RNDCLONE: TUMOR SUBCLONE RECONSTRUCTION BASED ON INTEGRATING DNA AND RNA SEQUENCE DATA

BY TIANJIAN ZHOU∗, SUBHAJIT SENGUPTA†, PETER MÜLLER‡ AND YUAN JI∗

University of Chicago∗, NorthShore University HealthSystem† and University of Texas at Austin‡

Tumor cell population consists of genetically heterogeneous subpopulations, known as subclones. Bulk sequencing data using high-throughput sequencing technology provide total and variant DNA and RNA read counts for many nucleotide loci as a mixture of signals from different subclones. We present RNDClone as a tool to deconvolute the mixture and reconstruct the subclones with distinct DNA genotypes and RNA expression profiles. In particular, we infer the number and population frequencies of subclones, as well as subclonal copy numbers, variant allele numbers and gene expression levels by jointly modeling DNA and RNA read counts from the same tumor samples based on generalized latent factor models. Incorporating data at the RNA level provides new insights into intra-tumor heterogeneity in addition to the existing DNA-based inference. Performance of RNDClone is assessed using simulated and real-world datasets, including an analysis of three samples from a lung cancer patient in The Cancer Genome Atlas (TCGA). A potential fatal subclone is identified from the primary tumor, which could explain the rapid prognosis and sudden death of the patient despite a promising diagnosis by conventional standards. The R package RNDClone is available at https://github.com/tianjianzhou/RNDClone.

1. Introduction. We develop a novel framework for statistical inference to understand intra-tumor heterogeneity. Biologically, the proposed approach is the first one to coherently combine information from both DNA and RNA level data using total and variant DNA and RNA read counts. Methodologically, the proposed inference is the first approach to formally cast the question as a generalized factor analysis problem. Adequate inference for tumor heterogeneity is a key ingredient for precision oncology.

1.1. Background. During tumorigenesis, tumor cells acquire and accumulate somatic mutations that give rise to genetically different cell sub-
populations [Nowell (1976), Heppner (1984), Shackleton et al. (2009)]. This phenomenon is known as intra-tumor heterogeneity. Each cell subpopulation, referred to as a subclone, consists of cells that have the same genetic architecture, such as point mutations and copy number aberrations (CNAs). These modifications must exert their effects through downstream cascades of molecular events, such as transcription or translation. Therefore, it is of great interest and importance to understand the downstream functional effects of the genetic modifications. For example, do subclonal DNA mutations affect mRNA expression? Is the effect subclonal as well? With DNA and RNA sequencing data on the same set of tumor samples, it is possible to simultaneously infer the subclones, their genetic architecture, and the impact on mRNA expression.

The problem of subclone reconstruction in this paper is about the identification of the number, population frequencies, genotypes, and gene expression profiles of the subclones from the matched DNA and RNA sequencing data. We take an important step beyond the previous work in subclone reconstruction based on DNA modifications only (Section 1.2), and ask the questions whether and how genetic changes affect transcriptomic landscape in a subclone-specific fashion. Knowledge of tumor subclones is clinically important because it provides information about tumor progression and further suggests personalized treatment strategy [Misale et al. (2012), Landau et al. (2013), Schmitt, Loeb and Salk (2016)]. Through a re-analysis of DNA and RNA data from a deceased stage 1 lung cancer patient, we show that subclone reconstruction might have altered the treatment strategy for the patient and might have potentially avoided the unexpected rapid disease progression that led to the fatality event.

The advent of next-generation sequencing (NGS) technology [Mardis (2008)] has enabled researchers to study the genomic landscape of tumor subclones with greater details. In NGS experiments for DNA or RNA molecules, fragmented DNA or cDNA (complementary DNA for RNA sequencing) molecules are extracted from the tumor cells and are sequenced using short or long reads that are mapped to the corresponding loci in the reference genome. We consider experiments in which DNA and RNA from the same tumor samples are sequenced. In particular, for DNA we consider whole-exome sequencing (WES) and for RNA, we consider RNA sequencing (RNA-seq) which covers the entire transcriptome. Comprehension of Figure 1 is essential in order to follow the upcoming discussion and requires domain knowledge in cancer biology, bioinformatics and statistics. We display four plots in the figure to simplify the discussion, which lay out the biological and statistical problems to be addressed. Figure 1(a) illustrates the biology of tumor subclonal evo-
lution, Figure 1(b) shows the heterogeneous tumor subclones with different somatic point mutations, copy number changes and gene expression levels, Figure 1(c) demonstrates the sequencing data from a tumor sample, and Figure 1(d) shows the statistical quantities (in matrix form) to be inferred that describe the subclone structure. We provide a detailed description of the four plots to motivate our problem next. At the DNA level, point mutations and copy number changes usually drive the subclonal expansion. This is also seen in Figure 1(b), such as the mutation from “C” to “G” at locus 2, with a copy number gain. The relative subclonal gene expression (RSGE) refers to the relative expression level of a certain gene in a specific subclone, which measures the relative abundance of the mRNA molecules produced by that gene. Here, “relative” means that the expression levels of different genes in different subclones are compared with each other. For example, if the RSGE of gene $g_1$ in subclone $c_1$ is two times larger than that of gene $g_2$ in subclone $c_2$, gene $g_1$ in subclone $c_1$ produces two times more mRNA molecules than gene $g_2$ in subclone $c_2$. Figure 1(b) shows hypothetical RSGEs for different genes in different subclones. Figure 1(c) shows that the DNA allele carrying the “G” nucleotide at locus 2 is captured by the short reads, which are mapped to the reference genome. In addition, the RNA-seq data show that both “G” and “C” alleles are expressed at this locus. However, this is not always the case. Consider locus $s = 4$. Although there is a point mutation from “T” to “C” in subclone 3 (Figure 1(b)), the RNA short reads only show the “U” allele but not the “C” allele, potentially indicating that the DNA mutation is not transcribed to RNA. Of course, proper modeling is needed to account for the variability of the data, which is the main goal of this paper. If done properly, using the short read data in Figure 1(c), the statistical model should provide the four matrices in Figure 1(d), which describes the subclonal architecture in both DNA and RNA level.

NGS data are usually over-dispersed and are subject to noise and artifacts. Furthermore, the observed data may be explained by different subclone structures, i.e. multiple solutions exist for the same subclone reconstruction problem. Therefore, proper statistical modeling and assumptions are necessary for valid subclone reconstruction, including a propagation of uncertainties that arise from this ambiguity. In this paper, we propose a Bayesian approach, named RNDClone, to reconstruct tumor subclones by integrating matched DNA and RNA sequencing data from the same tumor samples. We develop sampling models which take into account over-dispersion and noise in the data. We represent subclones as latent factors to capture the co-existence of multiple mutations in the same subclone. Importantly, we allow a simultaneous inference of the RSGEs so that the transcriptional im-
Fig 1: All loci are assumed to reside in exons. Panel (a) illustrates tumor cell evolution and emergence of three subclones from the original normal cell population. A star (in green) or a triangle (in red) represents distinct mutations. Panel (b) shows the genotypes of the three subclones at four loci belonging to three genes, as well as their population frequencies in the two samples. The bold italic letters (in red) represent somatic mutations including point mutations and CNAs. Panel (c) shows a number of DNA and RNA short reads mapped to the four loci in panel (b). The short line segments indicate the alignment of the read relative to the four loci. The bold italic letters (in red) indicate the short reads that bear variant sequences. Panel (d) demonstrates a mathematical representation of the three subclones in panel (b) with four matrices, where $L$ represents copy numbers, $Z$ represents variant allele numbers, $\Lambda$ represents RSGEs, and $W$ represents population frequencies.
RNDClone: Integrating DNA and RNA Sequence Data 5

Some methods are based on RNA sequence data only and handle inter-tumor heterogeneity, including csSAM [Shen-Orr et al. (2010)], CAM [Wang et al. (2016)] and BayCount [Xie, Zhou and Xu (2018)]. Yet, inference on intra-tumor heterogeneity based on integrating DNA and RNA data remains an underexplored topic. A few methods [Wilkerson et al. (2014), Radenbaugh et al. (2014)] have proposed integrated analyses for mutation detection using both DNA and RNA data. We take a different perspective. We aim to reconstruct the subclones that have distinct DNA mutations and RNA expression profiles, potentially better characterizing the impact of subclonal DNA mutations to the downstream RNA expression.

Statistically, we develop an approach based on latent-factor modeling, which has been widely used already in many applications [e.g., West (2003), Carvalho et al. (2008), Bhattacharya and Dunson (2011) and Gao, Brown and Engelhardt (2013)]. The idea of latent-factor modeling has been applied to infer intra-tumor heterogeneity based on DNA sequence data as seen in Zare et al. (2014), Lee et al. (2016), Marass et al. (2016), Zhou et al. (2019a), Zhou et al. (2019b) and Zeng, Warren and Zhao (2019). Interestingly, similar ideas have also been used in RNA subclone inference [Shen-Orr et al. (2010), Wang et al. (2016), Xie, Zhou and Xu (2018)] although the goal there is to infer inter-tumor heterogeneity.

The rest of the paper is structured as follows. In Section 2, we develop a statistical framework for RNDClone, including a sampling model and a prior model. In Section 3, we propose a scheme for posterior inference. In Section 4, we evaluate operating characteristics of RNDClone with two simulation studies. In Section 5, we apply RNDClone to the analysis of three samples from one patient in a lung cancer dataset from The Cancer Genome Atlas (TCGA). Finally, in Section 6, we conclude with a discussion.

2. Statistical Model.

2.1. Notation. We first introduce some notation to represent the observed data. Suppose $T$ tissue samples are dissected from the same patient, obtained either at different time points, at different spatial locations within the same tumor, or at different metastatic sites. Let $s = 1, \ldots, S$ index the loci of the nucleotides (base pairs) that are covered by short reads produced by NGS experiments. The observed data are collected into four $S \times T$ matrices, $N = [Nst], n = [nst], M = [Mst],$ and $m = [mst]$. We denote the data as $D = (N, n, M, m)$ in short. The values of $Nst, nst, Mst,$ and $mst$...
represent the total number of DNA reads, number of DNA reads that bear a variant sequence, total number of RNA reads, and number of RNA reads that bear a variant sequence at locus \( s \) for sample \( t \), respectively. The total read count is also referred to as read depth, and we refer to the reads that bear a variant sequence as \textit{variant reads} in short. In principle, each read can bear any of the four possible nucleotides, A, C, G and T at any locus. However, it is unlikely to observe more than two sequences across short reads at a single locus, as this would require repeated mutations at the same locus. Therefore, we only distinguish the reads according to whether a read possesses a reference or a variant sequence (compared to the reference genome). For example, in Figure 1(c), a total of \( N_{2t} = 5 \) DNA reads and \( M_{2t} = 6 \) RNA reads are mapped to the locus \( s = 2 \). Among all the reads, there are \( n_{2t} = 2 \) DNA variant reads and \( m_{2t} = 4 \) RNA variant reads.

2.2. Representation of Subclones. Next, we introduce a mathematical representation of subclones. Since we only consider intra-tumor heterogeneity where the \( T \) samples are from the same patient, we assume the samples share the same subclones. However, the population frequencies of the same subclone are allowed to vary across different samples. The same assumption is made by most existing methods and is thought to be realistic. Denote by \( C \) the number of subclones, where \( C \) is unknown and needs to be inferred. Tumor samples are in general not pure, in the sense that they contain some proportions of normal cells. Therefore, among the \( C \) subclones, we always include a first subclone of normal cells. The normal subclone does not possess any somatic mutation, as in Figure 1(b).

To represent the gene-level RSGE, suppose the \( S \) nucleotide loci reside in \( G \) genes. We index the genes by \( g = 1, \ldots, G \), where \( g = g(s) : \{1, \ldots, S\} \mapsto \{1, \ldots, G\} \) represents the gene in which nucleotide \( s \) resides. In WES and RNA-seq data, most of the genes possess up to one variant, while a few hyper-mutated genes possess multiple variants.

We encode the underlying subclone structure in the following four matrices, the construction of which is implicitly conditional on \( C \):

1. An \( S \times C \) matrix \( L = [l_{sc}] \), with \( l_{sc} \) representing the subclonal copy number of locus \( s \) in subclone \( c \), \( s = 1, \ldots, S, c = 1, \ldots, C \). The first subclone (normal cells) does not have any CNA, i.e. \( l_{s1} = 2 \) for all \( s \).
2. An \( S \times C \) matrix \( Z = [z_{sc}] \), with \( z_{sc} \) recording the number of variant alleles for locus \( s \) in subclone \( c \), \( z_{sc} \leq l_{sc}, s = 1, \ldots, S, c = 1, \ldots, C \). The first subclone (normal cells) does not have any mutation, i.e. \( z_{s1} = 0 \) for all \( s \).
3. A \( G \times C \) matrix \( \Lambda = [\lambda_{gc}] \), with \( \lambda_{gc} \) representing the RSGE of gene \( g \)
in subclone $c$, $g = 1, \ldots, G$, $c = 1, \ldots, C$.

4. A $T \times C$ matrix $W = [w_{tc}]$, where $w_{tc}$ represents the population frequency of subclone $c$ in sample $t$, $t = 1, \ldots, T$, $c = 1, \ldots, C$. The proportion of tumor cells in a tumor sample, $(1 - w_{t1})$, is called tumor purity.

Using the terminology of latent factor models, $L$, $Z$, $\Lambda$ are essentially factor matrices, and $W$ is the loading matrix. Figure 1(d) demonstrates the mathematical representation of the three subclones in Figure 1(b) using the four matrices. For example, $l_{22} = 3$ represents three copies of the base pair at locus 2 in subclone 2; $z_{22} = 2$ indicates that two out of the three alleles have a variant sequence.

Our goal is to infer the latent quantities of interest, $L$, $Z$, $\Lambda$, $W$ and $C$, from the observed data, $D = (N, n, M, m)$. Taking a Bayesian approach, the desired inference is achieved by sampling from the posterior distribution

$$p(L, Z, \Lambda, W, C | D) \propto p(D | L, Z, \Lambda, W, C)p(L, Z, \Lambda, W, C),$$

where the sampling model $p(D | L, Z, \Lambda, W, C)$ and the prior model $p(L, Z, \Lambda, W, C)$ are to be specified next.

The structure of the full inference model is summarized in Figure 2, which shows how the sampling distribution for the data $(N, n, M, m)$ is indexed by the number of subclones $C$, the latent structure $(L, Z, \Lambda, W)$ that describes the tumor heterogeneity, and additional hyperparameters. In particular, in Figure 2, $l_s = (l_{s1}, \ldots, l_{sC})$, $z_s = (z_{s1}, \ldots, z_{sC})$, $\lambda_g = (\lambda_{g1}, \ldots, \lambda_{gC})$, $w_t = (w_{t1}, \ldots, w_{tC})$, and $\zeta$, $\pi$, $\nu_{D0}$, $\nu_{R0}$, $\nu_{Dt}$, $\nu_{Rt}$, $\gamma_{Dt}$, $\gamma_{Rt}$ and $\phi_t$ represent additional hyperparameters, the meaning of which will be more clear later. We note that our notation is similar to that in Lee et al. (2016).

2.3. Sampling Model. Throughout the paper, we write $E(\cdot)$ for expectation and $\text{Var}(\cdot)$ for variance. We denote by $\text{Bin}(\cdot, \cdot)$ a binomial distribution, $\text{Neg-Bin}(\cdot, \cdot)$ a negative binomial distribution, $\text{Be-Bin}(\cdot, \cdot)$ a beta-binomial distribution, $\text{Be}(\cdot, \cdot)$ a beta distribution, $\text{Ga}(\cdot, \cdot)$ a gamma distribution, $\text{Dir}(\cdot, \ldots, \cdot)$ a Dirichlet distribution, and $\text{Unif}(\cdot, \cdot)$ a uniform distribution. Lastly, we use subscripts $D$ and $R$ to represent parameters related to DNA and RNA, respectively. We specify the sampling model $p(D | L, Z, \Lambda, W, C)$ as follows.

Sampling model for the total number of DNA reads $N_{st}$. We start with the sampling model for $N_{st}$. It is generally assumed that the total number of DNA short reads mapped to a genomic region (i.e. read depth) scales linearly with the number of times the region appears in the DNA sample
(i.e. copy number) [Magi et al. (2011)]. Based on this assumption, previous methods [Klambauer et al. (2012), Lee et al. (2016)] used a Poisson distribution to model $N_{st}$. The Poisson model relies on the assumption that the reads are randomly and independently sampled from any location of the test genome with equal probability, while in reality, the distribution of read counts is typically slightly over-dispersed [Magi et al. (2011)]. To account for this over-dispersion of read counts, we model $N_{st}$ using a negative-binomial distribution. Let $\phi_t$ denote the expected number of DNA reads in sample $t$ if there were no CNAs. We assume

$$ N_{st} \mid \phi_t, A_{st}, \gamma_{Dt} \sim \text{Neg-Bin}(\phi_t A_{st}/2, \gamma_{Dt}), $$

with

$$ \Pr(N_{st} \mid \phi_t, A_{st}, \gamma_{Dt}) = \frac{\Gamma(N_{st} + \gamma_{Dt}^{-1}) \left( \frac{1}{1 + \gamma_{Dt} \phi_t A_{st}/2} \right)^{1/\gamma_{Dt}}}{N_{st}! \Gamma(\gamma_{Dt}^{-1}) \left( 1 + \gamma_{Dt} \phi_t A_{st}/2 \right)^{N_{st}}}.$$ 

We have $E(N_{st}) = \phi_t A_{st}/2$ and $\text{Var}(N_{st}) = E(N_{st}) + \gamma_{Dt} E(N_{st})^2$, with $\gamma_{Dt}$ being a dispersion parameter. Here, $A_{st}$ is the average copy number for locus $s$ in sample $t$ and is modeled as

$$ A_{st} = w_t l_{D0} + \sum_{c=1}^{C} w_{tc} l_{sc}. $$
Specifically, \( w_{t0} \) represents the proportion of a “background” subclone in sample \( t \) with no biological meaning. The background subclone is only used as a mathematical tool to account for tiny subclones that are not detectable or cannot be inferred with sufficient statistical power, and also for noise and artifacts in the NGS data (sequencing errors, mapping errors, etc.). The term \( w_{t0}l_{D0} \) models random noise in the total DNA read counts, where \( l_{D0} \) can be viewed as the copy number in the background subclone. We assume the random noise does not differ across different loci, thus \( l_{D0} \) does not have an index \( s \).

Sampling model for the number of variant DNA reads \( n_{st} \). Conditional on \( N_{st} \), we model \( n_{st} \) as the number of successful trials from a beta-binomial distribution. See, for example, Marass et al. (2016). For each read, let \( \tilde{p}_{st} \) denote the probability that it bears a variant sequence. We assume

\[
\tilde{p}_{st} \mid p_{st}, \nu_{Dt} \sim \text{Be} \left( \nu_{Dt}^{-1} p_{st}, \nu_{Dt}^{-1} (1 - p_{st}) \right)
\]

Here \( \tilde{p}_{st} \) is centered at the variant allele fraction (VAF) in the cell population, denoted by \( p_{st} \), and \( \nu_{Dt} \) controls the variance of the beta distribution. We define \( p_{st} = \bar{A}_{st} / A_{st} \), where \( \bar{A}_{st} \) is the average number of variant alleles for locus \( s \) in sample \( t \), defined as

\[
\bar{A}_{st} = w_{t0}z_{D0} + \sum_{c=1}^{C} w_{tc}z_{sc}.
\]

To see this, recall that \( \bar{A}_{st} \) is the average number of variant alleles, and \( A_{st} \) is the average number of alleles. Therefore, the ratio \( \bar{A}_{st} / A_{st} \) is the fraction of variant alleles, i.e., the VAF. The term \( w_{t0}z_{D0} \) is used to account for random noise in variant DNA read counts, and \( z_{D0} \) can be viewed as the number of variant alleles in the background subclone. Again, we assume the random noise does not differ across different loci, thus \( z_{D0} \) does not have an index \( s \). Integrating out \( \tilde{p}_{st} \), \( n_{st} \) follows a beta-binomial distribution given by

\[
n_{st} \mid N_{st}, p_{st}, \nu_{Dt} \sim \text{Be-Bin} \left( N_{st}; p_{st}, \nu_{Dt} \right),
\]

with

\[
\Pr(n_{st} \mid N_{st}, p_{st}, \nu_{Dt}) = \frac{N_{st}!}{n_{st}!(N_{st} - n_{st})!} \times \frac{\Gamma(\nu_{Dt}^{-1}) \Gamma(\nu_{Dt}^{-1} p_{st} + n_{st}) \Gamma(\nu_{Dt}^{-1} (1 - p_{st}) + N_{st} - n_{st})}{\Gamma(\nu_{Dt}^{-1} p_{st}) \Gamma(\nu_{Dt}^{-1} (1 - p_{st}) + N_{st})},
\]
where \( E(n_{st} \mid N_{st}, p_{st}, \nu_{Dt}) = N_{st} p_{st} \) and \( \text{Var}(n_{st} \mid N_{st}, p_{st}, \nu_{Dt}) = N_{st} p_{st} (1 - p_{st}) (1 + N_{st} \nu_{Dt}) / (1 + \nu_{Dt}) \).

**Sampling model for the total number of RNA reads \( M_{st} \).** Next, we model \( M_{st} \). It follows a similar construction as \( N_{st} \). Suppose that the RSGE of gene \( g \) in subclone \( c \) is \( \lambda_{gc} \). We are only concerned about the relative expression levels. For example, \( \lambda_{g1c1} > \lambda_{g2c2} \) means that gene \( g_1 \) in subclone \( c_1 \) produces more RNA molecules than gene \( g_2 \) in subclone \( c_2 \). The average number of RNA copies in sample \( t \) that contain locus \( s \) is thus

\[
B_{st} = w_{0t} \lambda_{g(s)} l_{R0} + \sum_{c=1}^{C} w_{tc} \lambda_{g(s)c} l_{sc}.
\]

As before, we include \( w_{0t} \lambda_{g(s)} l_{R0} \) to account for random noise in the total RNA read counts. We allow the random noise to be different for DNA and RNA reads by allowing \( l_{R0} \) and \( l_{D0} \) to be different.

Again, assuming that the total number of RNA reads mapped to a genomic region has a linear relationship with the number of RNA copies of that region, we model \( M_{st} \) using a negative-binomial distribution,

\[
M_{st} \mid \psi_t, B_{st}, \gamma_{Rt} \sim \text{Neg-Bin}(\psi_t B_{st}/2, \gamma_{Rt}).
\]

Here \( \psi_t \) is the expected number of RNA reads at locus \( s \) in sample \( t \) if there were no CNAs at the locus and the RSGE for gene \( g(s) \) was 1.

**Sampling model for the number of variant RNA reads \( m_{st} \).** Finally, we model \( m_{st} \) conditional on \( M_{st} \). The average RNA copies in sample \( t \) that bear a variant sequence at locus \( s \) is given by

\[
\tilde{B}_{st} = w_{0t} \lambda_{g(s)0} l_{R0} + \sum_{c=1}^{C} w_{tc} \lambda_{g(s)c} z_{sc},
\]

where \( w_{0t} \lambda_{g(s)0} l_{R0} \) accounts for random noise in variant RNA read counts. We model \( m_{st} \) with a beta-binomial distribution,

\[
m_{st} \mid M_{st}, q_{st}, \nu_{Rt} \sim \text{Be-Bin}(M_{st}; q_{st}, \nu_{Rt}),
\]

where \( q_{st} \) is the fraction of RNA copies in sample \( t \) that bear a variant sequence at locus \( s \), \( q_{st} = \tilde{B}_{st} / B_{st} \).

**Generalized factor analysis.** Models (2.1), (2.2), (2.4), and (2.5) can be characterized as generalized latent factor models, where each observation
(N_{st}, n_{st}, M_{st}, m_{st}) is modeled by \( C \) interpretable and biologically meaningful latent factors, \((l_{sc}, z_{sc}, \lambda_{g(s)c})\) for \( c = 1, \ldots, C \), with the \( w_{tc} \) coefficients playing the role of factor loadings. In particular,

\[
E \left( \begin{pmatrix} N \\ n \\ M \\ m \end{pmatrix} \right) = f^{-1} \left[ \begin{pmatrix} L \\ Z \\ \Lambda^* \circ L \\ \Lambda^* \circ Z \end{pmatrix} \right] W^T,
\]

where \( \Lambda^* \) is a \( S \times C \) matrix with \( \lambda_{sc}^* = \lambda_{g(s)c} \), symbol \( \circ \) refers to the Hadamard product, i.e., entry-wise product, and \( f^{-1} \) is a linear transformation as in Equations (2.1), (2.2), (2.4) and (2.5).

Understanding the model as an instance of factor analysis, one might wonder why the factors \( L \) and \( Z \) should be constrained to integer scores. We strongly prefer the restriction to integer scores to maintain the interpretability of Equations (2.1) through (2.5) as mapping the relevant biology. While a relaxation would likely improve mixing of posterior Markov chain Monte Carlo simulation, the loss in ease of communication can not be justified by minor gains in computational efficiency, considering also that inference need not be carried out in real time.

2.4. Prior Model. We build a hierarchical prior model for the unknown parameters, including the key quantities of interest, \( L, Z, \Lambda, W \) and \( C \).

Prior for \( C \). We start with a truncated geometric prior for \( C \),

\[ Pr(C) \propto (1 - \alpha)^{C-1} \alpha, \quad 1 \leq C_{\text{min}} \leq C \leq C_{\text{max}}. \]

The hyperparameter \( \alpha \) (\( 0 < \alpha < 1 \)) implies a prior preference for a parsimonious model; larger \( \alpha \) represents more shrinkage. We assume that \( C \) is a priori restricted to \( C_{\text{min}} \leq C \leq C_{\text{max}} \). The bounds are only used to simplify implementation. Since we only consider heterogeneous cell samples, we can set \( C_{\text{min}} = 2 \). When computational resources allow, \( C_{\text{max}} \) can be set sufficiently large. Empirically, we find that the maximum number of subclones that can be inferred with sufficient statistical power is usually limited by the number of samples and sequencing depth. With the usual sequencing depths in WES experiments, setting \( C_{\text{max}} = T + 4 \) suffices, where \( T \) is the number of tissue samples. This can also be understood from the latent factor model perspective. The number of factors \( (C) \) that can be reliably inferred is usually less than the rank of the data matrix, which is bounded by the number of samples here.
Prior for $L$. Next, we construct the prior model for the copy numbers $L$. The first column of $L$ represents the normal subclone and is fixed at $l_{s1} = 2$ for all $s$. For locus $s$ of subclone $c (c \neq 1)$, we assume a truncated geometric-type prior for $l_{sc}$,

$$ \Pr(l_{sc} = k \mid \pi_c, C) \propto (1 - \pi_c)^{|k-2}| \pi_c, \quad 0 \leq K_{\text{min}} \leq k \leq K_{\text{max}}. $$

(2.6)

Here, $\pi_c$ is the probability for subclone $c$ having a copy number 2 at a locus, and $(1 - \pi_c)$ is the probability of a copy number change. We assume a beta prior,

$$ \pi_c \mid C \sim \text{Be}(a_\pi, b_\pi). $$

The prior bounds of $l_{sc}$, $K_{\text{min}}$ and $K_{\text{max}}$, are only used to simplify implementation. Suitable values for $K_{\text{min}}$ and $K_{\text{max}}$ can be explored by running copy number callers on the WES data or comparing the minimum and maximum read depths with the mean read depth. When such information is not available, we recommend setting $K_{\text{min}} = 0$ and choosing a sufficiently large $K_{\text{max}}$.

Prior for $Z$. Conditional on $L$, we define the prior model for the variant allele numbers $Z$, subject to $0 \leq z_{sc} \leq l_{sc}$. The first column of $Z$ corresponds to the normal subclone and is fixed at $z_{s1} = 0$ for all $s$. For locus $s$ of subclone $c (c \neq 1)$, we assume a truncated geometric prior for $z_{sc}$,

$$ \Pr(z_{sc} = k \mid l_{sc}, \zeta_c, C) \propto (1 - \zeta_c)^k \zeta_c, \quad 0 \leq k \leq l_{sc}, $$

and $z_{sc} = 0$ if $l_{sc} = 0$. Here $(1 - \zeta_c)$ is the probability of observing a mutation in subclone $c$ at a locus. We put a beta distribution prior on $\zeta_c$,

$$ \zeta_c \mid C \sim \text{Be}(a_\zeta, b_\zeta). $$

The prior for $L$ penalizes for deviations from copy number 2. Similarly, the prior for $Z$ penalizes for large numbers of variant alleles. Such prior specifications impose some sparsity structure on the latent factors to improve the identifiability of the latent subclones, especially when the number of samples ($T$) is small. If desired, the prior models could be modified without changing anything in the rest of the discussion. For example, one may use discrete uniform priors for $l_{sc}$ and $z_{sc}$.

Prior for $\Lambda$. The RSGEs $\Lambda$ should be non-negative, $\lambda_{gc} \geq 0$. We assume a gamma prior,

$$ \lambda_{gc} \mid C \overset{iid}{\sim} \text{Ga}(a_\lambda, b_\lambda). $$
Prior for \( W \). The population frequencies of the subclones satisfy \( \sum_{c=0}^{C} w_{tc} = 1 \) for all \( t \). Recall that \( w_{t1} \) stands for the proportion of the normal subclone (subclone 1), i.e. (1 - tumor purity). In addition, \( w_{t0} \) is the proportion of the “background” subclone that is only used to model tiny subclones and noise. We assume a beta-Dirichlet prior on \( w_t \), such that

\[
  w_{t1} \mid C \sim \text{Be}(a_w, b_w), \quad \text{and}
  (w_{t0}, w_{t2}, \ldots, w_{tC})/(1 - w_{t1}) \mid w_{t1}, C \sim \text{Dir}(d_0, d, \ldots, d).
\]

We set \( d_0 \ll d \) to reflect the nature of the background subclone. Informative prior can be elicited for \( w_{t1} \) based on information from some tumor purity caller.

Hyperpriors. We complete the model with priors for \( l_{D0}, z_{D0}, l_{R0}, z_{R0}, \phi_t, \psi_t, \gamma_{Dt}, \nu_{Dt}, \gamma_{Rt}, \nu_{Rt} \) and \( \lambda_{g0} \). We assume

\[
  l_{D0} \sim \text{Unif}(K_{\min}, K_{\max}), \quad z_{D0} \mid l_{D0} \sim \text{Unif}(0, l_{D0}),
  \phi_t \sim \text{Ga}(a_{\phi_t}, b_{\phi_t}), \quad \psi_t \sim \text{Ga}(a_{\psi_t}, b_{\psi_t}),
  \gamma_{Dt} \overset{iid}{\sim} \text{Ga}(a_{\gamma_{Dt}}, b_{\gamma_{Dt}}), \quad \nu_{Dt} \overset{iid}{\sim} \text{Ga}(a_{\nu_{Dt}}, b_{\nu_{Dt}}),
  \gamma_{Rt} \overset{iid}{\sim} \text{Ga}(a_{\gamma_{Rt}}, b_{\gamma_{Rt}}), \quad \nu_{Rt} \overset{iid}{\sim} \text{Ga}(a_{\nu_{Rt}}, b_{\nu_{Rt}}).
\]

Among these parameters, informative priors are necessary for \( \phi_t \) and \( \psi_t \). For example, the parameters \((\psi_t, \lambda_{gc})\) and \((\psi_t/a, a \cdot \lambda_{gc})\) lead to exactly the same data likelihood for any positive constant \( a \). To help identify the parameters, we can match the prior means of \( \phi_t \) and \( \psi_t \) with the mean DNA and RNA read depths in copy number neutral regions, respectively. This can be done by running a bioinformatics copy number caller on the DNA sequencing data. If such information is not available, we recommend setting the priors for \( \phi_t \) and \( \psi_t \) such that \( E(\phi_t) \) and \( E(\psi_t) \) are centered at the mean DNA and RNA read depths, respectively. Informative priors can also be elicited for the overdispersion parameters, \( \phi_t, \psi_t, \gamma_{Dt}, \nu_{Dt}, \gamma_{Rt}, \nu_{Rt} \). For example, the variance of total DNA read count is \( \text{Var}(N_{st}) = E(N_{st}) + \gamma_{Dt}E(N_{st})^2 \). We can choose the prior for \( \gamma_{Dt} \) according to the dispersion of \( N_{st} \) in copy number neutral regions. Finally, we set \( \lambda_{g0} = \left( \sum_{c=1}^{C} w_{tc} \lambda_{gc} \right) / \left( \sum_{c=1}^{C} w_{tc} \right) \) to let the magnitude of the random noise in the total RNA read counts (Equation 2.3) match the average expression level.

The RNDClone hierarchical model is summarized in Figure 2 using a graphical model representation.
2.5. Special Cases. In special cases, RNDClone reduces to a model based on DNA or RNA data only.

**DClone.** When only DNA data \((N, n)\) are available, we can still infer \(L, Z, W, C\) and DNA-related hyperparameters such as \(\phi_t, \gamma_{D}, \nu_{D}\). We cannot estimate \(\Lambda\) and RNA-related hyperparameters. In this case, we refer to our model as DClone. DClone is similar to some existing DNA-based methods such as Zeng, Warren and Zhao (2019) and Lee et al. (2016).

**RClone.** Typically, RNA-based methods only take as input the total RNA counts \(M\) and do not consider variant RNA counts. When only total RNA counts are available, we can still infer \(\Lambda, W, C\) and some RNA-related hyperparameters such as \(\psi_t, \gamma_R\). We cannot estimate \(L, Z\) and DNA-related hyperparameters. Without loss of generality, we can set \(l_{sc} = 2\) and \(z_{sc} = 0\) for all \(s\) and \(c\). In this case, we refer to our model as RClone. RClone is similar to some existing RNA-based methods such as Xie, Zhou and Xu (2018) although the goal there is to infer inter-tumor heterogeneity.

3. Posterior Inference. Taking a Bayesian approach, inference on the quantities of interest is contained in their posterior distribution. We use Markov chain Monte Carlo (MCMC) simulations to draw \(J\) samples \(\{L(j), Z(j), \Lambda(j), W(j), C(j), \ldots; j = 1, \ldots, J\}\) from the posterior distribution. Trans-dimensional MCMC and parallel tempering are needed to ensure proper convergence. The exact form of the posterior distribution is described in Supplementary Section S.1.1.

**Trans-dimensional MCMC.** Let \(x = (L, Z, \Lambda, W, \pi, \zeta, \phi, \psi, \gamma_D, \nu_D, \gamma_R, \nu_R, l_{D0}, z_{D0}, l_{R0}, z_{R0})\) denote all unknown parameters except the random number of subclones \(C\). Sampling \((x, C)\) involves trans-dimensional MCMC [Green (1995)], as the dimensions of \(L, Z, \Lambda, W, \pi\) and \(\zeta\) depend on \(C\). Prior to each MCMC transition, denote the current state by \((x, C)\). We propose a new \(\tilde{C}\) from \(q(\tilde{C} | C)\). We use a uniform proposal, \((\tilde{C} | C) \sim \text{Unif}(C_{\min}, \ldots, C_{\max})\). Next, we propose a new \(\tilde{x}\) whose dimension is consistent with \(\tilde{C}\) from \(q(\tilde{x} | \tilde{C}) = p_{\tau}(\tilde{x} | \tilde{C})\), where

\[
(3.1) \quad p_{\tau}(x | C) \propto p(x | C) p(D | x, C)^{\tau}, \quad 0 \leq \tau \leq 1.
\]

The proposal of \(\tilde{x}\) is motivated by the idea of power prior [Ibrahim and Chen (2000)], who used a fraction \(\tau\) of the historical data likelihood to define an informative prior based on historical data. The acceptance probability of the proposal \((\tilde{x}, \tilde{C})\) is calculated by

\[
p_{\text{acc}}(x, C, \tilde{x}, \tilde{C}) = 1 \wedge \frac{p(D | \tilde{x}, \tilde{C})^{1-\tau} p_{\tau}(\tilde{x} | \tilde{C}) p(C | \tilde{C})}{p(D | x, C)^{1-\tau} p_{\tau}(x | C) p(C)} \cdot \frac{q(\tilde{x} | \tilde{C}) q(C | \tilde{C})}{q(x | C) q(\tilde{C} | C)}.
\]
Here, $a \wedge b$ represents the minimum of $a$ and $b$. In summary, the effect of using $q(\tilde{x} \mid \tilde{C}) = p_{\tau}(\tilde{x} \mid \tilde{C})$ is to replace the original posterior ratio in the acceptance probability with a fractional $(1 - \tau)$ power likelihood ratio.

The purpose of using the power prior and power likelihood is to achieve reasonable acceptance probabilities for the proposals and a well mixing Markov chain. Since the likelihood is highly informative, commonly used trans-dimensional proposals such as the split-merge proposals [Richardson and Green (1997)] have a very low chance of being accepted and result in very slowly mixing Markov chains. On the other hand, proposals from the power prior are more likely to be accepted. Importantly, the conditional posterior of $x$ given $C$ under this trans-dimensional MCMC is exactly the same as what under the original model. Details in Supplementary Section S.1.2.

In contrast to many existing methods (such as Marass et al., 2016) which use model selection criteria to choose the optimal $C$, here, we use trans-dimensional MCMC to quantify the uncertainty associated with the estimate of $C$. The marginal posterior of $C$ implies a model comparison among different $C$’s.

Parallel tempering and Gibbs sampler. The previously described trans-dimensional MCMC scheme requires sampling from $p_{\tau}(x \mid C)$ in (3.1). We use (separate, up-front) MCMC simulation to generate from $p_{\tau}(x \mid C)$. However, the posterior surface of $p_{\tau}(x \mid C)$ is expected to be highly multi-modal. We therefore use parallel tempering [Geyer (1991)] to further improve the mixing of the Markov chain. Consider $I$ parallel Markov chains with decreasing temperatures $\{\Delta_1, \Delta_2, \ldots, \Delta_I\}$, where $\Delta_I = 1$. Let $x_i$ denote the state of the $i$-th chain. The target distribution for the $i$-th chain is

$$p_{\tau,i}(x_i \mid C) \propto p(x_i \mid C) p(D \mid x_i, C)^{\tau / \Delta_i},$$

thus the target distribution of the $I$-th chain is the original target distribution $p_{\tau}(x \mid C)$. At each MCMC iteration, we first independently update all $I$ chains. Gibbs sampling transition probabilities are used to update $x_i$. Details of the full conditionals are in Supplementary Section S.1.3. Then, for $i = 1, 2, \ldots, I - 1$, we propose a swap between $x_i$ and $x_{i+1}$ and accept the proposal with probability

$$p_{\text{swap}}(x_i, x_{i+1}) = 1 \wedge \left[ \frac{p(D \mid x_{i+1}, C)}{p(D \mid x_i, C)} \right]^{\frac{1}{\Delta_i} - \frac{1}{\Delta_{i+1}}}.$$

The value from the $I$-th chain, $x_I$, is kept.

To ensure that the Markov chain of $x$ reaches the stationary distribution, for every possible $C \in \{C_{\min}, \ldots, C_{\max}\}$, we run a sufficiently large number
Point estimate. Suppose we have obtained $J$ posterior samples of $(x, C)$, \{$(x^{(j)}, C^{(j)})$, $j = 1, \ldots, J$\}. As a point estimate for the number of subclones, $C$, we report the estimated posterior mode, $\hat{C} = \text{Mode}(\{C^{(j)}, j = 1, \ldots, J\})$. Conditional on $\hat{C}$, we again use the posterior mode as a point estimate for $x$, i.e., $\hat{x} = x^{(j)}$, where

$$j = \arg\max_j p(D | x^{(j)}, \hat{C})p(x^{(j)} | \hat{C}),$$

where the maximization is over all iterations $j$ with $C^{(j)} = \hat{C}$. That is, we report $\hat{x}$ based on the maximum a posteriori (MAP) estimator.

4. Simulation Studies. We carry out simulation studies to explore the power of RNDClone to recover under realistic sample size and signal the true number of subclones $C$, copy numbers $L$, variant allele numbers $Z$, RSGEs $\Lambda$ and cellular proportions $W$. We define subclone reconstruction errors by comparing the point estimates $\hat{C}, \hat{L}, \hat{Z}, \hat{\Lambda}$ and $\hat{W}$ with the simulation truth, similar to Marass et al. (2016). Let $C_{\text{err}} = |\hat{C} - C|$, $L_{\text{err}} = \frac{1}{S(C - 1)} \left( \sum_{s,c} I(\hat{l}_{\sigma(c)} \neq l_{sc}) \right)$, $Z_{\text{err}} = \frac{1}{S(C - 1)} \left( \sum_{s,c} I(\hat{z}_{\sigma(c)} \neq z_{sc}) \right)$, $\Lambda_{\text{err}} = \sum_{g,c} |\hat{\lambda}_{g\sigma(c)}^{\text{std}} - \lambda_{gc}^{\text{std}}|/(GC)$, and $W_{\text{err}} = \sum_{t,c} |\hat{w}_{t\sigma(c)} - w_{tc}|/(TC)$. Here $\sigma$ is a permutation of subclones that minimizes $Z_{\text{err}}$ to account for label-switching of subclones. The value $\lambda_{gc}^{\text{std}} = \lambda_{gc}/sd(\lambda_{gc})$ is the standardized RSGE. The reason for standardizing the RSGEs is to put them on the same scale to allow for comparison among different datasets. In some cases, RNDClone may fail to identify the correct number of subclones $C$. To avoid comparing two matrices with different dimensions and to ease the calculation of $L_{\text{err}}, Z_{\text{err}}, \Lambda_{\text{err}}$ and $W_{\text{err}}$, we always report point estimates for $\hat{L}, \hat{Z}, \hat{\Lambda}$ and $\hat{W}$ conditional on the correct $C$ in the simulation studies.

For all simulation studies, we fit the model with the following hyperparameters. We set $\alpha = 0.8$, $a_{\pi} = C - 1$, $b_{\pi} = 1$, $a_{\zeta} = C - 1$, $b_{\zeta} = 1$, $a_w = 1$, $b_w = 1$, $d = 1$, $d_0 = 0.03$, $a_{\lambda} = 1$, $b_{\lambda} = 1$, $b_{\phi_t} = 10$, $b_{\psi_t} = 10$, $a_{\gamma_d} = 1$, $a_{\nu_d} = 1$, $a_{\gamma_R} = 1$, and $a_{\nu_R} = 1$. Following the guidelines that we have introduced in

$J_0$ of burn-in iterations. A draw of $x$ after the burn-in period can be seen as a draw from $p_r(x | C)$. The detailed MCMC scheme is summarized in Supplementary Algorithm S.1.
Section 2.4, we set $a_{\phi_t}$ and $a_{\psi_t}$ such that $E(\phi_t)$ and $E(\psi_t)$ match the mean DNA and RNA read depths in copy number neutral regions, respectively; we set $b_{\gamma_D}$ and $b_{\nu_d}$ equal to 10 times the average DNA read depths, and $b_{\gamma_D}$ and $b_{\nu_d}$ equal to 10 times the average RNA read depths. We set $C_{\min} = 2$ and $C_{\max} = 7$ as the range of $C$. Based on empirical calibration, we set the power of the likelihood in the power prior $\tau = 0.99$ (Equation 3.1). We run MCMC simulation for 25,000 burn-in iterations and 5,000 trans-dimensional transitions.

4.1. Simulation 1. In simulation 1, we assess RNDClone under multiple scenarios with a range of values for the number of subclones ($C$), number of samples ($T$), average read depth ($E(\phi_t)$ and $E(\psi_t)$) and maximum copy number ($\text{max}(l_{sc})$).

Simulation 1(a). First, we validate RNDClone on simulated datasets with a range of values for the number of subclones ($C$) and samples ($T$). We consider 9 scenarios, one for each combination of $C \in \{3, 4, 5\}$ and $T \in \{3, 4, 5\}$. Suppose there are $S = 100$ loci. The average DNA and RNA read depths for sample $t$ in copy number neutral regions are generated from $\phi_t \sim \text{Ga}(200, 1)$ and $\psi_t \sim \text{Ga}(200, 1)$, with $E(\phi_t) = E(\psi_t) = 200$. For simplicity, we use in the simulation truth copy numbers ranging from 1 to 3, $l_{sc} \in \{1, 2, 3\}$. Accordingly, we set $K_{\min} = 1$ and $K_{\max} = 3$ as prior bounds for $l_{sc}$ (Equation 2.6). Next, typically, data include for each gene only one locus that carries a mutation. Mimicking this, we assume that the 100 loci span $G = 91$ genes, with 84 genes contain one locus, 5 genes contain two loci, and 2 genes contain three loci. The RSGEs $\lambda_{gc}$ are randomly generated from a gamma distribution, $\lambda_{gc} \sim \text{Ga}(1, 1)$. Finally, for each scenario, 50 hypothetical datasets $D = (N, n, M, m)$ are generated from the assumed sampling models (2.1), (2.2), (2.4) and (2.5). More details on the simulation parameters are reported in Supplementary Section S.2.1.

Table 1 reports the reconstruction errors under the 9 scenarios, averaged over the repeatedly simulated datasets. In all 9 cases, inference under RNDClone attains low reconstruction errors. The reconstruction errors on $C$, $Z$ and $W$ can be directly compared to those reported in Marass et al. (2016) (for example, Fig. 4). In general, the errors given by RNDClone are comparable with those in Marass et al. (2016), despite that RNDClone uses a more complex model which considers copy number aberrations and gene expression levels. We note that the reconstruction errors depend on the complexity of the subclone structure, for example, the number of subclones and the similarity among subclones. In general, the simulation truth is better recovered when the number of subclones $C$ is smaller (relative to the number
of samples \( T \). It is also easier to distinguish multiple subclones when their genotypes and gene expression profiles are less similar. Table 1 also reports the computation times for RNDClone (using an Intel E5-2680 v4 2.40 GHz processor). Running time increases with the number of samples \( T \).

### Table 1

**Simulation 1(a).** Reconstruction errors and computation times (in minutes) of RNDClone under 9 scenarios, one for each combination of \( C \in \{3,4,5\} \) and \( T \in \{3,4,5\} \). Values shown are averages over 50 repeat simulations, with numerical Monte Carlo standard errors in subscripts.

<table>
<thead>
<tr>
<th>( C )</th>
<th>( T )</th>
<th>( C_{err} )</th>
<th>( L_{err} )</th>
<th>( Z_{err} )</th>
<th>( \lambda_{err} )</th>
<th>( W_{err} )</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3</td>
<td>0.000.000</td>
<td>0.0190.001</td>
<td>0.0180.001</td>
<td>0.0980.001</td>
<td>0.0080.000</td>
<td>141</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0.000.000</td>
<td>0.0230.001</td>
<td>0.0230.001</td>
<td>0.0910.001</td>
<td>0.0080.000</td>
<td>171</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>0.080.004</td>
<td>0.0250.001</td>
<td>0.0250.001</td>
<td>0.0850.001</td>
<td>0.0070.000</td>
<td>203</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>0.000.000</td>
<td>0.0200.001</td>
<td>0.0020.000</td>
<td>0.2710.003</td>
<td>0.0050.000</td>
<td>142</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>0.000.000</td>
<td>0.0160.001</td>
<td>0.0010.000</td>
<td>0.2120.002</td>
<td>0.0050.000</td>
<td>174</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>0.000.000</td>
<td>0.0120.001</td>
<td>0.0030.001</td>
<td>0.1910.002</td>
<td>0.0070.000</td>
<td>203</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>0.000.000</td>
<td>0.0470.002</td>
<td>0.0040.000</td>
<td>0.2950.002</td>
<td>0.0100.000</td>
<td>141</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>0.000.000</td>
<td>0.0070.001</td>
<td>0.0010.000</td>
<td>0.2220.002</td>
<td>0.0050.000</td>
<td>171</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>0.000.000</td>
<td>0.0050.000</td>
<td>0.0020.000</td>
<td>0.1730.002</td>
<td>0.0070.000</td>
<td>204</td>
</tr>
</tbody>
</table>

**Simulation 1(b).** Next, we explore how read depth affects the performance of RNDClone. For each of the three simulation scenarios in Simulation 1(a) with \( C = 4 \) and \( T \in \{3,4,5\} \), we generate two more hypothetical scenarios with \( E(\phi_t) = E(\psi_t) \in \{100,400\} \). Table 2 reports the reconstruction errors under the 6 scenarios and also includes the errors under the 3 scenarios in Simulation 1(a) (with \( E(\phi_t) = E(\psi_t) = 200 \)) for comparison.

Again, in all scenarios, RNDClone maintains low reconstruction errors. In general, the simulation truth is better recovered with higher read depths \( E(\phi_t) \) and \( E(\psi_t) \) and a larger number of samples \( T \). This is similar to existing methods, and is well understood in the literature.

**Simulation 1(c).** In Simulations 1(a, b) we considered simulation truths with copy numbers ranging from 1 to 3. However, some cancer cells can undergo more extensive CNAs. We therefore include an additional simulation scenario with copy numbers ranging from 0 to 10. We consider \( S = 100 \) loci, \( C = 4 \) subclones, \( T = 3 \) samples and average read depths \( E(\phi_t) = E(\psi_t) = 200 \). We generate 50 datasets under this scenario and fit the datasets with RNDClone, setting \( K_{\min} = 0 \) and \( K_{\max} = 10 \) accordingly. The second row of Table 3 reports the reconstruction errors for this scenario. Again, the truth is recovered with small errors.

Since \( K_{\min} \) and \( K_{\max} \) are artificial prior bounds, it is possible that they are different from the true range of copy numbers. To explore how different
K is run with choosing a sufficiently large reconstruction errors are high. Therefore, we recommend setting K characterized in Table 3 (rows 3 and 4). When the true copy numbers are within we fit the 50 datasets generated under this scenario with previous scenario in Simulation 1(c) with copy numbers ranging from 0 to 10; while the true copy numbers range from 1 to 3. Next, we consider the pre-

choices of K_{min} and K_{max} affect the performance of RNDClone, we conduct more simulations under two additional scenarios. First, we consider one scenario in Simulation 1(a) with C = 4, T = 3 and E(\phi_t) = E(\psi_t) = 200; we fit the 50 datasets generated under this scenario with K_{min} = 0 and K_{max} = 10, while the true copy numbers range from 1 to 3. Next, we consider the previous scenario in Simulation 1(c) with copy numbers ranging from 0 to 10; we fit the 50 datasets generated under this scenario with K_{min} = 1 and K_{max} = 3. The reconstruction errors under these two scenarios are summarized in Table 3 (rows 3 and 4). When the true copy numbers are within the range of K_{min} and K_{max}, the truth can still be recovered. On the other hand, when the true copy numbers exceed the range of K_{min} and K_{max}, the reconstruction errors are high. Therefore, we recommend setting K_{min} = 0 and choosing a sufficiently large K_{max}. Increasing the range of K_{min} and K_{max} leads to longer computation time.

### Table 2

*Simulation 1(b). Reconstruction errors and computation times (in minutes) of RNDClone under 9 scenarios, one for each combination of T \in \{3, 4, 5\} and E(\phi_t) = E(\psi_t) \in \{100, 200, 400\}. Values shown are averages over 50 repeat simulations, with numerical Monte Carlo standard errors in subscripts.*

<table>
<thead>
<tr>
<th>T</th>
<th>E(\phi_t)</th>
<th>C_{err}</th>
<th>L_{err}</th>
<th>Z_{err}</th>
<th>A_{err}</th>
<th>W_{err}</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>100</td>
<td>0.000</td>
<td>0.066</td>
<td>0.008</td>
<td>0.323</td>
<td>0.013</td>
<td>141</td>
</tr>
<tr>
<td>3</td>
<td>200</td>
<td>0.000</td>
<td>0.020</td>
<td>0.002</td>
<td>0.271</td>
<td>0.008</td>
<td>142</td>
</tr>
<tr>
<td>3</td>
<td>400</td>
<td>0.000</td>
<td>0.006</td>
<td>0.003</td>
<td>0.221</td>
<td>0.006</td>
<td>142</td>
</tr>
<tr>
<td>4</td>
<td>100</td>
<td>0.000</td>
<td>0.063</td>
<td>0.006</td>
<td>0.278</td>
<td>0.010</td>
<td>171</td>
</tr>
<tr>
<td>4</td>
<td>200</td>
<td>0.000</td>
<td>0.016</td>
<td>0.001</td>
<td>0.212</td>
<td>0.008</td>
<td>174</td>
</tr>
<tr>
<td>4</td>
<td>400</td>
<td>0.000</td>
<td>0.004</td>
<td>0.003</td>
<td>0.157</td>
<td>0.006</td>
<td>174</td>
</tr>
<tr>
<td>5</td>
<td>100</td>
<td>0.000</td>
<td>0.045</td>
<td>0.004</td>
<td>0.244</td>
<td>0.010</td>
<td>202</td>
</tr>
<tr>
<td>5</td>
<td>200</td>
<td>0.000</td>
<td>0.012</td>
<td>0.003</td>
<td>0.191</td>
<td>0.007</td>
<td>203</td>
</tr>
<tr>
<td>5</td>
<td>400</td>
<td>0.127</td>
<td>0.026</td>
<td>0.025</td>
<td>0.172</td>
<td>0.014</td>
<td>206</td>
</tr>
</tbody>
</table>

### Table 3

*Simulation 1(c). Reconstruction errors and computation times (in minutes) of RNDClone under 2 scenarios with different copy number ranges. For each scenario, RNDClone is run with (K_{min}, K_{max}) = (1, 3) and (0, 10). Values shown are averages over 50 repeat simulations, with numerical Monte Carlo standard errors in subscripts.*

<table>
<thead>
<tr>
<th>min, max(l_{err})</th>
<th>K_{min}, K_{max}</th>
<th>C_{err}</th>
<th>L_{err}</th>
<th>Z_{err}</th>
<th>A_{err}</th>
<th>W_{err}</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 3</td>
<td>1, 3</td>
<td>0.002</td>
<td>0.020</td>
<td>0.002</td>
<td>0.271</td>
<td>0.008</td>
<td>142</td>
</tr>
<tr>
<td>0, 10</td>
<td>0, 10</td>
<td>0.000</td>
<td>0.105</td>
<td>0.038</td>
<td>0.303</td>
<td>0.012</td>
<td>224</td>
</tr>
<tr>
<td>1, 3</td>
<td>0, 10</td>
<td>0.000</td>
<td>0.025</td>
<td>0.003</td>
<td>0.272</td>
<td>0.009</td>
<td>220</td>
</tr>
<tr>
<td>0, 10</td>
<td>1, 3</td>
<td>1.14</td>
<td>0.552</td>
<td>0.231</td>
<td>0.559</td>
<td>0.061</td>
<td>142</td>
</tr>
</tbody>
</table>
Lastly, we note that increasing the number of genomic loci or having more loci on each gene allows more borrowing of information, thus also improves the accuracy of subclone reconstruction. Computation time will increase as the number of loci ($S$) increases.

4.2. Simulation 1: Comparison with Alternatives. There are no existing methods that simultaneously infer subclonal copy numbers, mutations and gene expressions. For comparison, we run SIFA [Zeng, Warren and Zhao (2019)], BayClone2 [Lee et al. (2016)] and BayCount [Xie, Zhou and Xu (2018)] on the same simulated datasets for the inference of DNA or RNA subclones, but not both. SIFA and BayClone2 use only DNA sequencing data to infer subclonal copy numbers and mutations, while BayCount uses only total RNA counts to infer subclonal RNA expression profiles.

SIFA and BayClone2 (DNA-based). SIFA is one of the most recently published methods for DNA-based subclone reconstruction. Both SIFA and BayClone2 characterize subclonal copy numbers and variant allele numbers by latent feature matrices (same as $L$ and $Z$) and model total and variant DNA read counts ($N$ and $n$) using Poisson and binomial distributions, respectively. We run SIFA and BayClone2 under the default hyperparameter settings. The reconstruction errors of SIFA under Simulation 1(a) scenarios are reported in Table 4. Additional details, including the results of SIFA under Simulation 1(b, c) scenarios, the results of BayClone2, and a discussion of the results, are presented in Supplementary Section S.2.2. In all scenarios, SIFA and BayClone2 have higher reconstruction errors than RNDClone.

BayCount (RNA-based). BayCount is one of the most recently published methods for RNA-based subclone inference. BayCount was developed to infer inter-tumor heterogeneity when gene expression data from multiple patients are available. Yet, when multiple tissue samples from the same patient are available, BayCount can also be used to measure the heterogeneity in gene expression across these samples. BayCount characterizes subclonal expression profiles by a latent factor matrix (similar to $\Lambda$) and models total RNA counts ($M$) using a negative binomial distribution. We run BayCount under its default hyperparameter setting. We rescale the outcome of BayCount and adjust for copy numbers so that it can be compared to $\Lambda$. The reconstruction errors of BayCount under Simulation 1(a) scenarios are reported in Table 4. Additional results are presented in Supplementary Section S.2.2. Overall, BayCount appears to have inflated errors compared with RNDClone.
RNDCLONE: INTEGRATING DNA AND RNA SEQUENCE DATA

Table 4
Reconstruction errors of SIFA and BayCount under the 9 scenarios in Simulation 1(a).
Values shown are averages over 50 repeat simulations, with numerical Monte Carlo standard errors in subscripts.

<table>
<thead>
<tr>
<th>C</th>
<th>T</th>
<th>SIFA</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>BayCount</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>C_{err}</td>
<td>\Lambda_{err}</td>
<td>Z_{err}</td>
<td>W_{err}</td>
<td></td>
<td>C_{err}</td>
<td>\Lambda_{err}</td>
<td>Z_{err}</td>
<td>W_{err}</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0.10</td>
<td>0.070</td>
<td>0.073</td>
<td>0.019</td>
<td>0.54</td>
<td>0.209</td>
<td>0.094</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0.32</td>
<td>0.066</td>
<td>0.065</td>
<td>0.014</td>
<td>0.16</td>
<td>0.177</td>
<td>0.083</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>0.39</td>
<td>0.102</td>
<td>0.136</td>
<td>0.030</td>
<td>0.90</td>
<td>0.150</td>
<td>0.065</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>1.42</td>
<td>0.126</td>
<td>0.192</td>
<td>0.088</td>
<td>0.78</td>
<td>0.516</td>
<td>0.080</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>2.57</td>
<td>0.125</td>
<td>0.197</td>
<td>0.081</td>
<td>0.84</td>
<td>0.477</td>
<td>0.080</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>1.54</td>
<td>0.108</td>
<td>0.204</td>
<td>0.081</td>
<td>1.74</td>
<td>0.425</td>
<td>0.065</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>0.44</td>
<td>0.073</td>
<td>0.052</td>
<td>0.038</td>
<td>1.14</td>
<td>0.538</td>
<td>0.106</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>0.86</td>
<td>0.057</td>
<td>0.087</td>
<td>0.048</td>
<td>1.12</td>
<td>0.453</td>
<td>0.092</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>0.74</td>
<td>0.068</td>
<td>0.112</td>
<td>0.059</td>
<td>1.32</td>
<td>0.472</td>
<td>0.103</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Self-comparison. We present a self-comparison of our own modeling approach applied separately to DNA and RNA data. As described in Section 2.5, when only DNA data (or RNA data) are available, RNDClone reduces to DClone (or RClone) and can still infer DNA-related (or RNA-related) parameters such as \( L \), \( Z \), \( W \) and \( C \) (or \( \Lambda \), \( W \) and \( C \)). The goal of this comparison is to better understand the advantage of jointly modeling DNA and RNA data.

The reconstruction errors of DClone and RClone under the 9 scenarios in Simulation 1(a) are reported in Table 5. Additional results are presented in Supplementary Section S.2.2. The reconstruction errors of RClone are generally higher than those of RNDClone. Although the reconstruction errors of DClone are comparable to those of RNDClone, using only DNA data, DClone does not provide estimations of the subclonal RNA expression profiles. One might think of combining the results from separate analyses post hoc to obtain a complete picture of the subclonal genotypes and gene expression profiles. However, such combination might not be straightforward, as DNA analysis and RNA analysis might yield different numbers of subclones and different subclone proportions, and the relationship between DNA and RNA subclones is also unknown. In contrast, inference under a joint model for DNA and RNA data provides coherent inference and easily interpretable results.

4.3. Simulation 2: Frequentist Coverage. In simulation 2, we generate simulated data by mimicking the actual TCGA lung cancer dataset in Section 5. Same as the lung cancer dataset, we consider \( S = 66 \) loci and \( T = 3 \)
Reconstruction errors of DClone and RClone under the 9 scenarios in Simulation 1(a).

Values shown are averages over 50 repeat simulations, with numerical Monte Carlo standard errors in subscripts.

<table>
<thead>
<tr>
<th>C</th>
<th>T</th>
<th>DClone</th>
<th>RClone</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$C_{\text{err}}$</td>
<td>$L_{\text{err}}$</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0.00 ± 0.00</td>
<td>0.01 ± 0.00</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0.00 ± 0.00</td>
<td>0.02 ± 0.00</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>0.04 ± 0.03</td>
<td>0.02 ± 0.01</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>0.00 ± 0.00</td>
<td>0.02 ± 0.00</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>0.00 ± 0.00</td>
<td>0.01 ± 0.00</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>0.02 ± 0.02</td>
<td>0.01 ± 0.01</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>0.00 ± 0.00</td>
<td>0.06 ± 0.00</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>0.00 ± 0.00</td>
<td>0.00 ± 0.00</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>0.00 ± 0.00</td>
<td>0.00 ± 0.00</td>
</tr>
</tbody>
</table>

samples. These loci reside on $G = 61$ genes. The number, genotypes and gene expression profiles of the subclones, as well as the expected read depths and overdispersion parameters, are generated by fitting the lung cancer dataset under RNDClone. See Section 5 for more detail of the lung cancer dataset. In particular, the number of subclones is $C = 3$, and the copy numbers are $l_{sc} \in \{0, 1, \ldots, 13\}$. Under this simulation truth, we repeatedly generate 50 datasets with the assumed sampling model to investigate the frequentist coverage properties of the Bayesian credible intervals given by RNDClone. This exercise gives us an assessment of the potential performance of RNDClone for the real data in Section 5.

We fit each simulated dataset with RNDClone using the same hyperparameter and MCMC setting described in the beginning of Section 4. Moreover, we set $K_{\text{min}} = 0$ and $K_{\text{max}} = 15$. The results are summarized in Table 6. The coverage probabilities are calculated as the proportion of the time that the 95% credible interval of the corresponding parameter contains the truth. For the multidimensional parameters $L$, $Z$, $\Lambda$ and $W$, the reported coverage probabilities are averages over all of their entries. The frequentist coverage does not give rise to concerns. An exact match is not expected due to a relatively small sample size ($T = 3$) and a large parameter space. In addition, we report the reconstruction errors and root mean squared errors (RMSEs) averaged over the repeated datasets. Again, for $L$, $Z$, $\Lambda$ and $W$, the RMSEs are averages over all of their entries. For example,

$$L_{\text{RMSE}} = \frac{1}{S(C - 1)} \sum_{s,c} \sqrt{\frac{1}{J} \sum_{j} \left( l_{sc(c)}^{(j)} - l_{sc} \right)^2}.$$
The reconstruction errors and RMSEs are small, indicating good recovery of the truth.

Table 6
Simulation 2. Frequentist coverage probabilities, reconstruction errors and root mean squared errors (RMSEs) for $C$, $L$, $Z$, $\Lambda$, $W$ and the log-likelihood. For the multidimensional parameters, the values are averages over all entries.

<table>
<thead>
<tr>
<th></th>
<th>$C$</th>
<th>$L$</th>
<th>$Z$</th>
<th>$\Lambda$</th>
<th>$W$</th>
<th>log-likelihood</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coverage</td>
<td>100%</td>
<td>95.7%</td>
<td>98.0%</td>
<td>85.4%</td>
<td>76.2%</td>
<td>92%</td>
</tr>
<tr>
<td>Rec. error</td>
<td>0</td>
<td>0.292</td>
<td>0.065</td>
<td>0.261</td>
<td>0.023</td>
<td>–</td>
</tr>
<tr>
<td>RMSE</td>
<td>0.714</td>
<td>0.682</td>
<td>0.133</td>
<td>0.319</td>
<td>0.024</td>
<td>–</td>
</tr>
</tbody>
</table>

5. TCGA Data Analysis. We apply RNDClone to the analysis of a TCGA (The Cancer Genome Atlas, [https://cancergenome.nih.gov/](https://cancergenome.nih.gov/)) lung adenocarcinoma (LUAD) dataset. We have $T = 3$ tumor samples from a stage 1B T2 patient with lung adenocarcinoma (sample ID: TCGA-44-2668) who has passed away two years after diagnosis. In particular, all the three samples are obtained from the primary tumor tissue and are sequenced using whole exome sequencing (WES) technology. We first retrieve the DNA sequence data by downloading the VCF (variant call format) files from the TCGA website. In particular, we choose the somatic variant callset generated by MuTect2 [Cibulskis et al. (2013)]. The average sequencing depth is around 100x. Next, to obtain the RNA sequence data, we download the corresponding RNA BAM (binary alignment map) files, which contain sorted and indexed RNA reads. Finally, we find the loci for which both DNA and RNA read data are available for the three samples. For each locus, we record the total numbers of mapped DNA and RNA reads, as well as numbers of mapped DNA and RNA reads that bear a variant sequence. We obtain a total of $S = 66$ loci across $G = 61$ genes. Figure 3 shows the histograms of DNA and RNA read depths and variant allele fractions. The average DNA read depths for the three samples are 141, 210 and 122, respectively. As shown in the simulation studies, RNDClone should provide useful inference with $T = 3$ samples and 100x–200x read depth.

We fit the dataset with the same hyperparameters described in the beginning of Section 4. To calibrate an informative prior for $\phi_t$, for each sample, we retrieve copy number estimates of specific DNA segments obtained from single nucleotide polymorphism (SNP) array analysis, using the TCGA SNP array data for the three samples. We then run a regression between the copy number estimates and the average read depths for the DNA segments. Lastly, we match the prior mean and variance of the average read depth in
copy number neutral regions (i.e. copy number = 2) with the values estimated from the regression analysis. We run MCMC simulation with 25,000 burn-in iterations and 5,000 trans-dimensional transitions. The computation takes around 150 minutes using a single core on an Intel Xeon (E5-2650 v4, 2.20GHz) computing node.

**Results.** The posterior mode of $C$ is $\hat{C} = 3$ (posterior distribution of $C$ is shown in Supplementary Figure S.3). That is, we infer that the tumor samples have three subclones (including one normal subclone). The estimated population frequencies of the three subclones across the three samples are
shown in Figure 4(d). For the upcoming biological interpretation, we focus on 15 genes that have high expression levels in at least one subclone. Figures 4(a, b) show the estimated copy numbers and variant allele numbers for the loci that reside on the 15 genes, and Figure 4(c) shows the estimated RSGEs for the 15 genes. The complete estimates are reported in Supplementary Figure S.4. A convergence diagnostic for the MCMC simulation and a check of model fit are reported later.

From Figure 4, we can see that RNDClone identifies two tumor subclones (subclones 2 and 3), aside from the normal cells (subclone 1). Subclone 3 exhibits copy number losses (Figure 4(a)) in many genes but no point mutations (Figure 4(b)). It has the largest population frequencies (Figure 4(d)) in all three samples. However, subclone 2 shows mostly copy number gains (Figure 4(a)) and extensive mutations in almost all the genes (Figure 4(b)). Its population frequencies are quite small as shown in Figure 4(d), indicating that this is a hyper-mutated small subclone.

The lung cancer patient was initially diagnosed with stage 1 disease and was declared a complete response, i.e., no detectable tumor, after a surgery. However, merely one year after the surgery, the patient relapsed with new tumor and died quickly after failing conventional treatment. This is unusual for a stage 1B patient. A possible explanation is that subclone 2 might have been harboring the fatal tumor cells that caused the disease relapse and death. The subclone-2 tumor could be highly malignant since most genes that we analyzed were mutated (Supplementary Figure S.4), including the well-known oncogene KRAS that usually leads to poor prognosis. In addition, Figure 4(c) shows that the RSGEs of subclone 2 are distinct from those of subclones 1 and 3, indicating that the hyper-mutated subclone 2 also alters the transcription profile of the cells. We speculate that subclone analysis (such as using RNDClone) could have been helpful for this patient if the analysis had been performed and subclone 2 had been identified. Perhaps additional chemotherapy would have been ordered by the treating clinicians after the surgery, or the patient might have been followed more frequently for potential relapse. Sadly, without knowing the malignant subclone 2, the stage 1B tumor, considered as early stage, could still be lethal.

Convergence diagnostic and test of fit. To assess the convergence of the MCMC simulation, we run RNDClone with three different starting values (random variate seeds). As a convenient scalar summary to track for the convergence assessment we use the log-likelihood values. Figure 5(a) shows the traceplot of the posterior samples of the log-likelihood from the three runs. To avoid complications related to the trans-dimensional nature of the MCMC implementation we first select from each of the three runs 1,000
iterations when the chain was imputing $C = 3$ subclones. We then evaluated diagnostics with these subchains. We evaluate the potential scale reduction factor [Gelman and Rubin (1992)] across the three chains. We find a factor of 1.01, with an upper confidence bound 1.05, indicating no evidence for lack of convergence.

Next, as an informal check of model fit, we inspect the histograms of standardized residuals, defined as $[a - E(a)]/\sqrt{\text{Var}(a)}$, where $a$ is $N_{st}$, $n_{st}$, $M_{st}$ or $m_{st}$, and $E(a)$ and $\text{Var}(a)$ are functions of the parameters. The residuals are then calculated and averaged over all posterior draws of the parameters. Figure 5(b) shows the histogram of the standardized residuals. The residuals are centered around zero with little mass beyond $\pm 2$, indicating a good model fit. We also use posterior predictive checking to examine the goodness-of-fit of the model. See, for example, Gelman et al. (2013) (Section 6.3) for a review. Using the $J$ posterior draws of the parameters, we generate replicated data $\{D^{\text{rep}}(j) = (N^{\text{rep}}(j), n^{\text{rep}}(j), M^{\text{rep}}(j), m^{\text{rep}}(j)), j = 1, \ldots, J\}$ from the RNDClone model. Since the data are multidimensional, for each entry $a$ ($a$ can be $N_{st}$, $n_{st}$, $M_{st}$ or $m_{st}$) we calculate the posterior predictive $p$-value by $p_{pp} = \frac{1}{J} \sum_j I(a^{\text{rep}}(j) > a)$. Figure 5(c) shows the histogram of the posterior predictive $p$-values for all entries of the data. In particular, 3.66% of the $p$-values are less than 0.05, and 2.40% of the $p$-values are greater than 0.95, which does not give rise to concerns about model fit.

Fig 5: Panel (a) shows the traceplot of the log-likelihood from 3 independent Markov chains from fitting the TCGA LUAD dataset with RNDClone using different random seeds. Panel (b) is a histogram of standardized residuals. Panel (c) is a histogram of posterior predictive $p$-values for the multidimensional data.

Comparison with alternatives. For comparison, we run SIFA and BayCount on the same dataset. The results under SIFA and BayCount, as well as a discussion of the results, are presented in Supplementary Section S.3.2.
6. Discussion. We have developed novel generalized latent factor models to reconstruct tumor subclones based on integrative analyses of both DNA and RNA sequence data. This is the first attempt to bridge existing DNA-based and RNA-based methods. We simultaneously infer subclonal genotype and gene expression profile. Such inference provides important clinical information about personalized treatment strategies.

Our modeling approach can be considered as a finite truncation (in $C$) of related Bayesian nonparametric models [Müeller, Quintana and Page (2018)]. The truncation is used to facilitate posterior simulation, following, e.g., the idea of truncated stick-breaking priors in Ishwaran and James (2001). Alternative split-merge proposals [Jain and Neal (2004), Griffiths and Ghahramani (2011)] in our application would be adding or deleting columns in $L$, $Z$, $\Lambda$ and $W$. We found that such proposals lead to very low acceptance probabilities in our application due to the highly peaked and multi-modal posterior landscape. We therefore developed the trans-dimensional and parallel-tempering MCMC algorithm to efficiently sample from the posterior space.

We have demonstrated the practical application of RNDClone on one real-world dataset and found some interesting results. Currently, publicly available synthetic or real-world datasets with both DNA and RNA sequencing counts are still rare. When more of such datasets become available, we plan to apply RNDClone to these datasets to further evaluate the validity of RNDClone.

It is generally assumed that tumor cells evolve along a phylogenetic tree. Each cell inherits the mutations of its parent and possibly also gains more mutations. An important extension of the current model is to explicitly model the potential phylogenetic relationship among the subclones, such as in Deshwar et al. (2015), Marass et al. (2016), Zeng, Warren and Zhao (2019) and Zhou et al. (2019b). However, modeling tumor phylogeny is very challenging when CNA data are included. Existing methods either assume copy number neutrality, or require copy number estimates from another software, or make strong assumptions about the occurrence of CNAs. Therefore, like many existing methods [Roth et al. (2014), Zare et al. (2014), Lee et al. (2016)] we chose not to explicitly model phylogeny. In simulation studies we find that RNDClone is able to recover an assumed true phylogenetic structure when it is included in the simulation scenario.

Currently, RNDClone characterizes tumor heterogeneity by point mutations, copy number aberrations and gene expression levels. A future direction is to incorporate data from other types of structural variations [Fan et al. (2014)], as well as DNA methylation data [Rhee et al. (2013)], in the analy-
sis. It is also of interest to integrate further downstream gene expression data such as proteomics data into the current scheme. The focus of RNDClone is inference on intra-tumor heterogeneity. Nevertheless, the proposed methodology can be easily extended to infer inter-tumor heterogeneity across different patients. Lastly, RNDClone utilizes NGS data from bulk sequencing experiments. Alternatively, single-cell sequencing data [Schmidt and Efferth (2016), Kuipers, Jahn and Beerenwinkel (2017)] provide genomic and transcriptomic information at the cellular level. It is of interest to develop similar methodologies for single-cell data, and there is some very recent progress [Campbell et al. (2019)].

SUPPLEMENTARY MATERIAL

Supplement to “RNDClone: Tumor Subclone Reconstruction Based on Integrating DNA and RNA Sequence Data” (). We provide the supplementary details referenced in the main text, including details of the MCMC implementation, simulation and TCGA data analysis.

References.


