{opt}} = \arg\min_{T, \Delta, A} \mathrm{BIC}(T, \Delta, A) \tag{7}$$

We refer to the earlier optimization problem as the *mutation phylogeny problem*. We propose two approaches to solve this problem, namely 'CITUP_qip' and 'CITUP_iter'. CITUP_qip uses an exact QIP formulation; while CITUP_iter implements an iterative heuristic. Detailed descriptions of these implementations for solving the mutation phylogeny problem are given below.

3.2 Algorithm overview

Given a fixed tree topology, define the *mutation assignment problem* as the problem of identifying A and Δ that minimize mutation frequency error (Equation 4). CITUP solves the mutation phylogeny problem by iterating through all tree topologies up to a fixed number of nodes $N_{\max}$, and solving the mutation assignment problem for each tree:

1. For each $T \in \mathbb{T}_N$, for each $N \in \{1..N_{\max}\}$
   a. Identify A and Δ that minimizes Equation (4).
   b. Calculate BIC for T using Equation (6).
2. Select T, A and Δ that minimize Equation (7).

We propose two methods for solving the mutation assignment problem: a QIP based approach (CITUP_qip), and an iterative heuristic approach (CITUP_iter) as explained later. Additional details of the algorithm and running configurations can be found in the Supplementary Materials.

3.3 QIP method

QIP-based approaches guarantee an optimal solution but limit the feasible problem size. To ensure a reasonable running time for the QIP approach on larger (>20 mutations) problem sizes, we first cluster the mutations into N sets by their mutation frequency, where N is the number of nodes in the current tree topology. We then limit the solution space for Δ by adding the constraint that all mutations in a cluster must be assigned, en masse, to a single node. We use multi-variate k-means clustering implemented in the python scikit learn package to cluster mutations.

Let $c : M \to \{1..N\}$ be a mapping from mutations to clusters. Let Δ' be an $N \times N$ binary matrix such that $\delta'_{c(i)v} = 1$ iff mutation i assigned to cluster $c(i)$ originated at node v, otherwise $\delta'_{c(i)v} = 0$. The total squared error given by Equation (4) can be rewritten as:

$$E(T, \Delta', A) = \sum_{i \in M} \sum_{s \in S} \sum_{v \in V} \delta'_{c(i)v} e_{ivs} \tag{8}$$

Requiring that each cluster must be assigned to exactly one node adds the constraint given by Equation (9).

$$\forall n \in \{1..N\} : \sum_{v \in V} \delta'_{nv} = 1 \tag{9}$$

Additionally, we require that all non-root nodes must have at least one cluster of mutations assigned to them, resulting in the constraint given by Equation (10),

$$\forall v \in V \setminus \{r\} : \sum_{n \in \{1..N\}} \delta'_{nv} \geq 1 \tag{10}$$

where r denotes the root node.

The QIP approach minimizes the squared error objective (Equation 8), subject to the subclone proportion constraints (Equation 1), the clone proportion constraints (Equation 2) and the cluster assignment constraints (Equations 9 and 10). In practice, we minimize a slightly modified (but equivalent) version of this objective function to speed up the process (Supplementary Methods Section 1.3).

3.4 Heuristic iterative method

We also propose a heuristic iterative method for solving the mutation assignment problem. The iterative heuristic is significantly faster than the QIP with only a small degradation in performance (Supplementary Figs 3 and 4).

In brief, the iterative heuristic solves two subproblems iteratively until convergence. Problem 1: given a fixed Δ calculate the (necessarily unique) A that minimizes Equation (4). Problem 2: with A fixed to the value calculated in the previous step, calculate the Δ that minimizes Equation (4). Each step is guaranteed to not increase the objective given by Equation (4), thus, the algorithm is guaranteed to converge to at least a local optimum.

Problem 1 is a convex quadratic programming problem and can be solved efficiently with existing convex optimization software. The objective given by Equation (4) is solved subject to constraints given by Equations (1) and (2). Problem 2 can be solved by independently assigning each mutation to the node v that minimizes Equation (3).

The iterative heuristic is not guaranteed to identify a globally optimal solution, and as such, results depend heavily on initialization. We mitigate this problem using multiple restarts with random initializations of Δ. A random Δ is generated by independently assigning each mutation to a node, with mutations assigned uniformly and at random to any node in the tree. We perform 1000 restarts with different random seeds and select the solution that maximizes Equation (4).

3.5 Evaluation criteria

We evaluate the performance of CITUP on the simulation sets using several measures. To compute these measures, we first obtain a matching between the predicted tree and the true tree as explained later.

Let $T = (V, E)$ denote the simulated tree, which we are trying to find, and let $T' = (V', E')$ denote the tree predicted by CITUP. We first check whether T and T' have identical topologies as a measure of success, which requires computing the correspondence of the nodes in each tree. To accomplish this, we first create a complete bipartite graph G, where one partition, denoted by A, consists of the nodes of T and the other partition, B consists of the nodes of T'. If $|V| \neq |V'|$, then we add dummy nodes to the partition with the fewer nodes until both partitions have exactly $\max(|V|, |V'|)$ nodes.

We denote by A_i the set of mutations assigned to node i in T. Similarly, we define B_j to be the set of mutations assigned to node j in T'. If i (or j) is a dummy node, then $A_i = \emptyset$ (respectively, $B_j = \emptyset$). For each edge (i, j) in G, we calculate its weight as the number of mutations that are assigned exactly one of i or j. We denote this weight by $c(i, j)$. We then search for a matching $f : A \rightarrow B$ that minimizes $\sum_{i \in A} c(i, f(i))$. This problem is a known as the 'minimum bipartite matching', for which efficient polynomial time algorithms exist (Kuhn, 2010). Once we obtain a one-to-one matching between the nodes of the two trees, we calculate the following scores:

1. *Correct tree proportion:* (**M0**) This is the proportion of correctly identified tree topologies to the total number of simulations in each experiment.
2. *Clone proportion error:* (**M1**) For this measure, we compute: $\left(\sum_{u \in T^*} |\beta_u^{T^*} - \beta_{g(u)}^{T^{**}}|\right) / |V^*|$. Here, T^* denotes smaller of the trees T and T' while T^{**} denotes the larger one. V^* is defined to be the set of nodes in T^* and β_n^X represents the frequency of clone n in tree X. If T^* is the true tree, we define $g \equiv f$. Otherwise, we set $g \equiv f^{-1}$.
3. *Misplaced mutation proportion:* (**M2**) Suppose a mutation m is assigned to a node v in the true tree T. If it is assigned to $f(v)$ in T', we say that m is correctly placed, otherwise we say it is misplaced. M2 is set to the number of misplaced mutations divided by the total number of mutations in the dataset. This measure essentially evaluates the mutation clustering accuracy.
4. *Phylogenetic accuracy:* (**M3**) For this measure, we count the number of phylogenetic relationships that are preserved. We use two types of mutually exclusive relationships: ancestor/descendant and non-ancestor/descendant. For example, if a mutation a emerges at a clone that is an ancestor of another clone where mutation b emerges, we say that a is an ancestor of b (or alternatively b is a descendant of a). If this relationship is reversed in the predictions, it is counted as non-preserved. If two mutations do not have an ancestor/descendant relationship, they are marked as a non-ancestor/descendant pair. If such a pair is predicted to have an ancestor/descendant relationship, this pair is also counted as non-preserved.

4 Results

4.1 Datasets

To evaluate our method, we use both simulated and real datasets. For simulations, we experiment with a variety of trees with differing number of subclones and model parameters. We report the performances of both CITUP_qip and CITUP_iter, using several measures that are explained in the following section. On these simulations, we compare the performance of CITUP to the performances of TrAp (Strino *et al.*, 2013) and PhyloSub (Jiao *et al.*, 2014), which can handle multi-sample datasets. Additionally, we report a separate comparison between CITUP and Rec-BTP (Hajirasouliha *et al.*, 2014) on a smaller set of single-sample simulations. We limit our comparison to these tools because our model does not support the type of input required by Schwartz and Shackney (2010) and Oesper *et al.* (2013). Although the method of Salari *et al.* (2013) also works with SNV data, their model is not directly comparable to ours due to incompatible assumptions and goals.

We also evaluate the utility of our method on two real datasets. The first dataset is taken from a CLL study by Schuh *et al.* (2012). This dataset contains targeted deep sequencing measurements of three CLL patients sampled at five time points. The second dataset consists of a study involving AML patients by Ding *et al.* (2012). This dataset features a large number of somatic indels and SNVs, however, only three sample points (designated as 'normal', 'tumor' and 'relapse') are available per patient. Because the simulations show the QIP and iterative versions to have similar performance, we only report the results of CITUP_qip on the real datasets.

4.2 Evaluation on simulated datasets

We evaluate the performances of CITUP_qip and CITUP_iter compared with TrAp and PhyloSub using a large set of simulations. For these simulations, we generate random tree topologies T with 3–6 subclones with 3–7 samples. The frequencies of subclones are simulated using a Dirichlet distribution with parameter α, ranging from 0.1 to 10.0. For each simulation, we generate a set of 500 mutations that are uniformly distributed to the subclones. The frequencies of these mutations are then altered through an additive Gaussian noise with deviation between 0.02 and 0.1.

We compare each true tree T with the trees obtained by the tools based on the four evaluation criteria introduced earlier. For CITUP_qip, we first cluster the mutations as described in Section 3. As the current version of TrAp does not have a module for clustering and we were unable to run it on the individual mutations, we use our own clustering method for TrAp as well. Because our model selection procedure is unlikely to work with TrAp's heuristic model, we had to provide TrAp with the clustering of the correct size. We emphasize that despite this significant advantage, TrAp performs worse than CITUP with respect to most of our criteria. For PhyloSub and CITUP_iter, we use the individual set of mutations.

Because all four methods can output multiple solutions, we devise the following protocol in order to compute the evaluation measures. For CITUP_qip, CITUP_iter and TrAp, we randomly choose up to three trees out of all (top scoring) solutions reported by the tool. If there are only one or two reported solutions, we pick only these. Because PhyloSub reports three solutions by default, we simply use these solutions. For each tool, if one of the chosen solutions has the correct tree topology, we use that solution to calculate all the measures for that tool. Otherwise, we select one of them randomly. Figure 1 summarizes the results of these simulations. Note that for each selection of parameters, we repeat the experiment 10 times.

The first column of the figure demonstrates the effect of the number of subclones/nodes on all four criteria. The number of nodes vary between 3 and 6—in all cases the number of samples is set to 4, the Gaussian noise deviation is set to 0.05 and the frequency imbalance, as determined by the parameter α of the Dirichlet distribution, is set to 1.0.

The second column demonstrates the effect of the number of samples on the four criteria. The number of samples now vary between 3 and 7—in all cases the number of nodes is set to 5, and again, the Gaussian noise deviation is set to 0.05 and α is set to 1.0.

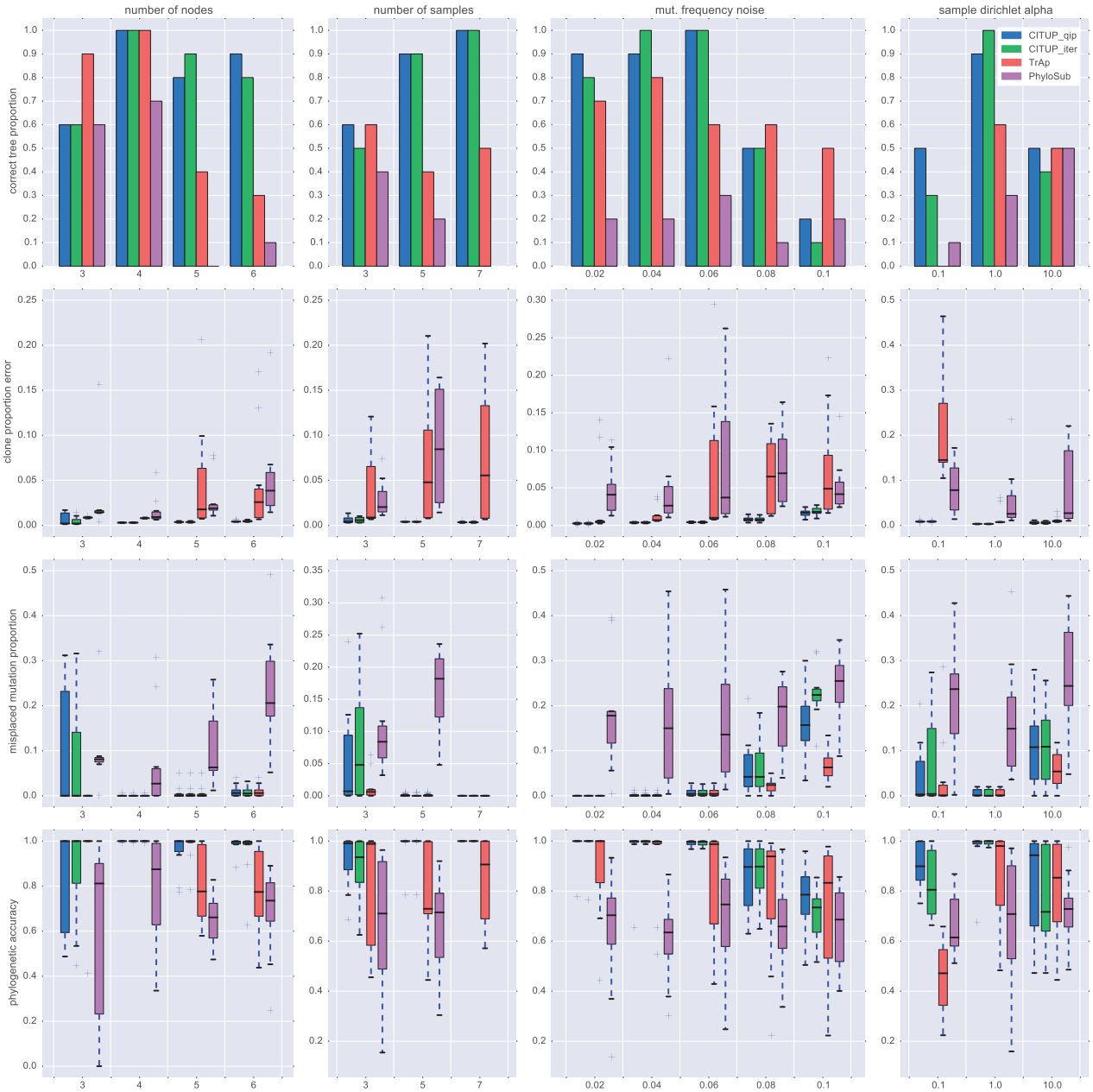

Fig. 1. Simulation results for TrAp, PhyloSub and CITUP (QIP and iterative procedures) under the four evaluation criteria. The rows depict measures **M0–M3**. The first column investigates the effect of the number of subclones/nodes in the dataset, the second investigate the effect of the number of samples, the third investigates the effect of noise added to the mutation frequencies and the fourth investigates the effect of non-uniformity among subclone frequencies. The figure is drawn using the boxplot function in Phyton's mathplot library: the line within each box is the mean and the box boundaries mark the 25 and 75% values. The extreme outliers are depicted with + symbols. Note that we were unable to run PhyloSub on seven samples, so the corresponding bars are absent from this column

We note that we were unable to run PhyloSub for seven samples due to limitations of this software. Hence, in this case the comparison is only between the other methods.

The third column depicts the effect of increasing noise (primarily due to sequence coverage variation). The Gaussian noise deviation now varies between 0.02 and 0.1—for four samples, five subclones and $\alpha = 1.0$. The fourth column depicts the effect of imbalance in subclones where α varies between 0.1 and 10.0, again for four samples, five subclones and noise deviation of 0.05.

From Figure 1, we see that both CITUP_qip and CITUP_iter find the correct tree topology more often than TrAp, despite the fact that TrAp is already provided with the correct number of clusters. In other words, while the other tools have to simultaneously identify the right tree size and topology, TrAp only has to find the right topology of the given tree size. Compared with CITUP and TrAp, PhyloSub performs poorly with respect to this measure. Similarly, CITUP performs typically better than the other tools in terms of phylogenetic accuracy with a score of 60% or more in most cases. This suggests that even when the correct tree is not found, the majority of phylogenetic relationships are preserved.

In estimating clonal frequencies, we see that CITUP outperforms both TrAp and PhyloSub, while TrAp performs best with respect

to the ratio of misplaced mutations. We remark that this is likely due to TrAp's unfair advantage of being given the clustering with the correct number of clusters. Note that this measure is evaluated by a one-to-one matching between the nodes of the predicted and the true tree using only the mutations assigned to (but not inherited by) the node. Hence, even when the predicted topology is not identical to the correct tree, this measure can have a perfect score as long as the initial clustering groups the mutations correctly. This, by definition, can only happen when the clustering is performed with the correct number of clusters. Indeed, Figure 1 shows that whenever CITUP identifies the correct tree topology (hence, the correct tree size) 10 out of 10 times, it performs on par with TrAp. This suggests that TrAp's apparent superiority to CITUP in this measure is simply due to the high accuracy of our clustering method.

Overall, we see that CITUP_qip and CITUP_iter perform similarly under most conditions, although CITUP_qip seems to be slightly more resilient to extreme values of simulation parameters (e.g. sample Dirichlet alpha and mutational frequency noise). Hence, we have chosen to proceed with CITUP_qip for the real datasets.

4.2.1 Comparison with Rec-BTP

We have also performed a separate comparison between CITUP_qip and Rec-BTP. Because Rec-BTP does not support multi-sample datasets, for these experiments we have simulated single-sample datasets with 500 mutations for 4–6 node trees. In each case, we generate 10 simulations adding up to 30 datasets in total. The topologies of the trees were chosen randomly as before. Because the current version of Rec-BTP does not report which mutations are assigned to each subclone, we were restricted to a limited evaluation of the performance of this tool. Briefly, we compare the results of the two methods based on (i) the number of subclones predicted and (ii) an Root Mean Square Deviation (RMSD) measure of the predicted subclonal frequencies similar to the one employed in Hajirasouliha *et al.* (2014). In terms of the first measure, CITUP was able to find the correct number of subclones in 50% of the simulations (15 out of 30). In contrast, Rec-BTP only identified the correct number of subclones in 23.3% of the cases (7 out of 30). CITUP also outperformed Rec-BTP with respect to the RMSD measure: the average RMSD values for CITUP and Rec-BTP in 30 simulations were 0.02 and 0.05, respectively. Further details can be found in Supplementary Results Section 2.5.

4.3 Results on CLL datasets

Next, we evaluate the performance of CITUP_qip on the CLL dataset of Schuh *et al.* (2012). This dataset consists of single nucleotide and small indel mutations as inferred from whole-genome sequencing (WGS) data from three CLL patients. Each patient is sampled at five time points while receiving a variety of treatments. The authors also perform targeted deep sequencing for a limited number of mutations found through WGS. Because the number of mutations are small for these datasets (i.e. only the frequencies of coding mutations were made available), we manually removed mutations that are not heterozygous as reported by Schuh *et al.* (2012).

Table 1 gives a summary of CITUP's performance on all three patients. The trees (Figs. 2, 3 and 4) and the clonal frequencies reported by CITUP for these patients match the results reported by Schuh *et al.* (2012) very closely: the mean absolute deviations are 0.0088, 0.0016 and 0.0048 for patients CLL003, CLL006 and CLL077, respectively. Note that while CITUP does not assign mutations to the root nodes in CLL003 and CLL077, the root node in CLL006 is assigned five mutations. This is in agreement with the observation in Schuh *et al.* (2012) that the normal contamination in this patient is insignificant and suggests that CITUP is able to automatically handle presence or absence of healthy cell contamination.

Table 1. Summary of CITUP's results on the CLL dataset

Patient	No. of mutations	No. of subclones	No. of solutions	Wall-clock time (min)
CLL003	19	5	1	1.64
CLL006	9	5	2	0.32
CLL077	15	5	1	0.84

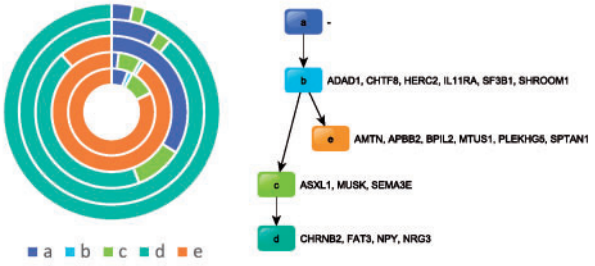

Fig. 2. CITUP predictions for patient CLL003. Left: estimated subclonal proportions for the five time points (ordered from inner to outer circles). Right: the predicted evolutionary tree and the mutations assigned to each subclone. Note that each node is also assumed to inherit mutations that emerge at its ancestors

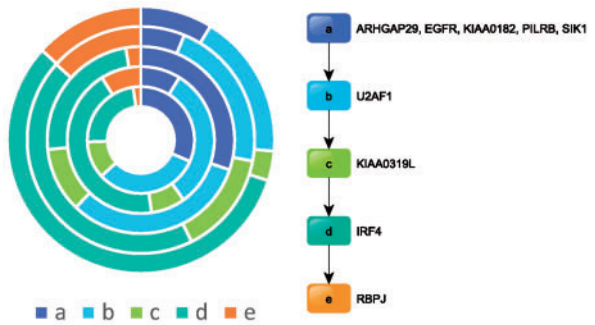

Fig. 3. CITUP predictions for patient CLL006. Left: estimated subclonal proportions for the five time points (ordered from inner to outer circles). Right: the predicted evolutionary tree and the mutations assigned to each subclone

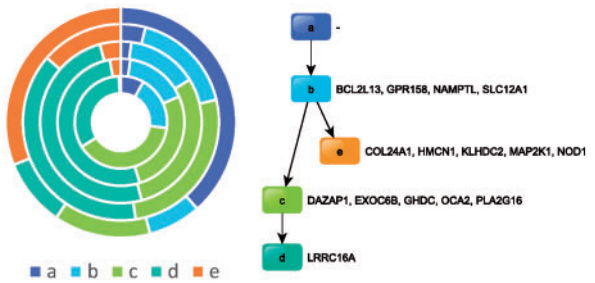

Fig. 4. CITUP predictions for patient CLL077. Left: estimated subclonal proportions for the five time points (ordered from inner to outer circles). Right: the predicted evolutionary tree and the mutations assigned to each subclone

Although CITUP finds two distinct topologies for patient CLL006—a chain topology and a branching topology, the clonal frequencies remain the same in both cases (Supplementary Figs. 8 and 9). We note that the number of deep sequencing mutations is quite small for this dataset, possibly resulting in an ambiguity with respect to the tree topology. To see if additional mutations can help identify the true tree, we also ran CITUP on the WGS predictions for this dataset, containing 16 mutations. In this case, CITUP reported a single solution with a chain topology (data not shown). Thus, we conclude that the true solution is likely to be one reported in Figure 3, which also matches the tree topology predicted by Schuh *et al.* (2012). The full set of predictions for CLL006 and the predicted subclonal frequency values for all three patients can be found in Supplementary Results Section 2.6.

Figure 2 suggests a switch between subclones 'd' and 'e' [referred to as subclones 4 and 2 in Schuh *et al.* (2012)] around time point 3. This is also in agreement with the disease progression as reported by Schuh *et al.* (2012), where the third time point is classified as 'complete response + minimal residual disease'. On the other hand, subclone 'd' simultaneously starts gaining dominance. The fourth and fifth time points (as well as the first two time points) are designated as 'progressive disease' suggesting that subclone 'd' replaces 'e' as the driver subclone while the tumor relapses. In contrast, Figures 4 and 3 imply a more stable subclonal composition over the time points. We note that the survival time of these patients are also longer than CLL003 (6+ and 9 versus 3 years) which may be linked to this slower pace of the clonal dynamics.

4.4 Results on AML datasets

We also evaluate CITUP_qip on an AML dataset (Ding *et al.*, 2012). This dataset contains sequencing data from primary tumor and relapse samples after chemotherapy treatment, in addition to matched normal tissue for each patient. Although the normal tissue is typically obtained to distinguish somatic mutations, we also include it as a sample since some of these tissues can contain various degrees of cancer contamination and thus can be helpful in identifying subclones. Similar to the CLL dataset, we preprocess the mutations based on their copy number analysis as reported by Ding *et al.* (2012). Briefly, we only keep autosomal mutations that are copy number neutral. A summary of CITUP's performance on eight patients taken from this dataset is given in Table 2. The full set of predictions for the AML datasets can be found in Supplementary Materials Section 2.6.

Due to the large number of mutations, CITUP_qip requires considerably more CPU time to run on this dataset compared with the CLL dataset. Nonetheless, we note that CITUP was able to optimize all but two datasets to an exact solution when a wall-clock time limit of 23 h is imposed for each dataset (see Supplementary Fig. 2).

Table 2. Summary of CITUP's results on the AML dataset

Patient	No. of mutations	No. of subclones	No. of solutions	Wall-clock time (hours)
UPN400220	265	7	1	1.71
UPN426980	822	7	1	23.00
UPN452198	97	5	4	0.14
UPN573988	144	3	2	1.02
UPN758168	412	7	2	3.33
UPN804168	589	8	1	6.89
UPN869586	1160	8	1	23.00
UPN933124	270	6	1	3.75

The number of subclones identified per patient is also higher than the number of subclones predicted for CLL patients. We believe this is likely due to the increased ability to detect subclones that differ by non-coding somatic mutations. To investigate this, we have also obtained CITUP_qip results on three of the AML datasets (UPN426980, UPN804168 and UPN869586) using coding mutations only. Although the number of subclones predicted were smaller in all three cases, the overall clonal architecture in the newly predicted trees were typically similar to the trees estimated from the full set of mutations (Supplementary Figs. 27–30). For instance, in UPN426980, the coding-only predictions could be obtained by merging two parent–child subclones (Supplementary Fig. 30).

Although it is unknown whether the non-coding mutations play an important role in cancer progression, some may be hitchhiker mutations which represent subclones that only differ by other types of aberrations such as gene fusions. Furthermore, some non-coding mutations may still be functional; for example, some intronic mutations are known to effect splicing (Lalonde *et al.*, 2011). Thus, we believe that phylogenetic trees derived from the full set of mutations may have better potential to represent the true cancer progression.

Because a full phylogenetic relationship analysis is absent from Ding *et al.* (2012) and the ground truth solutions are not known, we can not directly evaluate our predicted trees. Figure 5 shows, however, that the tumor purities inferred by CITUP generally agree with those reported by Ding *et al.* (2012) for primary and relapse samples. Note that since CITUP does not explicitly predict tumor purity, for each sample this value is estimated as $(1.0 - \alpha_{rs})$, where α_{rs} is the predicted subclonal frequency of the root node in that sample if the root node is not assigned any mutations. Otherwise, the tumor purity is considered to be 1.0 (assuming germline mutations have been excluded from the study).

The only striking difference between the tumor purities inferred by Ding *et al.* (2012) and CITUP is in the relapse sample of patient UPN869586. CITUP prediction for this patient is given in Supplementary Figure 13. The figure suggests that while the founder clone 'b' (and its descendants) is present at a lower abundance in the relapse sample, which may correspond to the tumor purity of 40% reported by Ding *et al.* (2012), CITUP predicts another emerging clone in the relapse sample (i.e. clone 'g'). Although no coding mutations is assigned to clone 'g', we have found that some of the mutations assigned to this clone are located in the intronic regions of several genes including *IL15* and *GPC5*. Interestingly, the tumor purity estimate in the relapse sample using coding-only mutations (Supplementary Fig. 26) for this patient is closer to the purity estimate reported by Ding *et al.* (2012). This is also observed in some of the near-optimal solutions in this dataset (Supplementary Fig. 34).

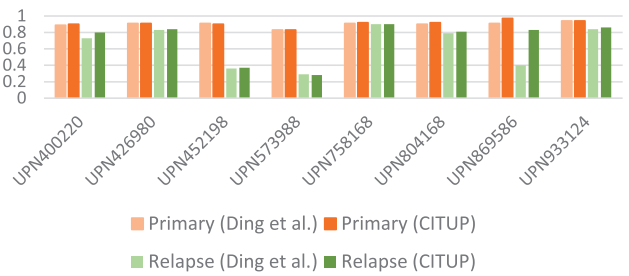

Fig. 5. Tumor purities predicted by Ding *et al.* (2012) and CITUP in primary and relapse samples of AML patients. For the three patients with multiple reported solutions, UPN758168 had the same root frequencies in both solutions. For UPN452198 and UPN573988, we pick the frequencies closest to the ones given in Ding *et al.* (2012)

5 Discussion

In this work, we present CITUP, a novel combinatorial algorithm to determine clonal frequencies in tumors as well as their evolutionary history using one or more samples from the same patient. Our comparisons to other state-of-the-art tools show that CITUP consistently reports fewer solutions and with better accuracy. This feature is important for real cancer datasets where additional experiments may be required to validate the predictions. For example, predictions that involve contradictory assignments reported by TrAp [referred to as 'non-sparse' solutions by Strino *et al.* (2013)], complicate the downstream analysis of identifying potential drivers of cancer. Similarly, the partial order plots reported by PhyloSub (Jiao *et al.*, 2014) can involve many connections, making it difficult to interpret the solutions reported by this tool.

Although our QIP framework is already able to handle a large number of mutations, and significantly faster than PhyloSub we acknowledge that it is considerably slower than TrAp. On the other hand, the iterative heuristic version of CITUP exhibits comparable accuracy, while achieving substantial reduction in computation time (Supplementary Figs. 3 and 4). Moreover, our ability to run CITUP separately on each tree topology means that parallel computing can be utilized to quickly obtain high accuracy results on large datasets.

We note that the current implementation of CITUP is primarily designed for deep sequencing experiments where a high sequence coverage (500–1000×) is implicit. Certain aspects of our model (e.g. Gaussian noise) may not be suitable for low coverage datasets (Supplementary Methods Section 1.2). Although coverage is certainly a limiting factor in detecting rare subclones, CITUP is still able to detect low frequency subclones when model assumptions hold (Supplementary Results Section 2.4). We also acknowledge that although CITUP can theoretically be used to find arbitrarily large trees, this may not be computationally feasible. On the other hand, the number of subclones considered in this work is in accordance with the numbers reported in the literature (Ding *et al.*, 2012; Schuh *et al.*, 2012; Zhang *et al.*, 2014). Moreover, our experiments with larger number of subclones show that CITUP approximates the real trees well even when the number of subclones is limited to smaller values (Supplementary Results Section 2.4).

As mentioned earlier, CITUP assumes infinite sites, which may be violated under certain conditions. For instance, lineages that die out before the first sampling of the tumor or emerge and disappear between two time points are not detectable. In addition, CITUP is only applicable to tumors with limited copy number changes. On the other hand, a reasonable proportion of cancers have low genome instability, making them amenable to analysis with CITUP. For example, in a recent survey of 12 cancer types, Ciriello *et al.* (2013) has found that copy number alterations and mutations are predominant in different subsets of tumors with several solid tumor types such as glioblastoma multiforme and kidney renal clear-cell carcinoma falling under the mutation-heavy class. Furthermore, this limitation of CITUP can be partially overcome by considering a restricted number of copy number corrected genotypes similar to the approach of PyClone (Roth *et al.*, 2014).

Acknowledgements

We thank Andrew Roth from BC Cancer Agency for helpful discussions and our reviewers for their suggestions and comments.

Funding

This work is funded by Genome Canada (BCB/SIP 176ISO) and National Science and Engineering Research Council (NSERC) Discovery (298339) grants to CSS, and NSERC CREATE (139277) fellowship to S.M.

Conflict of Interest: none declared.

References

Ciriello,G. *et al.* (2013) Emerging landscape of oncogenic signatures across human cancers. *Nat. Genet.*, **45**, 1127–1133.

Ding,L. *et al.* (2012) Clonal evolution in relapsed acute myeloid leukaemia revealed by whole-genome sequencing. *Nature*, **481**, 506–510.

Hajirasouliha,I. *et al.* (2014) A combinatorial approach for analyzing intra-tumor heterogeneity from high-throughput sequencing data. *Bioinformatics*, **30**, i78–i86.

Jiao,W. *et al.* (2014) Inferring clonal evolution of tumors from single nucleotide somatic mutations. *BMC Bioinformatics*, **15**, 35–40.

Kreso,A. *et al.* (2013) Variable clonal repopulation dynamics influence chemotherapy response in colorectal cancer. *Science*, **339**, 543–548.

Kuhn,H. (2010) The Hungarian method for the assignment problem. In: M., Junger *et al* (eds.) *50 Years of Integer Programming 1958–2008*. Springer, Berlin, pp. 29–47.

Lalonde,E. *et al.* (2011) RNA sequencing reveals the role of splicing polymorphisms in regulating human gene expression. *Genome Res.*, **21**, 545–554.

Merlo,L.M. *et al.* (2010) A comprehensive survey of clonal diversity measures in barrett's esophagus as biomarkers of progression to esophageal adenocarcinoma. *Cancer Prev. Res.*, **3**, 1388–1397.

Oesper,L. *et al.* (2013) Theta: inferring intra-tumor heterogeneity from high-throughput DNA sequencing data. *Genome Biol.*, **14**, R80–R100.

Roth,A. *et al.* (2014) PyClone: statistical inference of clonal population structure in cancer. *Nat. Methods*, **11**, 396–398.

Ruiz,C. *et al.* (2011) Advancing a clinically relevant perspective of the clonal nature of cancer. *Proc. Natl. Acad. Sci. USA.*, **108**, 12054–12059.

Salari,R. *et al.* (2013) Inference of tumor phylogenies with improved somatic mutation discovery. In: *Proceedings of the 17th International Conference on Research in Computational Molecular Biology*, RECOMB'13, pp. 249–263. Springer-Verlag, Berlin.

Schuh,A. *et al.* (2012) Monitoring chronic lymphocytic leukemia progression by whole genome sequencing reveals heterogeneous clonal evolution patterns. *Blood*, **120**, 4191–4196.

Schwartz,R. and Shackney,S. (2010) Applying unmixing to gene expression data for tumor phylogeny inference. *BMC Bioinformatics*, **11**, 42–61.

Strino,F. *et al.* (2013) Trap: a tree approach for fingerprinting subclonal tumor composition. *Nucleic Acids Res.*, **41**, e165–e179.

Zhang,J. *et al.* (2014) Intratumor heterogeneity in localized lung adenocarcinomas delineated by multiregion sequencing. *Science*, **346**, 256–259.