The Dissertation Committee for Tianjian Zhou
certifies that this is the approved version of the following dissertation:

Bayesian Nonparametric Models for Biomedical Data Analysis

Committee:

Peter Mueller, Supervisor

Michael Daniels

Yuan Ji

Sinead Williamson

Bayesian Nonparametric Models for Biomedical Data Analysis

by

Tianjian Zhou

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

August 2017

Dedicated to my family.

Acknowledgments

After completing my four-years life as a Ph.D. student, I would like to take this opportunity to thank many people who helped me.

First of all, I would like to thank my advisor and mentor, Professor Peter Mueller, without whom I would not be able to write this dissertation. His wonderful lectures have introduced me to Bayesian statistics and MCMC algorithms, and his guidance has prepared me for doing statistics research. He is experienced and is good at explaining complex ideas, so that he can always help me out when I get in trouble in research. He is kind, patient and considerate, so that I always feel supported in my Ph.D. life. I would also like to thank Peter's wife, Gautami Shah, who is also very kind and considerate. I will miss the great memories our group had together (at UT, restaurants, Peter's house, and so on).

Second, I would like to thank Professor Michael Daniels, who has mentored me in part of my dissertation (the missing data part) and has financially supported me in the last two years of my Ph.D. life through research assistantship (NIH CA 183854). I would also like to thank Mike for his comprehensive Statistical Modeling 1 class and Reading in Statistics class.

Third, I would like to thank Professor Yuan Ji, who has mentored me in part of my dissertation (the tumor heterogeneity part), has financially

supported me for two summers through internship positions at NorthShore University HealthSystem (NIH CA 132897) and has kindly provided me a postdoc position which I have accepted.

I would also like to thank several other professors in my department. I would like to thank Professor Sinead Williamson, who has served as my dissertation committee member and has provided insightful comments for my dissertation. I would like to thank Professor James Scott for his wonderful Statistical Modeling 2 class and Reading in Statistics class. I would like to thank Professor Carlos Carvalho for his wonderful Time Series and Dynamic Models class and Reading in Statistics class.

I would like to thank the Department of Statistics and Data Sciences for financially supporting me through teaching assistantships over the years. I would like to thank the staffs in our department, in particular Vicki Keller, for always being helpful and supportive.

I would like to thank my girlfriend, Lan Liang, who has supported me throughout the days and has made the last year of my Ph.D. life an unforgettable memory. I would like to thank Haoyu Zhang, my friend from college, for helping me throughout the four years and supporting my decisions. I would like to thank my collaborator and friend, Subhajit Sengupta, for helping me out when I was in trouble in the tumor heterogeneity project and supporting me when I did my internship at NorthShore. I would also like to thank my friends, Yanyan Dai, Yu Ding, Chao Ji, Yan Jin, Anastasiya Travina, Li Wang, Mengjie Wang, Yisi Wang, Carlos Pagani Zanini, Anao Zhang, and a lot more,

for the fantastic memories we had together.

Finally, I would like to thank my family members, who always trust me and support me in every decision I made. Without them, I am not able to become a grown man.

Bayesian Nonparametric Models for Biomedical Data Analysis

Publication No. _____

Tianjian Zhou, Ph.D.
The University of Texas at Austin, 2017

Supervisor: Peter Mueller

In this dissertation, we develop nonparametric Bayesian models for biomedical data analysis. In particular, we focus on inference for tumor heterogeneity and inference for missing data. First, we present a Bayesian feature allocation model for tumor subclone reconstruction using mutation pairs. The key innovation lies in the use of short reads mapped to pairs of proximal single nucleotide variants (SNVs). In contrast, most existing methods use only marginal reads for unpaired SNVs. In the same context of using mutation pairs, in order to recover the phylogenetic relationship of subclones, we then develop a Bayesian treed feature allocation model. In contrast to commonly used feature allocation models, we allow the latent features to be dependent, using a tree structure to introduce dependence. Finally, we propose a nonparametric Bayesian approach to monotone missing data in longitudinal studies with non-ignorable missingness. In contrast to most existing methods, our method allow for incorporating information from auxiliary covariates and

is able to capture complex structures among the response, missingness and auxiliary covariates. Our models are validated through simulation studies and are applied to real-world biomedical datasets.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

This dissertation develops nonparametric Bayesian models, corresponding Markov chain Monte Carlo (MCMC) algorithms, and applications for biomedical data analysis. Chapters 2 and 3 are about applications to genomic data analysis, and Chapter 4 discusses applications to longitudinal missing data analysis. Nonparametric Bayesian methods provide flexible and highly adaptable approaches for statistical inference. In applications with biostatistics data such methods can often better address biological research problems than more restrictive parametric methods.

In Chapter 2, we talk about inference on tumor heterogeneity. During tumor growth, tumor cells acquire somatic mutations that allow them to gain advantages compared to normal cells. As a result, tumor cell populations are typically heterogeneous consisting of multiple subpopulations with unique genomes, characterized by different subsets of mutations. This is known as tumor heterogeneity. The homogeneous subpopulations are known as subclones and are an important target in precision medicine. We propose a Bayesian feature allocation model to reconstruct tumor subclones using next-generation

sequencing (NGS) data. The key innovation is the use of (phased) pairs of proximal single nucleotide variants (SNVs) for the subclone reconstruction. We utilize parallel tempering to achieve a better mixing Markov chain with highly multi-modal posterior distributions. We also develop trans-dimensional MCMC algorithms with transition probabilities that are based on splitting the data into training and test data sets to efficiently implement trans-dimensional MCMC sampling. Through simulation studies we show that inference under our model outperforms models using only marginal SNVs by recovering the number of subclones as well as their structures more accurately. This is the case despite significantly smaller number of phased pairs than the number of marginal SNVs. Estimating our model for four lung cancer tissue samples, we successfully infer their subclone structures. For this work, I collaborate with Peter Mueller (The University of Texas at Austin), Subhajit Sengupta (NorthShore University HealthSystem) and Yuan Ji (NorthShore University HealthSystem and The University of Chicago).

In Chapter 3, we address another important aspect of statistical inference for tumor heterogeneity, aiming to recover the phylogenetic relationship of subclones. Such inference can significantly enrich our understanding of subclone evolution and cancer development. We develop a tree-based feature allocation model which explicitly models dependence structure among subclones. That is, in contrast to commonly used feature allocation models, we allow the latent features to be dependent, using a tree structure to introduce dependence. In the application to inference for tumor heterogeneity this tree

structure is interpreted as a phylogenetic tree of tumor cell subpopulations. We adapt our MCMC sampling techniques to efficiently search the tree space. We analyze a lung cancer data set and infer the underlying evolutionary process. For this work, I collaborate with Subhajit Sengupta, Peter Mueller and Yuan Ji.

In Chapter 4, we model missing data in longitudinal studies. In longitudinal clinical studies, the research objective is often to make inference on a subject's full data response conditional on covariates that are of primary interest; for example, to calculate the treatment effect of a test drug at the end of a study. The vector of responses for a research subject is often incomplete due to dropout. Dropout is typically non-ignorable and in such cases the joint distribution of the full data response and missingness needs to be modeled. In addition to the covariates that are of primary interest, we would often have access to some auxiliary covariates (often collected at baseline) that are not desired in the model for the primary research question. Such variables can often provide information about the missing responses and missing data mechanism. In this setting, auxiliary covariates should be incorporated in the joint model as well, and we should proceed with inference unconditional on these auxiliary covariates. As a result, we consider a joint model for the full data response, missingness and auxiliary covariates. In particular, we specify a nonparametric Bayesian model for the observed data via Gaussian process priors and Bayesian additive regression trees. These model specifications allow us to capture non-linear and non-additive effects, in contrast to existing

parametric methods. We then separately specify the conditional distribution of the missing data response given the observed data response, missingness and auxiliary covariates (i.e. the extrapolation distribution) using identifying restrictions. We introduce meaningful sensitivity parameters that allow for a simple sensitivity analysis. Informative priors on those sensitivity parameters can be elicited from subject-matter experts. We use Monte Carlo integration to compute the full data estimands. Our methodology is motivated by, and applied to, data from a clinical trial on treatments for schizophrenia. For this work, I collaborate with Michael Daniels (The University of Texas at Austin).

The remainder of this chapter is organized as follows, Section 1.2 contains basics of Bayesian inference and Bayesian nonparametrics. Sections 1.3, 1.4 and 1.5 present three classes of statistical models, discuss how nonparametric Bayesian methods can be used, and demonstrate applications related to succeeding chapters. These models include latent class models, latent feature models and regression.

1.2 Bayesian Nonparametrics

Bayesian Inference. By way of introducing notation, we briefly review the setup of *Bayesian inference*. Bayesian inference, named for Thomas Bayes, is a particular approach to statistical inference. Let y denote the observed *data*, θ denote the unobserved *parameters* of interest, and $\tilde{y}$ denote unknown but potentially observable quantities (such as a data point that is not yet observed) of interest. In the Bayesian framework, we update our belief on the

unobserved parameters according to evidences in the observed data based on Bayes' rule:

$$p(\theta \mid y) = \frac{p(y \mid \theta)p(\theta)}{p(y)}. \tag{1.1}$$

In equation (1.1), $p(\theta)$ is called the *prior distribution*, $p(y \mid \theta)$ is called the *sampling distribution* (when regarded as a function of y with fixed θ) or the *likelihood* (when regarded as a function of θ with fixed y). The denominator $p(y)$ is the marginal distribution of y, which is calculated by $p(y) = \int p(y \mid \theta)p(\theta)d\theta$. Inference on θ is given by the *posterior distribution* $p(\theta \mid y)$. We can then make inference on $\tilde{y}$ based on the posterior *predictive distribution*

$$p(\tilde{y} \mid y) = \int p(\tilde{y} \mid \theta)p(\theta \mid y)d\theta.$$

Bayesian statistical inference is stated in terms of probability statements conditional on the observed values of y. For a review of Bayesian statistics, see, for example, Gelman et al. (2014) or Hoff (2009).

Exchangeability. *Exchangeability* plays an important role in statistics. Suppose we have N random variables (which can be data points or parameters) with a joint distribution $p(y_1, \ldots, y_N)$. The random variables are called exchangeable if their joint distribution is invariant to permutation. Let $[N] = \{1, \ldots, N\}$ and denote by $\sigma : [N] \to [N]$ a permutation of $[N]$. (Finite) exchangeability states that

$$y_1, \ldots, y_N \overset{d}{=} y_{\sigma(1)}, \ldots, y_{\sigma(N)}$$

for any σ, where $\stackrel{d}{=}$ means equal in distribution. Furthermore, an infinite sequence of random variables $y_1, y_2, \ldots$ is called infinitely exchangeable if

$$y_1, y_2, \ldots \stackrel{d}{=} y_{\sigma(1)}, y_{\sigma(2)}, \ldots,$$

where $\sigma : \mathbb{N} \to \mathbb{N}$ is a finite permutation. That is, for some finite value N_σ, $\sigma(n) = n$ for all $N > N_\sigma$.

The importance of exchangeability is due to de Finetti's theorem (De Finetti, 1931, Hewitt and Savage, 1955, De Finetti, 1974), which states that [1] if $y_1, y_2, \ldots$ are infinitely exchangeable random variables, their joint distribution can be expressed as a mixture of independent and identical distributions

$$p(y_1, \ldots, y_N) = \int \left(\prod_{i=1}^{N} p(y_i \mid \theta) \right) p(\theta) d\theta. \tag{1.2}$$

The theorem can be rephrased from a more general perspective [2]. If $y_1, y_2, \ldots$ are infinitely exchangeable, there exists a random distribution F such that the sequence is composed of i.i.d. draws from it,

$$p(y_1, \ldots, y_N) = \int \prod_{i=1}^{N} F(y_i) dp(F). \tag{1.3}$$

That is, θ in Equation (1.2) can be interpreted as indexing a probability measure F, or θ can even be the probability measure F itself.

[1] This is a rephrased simpler version from Gelman et al. (2014). The original version is a statement about probability measure. De Finetti's original paper De Finetti (1931) is for the case of binary random variables, and Hewitt and Savage (1955) extended it to any real valued random variables.

[2] Another simpler version from Teh (2011).

Bayesian Nonparametrics. A model is called *parametric* if it only has a finite (and usually small) number of parameters, i.e. θ lives in a finite dimensional space. In contrast, a *nonparametric* model has a potentially infinite number of parameters, i.e. θ or F are in an infinite dimensional space. Thus, nonparametric Bayesian inference requires constructing probability distributions on an infinite dimensional parameter space. Such probability distributions are called *stochastic processes* with sample paths in the parameter space. Nonparametric Bayesian methods avoid the often restrictive assumptions of parametric models and provide flexible and highly adaptable approaches for statistical modeling. Reviews of Bayesian nonparametrics include Hjort et al. (2010), Ghosh and Ramamoorthi (2003), Müller et al. (2015), Walker et al. (1999), Müller and Quintana (2004), Orbanz and Teh (2011), Gershman and Blei (2012) and Orbanz (2014) [3].

Nonparametric Bayesian approaches have been widely used in many statistical inference problems, including density estimation, clustering, feature allocation, regression, classification, and graphical models. In the next sections, we give examples to show how Bayesian nonparametric approaches can be applied to address those important problems.

[3]Hjort et al. (2010) includes a set of introductory and overview papers, Ghosh and Ramamoorthi (2003) focuses on posterior convergence, and Müller et al. (2015) has more discussion on data analysis problems.

1.3 Latent Class Models

Suppose we have N objects $y_1, \ldots, y_N$. In a *latent class model* (for a review, see Griffiths and Ghahramani, 2011), each object y_n belongs to a latent class $c_n = k$, $k = 1, \ldots, K$, where K is the number of possible classes, and $K = \infty$ is allowed. When $K = \infty$, the model has an infinite number of classes and thus has an infinite number of parameters. In this case, the model is nonparametric. We are interested in how the classes are related to the objects, $p(\boldsymbol{y} \mid \boldsymbol{c})$, and the distribution over class assignments, $p(\boldsymbol{c})$. For $p(\boldsymbol{y} \mid \boldsymbol{c})$, we assume conditional independence,

$$\begin{aligned} p(\boldsymbol{y} \mid \boldsymbol{c}) &= \prod_{n=1}^{N} p(y_n \mid c_n), \\ p(y_n \mid c_n = k, \mu_k^*) &= G(\cdot \mid \mu_k^*). \end{aligned} \tag{1.4}$$

That is, for an object belonging to class k, we assume it has a distribution G with parameter μ_k^*. We then put some prior distribution F_0 on the μ_k^*'s,

$$\mu_1^*, \ldots, \mu_K^* \overset{iid}{\sim} F_0. \tag{1.5}$$

Next, we specify $p(\boldsymbol{c})$. Specifying $p(\boldsymbol{c})$ is equivalent to defining a distribution on a *random partition* of the index set $[N] := \{1, 2, \ldots, N\}$. The formal definition of random partition follows in Section 1.3.1. The equivalence can be seen by noticing that the unique values of $\boldsymbol{c}$ correspond to a partition of $[N]$, $f_N = \{A_1, \ldots, A_K\}$, with $n \in A_k$ if $c_n = k$, and vice versa.

1.3.1 Random Partition

We briefly summarize the definition of random partitions as given in Broderick, Jordan, and Pitman (2013). See there for more details and discussion. Let $[N] := \{1, 2, \ldots, N\}$ denote the index set of N objects. A *partition* f_N of $[N]$ is a collection of mutually exclusive, exhaustive, nonempty subsets $A_1, \ldots, A_K$ of $[N]$ called *blocks*. Let $f_N = \{A_1, \ldots, A_K\}$, where K is the number of blocks. Here $N = \infty$ is allowed, in which case the index set becomes $\mathbb{N} = \{1, 2, 3, \ldots\}$, and $K = \infty$ is also allowed.

Let $\mathcal{F}_N$ be the space of all partitions of $[N]$. A *random partition* F_N of $[N]$ is a random element of $\mathcal{F}_N$. The probability $p(F_N = f_N)$ is called the partition probability function of F_N.

Exchangeability of a random partition can be defined as follows. Let $\sigma : \mathbb{N} \to \mathbb{N}$ be a finite permutation. That is, for some finite value N_σ, $\sigma(n) = n$ for all $N > N_\sigma$. Furthermore, for any block $A \subset \mathbb{N}$, denote the permutation applied to the block as $\sigma(A) := \{\sigma(n) : n \in A\}$. For any partition Π_N, denote the permutation applied to the partition as $\sigma(\Pi_N) := \{\sigma(A) : A \in \Pi_N\}$. A random partition Π_N is called *exchangeable* if $\Pi_N \stackrel{d}{=} \sigma(\Pi_N)$ for every permutation of $[N]$. The importance of exchangeability has been stated in Section 1.2.

1.3.2 Chinese Restaurant Process

The Chinese restaurant process (CRP) defines an exchangeable random partition. The CRP can be derived in multiple ways, such as from the Dirichlet

process (Blackwell and MacQueen, 1973), or by taking limit of a finite mixture model (Green and Richardson, 2001, Neal, 1992, 2000). We take the latter approach, following Griffiths and Ghahramani (2011).

A Finite Mixture Model. We assume object i belongs to class k with probability π_k,

$$p(c_n = k \mid \pi_k) = \pi_k.$$

Note that the distribution of $\boldsymbol{c}$ always induces a distribution on a random partition of $[N]$. In the above case, the assignment of an object to a class is independent of the assignments of the other objects conditional on $\boldsymbol{\pi}$, and the latent class model is called a *mixture model*. To complete the model, we put a symmetric Dirichlet distribution prior on $(\pi_1, \ldots, \pi_K)$,

$$(\pi_1, \ldots, \pi_K) \sim \mathrm{Dir}(\alpha/K, \ldots, \alpha/K),$$

$\pi_k \geq 0$ and $\sum_{k=1}^{K} \pi_k = 1$.

Integrating out the π_k's, the marginal distribution of $\boldsymbol{c}$ is

$$p(\boldsymbol{c}) = \frac{\prod_{k=1}^{K} \Gamma(m_k + \frac{\alpha}{K})}{\Gamma\left(\frac{\alpha}{K}\right)^K} \frac{\Gamma(\alpha)}{\Gamma(N+\alpha)}, \tag{1.6}$$

where $m_k = \sum_{n=1}^{N} I(c_n = k)$ is the number of objects assigned to class k. This distribution is exchangeable, since it only depends on the counts and does not depend on the ordering of objects.

Equivalence Classes. A partition $\boldsymbol{c}$ includes an ordering of the K blocks by increasing labels $k = 1, \ldots, K$. In many applications, we are only interested in the division of objects, and the ordering of the blocks does not matter. For example, $f_3 = \{\{1\}, \{2, 3\}\}$ and $f_3' = \{\{2, 3\}, \{1\}\}$ correspond to the same division of objects, where the only difference is the choice of labels of the blocks. If the order of the blocks is not identifiable, it is helpful to define an equivalence class of assignment vectors, denoted by $[\boldsymbol{c}]$, with two assignment vectors $\boldsymbol{c}$ and $\boldsymbol{c}'$ belonging to the same equivalence class if they imply the same division of objects.

We therefore focus on the equivalence classes $[\boldsymbol{c}]$. Let K_+ be the number of classes for which $m_k > 0$, and K_0 be the number of classes for which $m_k = 0$, so $K = K_0 + K_+$. The cardinality of $[\boldsymbol{c}]$ is $(K!/K_0!)$. Taking the summation over all assignment vectors that belong to the same equivalence class, and expanding (1.6), we obtain

$$\begin{aligned} p([\boldsymbol{c}]) &= \sum_{\boldsymbol{c} \in [\boldsymbol{c}]} p(\boldsymbol{c}) \\ &= \frac{K!}{K_0!} \left(\frac{\alpha}{K}\right)^{K_+} \left(\prod_{k=1}^{K_+} \prod_{j=1}^{m_k - 1} \left(j + \frac{\alpha}{K}\right) \right) \frac{\Gamma(\alpha)}{\Gamma(N + \alpha)}. \end{aligned} \tag{1.7}$$

Taking the Infinite Limit. When the number of classes $K \to \infty$, taking the limit in Equation (1.7), we get

$$\lim_{K \to \infty} p([\boldsymbol{c}]) = \alpha^{K_+} \left(\prod_{k=1}^{K_+} (m_k - 1)! \right) \frac{\Gamma(\alpha)}{\Gamma(N + \alpha)}. \tag{1.8}$$

See details in Griffiths and Ghahramani (2011). Note that this distribution is still exchangeable, just as in the finite case.

Chinese Restaurant Analogy. The Chinese restaurant analogy (Aldous, 1985) [4] comes from the fact that the distribution in Equation (1.8) can be described by a Chinese restaurant metaphor. Let the N objects be customers in a restaurant, and the K classes be tables at which they sit, $K = \infty$. The customers enter the restaurant one by one, and each chooses a table at random. At time 1, the first customer comes in and chooses the first table to sit. At time n, the n-th customer comes in, chooses an occupied table with probability proportional to the number of customers sitting at that table, or the first unoccupied table with probability proportional to α, $n = 2, \ldots, N$. The customers and tables form a partition of $\boldsymbol{c}$, if we treat the tables as partition blocks. Denote by $c_n = k$ the event that customer n sits at table k, m_k the number of customers sitting at table k after time $n-1$, and $K_+ = \max\{c_1, \ldots, c_{n-1}\}$. Mathematically, the CRP can be written as

$$c_n \mid c_1, \ldots, c_{n-1} = \begin{cases} k, & \text{w. pr. } \frac{m_k}{\alpha+n-1}, \text{ for } k \leq K_+; \\ K_+ + 1, & \text{w. pr. } \frac{\alpha}{\alpha+n-1}, \end{cases} \tag{1.9}$$

for $n = 1, \ldots, N$. The probability of a partition of $\boldsymbol{c}$ given by Equation (1.9) is identical to what given in Equation (1.8).

Dirichlet Process. The CRP defines an exchangeable random partition of $[N]$. We can further extend he model by assigning each table k a value μ_k^*,

[4] Aldous credits this analogy to Jim Pitman and Lester Dubins.

with μ_k^* generated from some fixed distribution F_0. We then assign customer n a value $\mu_n = \mu_k^*$ if the customer sits at table k. The predictive distribution of μ_n given the values $\mu_1, \ldots, \mu_{n-1}$ of the first $n-1$ customers is

$$\mu_n \mid \mu_1, \ldots, \mu_{n-1} = \begin{cases} \mu_k^*, & \text{w. pr. } \frac{m_k}{\alpha+n-1}, \text{ for } k \leq K_+; \\ \mu_{K_+ +1}^* \sim F_0, & \text{w. pr. } \frac{\alpha}{\alpha+n-1}. \end{cases} \tag{1.10}$$

It can be shown that the random sequence $\mu_1, \mu_2, \ldots$ is infinitely exchangeable. By Equation (1.3), there exists a random distribution F such that $\mu_n \mid F \overset{iid}{\sim} F$ and $F \sim \upsilon$. Here υ is a prior over the random distribution F, which is known as the Dirichlet process (DP) (Ferguson, 1973). We denote by $\mathcal{DP}(\alpha, F_0)$ a DP with concentration parameter α and base distribution F_0. Equation (1.10) is also called the Pólya urn representation (Blackwell and MacQueen, 1973) of the DP. Using the notion of DP, we can re-parameterize Equations (1.4), (1.5) and (1.9) with a hierarchical model

$$\begin{aligned} y_n \mid \mu_n &\sim G(\cdot \mid \mu_n) \\ \mu_n \mid F &\sim F \\ F \mid \alpha, F_0 &\sim \mathcal{DP}(\alpha, F_0) \end{aligned} \tag{1.11}$$

The model (1.11) is called a Dirichlet process mixture (DPM) model.

The DP is probably the most popular nonparametric Bayesian model. Discussions and extensions of the DP include Blackwell (1973), Blackwell and MacQueen (1973), Antoniak (1974), Lo et al. (1984), Sethuraman (1994), Pitman and Yor (1997), MacEachern (2000), Müller et al. (2004), Teh et al. (2006), Rodriguez et al. (2008), Lijoi and Prünster (2010), Adams et al. (2010), Teh (2011), De Blasi et al. (2015), where Sethuraman (1994) proposes the

stick-breaking construction of the DP, Pitman and Yor (1997) extend the DP to the Pitman-Yor process, MacEachern (2000) proposes the dependent DP, Teh et al. (2006) develop the hierarchical DP, and Rodriguez et al. (2008) develop the nested DP. Literature about posterior inference methods for the DP includes West et al. (1994), Escobar and West (1995), MacEachern and Müller (1998), Neal (2000), Rasmussen (2000), Ishwaran and James (2001), Jain and Neal (2004), and Blei and Jordan (2006).

1.3.3 Related Applications

Latent class models, in particular, the DPM model and its variations, have been extensively used in many data analysis problems. We highlight their applications to inference for tumor heterogeneity and inference for missing data because of the relevance to the following chapters.

Inference for Tumor Heterogeneity. Tumor cell populations are typically heterogeneous consisting of multiple homogeneous subpopulations with unique genomes. Such subpopulations are known as subclones. Our goal is reconstructing such subclones from next-generation sequencing (NGS) data (Mardis, 2008). See more details in Chapters 2 and 3. One approach to this problem is to model the observed read count data using a latent class model. This approach is taken by PyClone (Roth et al., 2014) and PhyloWGS (Jiao et al., 2014, Deshwar et al., 2015). Tumor evolution is a complex process involving many biological details, such as tumor purity, copy number variations

and tumor phylogeny. Also, NGS data are subject to sequencing error and are often overdispersed. For simplicity of illustration we consider only one pure tumor tissue sample, ignore all the complexities mentioned above and also ignore the zygosity of the mutation sites. For detailed discussions see, for example, Roth et al. (2014), Jiao et al. (2014), Deshwar et al. (2015) and Chapters 2 and 3.

Consider S single nucleotide variants (SNVs). Here SNVs refer to the loci of the nucleotides (base pairs) for which we record variants. Variants are defined relative to some reference genome. The SNVs are the objects in the latent class model. In an NGS experiment, DNA fragments are first produced by extracting the DNA molecules from the cells in a tumor sample. The fragments are then sequenced using short reads. The short reads are mapped to the reference genome, and counts are recorded for each locus (i.e. base pair). In the end, for each SNV locus s ($s = 1, \ldots, S$), denote by N_s and n_s the total number of reads and number of variant reads covering the locus, respectively. The total number of reads N_s is usually treated as a fixed number. PyClone uses a DPM model for n_s,

$$
\begin{aligned}
n_s \mid p_s &\sim \text{Binom}(N_s; p_s), \\
p_s \mid F &\sim F, \\
F \mid \alpha, F_0 &\sim \mathcal{DP}(\alpha, F_0),
\end{aligned}
$$

where the base distribution F_0 is chosen to be $\text{Unif}(0, 1)$. Here p_s is known as the cellular prevalence of mutation s, i.e. the fraction of cancer cells harbour-

ing a mutation. The DP prior for F allows multiple mutations to share the same cellular prevalence. The critical step towards subclone reconstruction is the following. Mutations having the same cellular prevalence are thought of having occurred at the same point in the clonal phylogeny. Thus, latent classes of mutations can be used as markers of subclone populations (Roth et al., 2014). We note that this is essentially an application of latent class models to *clustering*. PhyloWGS, on the other hand, uses the tree-structured stick breaking process (TSSB) (Adams et al., 2010) as the prior for F,

$$F \mid \alpha, \gamma, F_0 \sim \text{TSSB}(\alpha, \gamma, F_0),$$

which allows it to infer tumor phylogeny.

One restriction of the latent class model is that each object can only belong to one class. In the tumor heterogeneity application, this restriction implies that each mutation can only occur once in the clonal phylogeny. Therefore, subclone reconstruction methods based on latent class models usually rely on the infinite site assumption (ISA) (Kimura, 1969), which can be summarized as (Roth et al., 2014)

1. Subclone populations follow a *perfect* phylogeny. That is, no SNV site mutates more than once in its evolutionary history;

2. Subclone populations follow a *persistent* phylogeny. That is, mutations do not disappear or revert.

However, ISA is not necessarily valid, in which case we should model the observed read count data using latent feature models. See Section 1.4.5.

Inference for Missing Data. Missing data are very common in real studies. Missingness is typically non-ignorable (Rubin, 1976, Little and Rubin, 2014), and in such cases the joint distribution of the full data response and missingness needs to be modeled. We focus on the missing outcome case. See Linero and Daniels (2017) for a general review of Bayesian nonparametric approach to missing outcome data. Let Y_{ij} denote the outcome that was planned to be collected for subject i at time j, and R_{ij} be the missingness indicator with $R_{ij} = 1$ or 0 accordingly as Y_{ij} is observed or not, $i = 1, \ldots, N$, $j = 1, \ldots, J$. Let $\boldsymbol{X}_i$ denote the covariates that are of primary interest to the study. In longitudinal clinical trial setting, $\boldsymbol{X}_i$ is usually an indicator of treatment. We often treat $\boldsymbol{X}_i$ as fixed and do not proceed with inference on it. The full data for subject i are $(\boldsymbol{Y}_i, \boldsymbol{R}_i, \boldsymbol{X}_i)$. The observed data for subject i are $(\boldsymbol{Y}_{i,\text{obs}}, \boldsymbol{R}_i, \boldsymbol{X}_i)$, where $\boldsymbol{Y}_{i,\text{obs}} = (Y_{ij} \mid j : R_{ij} = 1)$. We assume $(\boldsymbol{Y}_i, \boldsymbol{R}_i \mid \boldsymbol{X}_i) \overset{iid}{\sim} p(\boldsymbol{y}, \boldsymbol{r} \mid \boldsymbol{x})$. We stratify the model by $\boldsymbol{x}$ and suppress the conditional on $\boldsymbol{x}$ hereafter to simplify notation. The extrapolation factorization (Daniels and Hogan, 2008) factorizes

$$p(\boldsymbol{y}, \boldsymbol{r}) = p(\boldsymbol{y}_{\text{mis}} \mid \boldsymbol{y}_{\text{obs}}, \boldsymbol{r}) p(\boldsymbol{y}_{\text{obs}}, \boldsymbol{r}).$$

The observed data distribution $p(\boldsymbol{y}_{\text{obs}}, \boldsymbol{r})$ is identified by the observed data, while the extrapolation distribution $p(\boldsymbol{y}_{\text{mis}} \mid \boldsymbol{y}_{\text{obs}}, \boldsymbol{r})$ is not. Identifying the extrapolation distribution relies on untestable assumptions such as parametric models for the full data distribution or identifying restrictions. See Chapter 4 or Linero and Daniels (2017).

For current discussion, we focus on specifying $p(\boldsymbol{y}_{\text{obs}}, \boldsymbol{r})$. One way of specifying $p(\boldsymbol{y}_{\text{obs}}, \boldsymbol{r})$ is specifying $p(\boldsymbol{y}, \boldsymbol{r})$, and set

$$p(\boldsymbol{y}_{\text{obs}}, \boldsymbol{r}) = \int p(\boldsymbol{y}, \boldsymbol{r}) d\boldsymbol{y}_{\text{mis}}.$$

This is known as the working model idea (Linero and Daniels, 2015, Linero, 2017, Linero and Daniels, 2017). Linero and Daniels (2017) discuss a Bayesian nonparametric framework for modeling a complex joint distribution of outcome and missingness, which sets

$$p(\boldsymbol{y}, \boldsymbol{r}) = \sum_{k=1}^{K} \pi_k \, G(\cdot \mid \boldsymbol{\mu}_k). \tag{1.12}$$

Equation (1.12) is another way of writing a latent class model

$$\boldsymbol{Y}_i, \boldsymbol{R}_i \mid c_i = k, \boldsymbol{\mu}_k \sim G(\cdot \mid \boldsymbol{\mu}_k),$$
$$p(c_i = k \mid \pi_k) = \pi_k.$$

When $K = \infty$, mixture models of the form (1.12) can approximate any joint distribution for $(\boldsymbol{Y}_i, \boldsymbol{R}_i)$ (subject to technical constraints). We note that this is essentially an application of latent class models to *density estimation*.

Linero and Daniels (2017) use a model of the form (1.12) to analyze data from the Breast Cancer Prevention Trial. In this trial, $Y_{ij} = 1$ or 0 represent subject i is depressed or not at time j. They model

$$p(\boldsymbol{y}, \boldsymbol{r}) = \sum_{k=1}^{\infty} \pi_k \left\{ \prod_{j=1}^{J} \gamma_{kj}^{r_j} (1 - \gamma_{kj})^{1 - r_j} \right\} \left\{ \prod_{j=1}^{J} \beta_{kj}^{y_j} (1 - \beta_{kj})^{1 - r_j} \right\}.$$

Linero and Daniels (2015) discussion is another example of applying latent class models to inference for missing data. They analyze data from an

acute schizophrenia clinical trial, where the outcome Y_{ij} is a continuous variable called the positive and negative syndrome scale (PANSS) score. Missingness is monotone in this application. That means, if Y_{ij} was unobserved then $Y_{i,j+1}$ was also unobserved. Let S_i denote the dropout time, i.e. if $S_i = j$ then Y_{ij} was observed but $Y_{i,j+1}$ was not. For monotone missingness, S captures all the information about missingness. Denote by $\bar{\boldsymbol{Y}}_{ij} = (Y_{i1}, \ldots, Y_{ij})$ the history of outcomes through the first j times. They model

$$p(\boldsymbol{y}, s) = \sum_{k=1}^{\infty} \pi_k f(\boldsymbol{y} \mid \boldsymbol{\mu}_k, \Sigma_k) g(s \mid \boldsymbol{y}, \boldsymbol{\zeta}_k, \boldsymbol{\gamma}_k),$$

with

$$f(\boldsymbol{y} \mid \boldsymbol{\mu}, \Sigma) = N(\boldsymbol{y}; \boldsymbol{\mu}, \Sigma),$$
$$\text{logit}[g(S = j \mid S \geq j, \boldsymbol{y}, \boldsymbol{\zeta}, \boldsymbol{\gamma})] = \zeta_j + \boldsymbol{\gamma}_j^T \bar{\boldsymbol{y}}_j.$$

1.4 Latent Feature Models

Suppose we have N objects, $y_1, \ldots, y_N$. In a *latent feature model* (for a review, see Griffiths and Ghahramani, 2011), each object y_n is represented by a vector of latent feature values $\boldsymbol{d}_n = (d_{n1}, \ldots, d_{nK})$, where K is the number of features. Similar to the latent class model case, $K = \infty$ is allowed, in which case the model is nonparametric. Examples of latent feature models include probabilistic principle component analysis (Tipping and Bishop, 1999) and factor analysis (Roweis and Ghahramani, 1999). We focus on the case that there is no upper bound on the number of features.

We can break the vector $\boldsymbol{d}_n$ into two components: a binary vector $\boldsymbol{z}_n$ with $z_{nk} = 1$ if object n has feature k and 0 otherwise, and a second vector $\boldsymbol{v} = (v_1, \ldots, v_K)$ indicating the value of each feature. The vector $\boldsymbol{d}_n$ can be expressed as the elementwise product of $\boldsymbol{z}_n$ and $\boldsymbol{v}$, i.e.

$$d_{nk} = z_{nk} v_k, \quad k = 1, \ldots, K. \tag{1.13}$$

Let $\boldsymbol{D} = (\boldsymbol{d}_1, \ldots, \boldsymbol{d}_N)^T$ and $\boldsymbol{Z} = (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N)^T$ denote $N \times K$ matrices with columns $\boldsymbol{d}_n$ and $\boldsymbol{z}_n$, respectively. We are interested in how the feature values are related to the data, $p(\boldsymbol{y} \mid \boldsymbol{D})$, and the distribution over feature values, $p(\boldsymbol{D})$. For $p(\boldsymbol{y} \mid \boldsymbol{D})$, we generally assume conditional independence

$$\begin{aligned} p(\boldsymbol{y} \mid \boldsymbol{D}) &= \prod_{n=1}^{N} p(y_n \mid \boldsymbol{d}_n), \\ p(y_n \mid \boldsymbol{d}_n) &\sim G(\cdot \mid \boldsymbol{d}_n). \end{aligned} \tag{1.14}$$

The distribution $p(\boldsymbol{D})$ can be broken into two components, with $p(\boldsymbol{D})$ being determined by $p(\boldsymbol{Z})$ and $p(\boldsymbol{v})$. We assume an independent prior on $\boldsymbol{v}$,

$$v_1, \ldots, v_K \overset{iid}{\sim} F_0.$$

We then focus on defining a prior on $\boldsymbol{Z}$. Specifying $p(\boldsymbol{Z})$ is equivalent to defining a distribution on a *random feature allocation* of the index set $[N]$. The formal definition of random feature allocation follows in Section 1.4.1. The equivalence can be seen by noticing that the values of $\boldsymbol{Z}$ correspond to a feature allocation $f_N = \{A_1, \ldots, A_K\}$, with $n \in A_k$ if $z_{nk} = 1$ and $n \notin A_k$ if $z_{nk} = 0$, and vice versa.

1.4.1 Random Feature Allocation

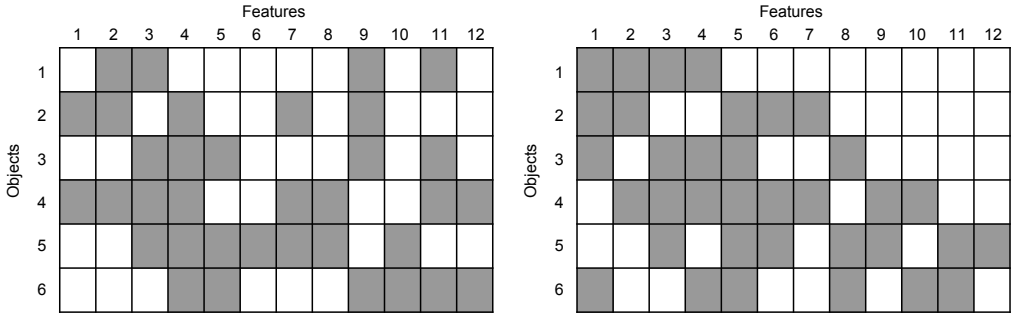
We briefly summarize the definition of random feature allocations from Broderick, Jordan, and Pitman (2013). See there for more details and discussion. Feature allocations could be seen as a generalization of partitions which relaxes the restriction to mutually exclusive and exhaustive subsets. Consider an index set $[N]$. A *feature allocation* f_N of $[N]$ is a multiset of non-empty subsets of $[N]$ called *features*, such that no index n belongs to infinitely many features. Denote by $f_N = \{A_1, \ldots, A_K\}$, where K is the number of features. Here $N = \infty$ and $K = \infty$ are allowed. For example, a feature allocation of [6] is $f_6 = \{\{1, 4\}, \{1, 3, 6\}, \{4\}, \{4\}, \{4\}\}$.

Let $\mathcal{F}_N$ be the space of all feature allocations of $[N]$. A *random feature allocation* F_N of $[N]$ is a random element of $\mathcal{F}_N$.

Let $\sigma : \mathbb{N} \to \mathbb{N}$ be a finite permutation. That is, for some finite value N_σ, $\sigma(n) = n$ for all $N > N_\sigma$. Furthermore, for any feature $A \subset \mathbb{N}$, denote the permutation applied to the feature as $\sigma(A) := \{\sigma(n) : n \in A\}$. For any feature allocation F_N, denote the permutation applied to the feature allocation as $\sigma(F_N) := \{\sigma(A) : A \in F_N\}$. Let F_N be a random feature allocation of $[N]$. A random feature allocation F_N is called *exchangeable* if $F_N \stackrel{d}{=} \sigma(F_N)$ for every permutation of $[N]$.

Matrix Representation. Suppose N objects and K features are present. A feature allocation can be represented by a $N \times K$ binary matrix, denoted by $\boldsymbol{Z}$. Rows of $\boldsymbol{Z}$ correspond to the index set $[N]$, and columns of $\boldsymbol{Z}$ correspond to

features. Each element z_{nk}, $n = 1, \ldots, N$, $k = 1, \ldots, K$, is a binary indicator, where $z_{nk} = 0$ or 1 indicates index n belongs or does not belong to feature k, i.e. $n \notin A_k$ or $n \in A_k$, respectively. For example, Figure 1.1(a) shows a feature allocation of [6] with 12 features, $f_6 = \{\{2,4\}, \{1,2,4\}, \{1,3,4,5\}, \{2,3,4,5,6\}, \{3,5,6\}, \{5\}, \{2,4,5\}, \{4,5\}, \{1,2,3,6\}, \{5,6\}, \{1,3,4,6\}, \{4,6\}\}$.



(a) (b)

Figure 1.1: An example of binary matrix representation of feature allocation. A shaded rectangle indicates the corresponding matrix element $z_{nk} = 1$. The binary matrix on (a) is transformed into the left-ordered binary matrix on (b) by the function $lof(\cdot)$.

1.4.2 Indian Buffet Process

The Indian buffet process (IBP) (Griffiths and Ghahramani, 2006, 2011)) is a popular example of an exchangeable random feature allocation. Using the matrix representation of feature allocation, the IBP defines a distribution on binary matrices (with an unbounded random number of columns).

A Finite Feature Model. The IBP can be defined as the limit of a finite feature model. Suppose we have N objects and K features. We use a binary variable z_{nk} to indicate object n has feature k, thus z_{nk} form a binary $N \times K$ matrix $\boldsymbol{Z}$. We assume that each object possesses feature k with probability π_k, and the features are independent. Furthermore, beta distribution priors are put on π_k's. That is,

$$z_{nk} \mid \pi_k \sim \text{Bernulli}(\pi_k);$$
$$\pi_k \sim \text{Beta}(\alpha/K, 1).$$

Integrating out the π_k's, the marginal distribution of $\boldsymbol{Z}$ is

$$p(\boldsymbol{Z}) = \prod_{k=1}^{K} \frac{\frac{\alpha}{K}\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \frac{\alpha}{K})},$$

where $m_k = \sum_{n=1}^{N} z_{nk}$ is the number of objects possessing feature k. This distribution is exchangeable, since it only depends on the counts and does not depend on the ordering of the objects.

Left-ordered Constraint and Equivalence Classes. A feature allocation indicates an ordering of the K features. In many applications, the ordering of the features is not identifiable. When the labels of the features are arbitrary, it is helpful to define an equivalence class of binary matrices, denoted by $[\boldsymbol{Z}]$. We first introduce an order constraint on binary matrices called the *left-ordered constraint*. For a binary matrix $\boldsymbol{Z}$, its corresponding left-ordered binary matrix, denoted by $lof(\boldsymbol{Z})$, is obtained by ordering the columns of $\boldsymbol{Z}$

from left to right by the magnitude of the binary number expressed by that column, taking the first row as the most significant bit. For example, Figure 1.1(b) shows the corresponding left-ordered binary matrix of Figure 1.1(a). In the first row of the left-ordered matrix, the columns for which $z_{1k} = 1$ are grouped at the left. In the second row, the columns for which $z_{2k} = 1$ are grouped at the left of the sets for which $z_{1k} = 1$. This grouping structure persists throughout the matrix.

We can then define equivalence classes with respect to the function $lof(\cdot)$. This function maps binary matrices to left-ordered binary matrices, as described before. The function $lof(\cdot)$ is many-to-one: many binary matrices reduce to the same left-ordered form, and there is a unique left-ordered form for every binary matrix. Any two binary matrices $\boldsymbol{Y}$ and $\boldsymbol{Z}$ are $lof(\cdot)$ equivalent if $lof(\boldsymbol{Y}) = lof(\boldsymbol{Z})$. In models where feature order is not identifiable, performing inference at the level of lof-equivalence classes is appropriate. The probability of a particular lof-equivalence class of binary matrices $[\boldsymbol{Z}]$ is $p([\boldsymbol{Z}]) = \sum_{\boldsymbol{Z} \in [\boldsymbol{Z}]} p(\boldsymbol{Z})$.

The matrix left-ordered form motivates the following definition. The *history* of feature k at object n is defined to be $(z_{1k}, \ldots, z_{(n-1)k})$. When n is not specified, history refers to the full history of feature k, $(z_{1k}, ..., z_{Nk})$. The histories of features are individuated using the decimal equivalent of the binary numbers corresponding to the column entries. For example, at object 3, features can have one of four histories: 0, corresponding to a feature with no previous assignments, 1, being a feature for which $z_{2k} = 1$ but $z_{1k} = 0$, 2,

being a feature for which $z_{1k} = 1$ but $z_{2k} = 0$, and 3, being a feature possessed by both previous objects were assigned. The number of features possessing the history h is denoted by K_h, with K_0 being the number of features for which $m_k = 0$ and $K_+ = \sum_{h=1}^{2^N-1} K_h$ being the number of features for which $m_k > 0$, so $K = K_0 + K_+$. The function lof thus places the columns of a matrix in ascending order of their histories.

Using the notion above, the cardinality of $[\boldsymbol{Z}]$ is $\left(K!/\prod_{h=0}^{2^N-1} K_h!\right)$. Thus,

$$p([\boldsymbol{Z}]) = \frac{K!}{\prod_{h=0}^{2^N-1} K_h!} \cdot \prod_{k=1}^{K} \frac{\frac{\alpha}{K}\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \frac{\alpha}{K})}. \tag{1.15}$$

Taking the Infinite Limit. Taking the limit $K \to \infty$ in Equation (1.15),

$$\lim_{K\to\infty} p([\boldsymbol{Z}]) = \frac{\alpha^{K_+}}{\prod_{h=1}^{2^N-1} K_h!} \cdot \exp\{-\alpha H_N\} \cdot \prod_{k=1}^{K_+} \frac{(N - m_k)!(m_k - 1)!}{N!}, \tag{1.16}$$

where H_N is the N-th harmonic number, $H_N = \sum_{j=1}^{N} 1/j$. See Griffiths and Ghahramani (2011) for details. This distribution is still exchangeable. In practice, we usually drop all columns with all zeros, since they corresponds to the features that no object possesses, and it should not be included in the feature allocation as features are non-empty sets. It can be proved that we can obtain a matrix with finite columns with probability 1 by deleting the columns with all zeros.

Indian Buffet Analogy. The probability distribution defined in Equation (1.16) can be derived from a simple stochastic process, which is referred to

as the IBP. Think about an Indian buffet where customers (objects) choose dishes (features). In the buffet, N customers enter one after another, and each customer encounters infinitely many dishes arranged in a line. The first customer starts at the left of the buffet and takes a serving from each dish, stopping after a Poisson(α) number of dishes. The n-th customer moves along the buffet, sampling dishes in proportion to their popularity, serving himself with probability m_k/n, where m_k is the number of previous customers who have sampled a dish. Having reached the end of all previously sampled dishes, the n-th customer then tries a Poisson(α/n) number of new dishes. We use a binary matrix $\boldsymbol{Z}$ with N rows and infinitely many columns to indicate which customers chose which dishes, where $z_{nk} = 1$ if the n-th customer sampled the k-th dish. The matrices produced by this process are generally not in left-ordered form, and customers are not exchangeable under this distribution. However, if we only record the lof-equivalence classes of the matrices generated by this process, one obtains the exchangeable distribution $p([\boldsymbol{Z}])$ given by Equation (1.16).

Beta Process. Similar to the relationship between the DP and the CRP, the de Finetti's measure underlying the exchangeable distribution produced by the IBP is the beta process (BP) (Hjort, 1990). See full details in Thibaux and Jordan (2007).

Other discussions of the IBP and the BP include Teh et al. (2007), Teh and Gorur (2009), Doshi et al. (2009), Paisley et al. (2010), Williamson et al.

(2010), Knowles and Ghahramani (2011), and Miller et al. (2012).

1.4.3 Latent Categorical Feature Models

In a latent feature model of the form (1.13) and (1.14), a feature can either have the same effect on the objects possessing it, or have no effect on the objects not possessing it. This is sometimes too restrictive. One relaxation is to assume different effects of each feature on different objects, as seen in Griffiths and Ghahramani (2011). That is, in Equation (1.13), v_k can depend on n, and $d_{nk} = z_{nk}v_{nk}$. We can then calibrate prior on v_{nk} according to the specific application.

In some applications, for example, the tumor heterogeneity application in Chapters 2 and 3, it is natural to categorize the features. To elaborate, let $z_{nk} = 0, 1, \ldots, Q$ indicating object n does not possess feature k, if $z_{nk} = 0$, or possesses category q of feature k, if $z_{nk} = q$, $q = 1, \ldots, Q$. A feature has the same effect on the objects possessing the same category of it, while the feature has different effects on the objects possessing different categories of it. That is, in Equation (1.13),

$$d_{nk} = g(z_{nk}) \cdot v_k, \quad k = 1, \ldots, K, \tag{1.17}$$

where $g : \{0, \ldots, Q\} \mapsto \mathbb{R}$ represents those $(Q + 1)$ types of effects. We do not restrict $g(0) \equiv 0$. That is, a feature can have effect on the objects that do not possess it. We will see this is the case in the application in Chapters 2 and 3. We hereafter refer to this type of latent feature models (Equations (1.14) and

(1.17)) as *latent categorical feature models*. To complete the model, we need to define a distribution on categorical valued matrices.

1.4.4 Categorical Indian Buffet Process

The categorical Indian buffet process (cIBP) (Sengupta, 2013) is a categorical extension of the IBP, which defines a distribution on categorical valued matrices (with a random and unbounded number of columns). Each entry of the matrix $\boldsymbol{Z}$ can take values from a set of integers $\{0, 1, \ldots, Q\}$, where Q is fixed. Here $z_{nk} = 0$ indicates object n does not possess feature k, and $z_{nk} = q$, $q = 1, \ldots, Q$ represents object n possesses category q of feature k.

A Finite Feature Model. Similar to the construction of the IBP, the cIBP can be derived as a limit of finite feature allocation models. Assume that each object possesses category q of feature k with probability π_{kq}, i.e. $Pr(z_{nk} = q) = \pi_{kq}$, and the features are independent. Furthermore, beta-Dirichlet distribution (Kim et al., 2012) priors are put on $\boldsymbol{\pi}_k$'s. That is, $\pi_{k0} = Pr(z_{nk} = 0)$ follows a beta distribution with parameters 1 and α/K, i.e. $(1 - \pi_{k0}) = Pr(z_{nk} \neq 0) \sim \text{Beta}(\alpha/K, 1)$. Let $\tilde{\pi}_{kq} = \pi_{kq}/(1 - \pi_{k0})$, $q = 1, \cdots, Q$. Then $(\tilde{\pi}_{k1}, \ldots, \tilde{\pi}_{kQ})$ follows a Dirichlet distribution with parameters $(\beta_1, \ldots, \beta_Q)$. In summary,

$$
\begin{aligned}
z_{nk} \mid \boldsymbol{\pi}_k &\sim \text{Categorical}(\boldsymbol{\pi}_k), \\
\boldsymbol{\pi}_k &\sim \text{Beta-Dirichlet}(\alpha/K, 1, \beta_1, \ldots, \beta_Q).
\end{aligned}
$$

For simplicity we only consider a symmetric Dirichlet distribution $\beta_1 = \cdots = \beta_Q = \beta$, which is sufficient in most cases. Integrating out the $\boldsymbol{\pi}_k$'s, the marginal distribution of $\boldsymbol{Z}$ is

$$p(\boldsymbol{Z}) = \prod_{k=1}^{K} \left\{ \frac{\Gamma(Q\beta)\Gamma(\frac{\alpha}{K}+1)}{\Gamma(\frac{\alpha}{K})(\Gamma(\beta))^Q} \cdot \frac{\left[\prod_{q=1}^{Q}\Gamma(\beta+m_{kq})\right]\Gamma(N-m_{k\cdot}+1)\Gamma(\frac{\alpha}{K}+m_{k\cdot})}{\Gamma(\frac{\alpha}{K}+N+1)\Gamma(Q\beta+m_{k\cdot})} \right\}, \quad (1.18)$$

where $m_{kq} = \sum_{n=1}^{N} I(z_{nk} = q)$ denotes the number of objects from total N objects possessing category q of feature k, and $m_{k\cdot} = \sum_{q=1}^{Q} m_{kq}$ denotes the number of objects possessing feature k. The distribution is exchangeable since it depends on the counts m_{kq} only.

Left-ordered Constraint and Equivalent Classes. Similar to what was defined for binary matrices, we can define *left-ordered form* and *history* on $(Q+1)$-nary matrices. A left-ordered $(Q+1)$-nary matrix or $lof(\boldsymbol{Z})$ is obtained by ordering the columns of $\boldsymbol{Z}$ from left to right by the magnitude of the $(Q+1)$-nary number (i.e represented in base $(Q+1)$) expressed by that column taking first row as the most significant bit. Figure 1.2(a) shows an example of $(Q+1)$-nary matrix, where $Q = 3$, and Figure 1.2(b) shows the corresponding left-ordered matrix. The lof-equivalence class of matrix $\boldsymbol{Z}$ is still denoted by $[\boldsymbol{Z}]$. The *history* of feature k at object n is defined to be the decimal equivalent of the $(Q+1)$-nary number represented by the vector $(z_{1k}, \ldots, z_{(n-1)k})$, and the full history of feature k refers to the decimal equivalent of the $(Q+1)$-nary number of $(z_{1k}, \ldots, z_{Nk})$. The number of features having history h is

denoted by K_h, with K_0 being the number of features for which $m_{k\cdot} = 0$ and $K_+ = \sum_{h=1}^{(Q+1)^N-1} K_h$ being the number of features for which $m_{k\cdot} > 0$, and $K = K_0 + K_+$.

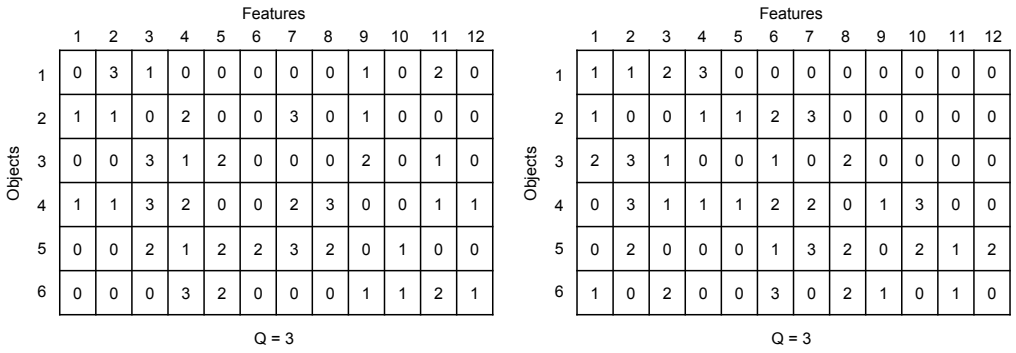


Figure 1.2: An example of $(Q+1)$-nary matrix with $Q = 3$. The matrix on (a) is transformed into the left-ordered matrix on (b) by the function $lof(\cdot)$.

Using the notions above, the cardinality of $[\boldsymbol{Z}]$ is $\left(K!/\prod_{h=0}^{(Q+1)^N-1} K_h!\right)$, and

$$p([\boldsymbol{Z}]) = \sum_{\boldsymbol{Z}\in[\boldsymbol{Z}]} p(\boldsymbol{Z}) = \frac{K!}{\prod_{h=0}^{(Q+1)^N-1} K_h!} \cdot p(\boldsymbol{Z}). \tag{1.19}$$

Taking the Infinite Limit. Taking the limit $K \to \infty$ in Equations (1.19) and (1.18), we obtain

$$\lim_{K\to\infty} p([\boldsymbol{Z}]) = \frac{(\alpha/Q)^{K_+}}{\prod_{h=1}^{(Q+1)^N-1} K_h!} \cdot \exp\{-\alpha H_N\} \cdot \prod_{k=1}^{K_+} \left\{ \frac{(N-m_{k\cdot})!(m_{k\cdot}-1)!}{N!} \cdot \frac{1}{\prod_{j=1}^{m_{k\cdot}-1}(j+Q\beta)} \frac{1}{\beta} \prod_{q=1}^{Q} \frac{\Gamma(\beta+m_{kq})}{\Gamma(\beta)} \right\}. \tag{1.20}$$

Details in Sengupta (2013). This distribution is exchangeable as in the finite case. The last step is dropping the columns with all zeros.

Indian Buffet Analogy. The probability distribution defined in Equation (1.20) can also be derived from a stochastic process similar to the IBP, which is referred to as the cIBP. The customers are the objects and the dishes are the features. The Q categories of a feature can be seen as different spice levels of a dish. Start with N customers and an infinite number of dishes. As the i-th customer walks in, he/she chooses the k-th dish with a particular spice level q with probability $[m_{k\cdot} \cdot (\beta_q + m_{kq})] \,/\, [i \cdot (\beta_* + m_{k\cdot})]$, where m_{kq} denotes the number of customers who have tasted the dish k with spice level q, $m_{k\cdot} = \sum_{q=1}^{Q} m_{kq}$ denotes the number of customers who have tasted the dish k in total, and $\beta_* = \sum_{q=1}^{Q} \beta_q$. Then the i-th customer tastes new dishes with spice level q determined by a draw from Poisson$[(\beta_q \cdot \alpha)/(\beta_* \cdot i)]$. Using a $(Q+1)$-nary matrix $\boldsymbol{Z}$ with N rows and infinitely many columns to indicate which customers chose which dishes, where $z_{nk} = 0$ if the n-th customer did not choose the k-th dish, and $z_{nk} = q$ if the n-th customer chose the k-th dish with spice level q. If one only pays attention to the lof-equivalence classes of the matrices generated by this process, one obtains the exchangeable distribution $p([\boldsymbol{Z}])$ given by Equation (1.20).

1.4.5 Inference for Tumor Heterogeneity Using Feature Allocation Models

We continue the discussion of inference for tumor heterogeneity in Section 1.3.3. Recall that one approach to this problem is to model the read counts using a latent class model, where the SNVs are the objects, and the subclones are the classes. We have discussed in Section 1.3.3 that subclone reconstruction methods based on latent class models usually rely on the ISA. However, ISA is not necessarily valid because multiple tumor subclones might acquire the same mutation in convergent evolution. See more discussions in Marass et al. (2017). Such concerns inspire another approach to this problem: modeling the read counts using a latent feature model. Clomial (Zare et al., 2014), Lee et al. (2015), BayClone (Sengupta et al., 2015), Cloe (Marass et al., 2017) and PairClone (Chapters 2, 3) take this approach. We briefly summarize BayClone and Cloe here.

Still, ignoring many biological details, consider S SNVs. Let N_s and n_s denote the total and variant read counts covering locus s, respectively, $s = 1, \ldots, S$. BayClone models the variant read counts using a latent (categorical) feature model

$$
\begin{aligned}
& n_s \mid p_s \sim \text{Binom}(N_s; p_s), \\
& p_s \mid \boldsymbol{w}, \boldsymbol{Z} = \sum_{c=1}^{C} w_c \, g(z_{sc}), \\
& p(z_{sc} = q \mid \pi_{cq}) = \pi_{cq}, \\
& (\pi_{c0}, \ldots, \pi_{cQ}) \sim \text{Beta-Dirichlet}(\alpha/C, 1, \beta, \ldots, \beta)
\end{aligned}
$$

where $c = 1, \ldots, C$ represent the C subclones (i.e. C features), w_c is the cellular proportion of subclone c (i.e. value of feature c), and z_{sc} is the genotype, with $z_{sc} = 0$, 1 or 2 corresponding to a homozygous wild-type, heterozygous variant or homozygous variant at site s of subclone c. The mapping g maps $g(0) = 0$, $g(1) = 0.5$ and $g(2) = 1$, which indicates the contributions of three different genotypes to the expected variant allele fraction p_s.

Cloe, on the other hand, uses a column-dependent phylogenetic prior for $\boldsymbol{Z}$, which allows it to infer tumor phylogeny.

1.5 Regression

Given two or more observables, *regression analysis* is concerned with the relationship between a dependent variable (or response), denote by y, and one or more independent variables (or predictors), denote by $\boldsymbol{x}$. We usually treat $\boldsymbol{x}$ as fixed quantities and treat y as random variables. We focus on the case that $y \in \mathbb{R}$ is continuous and is normally distributed. For more general cases (e.g. generalized linear models) see, for example, Dobson and Barnett (2008) or Dey et al. (2000). Suppose we have N observations $\{(y_n, \boldsymbol{x}_n) : n = 1, \ldots, N\}$. For observation n, we assume

$$y_n = f(\boldsymbol{x}_n) + \varepsilon_n, \tag{1.21}$$

where

$$\varepsilon_n \overset{iid}{\sim} N(0, \sigma^2)$$

is normally distributed random error.

In the linear regression setting, the function f in Equation (1.21) is specified as a linear function

$$f(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{\beta},$$

where $\boldsymbol{\beta}$ is a parameter vector, and the elements of $\boldsymbol{\beta}$ are called regression coefficients. The model is completed with a prior on $\boldsymbol{\beta}$, which is usually a normal conjugate prior. Linear regression model is a traditional statistical model and has been studied extensively. For a review, see, for example, Gelman et al. (2014) or Christensen (2011).

In many applications, the linearity assumption is too restrictive. One possible generalization using nonparametric Bayesian models is to replace the linear model with a less restrictive flexible prior on f. A popular prior specification for f is the Gaussian process, which we will describe in detail in Section 1.5.1. Another popular specification for f is based on a function basis $\mathcal{G} = \{g_1, g_2, \ldots\}$ and a representation of f as $f(\boldsymbol{x}) = \sum_j \beta_j g_j(\boldsymbol{x})$. A prior model for $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots)$ induces a prior model for f. See Müller and Quintana (2004) for a review.

1.5.1 Gaussian Process

The Gaussian process (GP) (O'Hagan, 1978, Neal, 1998, Rasmussen and Williams, 2006) defines a distribution over random functions (stochastic processes). A GP is a collection of random variables $\{f(\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{X}\}$ such

that, for any finite number of indices $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathcal{X}$, the joint distribution of $[f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N)]^T$ is multivariate normal.

A GP is completely characterized by its mean function $m(\boldsymbol{x}) : \mathcal{X} \to \mathbb{R}$ and covariance function $C(\boldsymbol{x}, \boldsymbol{x}') : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$, where

$$m(\boldsymbol{x}) = \mathrm{E}[f(\boldsymbol{x})],$$
$$C(\boldsymbol{x}, \boldsymbol{x}') = \mathrm{Cov}[f(\boldsymbol{x}), f(\boldsymbol{x}')].$$

We denote by $\mathcal{GP}(m(\boldsymbol{x}), C(\boldsymbol{x}, \boldsymbol{x}'))$ a GP with mean function m and covariance function C, and write $f(\boldsymbol{x}) \sim \mathcal{GP}(m(\boldsymbol{x}), C(\boldsymbol{x}, \boldsymbol{x}'))$ if f has a GP prior.

A common choice of the mean function is the linear function,

$$m(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{\beta},$$

which means we center the GP on a linear model. Suppose $\boldsymbol{x} = (x_1, \ldots, x_p)$. A common choice of the covariance function is the squared exponential covariance function,

$$C(\boldsymbol{x}, \boldsymbol{x}') = \tau^2 \exp\left[-\sum_{j=1}^{p} \frac{(x_j - x_j')^2}{2l_j^2}\right] + \tau_0^2 \delta(\boldsymbol{x}, \boldsymbol{x}'),$$

where $\tau^2, \tau_0^2, l_1^2, \ldots, l_p^2$ are hyperparameters, and $\delta(\boldsymbol{x}, \boldsymbol{x}')$ is the Kronecker delta function that takes the value 1 if $\boldsymbol{x} = \boldsymbol{x}'$ and 0 otherwise. Here τ^2 controls the magnitude of C, and $l_1^2, \ldots, l_p^2$ (called length scales) control the smoothness of C. The function $\delta(\boldsymbol{x}, \boldsymbol{x}')$ is used to introduce small nugget for the diagonal covariances, which overcomes near-singularity of the covariance matrices and improves numerical stability. The nugget term τ_0^2 is usually chosen small, e.g.

$\tau_0^2 = 0.01$. For simplicity, sometimes we set $l_1^2 = \ldots = l_p^2 = l^2$, in which case C is called isotropic. For a detailed discussions of different covariance functions, see Rasmussen and Williams (2006) (Chapter 4).

Basis Expansions. An alternative way to derive the GP is using basis expansions. Consider H basis functions $\phi_1(\boldsymbol{x}), \ldots, \phi_H(\boldsymbol{x})$ and let $\boldsymbol{\phi}(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}), \ldots, \phi_H(\boldsymbol{x})]^T$. Let

$$f(\boldsymbol{x}) = \boldsymbol{\phi}(\boldsymbol{x})^T \boldsymbol{\beta}, \quad \boldsymbol{\beta} \sim N(\boldsymbol{\beta}_0, \Sigma_\beta). \tag{1.22}$$

Integrating out the $\boldsymbol{\beta}$, for any finite number of indices $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$, we have

$$[f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N)]^T \sim N[(m(\boldsymbol{x}_1), \ldots, m(\boldsymbol{x}_N))^T, S],$$

where

$$m(\boldsymbol{x}) = \boldsymbol{\phi}(\boldsymbol{x})^T \boldsymbol{\beta}_0,$$
$$C(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\phi}(\boldsymbol{x})^T \Sigma_\beta \boldsymbol{\phi}(\boldsymbol{x}'),$$

and S is a covariance matrix with (i, j)-th element $S_{ij} = C(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Therefore, f is a GP. When $\boldsymbol{\phi}(\boldsymbol{x}) = \boldsymbol{x}$, Equation (1.22) reduces to a linear model, and we can see linear regression is a special case of GP with covariance function $C(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^T \Sigma_\beta \boldsymbol{x}'$. The number of basis functions H needs not to be finite. For example, when

$$\phi_h(x) = \exp\left[-\frac{(x-h)^2}{2l^2}\right],$$

and $H \to \infty$ (consider scalar x for simplicity), Equation (1.22) leads to a GP with squared exponential covariance function (details in Rasmussen and Williams, 2006).

Inference. We are usually interested in predicting the value of f at some location $\tilde{\boldsymbol{x}}$ given the observed data $\{(y_n, \boldsymbol{x}_n) : n = 1, \ldots, N\}$. Denote by $\boldsymbol{y} = (y_1, \ldots, y_N)^T$, $\boldsymbol{m} = [m(\boldsymbol{x}_1), \ldots, m(\boldsymbol{x}_N)]^T$, $\tilde{f} = f(\tilde{\boldsymbol{x}})$ and $\tilde{m} = m(\tilde{\boldsymbol{x}})$. The joint distribution of $\boldsymbol{y}$ and $\tilde{f}$ is

$$\begin{pmatrix} \boldsymbol{y} \\ \tilde{f} \end{pmatrix} \sim N\left[\begin{pmatrix} \boldsymbol{m} \\ \tilde{m} \end{pmatrix}, \begin{pmatrix} C(X,X) + \sigma^2 I & C(X, \tilde{\boldsymbol{x}}) \\ C(\tilde{\boldsymbol{x}}, X) & C(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{x}}) \end{pmatrix}\right].$$

The posterior predictive distribution of $\tilde{f}$ is thus

$$\tilde{f} \mid \boldsymbol{y} \sim N\Big[\tilde{m} + C(\tilde{\boldsymbol{x}}, X)[C(X,X) + \sigma^2 I]^{-1}(\boldsymbol{y} - \boldsymbol{m}),$$
$$C(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{x}}) - C(\tilde{\boldsymbol{x}}, X)[C(X,X) + \sigma^2 I]^{-1} C(X, \tilde{\boldsymbol{x}})\Big].$$

We can then add one more step to predict the response value $\tilde{y}$ at $\tilde{\boldsymbol{x}}$,

$$\tilde{y} \mid \tilde{f} \sim N(\tilde{f}, \sigma^2).$$

Recent literature on results and generalizations of GP priors includes the following. Neal (1995) reveals the connection between neural networks (with one hidden layer and an infinite number of units) and Gaussian processes, Ghosal and Roy (2006) and Choi and Schervish (2007) discuss posterior consistency, Gramacy and Lee (2008) develop treed Gaussian processes, and Banerjee et al. (2008, 2013), Hensman et al. (2013) and Datta et al. (2016) develop efficient computational algorithms.

1.5.2 Inference for Missing Data Using Nonparametric Regression Models

We continue the discussion of inference for monotone missing data in Section 1.3.3. Recall that one approach to this problem is to model the joint

distribution of the full data response $\boldsymbol{Y}_i$ and dropout S_i (conditional on the covariates of primary interest $\boldsymbol{X}_i$) using a latent class model. In many cases, in addition to $(\boldsymbol{Y}_i, S_i, \boldsymbol{X}_i)$, we would have access to a set of *auxiliary covariates*, denoted by $\boldsymbol{V}_i$. Such covariates, although are not of direct interest, can often provide information about the missing responses and missing data mechanism. See Daniels and Hogan (2008) and Daniels et al. (2014) for more discussion. In this setting, we should incorporate $\boldsymbol{V}_i$ and consider a joint model for $(\boldsymbol{Y}_i, S_i, \boldsymbol{V}_i \mid \boldsymbol{X}_i)$, denoted by $p(\boldsymbol{y}, s, \boldsymbol{v} \mid \boldsymbol{x})$. Here we proceed with inference unconditional on $\boldsymbol{v}$, because the primary interest is in $p(\boldsymbol{y}, s \mid \boldsymbol{x})$, and

$$p(\boldsymbol{y}, s \mid \boldsymbol{x}) = \int p(\boldsymbol{y}, s, \boldsymbol{v} \mid \boldsymbol{x}) d\boldsymbol{v}.$$

Still, we stratify the model by $\boldsymbol{x}$ and suppress the conditional on $\boldsymbol{x}$. Under the extrapolation factorization,

$$p(\boldsymbol{y}, s, \boldsymbol{v}) = p(\boldsymbol{y}_{\text{mis}} \mid \boldsymbol{y}_{\text{obs}}, s, \boldsymbol{v}) p(\boldsymbol{y}_{\text{obs}}, s, \boldsymbol{v}).$$

In Chapter 4, we specify $p(\boldsymbol{y}_{\text{obs}}, s, \boldsymbol{v})$ based on pattern-mixture modeling (Little, 1993),

$$p(\boldsymbol{y}_{\text{obs}}, s, \boldsymbol{v}) = p(\boldsymbol{y}_{\text{obs}} \mid s, \boldsymbol{v}) p(s \mid \boldsymbol{v}) p(\boldsymbol{v}).$$

The models $p(\boldsymbol{y}_{\text{obs}} \mid s, \boldsymbol{v})$ and $p(s \mid \boldsymbol{v})$ are regression models. We then specify

$$[Y_j \mid \bar{Y}_{j-1} = \bar{\boldsymbol{y}}_{j-1}, S = s, \boldsymbol{V} = \boldsymbol{v}] = a(\boldsymbol{v}, j, s) + \bar{\boldsymbol{y}}'_{j-1} \boldsymbol{\Phi}_{js} + \varepsilon_{js}, \quad (j = 1, \ldots, s);$$
$$p(S = k \mid S \geq k, \boldsymbol{v}, \boldsymbol{f}) = F_N(f_k(\boldsymbol{v})),$$

where

$$a(\boldsymbol{v}, j, s) \sim \mathcal{GP}(\mu, C),$$

F_N denotes the standard normal cdf (probit link), and $f_k(\boldsymbol{v})$ is the Bayesian additive regression trees (BART) model (Chipman et al., 2010). BART is also a popular Bayesian nonparametric model for regression. See Chapter 4 for further details.

1.6 Contributions

This dissertation makes the following contributions in methodology and applications.

In Chapter 2, we propose a Bayesian feature allocation model for tumor subclone reconstruction using mutation pairs. With respect to methodology, we develop a feature allocation model with categorical matrix-valued features. We also develop a trans-dimensional MCMC algorithm based on splitting the data into training and test data sets, which is specially tailored to the feature allocation model. In terms of application, we model subclones characterized by phased pairs of (diploid) variant alleles. Our approach is a substantial improvement over current methods which all work with marginal counts only. We make inference for tumor heterogeneity on the basis of the proposed model and show that the model with (few) phased mutation pairs provides more accurate inference than current models with (far more) marginal SNVs. We also develop an open source software package `PairClone` which is available at

`http://www.compgenome.org/pairclone`.

In Chapter 3, we propose a Bayesian treed feature allocation model for tumor subclone reconstruction using mutation pairs. Regarding methodology, we develop a feature allocation model with *a priori* dependent features, where the dependence is modeled with a tree prior. We develop a computationally efficient posterior simulation method on the tree. In terms of application, our model allows for inference for phylogenetic trees of tumor cell subpopulations. We also develop an open source software package `PairCloneTree` which is available at `http://www.compgenome.org/pairclonetree`. This project uses the same data as the first project. However, the tree-based prior on dependent features is entirely different from the IBP model.

In Chapter 4, we propose a nonparametric Bayesian approach to monotone missing data in longitudinal studies. With regard to methodology, we develop nonparametric Bayesian regression models and shrinkage priors for observed data responses and dropout across dropout times and patterns. Such models can effectively capture non-linear and non-additive relationships and allow for borrow of information across times and patterns. In particular, the model is built on a GP prior for a nonparametric regression of observed data responses (conditional on dropout pattern and auxiliary covariates), combined with a BART model for dropout as a function of auxiliary covariates. In terms of application, our model allows for utilizing information from auxiliary covariates that are not desired in the primary research question. The inclusion of such auxiliary covariates can ideally reduce the extent of sensitivity analysis

that is needed for drawing accurate inferences.

Chapter 2

A Bayesian Feature Allocation Model for Tumor Subclone Reconstruction Using Mutation Pairs

Tumor cell populations can be thought of as being composed of heterogeneous cell subpopulations, with each subpopulation being characterized by overlapping sets of single nucleotide variants (SNVs). Such subpopulations are known as subclones and are an important target for precision medicine. Reconstructing such subclones from next-generation sequencing (NGS) data is one of the major challenges in precision medicine. We present PairClone as a new tool to implement this reconstruction. The main idea of PairClone is to model short reads mapped to pairs of proximal SNVs. In contrast, most existing methods use only marginal reads for unpaired SNVs. Using Bayesian nonparametric models, we estimate posterior probabilities of the number, genotypes and population frequencies of subclones in one or more tumor sample. We use the categorical Indian buffet process (cIBP) as a prior probability model for subclones that are represented as vectors of categorical matrices that record the corresponding sets of mutation pairs. Performance of PairClone is assessed using simulated and real datasets. An open source software package can be obtained at `http://www.compgenome.org/pairclone`.

2.1 Introduction

We explain intra-tumor heterogeneity by representing tumor cell populations as a mixture of subclones. We reconstruct unobserved subclones by utilizing information from pairs of proximal mutations that are obtained from next-generation sequencing (NGS) data. We exploit the fact that some short reads in NGS data cover pairs of phased mutations that reside on two sufficiently proximal loci. Therefore haplotypes of the mutation pairs can be observed and used for subclonal inference.

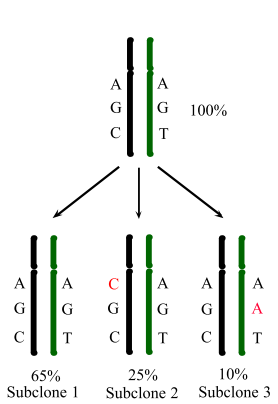
We develop a suitable sampling model that represents the paired nature of the data, and construct a nonparametric Bayesian feature allocation model as a prior for the hypothetical subclones. Both models together allow us to develop a fully probabilistic description of the composition of the tumor as a mixture of homogeneous underlying subclones, including the genotypes and number of such subclones.

2.1.1 Background

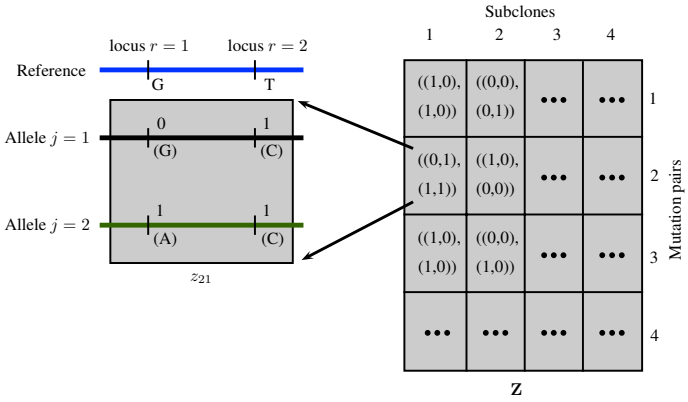
NGS technology (Mardis, 2008) has enabled researchers to develop bioinformatics tools that are being used to understand the landscape of tumors within and across different samples. An important related task is to reconstruct cellular subpopulations in one or more tumor samples, known as subclones. Mixtures of such subclones with varying population frequencies across spatial locations in the same tumor, across tumors from different time points, or across tumors from the primary and metastatic sites can provide

information about the mechanisms of tumor evolution and metastasis. Heterogeneity of cell populations is seen, for example, in varying frequencies of distinct somatic mutations. The hypothetical tumor subclones are homogeneous. That is, a subclone is characterized by unique genomic variants in its genome (Marjanovic et al., 2013, Almendro et al., 2013, Polyak, 2011, Stingl and Caldas, 2007, Shackleton et al., 2009, Dexter et al., 1978). Such subclones arise as the result of cellular evolution, which can be described by a phylogenetic tree that records how a sequence of somatic mutations gives rise to different cell subpopulations. Figure 2.1(a) provides a stylized and simple illustration in which a homogeneous sample with one original normal clone evolves into a heterogeneous sample with three subclones. Subclone 1 is the original parent cell population, and subclones 2 and 3 are descendant subclones of subclone 1, each possessing somatic mutations marked by the red letters. Each subclone possesses two homologous chromosomes (in black and green), and each chromosome in Figure 2.1(a) is marked by a triplet of letters representing the nucleotide on the three genomic loci. Together, the three subclones include four different haplotypes, (A, G, C), (A, G, T), (C, G, C), and (A, A, T), at these three genomic loci. In addition, each subclone has a different population frequency shown as the percentage values in Figure 2.1(a).

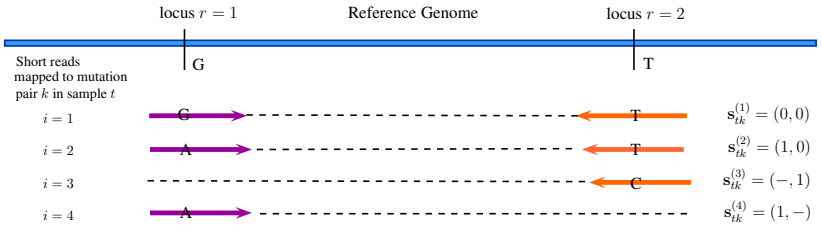
We use NGS data to infer such tumor heterogeneity. In an NGS experiment, DNA fragments are first produced by extracting the DNA molecules from the cells in a tumor sample. The fragments are then sequenced using short reads. For the three subclones in Figure 2.1(a), there are the four afore-



(a) Tumor evolution



(b) Subclone structure matrix $\boldsymbol{Z}$



(c) Paired-end reads data

Figure 2.1: (a) Illustration of tumor evolution, emergence of subclones and their population frequencies. (b) Illustration of the subclone structure matrix $\boldsymbol{Z}$. Right panel: A subclone is represented by one column of $\boldsymbol{Z}$. Each element of a column represents the subclonal genotypes for a mutation pair. For example, the genotypes for mutation pair 2 in subclone 1 is ((0, 1), (1, 1)), which is shown in detail on the left panel. Left panel: The reference genome for mutation pair 2 is (G, T) and the corresponding genotype of subclone 1 is ((G, C), (A, C)), which gives rise to $\boldsymbol{z}_{21} = ((0, 1), (1, 1))$. (c) Illustration of paired-end reads data for a mutation pair. Shown are four short reads mapped to mutation pair k in sample t. Some reads are mapped to both loci of the mutation pair, and others are mapped to only one of the two loci. The two ends of the same read are marked with opposing arrows in purple and orange.

mentioned haplotypes at the three loci. Consequently, short reads that cover some of these three loci may manifest different alleles. For example, if a large number of reads cover the first two loci, we might observe the four alleles (A, G), (C, G), (A, T) and (C, T) for the mutation pair. Observing four alleles is direct evidence for the presence of subclones (Sengupta et al., 2016). This is because in the absence of copy number variations there can be only two haploid genomes at any locus for a homogeneous human sample. Therefore, one can use mutation pairs in copy neutral regions to develop statistical inference for the presence and frequency of subclones. This is the goal of the paper.

Almost all mutation-based subclone-calling methods in the literature use only single nucleotide variants (SNVs) (Oesper et al., 2013, Strino et al., 2013, Jiao et al., 2014, Miller et al., 2014, Roth et al., 2014, Zare et al., 2014, Deshwar et al., 2015, Sengupta et al., 2015, Lee et al., 2015, 2016). Instead of examining mutation pairs, SNV-based methods use marginal counts for each recorded locus only. Consider, for example, the first locus in Figure 2.1(a). At this locus, the reference genome has an "A" nucleotide while subclones 2 and 3 have a "C" nucleotide. In the entire sample, the "C" nucleotide is roughly present in 17.5% of the DNA molecules based on the population frequencies illustrated in Figure 2.1(a). The percentage of a mutated allele is called variant allele fraction (VAF). If a sample is homogeneous and assuming no copy number variations at the locus, the population frequency for the "C" nucleotide should be close to 0, 50%, or 100%, depending on the heterozygosity of the locus. Therefore, if the population frequency of "C" deviates from

0%, 50%, or 100%, the sample is likely to be heterogeneous. Based on this argument, SNV-based subclone callers search for SNVs with VAFs that are different from these frequencies (0, 50%, 100%), which are evidence for the presence of different (homogeneous) subpopulations. In the event of copy number variations, a similar but slightly more sophisticated reasoning can be applied, see for example, Lee et al. (2016).

2.1.2 Using mutation pairs

NGS data usually contain substantially fewer mutation pairs than marginal SNVs. However mutation pairs carry important phasing information that improves the accuracy of subclone reconstruction. We find that the phased data for (far fewer) pairs provide more information than the (far more) marginal VAFs. For example, imagine a tumor sample that is a mixture of subclones 2 and 3 in Figure 2.1(a). Suppose a sufficient amount of short reads cover the first two loci, we should observe relatively large reads counts for four alleles (C, G), (A, T), (C, T) and (A, G). One can then reliably infer that there are heterogeneous cell subpopulations in the tumor sample. In contrast, if we ignore the phasing information and only consider the (marginal) VAFs for each SNV, then the observed VAFs for both SNVs are 50%, which could be heterogeneous mutations from a single cell population. See Simulation 1 for an illustration. In summary, we leverage the power of using mutation pairs over marginal SNVs by incorporating partial phasing information in our model. Besides the simulation study we will later also empirically confirm these

considerations in actual data analysis.

The relative advantage of using mutation pairs over marginal SNV's can also be understood as a special case of a more general theme. In biomedical data it is often important to avoid overinterpretation of noisy data and to distill a relatively weak signal. A typical example is the probability of expression (POE) model of Parmigiani et al. (2002). Similarly, the modeling of mutation pairs is a way to extract the pertinent information from the massive noisy data. Due to noise and artifacts in NGS data, such as base-calling or mapping error, many called SNVs might record unusual population frequencies, for reasons unrelated to the presence of subclones (Li, 2014). Direct modeling of all marginal read counts one ends up with noise swamping the desired signal (Nik-Zainal et al., 2012, Jiao et al., 2014). See our analysis of a real data set in Section 2.6 for an example. To mitigate this challenge, most methods use clustering of the VAFs, including, for example, Roth et al. (2014). One would then use the resulting cluster centers to infer subclones, which is one way of extracting more concise information. In addition, the vast majority of the methods in the literature show that even though a tumor sample could possess thousands to millions of SNVs, the number of inferred subclones usually is in the low single digit, no more than 10. To this end, we propose instead an alternative approach to extract useful information by modeling (fewer) mutation pairs, as mutation pairs contain more information and are of higher quality. We show in our numerical examples later that with a few dozens of these mutation pairs, the inference on the subclones is strikingly similar to

cluster-based subclone callers using much more SNVs.

Finally, using mutation pairs does not exclude the possibility of making use of marginal SNVs. In Section 2.5.1, we show it is straightforward to jointly model mutation pairs and SNVs. Other biological complexities, such as tumor purity and copy number variations, can also be incorporated in our model. See Sections 2.5.2 and 2.5.3 for more details. Incorporating CNVs greatly increases the complexity in modeling and is not addressed by most existing methods. Thus, we mainly concern ourselves with mutation pairs in copy number neutral regions and leave incorporation of CNVs to future work.

2.1.3 Representation of subclones

We construct a $K \times C$ categorical valued matrix $\boldsymbol{Z}$ (Figure 2.1(b)) to represent the subclone structure. Rows of $\boldsymbol{Z}$ are indexed by k and represent mutation pairs, and columns of $\boldsymbol{Z}$, denoted by $\boldsymbol{z}_c = (\boldsymbol{z}_{1c}, \cdots, \boldsymbol{z}_{Kc})$, record the phased mutation pairs on the two homologous chromosomes of subclone c, $c = 1, \ldots, C$. As in Figure 2.1(b), let $j = 1, 2$ index the two homologous chromosomes, $r = 1, 2$ index the two mutation loci, $\boldsymbol{z}_{kc} = (\boldsymbol{z}_{kcj}, j = 1, 2)$ be the genotype consisting of two alleles for mutation pair k in subclone c, and $\boldsymbol{z}_{kcj} = (z_{kcjr}, r = 1, 2)$ denote the allele of the j-th homologous chromosome. Therefore, each entry $\boldsymbol{z}_{kc}$ of the matrix $\boldsymbol{Z}$ is a 2×2 binary submatrix itself. For example, in Figure 2.1(b) the entry z_{21} is a pair of 2-dimension binary row vectors, $(0, 1)$ and $(1, 1)$, representing the genotypes for both alleles at mutation pair $k = 2$ of subclone $c = 1$; each vector indicates the allele for the

mutation pair on a homologous chromosome. The first vector $(0, 1)$ indicates that locus $r = 1$ harbors no mutation (0) and locus $r = 2$ harbors a mutation (1). Similarly, the second vector $(1, 1)$ marks two mutations on both loci.

In summary, each entry of $\boldsymbol{Z}$,

$$\boldsymbol{z}_{kc} = (\boldsymbol{z}_{kc1}, \boldsymbol{z}_{kc2}) = ((z_{kc11}, z_{kc12}), (z_{kc21}, z_{kc22}))$$

is a 2×2 matrix (with the two row vectors horizontally displayed for convenience). Each z_{kcjr} is a binary indicator and $z_{kcjr} = 1$ (or 0) indicates a mutation (or reference). Thus, $\boldsymbol{z}_{kc}$ can take $Q = 16$ possible values. That is, $\boldsymbol{z}_{kc} \in \{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(16)}\} = \{(00, 00), (00, 01), \ldots, (11, 11)\}$, where we write 00 short for $(0, 0)$ etc., and $z^{(1)} = (00, 00)$ refers to the genotype on the reference genome. Formally, $\boldsymbol{z}_{kc}$ is a 2×2 binary matrix, and $\boldsymbol{Z}$ is a matrix of such binary matrices. Moreover, we can collapse some $\boldsymbol{z}^{(q)}$ values as we do not have phasing across mutation pairs. For example, $\boldsymbol{z}_{kc} = (01, 10)$ and $\boldsymbol{z}_{kc} = (10, 01)$, etc. have mirrored rows and are indistinguishable in defining a subclone (a column of $\boldsymbol{Z}$). (More details in Section 2.2.2). Typically distinct mutation pairs are distant from each other, and in NGS data they are almost never phased. Therefore, we can reduce the number of possible outcomes of z_{kc} to $Q = 10$, due to the mirrored outcomes. We list them below for later reference: $z^{(1)} = (00, 00), z^{(2)} = (00, 01), z^{(3)} = (00, 10), z^{(4)} = (00, 11), z^{(5)} = (01, 01), z^{(6)} = (01, 10), z^{(7)} = (01, 11), z^{(8)} = (10, 10), z^{(9)} = (10, 11)$ and $z^{(10)} = (11, 11)$. In summary, the entire matrix $\boldsymbol{Z}$ fully specifies the genomes of each subclone at all the mutation pairs.

Suppose T tumor samples are available from the same patient, obtained either at different time points (such as initial diagnosis and relapses), at the same time but from different spatial locations within the same tumor, or from tumors at different metastatic sites. We assume those T samples share the same subclones, while the subclonal population frequencies may vary across samples. For clinical decisions it can be important to know the population frequencies of the subclones. To facilitate such inference, we introduce a $T \times (C+1)$ matrix $\boldsymbol{w}$ to represent the population frequencies of subclones. The element w_{tc} refers to the proportion of subclone c in sample t, where $0 < w_{tc} < 1$ for all t and c, and $\sum_{c=0}^{C} w_{tc} = 1$. A background subclone, which has no biological meaning and is indexed by $c = 0$, is included to account for artifacts and experimental noise. We will discuss more about this later.

The remainder of this article is organized as follows. In Sections 2.2 and 2.3, we propose a Bayesian feature allocation model and the corresponding posterior inference scheme to estimate the latent subclone structure. In Section 2.4, we evaluate the model with three simulation studies. Section 2.5 extends the models to accommodate other biological complexities and present additional simulation results. Section 2.6 reports the analysis results for a lung cancer patient with multiple tumor biopsies. We conclude with a final discussion in Section 2.7.

2.2 The PairClone Model

2.2.1 Sampling Model

Suppose paired-end short reads data are obtained by deep DNA sequencing of multiple tumor samples. In such data, a short read is obtained by sequencing two ends of the same DNA fragment. Usually a DNA fragment is much longer than a short read, and the two ends do not overlap and must be mapped separately. However, since the paired-end reads are from the same DNA fragment, they are naturally phased and can be used for inference of alleles and subclones. We use `LocHap` (Sengupta et al., 2016) to find pairs of mutations that are no more than a fixed number, say 500, base pairs apart. Such mutation pairs can be mapped by paired-end reads, making them eligible for PairClone analysis. See Figure 2.1(c) for an example. For each mutation pair, a number of short reads are mapped to at least one of the two loci. Denote the two sequences on short read i mapped to mutation pair k in tissue sample t by $\boldsymbol{s}_{tk}^{(i)} = \left(s_{tkr}^{(i)}, r = 1, 2\right) = \left(s_{tk1}^{(i)}, s_{tk2}^{(i)}\right)$, where $r = 1, 2$ index the two loci, $s_{tkr}^{(i)} = 0$ or 1 indicates that the short read sequence is a reference or mutation. Theoretically, each $s_{tkr}^{(i)}$ can take four values, A, C, G, T, the four nucleotide sequences. However, at a single locus, the probability of observing more than two sequences across short reads is negligible since it would require the same locus to be mutated twice throughout the life span of the person or tumor, which is unlikely. We therefore code $s_{tkr}^{(i)}$ as a binary value. Also, sometimes a short read may cover only one of the two loci in a pair, and we use $s_{tkr}^{(i)} = -$ to represent a missing base when there is no overlap between a

short read and the corresponding SNV. Therefore, $s_{tkr}^{(i)} \in \{0, 1, -\}$. For example, in Figure 2.1(c) locus $r = 1$, $s_{tk1}^{(1)} = 0$ for read $i = 1$, $s_{tk1}^{(2)} = 1$ for read $i = 2$, and $s_{tk1}^{(3)} = -$ for read $i = 3$. Reads that are not mapped to either locus are excluded from analysis since they do not provide any information for subclones. Altogether, $\boldsymbol{s}_{tk}^{(i)}$ can take $G = 8$ possible values, and its sample space is denoted by $\mathcal{H} = \{\boldsymbol{h}_1, \ldots, \boldsymbol{h}_G\} = \{00, 01, 10, 11, -0, -1, 0-, 1-\}$. Each value corresponds to an allele of two loci, with $-$ being a special "missing" coverage. For mutation pair k in sample t, the number of short reads bearing allele $\boldsymbol{h}_g$ is denoted by $n_{tkg} = \sum_i I\left(\boldsymbol{s}_{tk}^{(i)} = \boldsymbol{h}_g\right)$, where $I(\cdot)$ is the indicator function, and the total number of reads mapped to the mutation pair is then $N_{tk} = \sum_g n_{tkg}$. Finally, depending upon whether a read covers both loci or only one locus we distinguish three cases: (i) a read maps to both loci (complete), taking values $\boldsymbol{s}_{tk}^{(i)} \in \{\boldsymbol{h}_1, \ldots, \boldsymbol{h}_4\}$; (ii) a read maps to the second locus only (left missing), $\boldsymbol{s}_{tk}^{(i)} \in \{\boldsymbol{h}_5, \boldsymbol{h}_6\}$; and (iii) a read maps to the first locus only (right missing), $\boldsymbol{s}_{tk}^{(i)} \in \{\boldsymbol{h}_7, \boldsymbol{h}_8\}$. We assume a multinomial sampling model for the observed read counts

$$(n_{tk1}, \ldots, n_{tk8}) \mid N_{tk} \sim \mathrm{Mn}(N_{tk};\ p_{tk1}, \ldots, p_{tk8}). \tag{2.1}$$

Here $\boldsymbol{p} = \{p_{tkg}, g = 1, \ldots, 8\}$ are the probabilities for the 8 possible values of $\boldsymbol{s}_{tk}^{(i)}$. For the upcoming discussion, we separate out the probabilities for the three missingness cases. Let $v_{tk1}, v_{tk2}, v_{tk3}$ denote the probabilities of observing a short read satisfying cases (i), (ii) and (iii), respectively. We write $p_{tkg} = v_{tk1}\,\tilde{p}_{tkg}, g = 1, \ldots, 4$, $p_{tkg} = v_{tk2}\,\tilde{p}_{tkg}, g = 5, 6$, and $p_{tkg} = v_{tk3}\,\tilde{p}_{tkg}, g = 7, 8$. Here $\tilde{p}_{tkg}$ are the probabilities conditional on case (i),

(ii) or (iii). That is, $\sum_{g=1}^{4} \tilde{p}_{tkg} = \sum_{g=5,6} \tilde{p}_{tkg} = \sum_{g=7,8} \tilde{p}_{tkg} = 1$. We still use a single running index, $g = 1, \ldots, 8$, to match the notation in p_{tkg}. Below we link the multinomial sampling model with the underlying subclone structure by expressing $\tilde{p}_{tkg}$ in terms of $\boldsymbol{Z}$ and $\boldsymbol{w}$. Regarding $v_{tk1}, v_{tk2}, v_{tk3}$ we assume non-informative missingness and therefore do not proceed with inference on them (and v's remain constant factors in the likelihood).

2.2.2 Prior Model

Construction of $\tilde{p}_{tkg}$. The construction of a prior model for $\tilde{p}_{tkg}$ is based on the following generative model. To generate a short read, we first select a subclone c from which the read arises, using the population frequencies w_{tc} for sample t. Next we select with probability 0.5 one of the two DNA strands, $j = 1, 2$. Finally, we record the read $\boldsymbol{h}_g$, $g = 1, 2, 3$ or 4, corresponding to the chosen allele $\boldsymbol{z}_{kcj} = (z_{kcj1}, z_{kcj2})$. In the case of left (or right) missing locus we observe $\boldsymbol{h}_g$, $g = 5$ or 6 (or $g = 7$ or 8), corresponding to the observed locus of the chosen allele. Reflecting these three generative steps, we denote the probability of observing a short read $\boldsymbol{h}_g$ that bears sequence $\boldsymbol{z}_{kcj}$ by

$$A(\boldsymbol{h}_g, \boldsymbol{z}_{kc}) = \sum_{j=1}^{2} 0.5 \times I(h_{g1} = z_{kcj1})\, I(h_{g2} = z_{kcj2}), \tag{2.2}$$

with the understanding that $I(- = z_{kcjr}) \equiv 1$ for missing reads. Implicit in (2.2) is the restriction $A(\boldsymbol{h}_g, \boldsymbol{z}_{kc}) \in \{0, 0.5, 1\}$, depending on the arguments.

Finally, using the definition of $A(\cdot)$ we model the probability of observ-

ing a short read $\boldsymbol{h}_g$ as

$$\tilde{p}_{tkg} = \sum_{c=1}^{C} w_{tc}\, A(\boldsymbol{h}_g, \boldsymbol{z}_{kc}) + w_{t0}\, \rho_g. \tag{2.3}$$

In (2.3) we include $w_{t0}\rho_g$ to model a background subclone denoted by $c = 0$ with population frequency w_{t0}. The background subclone does not exist and has no biological interpretation. It is only used as a mathematical device to account for noise and artifacts in the NGS data (sequencing errors, mapping errors, etc.). The weights ρ_g are the conditional probabilities of observing a short read $\boldsymbol{s}_{tk}^{(i)}$ harboring allele $\boldsymbol{h}_g$ if the recorded read were due to experimental noise. Note that $\rho_1 + \ldots + \rho_4 = \rho_5 + \rho_6 = \rho_7 + \rho_8 = 1$.

Prior for C. We assume a geometric prior distribution on C, $C \sim \text{Geom}(r)$, to describe the random number of subclones (columns of $\boldsymbol{Z}$), $p(C) = (1 - r)^C r, C \in \{1, 2, 3, \ldots\}$. *A priori* $E(C) = 1/r$.

Prior for $\boldsymbol{Z}$. We use the finite version of the categorical Indian buffet process (cIBP) (Sengupta, 2013) as the prior for the latent categorical matrix $\boldsymbol{Z}$. The cIBP is a categorical extension of the Indian buffet process (Griffiths and Ghahramani, 2011) and defines feature allocation (Broderick et al., 2013) for categorical matrices. In our application, the mutation pairs are the objects, and the subclones are the latent features chosen by the objects. The number of subclones C is random, with the geometric prior $p(C)$. Conditional on C, we now introduce for each column of $\boldsymbol{Z}$ a vector $\boldsymbol{\pi}_c = (\pi_{c1}, \pi_{c2}, \ldots, \pi_{cQ})$,

where $p(\boldsymbol{z}_{kc} = \boldsymbol{z}^{(q)}) = \pi_{cq}$, and $\sum_{q=1}^{Q} \pi_{cq} = 1$. Recall that $\boldsymbol{z}^{(q)}$ are the possible genotypes for the mutation pairs defined in Section 2.1.3, $q = 1, \ldots, Q$, for $Q = 10$ possible genotypes.

As prior model for $\boldsymbol{\pi}_c$, we use a Beta-Dirichlet distribution (Kim et al., 2012). Let $\tilde{\pi}_{cq} = \pi_{cq}/(1 - \pi_{c1})$, $q = 2, \ldots, Q$. Conditional on C, $\pi_{c1} \sim \text{Be}(1, \alpha/C)$ follows a beta distribution, and $(\tilde{\pi}_{c2}, \ldots, \tilde{\pi}_{cQ}) \sim \text{Dir}(\gamma_2, \ldots, \gamma_Q)$ follows a Dirichlet distribution. Here $\boldsymbol{z}_{kc} = \boldsymbol{z}^{(1)}$ corresponds to the situation that subclone c is not chosen by mutation pair k, because $\boldsymbol{z}^{(1)}$ refers to the reference genome. We write

$$\boldsymbol{\pi}_c \mid C \sim \text{Beta-Dirichlet}(\alpha/C, 1, \gamma_2, \ldots, \gamma_Q).$$

This construction includes a positive probability for all-zero columns $\boldsymbol{z}_c = \mathbf{0}$. In our application, $\boldsymbol{z}_c = \mathbf{0}$ refers to normal cells with no somatic mutations, which could be included in the cell subpopulations.

In the definition of the cIBP prior, we would have one more step of dropping all zero columns. This leaves a categorical matrix $\boldsymbol{Z}$ with at most C columns. As shown in Sengupta (2013), the marginal limiting distribution of $\boldsymbol{Z}$ follows the cIBP as $C \to \infty$.

Prior for $\boldsymbol{w}$. We assume $\boldsymbol{w}_t$ follows a Dirichlet prior,

$$\boldsymbol{w}_t \mid C \overset{iid}{\sim} \text{Dirichlet}(d_0, d, \cdots, d),$$

for $t = 1, \cdots, T$. We set $d_0 < d$ to reflect the nature of $c = 0$ as a background noise and model mis-specification term.

Prior for $\boldsymbol{\rho}$. We complete the model with a prior for $\boldsymbol{\rho} = \{\rho_g\}$. Recall ρ_g is the conditional probability of observing a short read with allele $\boldsymbol{h}_g$ due to experimental noise. We consider complete read, left missing read and right missing read separately, and assume

$$\rho_{g_1} \sim \text{Dirichlet}(d_1, \ldots, d_1); \quad \rho_{g_2} \sim \text{Dirichlet}(2d_1, 2d_1); \quad \rho_{g_3} \sim \text{Dirichlet}(2d_1, 2d_1),$$

where $g_1 = \{1, 2, 3, 4\}$, $g_2 = \{5, 6\}$ and $g_3 = \{7, 8\}$.

2.3 Posterior Inference

Let $\boldsymbol{x} = (\boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{w}, \boldsymbol{\rho})$ denote the unknown parameters except C, where $\boldsymbol{Z} = \{z_{kc}\}$, $\boldsymbol{\pi} = \{\pi_{cq}\}$, $\boldsymbol{w} = \{w_{tc}\}$, and $\boldsymbol{\rho} = \{\rho_g\}$. We use Markov chain Monte Carlo (MCMC) simulations to generate samples from the posterior $\boldsymbol{x}^{(l)} \overset{iid}{\sim} p(\boldsymbol{x} \mid \boldsymbol{n}, C)$, $l = 1, \ldots, L$. With fixed C such MCMC simulation is straightforward. See, for example, Brooks et al. (2011) for a review of MCMC. Gibbs sampling transition probabilities are used to update $\boldsymbol{Z}$ and $\boldsymbol{\pi}$, and Metropolis-Hastings transition probabilities are used to update $\boldsymbol{w}$ and $\boldsymbol{\rho}$. Since $p(\boldsymbol{x} \mid \boldsymbol{n}, C)$ is expected to be highly multi-modal, we use additional parallel tempering to improve mixing of the Markov chain. Details of MCMC simulation and parallel tempering are described in Appendix A.1.

Updating C. Updating the value of C is more difficult as it involves trans-dimensional MCMC (Green, 1995). At each iteration, we propose a new value $\tilde{C}$ by generating from a proposal distribution $q(\tilde{C} \mid C)$. In the later examples

we assume that C is *a priori* restricted to $C_{\min} \leq C \leq C_{\max}$, and use a uniform proposal $q(\tilde{C} \mid C) \sim \text{Unif}\{C_{\min}, \ldots, C_{\max}\}$.

Next, we split the data into a training set $\boldsymbol{n}'$ and a test set $\boldsymbol{n}''$ with $n'_{tkg} = bn_{tkg}$ and $n''_{tkg} = (1-b)n_{tkg}$, respectively, for $b \in (0,1)$. Denote by $p_b(\boldsymbol{x} \mid C) = p(\boldsymbol{x} \mid \boldsymbol{n}', C)$ the posterior of $\boldsymbol{x}$ conditional on C evaluated on the training set only. We use p_b in two instances. First, we replace the original prior $p(\boldsymbol{x} \mid C)$ by $p_b(\boldsymbol{x} \mid C)$, and second, we use p_b as a proposal distribution for $\tilde{\boldsymbol{x}}$, as $q(\tilde{\boldsymbol{x}} \mid \tilde{C}) = p_b(\tilde{\boldsymbol{x}} \mid \tilde{C})$. Finally, we evaluate the acceptance probability of $(\tilde{C}, \tilde{\boldsymbol{x}})$ on the test data by

$$p_{\text{acc}}(C, \boldsymbol{x}, \tilde{C}, \tilde{\boldsymbol{x}}) = 1 \wedge \frac{p(\boldsymbol{n}'' \mid \tilde{\boldsymbol{x}}, \tilde{C})}{p(\boldsymbol{n}'' \mid \boldsymbol{x}, C)} \cdot \frac{p(\tilde{C}) \cancel{p_b(\tilde{\boldsymbol{x}} \mid \tilde{C})}}{p(C) \cancel{p_b(\boldsymbol{x} \mid C)}} \cdot \frac{q(C \mid \tilde{C}) \cancel{q(\boldsymbol{x} \mid C)}}{q(\tilde{C} \mid C) \cancel{q(\tilde{\boldsymbol{x}} \mid \tilde{C})}}. \quad (2.4)$$

The use of the prior $p_b(\tilde{\boldsymbol{x}} \mid \tilde{C})$ is similar to the construction of the fractional Bayes factor (FBF) (O'Hagan, 1995) which uses a fraction of the data to define an informative prior that allows the evaluation of Bayes factors. In contrast, here p_b is used as an informative proposal distribution for $\tilde{\boldsymbol{x}}$. Without the use of a training sample it would be difficult to generate proposals $\tilde{\boldsymbol{x}}$ with reasonable acceptance rate. In other words, we use p_b to achieve a better mixing Markov chain Monte Carlo simulation. The use of the same p_b to replace the original prior avoids the otherwise prohibitive evaluation of p_b in the acceptance probability (2.4). See more details in Appendix A.2 and A.3.

Point estimates for parameters. We use the posterior mode $\hat{C}$ as a point estimate of C. Conditional on $\hat{C}$, we follow Lee et al. (2015) to find a point

estimate of $\boldsymbol{Z}$. For any two $K \times \hat{C}$ matrices $\boldsymbol{Z}$ and $\boldsymbol{Z}'$, a distance between the c-th column of $\boldsymbol{Z}$ and the c'-th column of $\boldsymbol{Z}'$ is defined by $\mathcal{D}_{cc'}(\boldsymbol{Z}, \boldsymbol{Z}') = \sum_{k=1}^{K} \|\boldsymbol{z}_{kc} - \boldsymbol{z}'_{kc'}\|_1$, where $1 \leq c, c' \leq \hat{C}$, and we take the vectorized form of $\boldsymbol{z}_{kc}$ and $\boldsymbol{z}'_{kc'}$ to compute L^1 distance between them. Then, we define the distance between $\boldsymbol{Z}$ and $\boldsymbol{Z}'$ as $d(\boldsymbol{Z}, \boldsymbol{Z}') = \min_{\boldsymbol{\sigma}} \sum_{c=1}^{\hat{C}} \mathcal{D}_{c,\boldsymbol{\sigma}_c}(\boldsymbol{Z}, \boldsymbol{Z}')$, where $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_{\hat{C}})$ is a permutation of $\{1, \ldots, \hat{C}\}$, and the minimum is taken over all possible permutations. This addresses the potential label-switching issue across the columns of $\boldsymbol{Z}$. Let $\{\boldsymbol{Z}^{(l)}, l = 1, \ldots, L\}$ be a set of posterior Monte Carlo samples of $\boldsymbol{Z}$. A posterior point estimate for $\boldsymbol{Z}$, denoted by $\hat{\boldsymbol{Z}}$, is reported as $\hat{\boldsymbol{Z}} = \boldsymbol{Z}^{(\hat{l})}$, where

$$\hat{l} = \underset{l \in \{1,\ldots,L\}}{\arg\min} \sum_{l'=1}^{L} d(\boldsymbol{Z}^{(l)}, \boldsymbol{Z}^{(l')}).$$

Based on $\hat{l}$, we report posterior point estimates of $\boldsymbol{w}$ and $\boldsymbol{\rho}$, given by $\hat{\boldsymbol{w}} = \boldsymbol{w}^{(\hat{l})}$ and $\hat{\boldsymbol{\rho}} = \boldsymbol{\rho}^{(\hat{l})}$, respectively.

2.4 Simulation Studies

We evaluate the proposed model with three simulation studies. In the first simulation we use single sample data ($T = 1$), since in most current applications only a single sample is available for analysis. Inferring subclonal structure accurately under only one sample is a major challenge, and not completely resolved in the current literature. The single sample does not rule out meaningful inference, as the relevant sample size is the number of SNVs or mutation pairs, or the (even larger) number of reads. In the second and third

simulations we consider multi-sample data, similar to the lung cancer data that we analyze later. In all simulations, we assume the missing probabilities v_{tk2} and v_{tk3} to be 30% or 35%. Recall that these probabilities represent the probabilities that a short read will only cover one of the two loci in the mutation pair.

2.4.1 Simulation 1

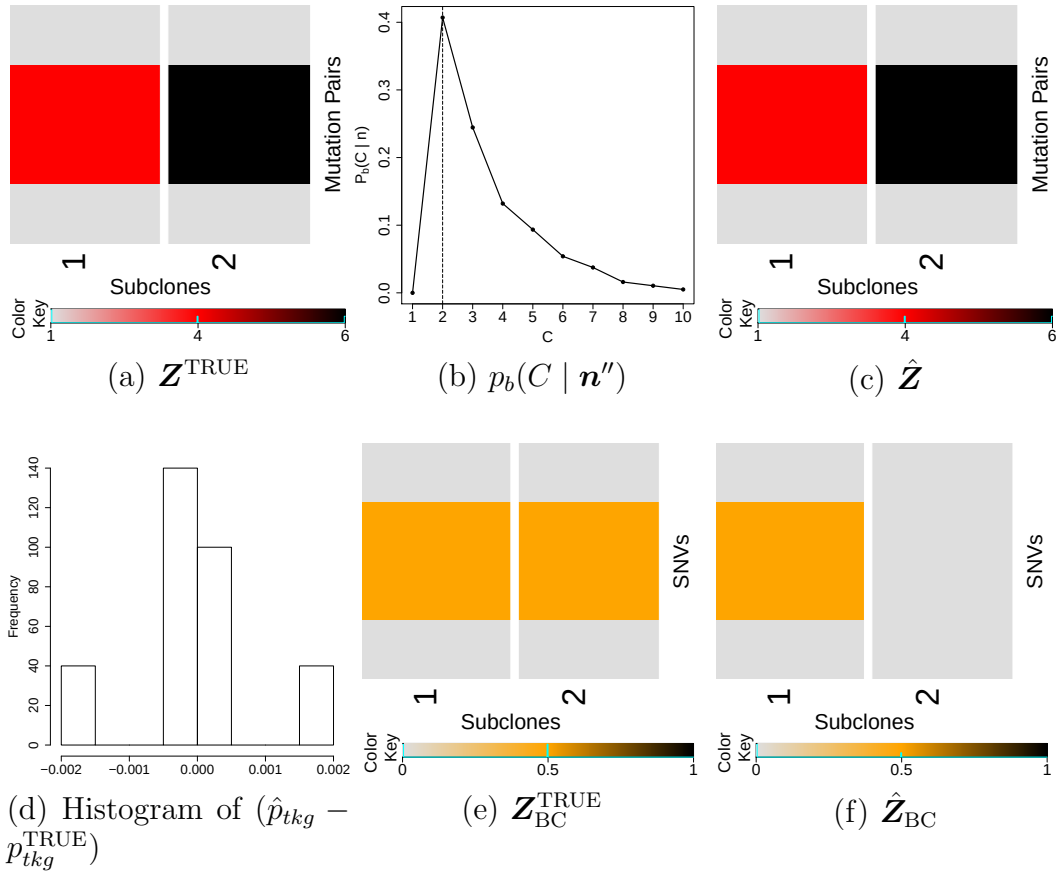
Setup. In the first simulation, we illustrate the advantage of using mutation pair data over marginal SNV counts. We generate hypothetical short reads data for $T = 1$ sample and $K = 40$ mutation pairs. Based on our own experiences, for a whole-exome sequencing data set, we usually obtain dozens of mutation pairs with decent coverage. See Sengupta et al. (2016) for a discussion. We assume there are $C^{\text{TRUE}} = 2$ latent subclones, and set their population frequencies as $\boldsymbol{w}^{\text{TRUE}} = (1.0 \times 10^{-7}, 0.8, 0.2)$, where 1.0×10^{-7} refers to the proportion of the hypothetical background subclone $c = 0$. The subclone matrix $\boldsymbol{Z}^{\text{TRUE}}$ is shown in Figure 2.2(a) (as a heat map). Light grey, red and black colors are used to represent genotypes $\boldsymbol{z}^{(1)}$, $\boldsymbol{z}^{(4)}$ and $\boldsymbol{z}^{(6)}$. For example, subclone 1 has genotype $\boldsymbol{z}^{(1)}$ (wild type) for mutation pairs 1–10 and 31 – 40, and $\boldsymbol{z}^{(4)}$ for mutation pairs 11–30. We generate $\boldsymbol{\rho}^{\text{TRUE}}$ from its prior with hyperparameter $d_1 = 1$. Next we set the probabilities of observing left and right missing reads as $v_{tk2} = v_{tk3} = 0.3$ for all k and t, to mimic a typical missing rate observed in the real data. We calculate multinomial probabilities $\{p_{tkg}^{\text{TRUE}}\}$ shown in equations (2.3) and (2.2) from the simulated $\boldsymbol{Z}^{\text{TRUE}}$, $\boldsymbol{w}^{\text{TRUE}}$

and $\boldsymbol{\rho}^{\text{TRUE}}$. Total read counts N_{tk} are generated as random numbers ranging from 400 to 600, and finally we generate read counts n_{tkg} from the multinomial distribution given N_{tk} as shown in equation (2.1).

We fit the model with hyperparameters fixed as follows: $\alpha = 4$, $\gamma_2 = \cdots = \gamma_Q = 2$, $d = 0.5$, $d_0 = 0.03$, $d_1 = 1$, and $r = 0.4$. We set $C_{\min} = 1$ and $C_{\max} = 10$ as the range of C. The fraction b needs to be calibrated. We choose b such that the test sample size $(1-b)\sum_{t=1}^{T}\sum_{k=1}^{K} N_{tk}$ is approximately equal to $160/T$. See Section A.3 for a discussion of this choice.

We run MCMC simulation for $30,000$ iterations, discarding the first $10,000$ iterations as initial burn-in, and keep one sample every 10 iterations. The initial values are randomly generated from the priors.

Results. Figure 2.2(b) shows $p_b(C \mid \boldsymbol{n}'')$, where the vertical dashed line marks the simulation truth. The posterior mode $\hat{C} = 2$ recovers the truth. Figure 2.2(c) shows the point estimate of $\boldsymbol{Z}^{\text{TRUE}}$, given by $\hat{\boldsymbol{Z}}$. The true subclone structure is perfectly recovered. The estimated subclone weights are $\hat{\boldsymbol{w}} = (2.27 \times 10^{-116}, 0.8099, 0.1901)$, which is also very close to the truth. We use $\hat{\boldsymbol{Z}}$ and $\hat{\boldsymbol{w}}$ to calculate estimated multinomial probabilities, denoted by $\{\hat{p}_{tkg}\}$. Figure 2.2(d) shows a histogram of the differences $(\hat{p}_{tkg} - p_{tkg}^{\text{TRUE}})$ as a residual plot to assess model fitting. The histogram is centered at zero with little variation, indicating a reasonably good model fit. In summary, this simulation shows that the proposed inference can almost perfectly recover the truth in a simple scenario with a single sample.



(a) $\boldsymbol{Z}^{\text{TRUE}}$ (b) $p_b(C \mid \boldsymbol{n}'')$ (c) $\hat{\boldsymbol{Z}}$

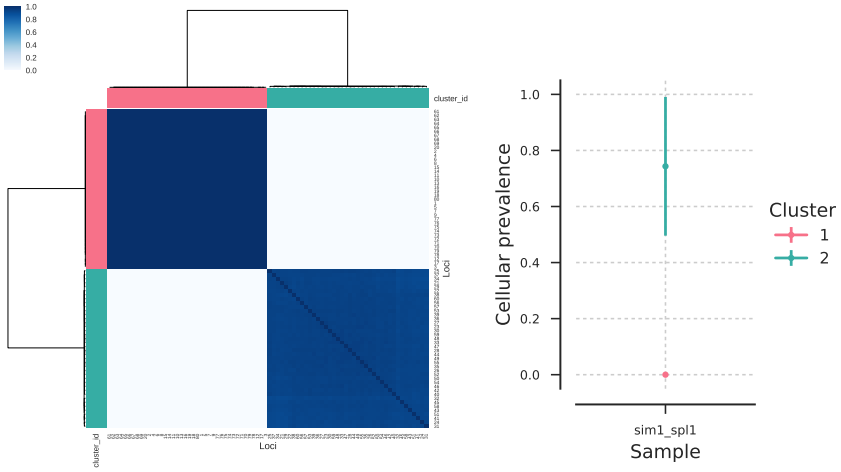
(d) Histogram of $(\hat{p}_{tkg} - p_{tkg}^{\text{TRUE}})$ (e) $\boldsymbol{Z}_{\text{BC}}^{\text{TRUE}}$ (f) $\hat{\boldsymbol{Z}}_{\text{BC}}$

Figure 2.2: Simulation 1. Simulation truth $\boldsymbol{Z}^{\text{TRUE}}$ (a, e), and posterior inference under PairClone (b, c, d) and under BayClone (f).

2.4.2 Comparison with BayClone and PyClone

We compare the proposed inference under PairClone versus inference under SNV-based subclone callers ,i.e., based on marginal (un-paired) counts of point mutations, including BayClone(Sengupta et al., 2015) and PyClone (Roth et al., 2014).

BayClone infers the subclone structure based on marginal allele fre-



(a) Posterior similarity matrix (b) Cellular prevalence

Figure 2.3: Simulation 1. Posterior inference under PyClone.

quencies of the recorded SNVs, and chooses the number of subclones based on log pseudo marginal likelihood (LPML) model comparison. Under the LPML criterion, the estimated number of subclones reported by BayClone is $\hat{C} = 2$, which also recovers the truth. Figure 2.2(e) displays the true genotypes of the unpaired SNVs, denoted by $\boldsymbol{Z}_{\mathrm{BC}}^{\mathrm{TRUE}}$, based on the true genotypes in Figure 2.2 for the mutation pairs. That is, we derive the corresponding marginal genotype for each SNV in the mutation pair based on the truth $\boldsymbol{Z}^{\mathrm{TRUE}}$. Figure 2.2(f) shows the heat map of estimated matrix $\hat{\boldsymbol{Z}}_{\mathrm{BC}}$, where $z_{sc} = 0$ (light grey), 0.5 (orange) and 1 (black) refer to homozygous wild-type, heterozygous variant and homozygous variant at SNV locus s, respectively. The estimated subclone proportions are $\hat{\boldsymbol{w}}_{\mathrm{BC}} = (0.008, 0.988, 0.004)$.

PyClone, on the other hand, clusters mutations based on allele fre-

quencies of the recorded SNVs using the implied clustering under a Dirichlet process mixture model. PyClone does not report subclonal genotypes and thus is not directly comparable with PairClone. Posterior inference is summarized in Figure 2.3. Panel (a) indicates that the 80 SNV loci form two clusters, with one cluster corresponding to loci 1–20 and 61–80, and the other cluster corresponding to loci 21–60, which agrees with the truth. Panel (b) shows the cellular prevalence of the two clusters across samples, where the middle point represents the posterior mean, and the error bar indicates posterior standard deviation. The cellular prevalence is defined as fraction of clonal population harbouring a mutation. In the PyClone MCMC samples, the estimated cellular prevalence of cluster 2 fluctuates between 0.5 and 1 and thus includes high posterior uncertainty, while the true cellular prevalence of cluster 2 is 1.

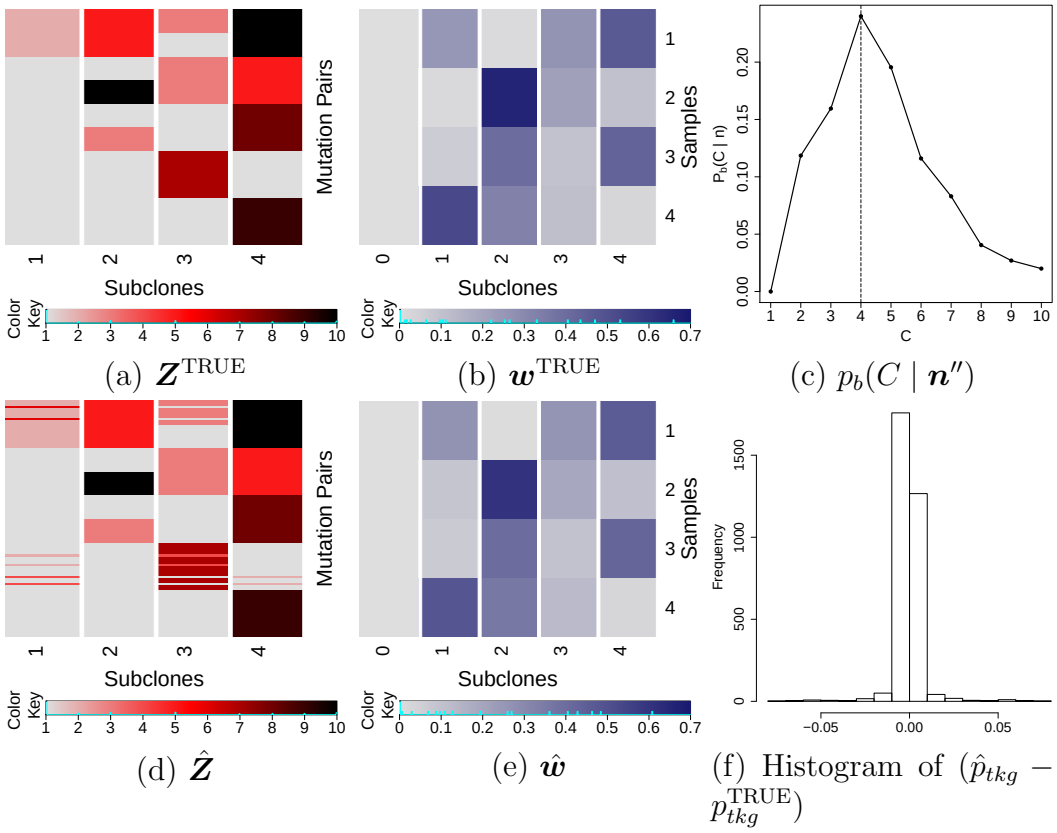
The estimates under SNV-based subclone callers do not fully recover the simulation truth. The main reason is probably that the phasing information of paired SNVs is lost in the marginal counts that are used in BayClone and PyClone, making the subclone estimation less accurate than under PairClone. For example, the two subclones with genotypes $\boldsymbol{z}^{(4)} = (00, 11)$ and $\boldsymbol{z}^{(6)} = (01, 10)$ lead to exactly the same allele frequency (50%) for both loci. BayClone can not distinguish between these two different subclones based on the 50% allele frequency for each locus. Although BayClone correctly reports the number of subclones, inference mistakenly includes a normal subclone with negligible weight, and thus fails to recover the true population frequencies. On the other hand, PyClone can not identify if cluster 2 contains homozygous

(corresponding to cellular prevalence of 0.5) or heterozygous (corresponding to cellular prevalence of 1) variants. In contrast, using the phasing information, PairClone is able to infer two subclones having genotypes $(00, 11)$ and $(01, 10)$ for mutation pairs 11–30, and we know cluster 2 contains only heterozygous variants for sure.

2.4.3 Simulation 2

In the second simulation, we consider data with $K = 100$ mutation pairs and a more complicated subclonal structure with $C^{\text{TRUE}} = 4$ latent subclones. We generate hypothetical data for $T = 4$ samples. The subclone matrix $\boldsymbol{Z}^{\text{TRUE}}$ is shown in Figure 2.4(a). Colors on a scale from light grey to red, to black (see the scale in the figure) are used to represent genotype $\boldsymbol{z}^{(q)}$ with $q = 1, \ldots, 10$. For example, subclone 4 has genotype $\boldsymbol{z}^{(10)}$ for mutation pairs 1–20, $\boldsymbol{z}^{(5)}$ for mutation pairs 21–40, $\boldsymbol{z}^{(8)}$ for mutation pairs 41–60, $\boldsymbol{z}^{(1)}$ for mutation pairs 61–80, and $\boldsymbol{z}^{(9)}$ for mutation pairs 81–100. For each sample t, we generate the subclone proportions from a Dirichlet distribution, $\boldsymbol{w}_t^{\text{TRUE}} \sim \text{Dir}(0.01, \sigma(20, 10, 5, 2))$, where $\sigma(20, 10, 5, 2)$ is a random permutation of $(20, 10, 5, 2)$. The subclone proportion matrix $\boldsymbol{w}^{\text{TRUE}}$ is shown in Figure 2.4(b), where darker blue color indicates higher abundance of a subclone in a sample, and light grey color represents low abundance. The parameters $\boldsymbol{\rho}^{\text{TRUE}}$ and N_{tk} are generated using the same approach as before, and we use $v_{tk2} = v_{tk3} = 0.3$ for $k = 1, \ldots, 50$ and all t, and $v_{tk2} = v_{tk3} = 0.35$ for $k = 51, \ldots, 100$ and all t. Finally, we calculate $\{p_{tkg}^{\text{TRUE}}\}$ and generate read

counts n_{tkg} from equation (2.1) similar to previous simulation.



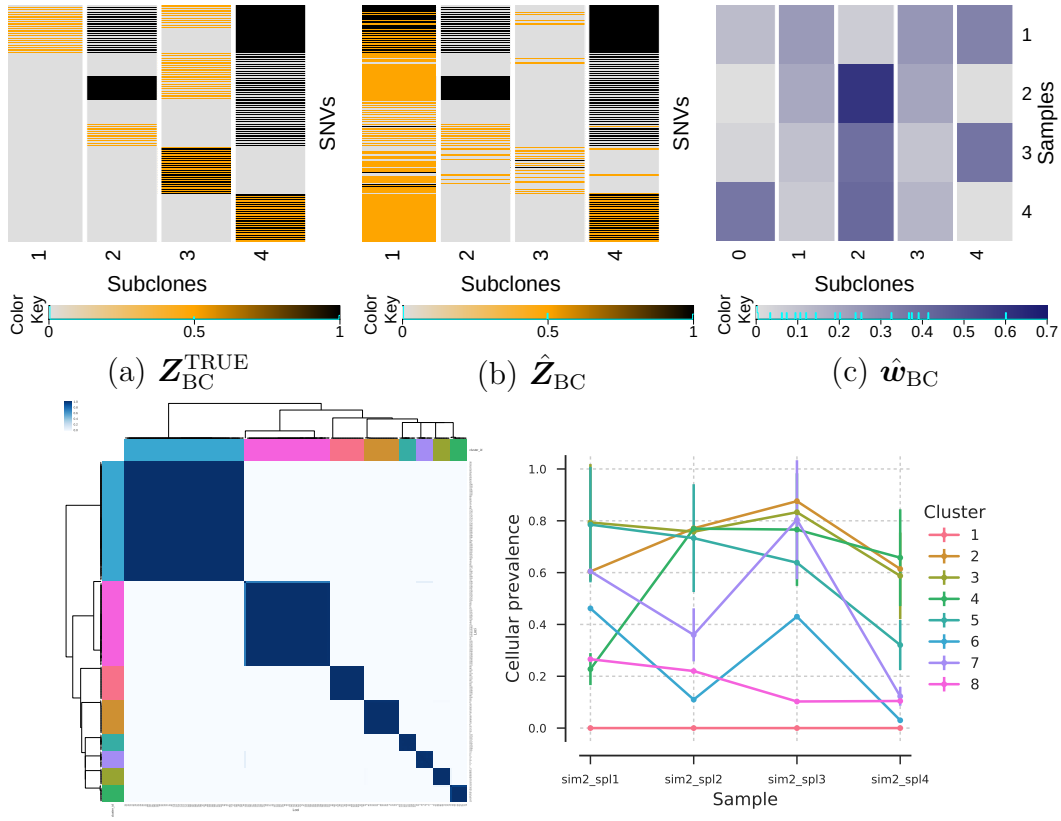
(a) $\boldsymbol{Z}^{\text{TRUE}}$ (b) $\boldsymbol{w}^{\text{TRUE}}$ (c) $p_b(C \mid \boldsymbol{n}'')$

(d) $\hat{\boldsymbol{Z}}$ (e) $\hat{\boldsymbol{w}}$ (f) Histogram of $(\hat{p}_{tkg} - p_{tkg}^{\text{TRUE}})$

Figure 2.4: Simulation 2. Simulation truth $\boldsymbol{Z}^{\text{TRUE}}$ and $\boldsymbol{w}^{\text{TRUE}}$ (a, b), and posterior inference under PairClone (c, d, e, f).

We fit the model with the same set of hyperparameters and MCMC parameters as in simulation 1. Figure 2.4(c) shows $p_b(C \mid \boldsymbol{n}'')$. Again, the posterior mode $\hat{C} = 4$ recovers the truth. Figure 2.4(d) shows the estimate $\hat{\boldsymbol{Z}}$; the truth is nicely approximated. Some mismatches are expected under this more complex subclone structure. The estimated subclone proportions $\hat{\boldsymbol{w}}$ are shown in Figure 2.4(e), again close to the truth. Figure 2.4(f) shows the

histogram of $(\hat{p}_{tkg} - p_{tkg}^{\text{TRUE}})$ which indicates a good model fit.



(a) $\boldsymbol{Z}_{\text{BC}}^{\text{TRUE}}$ (b) $\hat{\boldsymbol{Z}}_{\text{BC}}$ (c) $\hat{\boldsymbol{w}}_{\text{BC}}$

(d) Posterior similarity matrix (e) Cellular prevalence

Figure 2.5: Simulation 2. Posterior inference under BayClone (a, b, c) and PyClone (d, e).

For comparison, we again fit the same simulated data with BayClone and PyClone. BayClone chooses the model with 4 subclones, which still recovers the truth. However, using only SNV data, BayClone can not see the connection between adjacent SNVs, and inference fails to recover $\boldsymbol{w}_{\text{BC}}^{\text{TRUE}}$ and therefore $\boldsymbol{Z}_{\text{BC}}^{\text{TRUE}}$, even approximately. PyClone infers 8 clusters for the 200

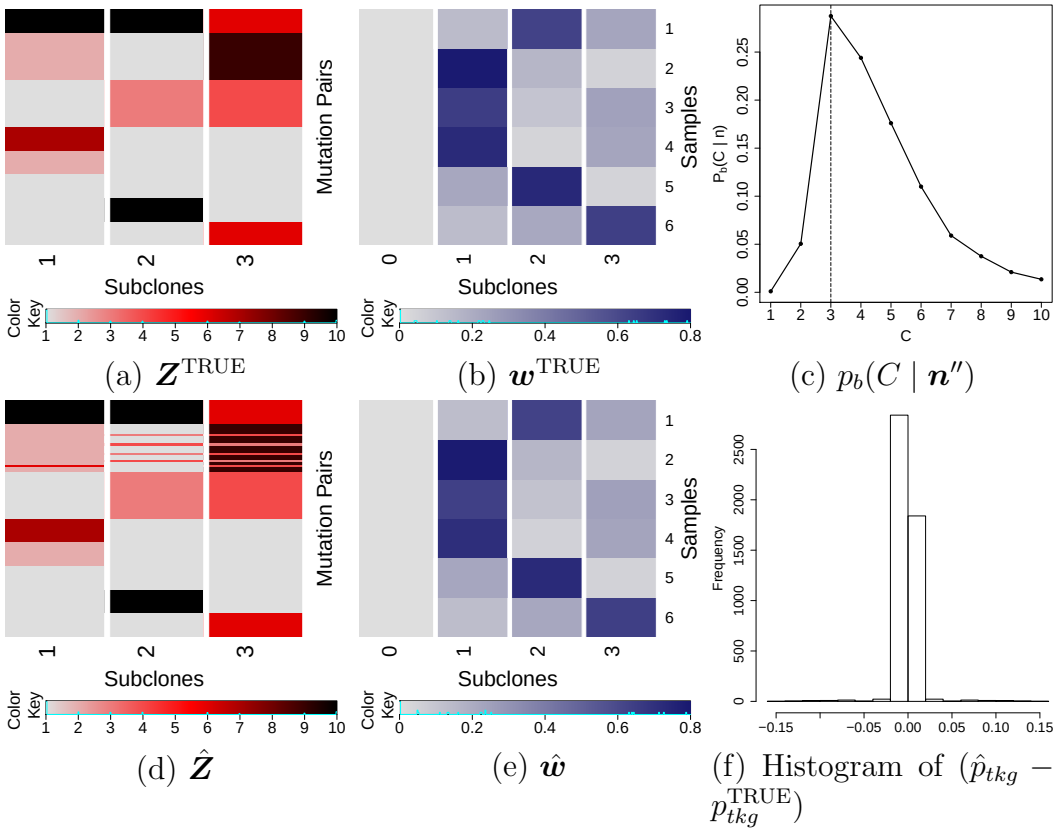
loci, which reasonably recovers the truth. However, since the underlying subclone structure is more complex, the PyClone cellular prevalence is not directly comparable to PairClone outputs.

2.4.4 Simulation 3

In the last simulation we use $T = 6$ samples with $C^{\text{TRUE}} = 3$ and latent subclones. We still consider $K = 100$ mutation pairs. The subclone matrix $\boldsymbol{Z}^{\text{TRUE}}$ is shown in Figure 2.6(a). For each sample t, we generate the subclone proportions from $\boldsymbol{w}_t^{\text{TRUE}} \sim \text{Dir}(0.01, \sigma(14, 6, 3))$, where $\sigma(14, 6, 3)$ is a random permutation of $(14, 6, 3)$. The proportions $\boldsymbol{w}^{\text{TRUE}}$ are shown in Figure 2.6(b). The parameters $\boldsymbol{\rho}^{\text{TRUE}}$ and N_{tk} are generated using the same approach as before, and we use the same v_{tk2} and v_{tk3} as in Simulation 2. Finally, we calculate $\{p_{tkg}^{\text{TRUE}}\}$ and generate read counts n_{tkg} from equation (2.1) similar to simulation 1.

We fit the model with the same hyperparameter and the same MCMC tuning parameters as in simulation 1. We now use a smaller test sample size, i.e., a smaller fraction b in the transdimensional MCMC. See Section A.3 for a discussion.

Figure 2.6(c) shows $p_b(C \mid \boldsymbol{n}'')$, with the posterior mode $\hat{C} = 3$ recovering the truth. Figures 2.6(d, e) show $\hat{\boldsymbol{Z}}$ and $\hat{\boldsymbol{w}}$. Comparing with panels (a) and (b) we can see an almost perfect recovery of the truth. Figure 2.6(f) shows a histogram of the residuals $(\hat{p}_{tkg} - p_{tkg}^{\text{TRUE}})$. The plot indicates a good model fit.

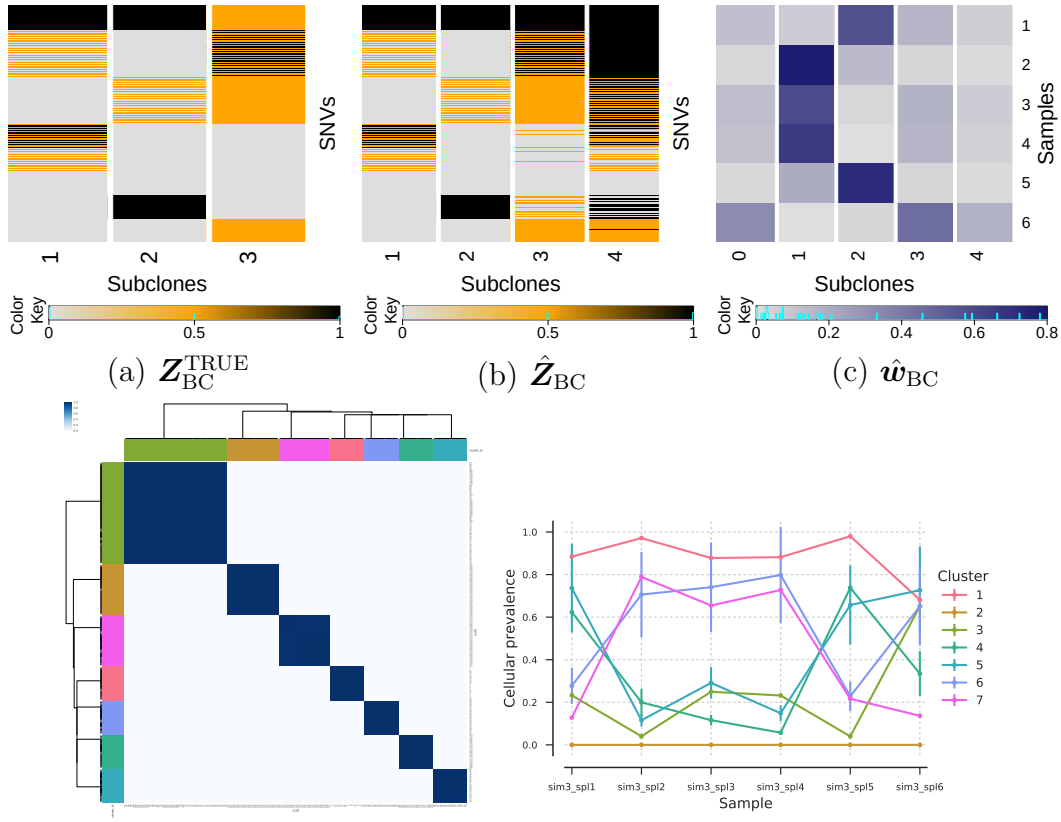


(a) $\boldsymbol{Z}^{\text{TRUE}}$ (b) $\boldsymbol{w}^{\text{TRUE}}$ (c) $p_b(C \mid \boldsymbol{n}'')$

(d) $\hat{\boldsymbol{Z}}$ (e) $\hat{\boldsymbol{w}}$ (f) Histogram of $(\hat{p}_{tkg} - p_{tkg}^{\text{TRUE}})$

Figure 2.6: Simulation 3. Simulation truth $\boldsymbol{Z}^{\text{TRUE}}$ and $\boldsymbol{w}^{\text{TRUE}}$ (a, b), and posterior inference under PairClone (c, d, e, f).

We again compare with inference under BayClone and PyClone. In this case, BayClone chooses the model with 4 subclones, failing to recover the truth. PyClone infers 7 clusters for the 200 loci, which reasonably recovers the truth, but the result is still not directly comparable.



(a) $\boldsymbol{Z}_{\text{BC}}^{\text{TRUE}}$ (b) $\hat{\boldsymbol{Z}}_{\text{BC}}$ (c) $\hat{\boldsymbol{w}}_{\text{BC}}$

(d) Posterior similarity matrix (e) Cellular prevalence

Figure 2.7: Simulation 3. Posterior inference under BayClone (a, b, c) and PyClone (d, e).

2.5 PairClone Extensions

2.5.1 Incorporating Marginal Read Counts

Most somatic mutations are not part of the paired reads that we use in PairClone. We refer to these single mutations as SNVs (single nucleotide variants) and consider the following simple extension to incorporate marginal counts for SNVs in PairClone. We introduce a new $S \times C$ matrix $\boldsymbol{Z}^S$ to

represent the genotype of the C subclones for these additional SNVs. To avoid confusion, we denote the earlier $K \times C$ subclone matrix by $\boldsymbol{Z}^P$ in this section. The (s, c) element of $\boldsymbol{Z}^S$ reports the genotype of SNV s in subclone c, with $z^S_{sc} \in \{0, 0.5, 1\}$ denoting homozygous wild-type (0), heterozygous variant (0.5), and homozygous variant (1), respectively. The c-th column of $\boldsymbol{Z}^P$ and $\boldsymbol{Z}^S$ together define subclone c. We continue to assume copy number neutrality in all SNVs and mutation pairs (we discuss an extension to incorporating subclonal copy number variations in the next subsection). The marginal read counts are easiest incorporated in the PairClone model by recording them as right (or left) missing reads (as described in Section 2.2.1) for hypothetical pairs, $k = K + 1, \ldots, k + S$. Let $\tilde{N}_{ts}$ and $\tilde{n}_{ts}$ denote the total count and the number of reads bearing a variant allele, respectively, for SNV s in sample t. Treating s as a mutation pair $k = K + s$ with missing second read, we record $n_{tk8} = \tilde{n}_{ts}$, $n_{tk1} = \ldots = n_{tk7} = 0$ and $N_{tk} = \tilde{N}_{ts}$. We then proceed as before, now with $K + S$ mutation pairs. Inference reports an augmented $(K + S) \times C$ subclone matrix $\tilde{\boldsymbol{Z}}^P$. We record the first K rows of $\tilde{\boldsymbol{Z}}^P$ as $\boldsymbol{Z}^P$, and transform the remaining S rows to $\boldsymbol{Z}^S$ by only recording the genotypes of the observed loci.

We evaluate the proposed modeling approach with a simulation study. The simulation setting is the same as simulation 3 in Section 2.4, except that we discard the phasing information of mutation pairs $51 - 100$ and only record their marginal read counts. Figure 2.8(a)–(f) summarizes the simulation results. Panels (a, b) show the simulation truth for the mutation pairs and SNVs,

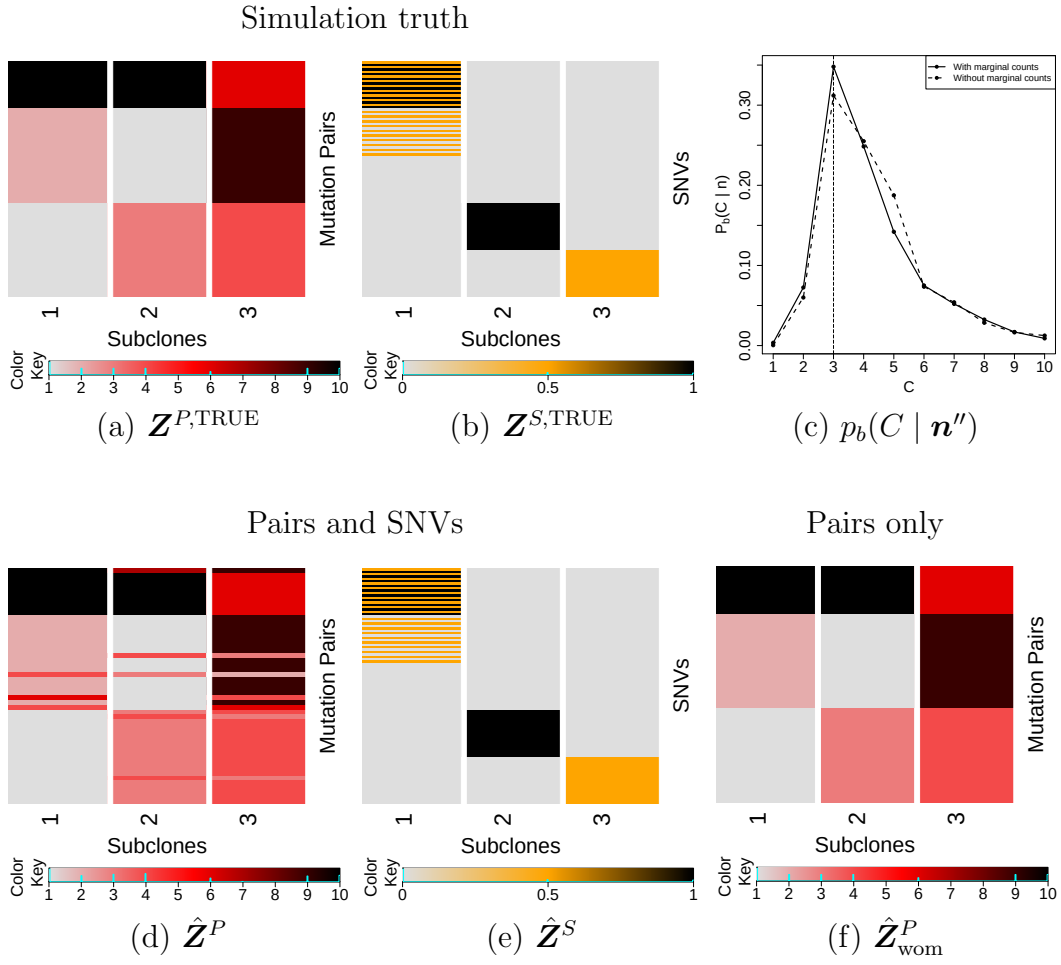
respectively. Panel (c) shows the posterior $p(C \mid \boldsymbol{n}'')$ and panels (d, e) show the estimated genotypes $\hat{\boldsymbol{Z}}^P$ and $\hat{\boldsymbol{Z}}^S$. Inference for the weights w_{tc} recovers the simulation truth (not shown). The result compares favorably to inference under BayClone (Figure 2.7(b)), due to the additional phasing information for the first 50 mutation pairs.

For a direct evaluation of the information in the additional marginal counts we also evaluate posterior inference with only the first 50 mutation pairs, shown in Figure 2.8 (c, f). Comparison with Figure 2.8 (c, d) shows that the additional marginal counts do not noticeably improve inference on tumor heterogeneity.

2.5.2 Incorporating Tumor Purity

Usually, tumor samples are not pure in the sense that they contain certain proportions of normal cells. Tumor purity refers to the fraction of tumor cells in a tumor sample. To explicitly model tumor purity, we introduce a normal subclone, the proportion of which in sample t is denoted by $w_{t\star}$, $t = 1, \ldots, T$. The normal subclone does not possess any mutation (since we only consider somatic mutations). The tumor purity for sample t is thus $(1 - w_{t\star})$. The normal subclone is denoted by $\boldsymbol{z}_*$, with $\boldsymbol{z}_{k*} = \boldsymbol{z}^{(1)}$ for all k. The remaining subclones are still denoted by $\boldsymbol{z}_c$, $c = 1, \ldots, C$, with proportion w_{tc} in sample t, and $\sum_{c=0}^{C} w_{tc} + w_{t\star} = 1$.

The probability model needs to be slightly modified to accommodate the normal subclone. The sampling model remains unchanged as (2.1). Same



(a) $\boldsymbol{Z}^{P,\text{TRUE}}$ (b) $\boldsymbol{Z}^{S,\text{TRUE}}$ (c) $p_b(C \mid \boldsymbol{n}'')$

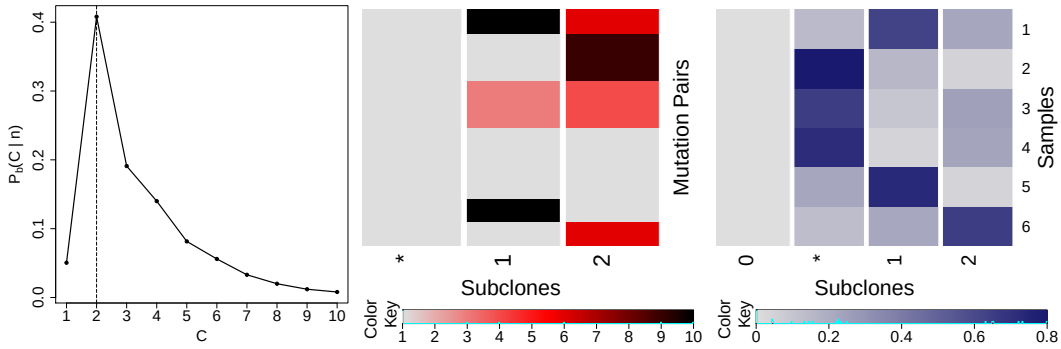
(d) $\hat{\boldsymbol{Z}}^{P}$ (e) $\hat{\boldsymbol{Z}}^{S}$ (f) $\hat{\boldsymbol{Z}}^{P}_{\text{wom}}$

Figure 2.8: Summary of simulation results using additional marginal read counts. Simulation truth $\boldsymbol{Z}^{P,\text{TRUE}}$ and $\boldsymbol{Z}^{S,\text{TRUE}}$ (a, b), posterior inference with marginal read counts incorporated (c, d, e), and posterior inference without marginal read counts (c, f).

for the prior models for $\boldsymbol{Z}$, $\boldsymbol{\rho}$ and C. We only change the construction of $\tilde{p}_{tkg}$ and $p(\boldsymbol{w})$ as follows. With a new normal subclone, the probability of observing a short read $\boldsymbol{h}_g$ becomes $\tilde{p}_{tkg} = \sum_{c=1}^{C} w_{tc}\, A(\boldsymbol{h}_g, \boldsymbol{z}_{kc}) + w_{t\star}\, A(\boldsymbol{h}_g, \boldsymbol{z}^{(1)}) + w_{t0}\, \rho_g$, based on the same generative model described in Section 2.2.2. Let $\tilde{w}_{tc} =$

$w_{tc}/(1 - w_{t\star})$. We use a Beta-Dirichlet prior, $w_{t\star} \overset{iid}{\sim} \text{Be}(d_1^*, d_2^*)$, and $\tilde{\boldsymbol{w}}_t \overset{iid}{\sim} \text{Dir}(d_0, d, \ldots, d)$. An informative prior for $w_{t\star}$ could be based on an estimate from a purity caller, for example, Van Loo et al. (2010) or Carter et al. (2012).

We evaluate the modified model with a simulation study. The simulation setting is the same as simulation 3 in Section 2.4, except that we substitute the first subclone with a normal subclone. We use exactly the same hyperparameters as those in simulations 2 and 3, and in addition we take $d_1^* = d_2^* = 1$. Figure 2.9 summarizes inference results. Columns in panels (b) and (c) marked with "*" correspond to the normal subclone. Panel (a) shows $p_b(C \mid \boldsymbol{n}'')$. Posterior inference recovers the simulation truth, with posterior mode $\hat{C} = 2$. Panel (b) shows $\hat{\boldsymbol{Z}}$. Comparing with subclones 2 and 3 in Figure 2.6(a) we find a good recovery of the simulation truth. Panel (c) shows $\hat{\boldsymbol{w}}$, which can be compared with Figure 2.6(b).



(a) $p_b(C \mid \boldsymbol{n}'')$ (b) $\hat{\boldsymbol{Z}}$ (c) $\hat{\boldsymbol{w}}$

Figure 2.9: Summary of simulation results with tumor purity incorporated.

2.5.3 Incorporating Copy Number Changes

Tumor cells not only harbor sequence mutations such as SNVs and mutation pairs, they often undergo copy number changes and produce copy number variants (CNVs). Genomic regions with CNVs have copy number $\neq 2$. We briefly outline an extension of PairClone that includes CNVs in the inference, but do not implement inference under this model, as it would require significantly more complex modeling. In addition to $\boldsymbol{Z}$ which describes sequence variation we introduce a $K \times C$ matrix $\boldsymbol{L}$ to represent subclonal copy number variation with ℓ_{kc} reporting the copy number for mutation pair k in subclone c. We use $\boldsymbol{L}$ to augment the sampling model to include the total read count N_{tk}. Earlier in (2.1), the multinomial sample size N_{tk} was considered fixed. We now add a sampling model. Following Lee et al. (2016) we assume

$$N_{tk} \mid \phi_t, M_{tk} \sim \text{Poisson}(\phi_t M_{tk}/2)$$

Here, ϕ_t is the expected number of reads in sample t under copy-neutral conditions, and M_{tk} is a weighted average copy number across subclones,

$$M_{tk} = \sum_{c=1}^{C} w_{tc}\ell_{kc} + w_{t0}\ell_{k0}.$$

The last term $w_{t0}\ell_{k0}$ accounts for noise and artifacts, where w_{t0} and ℓ_{k0} are the population frequency and copy number of the background subclone, respectively. We assume no CNVs for the background subclone, that is, $\ell_{k0} = 2$ for all k. We complete the model with a prior $p(\boldsymbol{L})$. Assuming $\ell_{kc} \in \{0, \ldots, Q\}$, i.e., a maximum copy number Q, we use another instance of a finite cIBP.

For each column of $\boldsymbol{L}$, we introduce $\boldsymbol{\pi}_c = (\pi_{c0}, \pi_{c1}, \ldots, \pi_{cQ})$ and assume $p(\ell_{kc} = q) = \pi_{cq}$, again with a Beta-Dirichlet prior for $\boldsymbol{\pi}_c$.

Recall the construction of $\tilde{p}_{tkg}$ in (2.3), including in particular the generative model. This generative model is now updated to include the varying ℓ_{tc}. To generate a short read for mutation pair k, we first select a subclone c from which the read arises, using the population frequencies $w_{tc}\ell_{kc}/\sum_{c=0}^{C} w_{tc}\ell_{kc}$ for sample t. Next we select with probability z_{kcj}/ℓ_{kc} one of the four possible alleles, $\boldsymbol{h}_g$, $g = 1, 2, 3$ or 4, where we now use $\boldsymbol{z}_{kc} = (z_{kcj}, j = 1, \ldots, 4)$ to denote numbers of alleles having genotypes 00, 01, 10 or 11, and $\sum_j z_{kcj} = \ell_{kc}$. In the case of left (or right) missing locus we observe $\boldsymbol{h}_g$, $g = 5$ or 6 (or $g = 7$ or 8), corresponding to the observed locus of the chosen allele, similar to before. In summary, the probability of observing a short read $\boldsymbol{h}_g$ can be written as

$$\tilde{p}_{tkg} = \sum_{c=0}^{C} \left[\frac{w_{tc}\ell_{kc}}{\sum_{c=1}^{C} w_{tc}\ell_{kc} + w_{t0}\ell_{k0}} \cdot \frac{A(\boldsymbol{h}_g, \boldsymbol{z}_{kc})}{\ell_{kc}} \right] = \frac{\sum_{c=0}^{C} w_{tc} A(\boldsymbol{h}_g, \boldsymbol{z}_{kc})}{M_{tk}},$$

where $A(\cdot)$ corresponds to the described generative model.

2.6 Lung Cancer Data

2.6.1 Using PairClone

We apply PairClone to analyze whole-exome in-house data. Whole-exome sequencing data is generated from four ($T = 4$) surgically dissected tumor samples taken from a single patient diagnosed with lung adenocarcinoma. The resected tumor is divided into two portions. One portion is flash frozen and another portion is formalin fixed and paraffin embedded (FFPE).

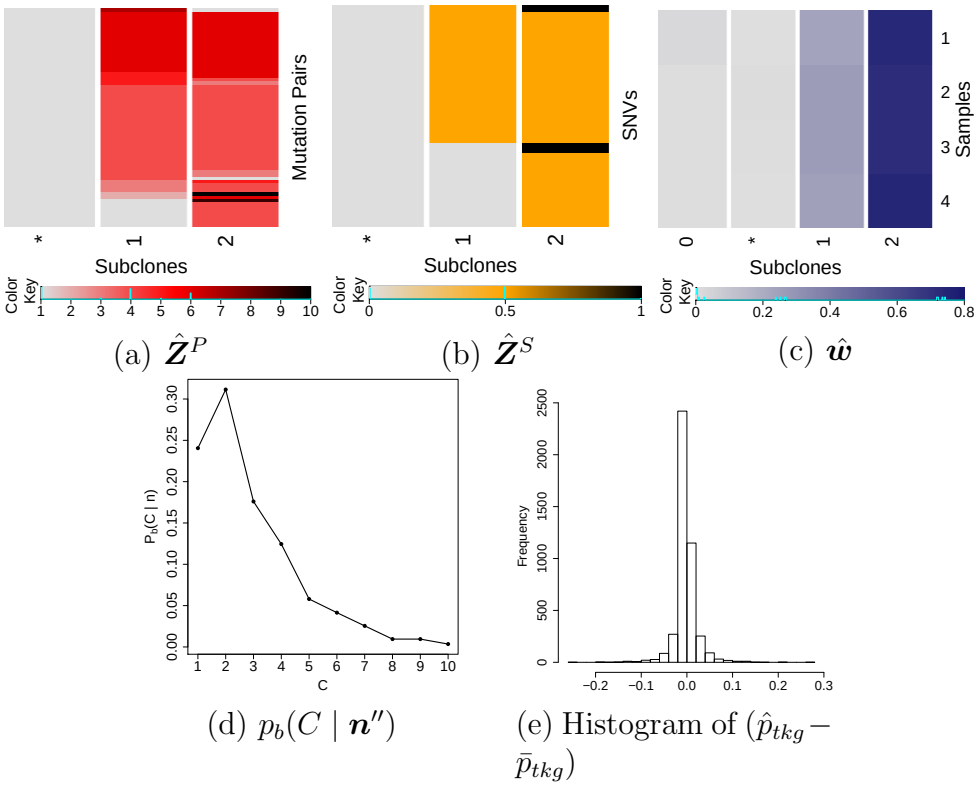
Four different samples (two from each portion) are taken. DNA is extracted from all four samples. Agilent SureSelect v5+UTR probe kit (targeting coding regions plus UTRs) is used for exome capture. The exome library is sequenced in paired-end fashion on an Illumina HiSeq 2000 platform. About 60 million reads are obtained in FASTQ file format, each of which is 100 bases long. We map paired-end reads to the human genome (version HG19) (Church et al., 2011) using BWA (Li and Durbin, 2009) to generate BAM files for each individual sample. After mapping the mean coverage of the samples is around 70 fold. We call variants using UnifiedGenotyper from GATK toolchain (McKenna et al., 2010) and generate a single VCF file for all of them. A total of nearly $115,000$ SNVs and small indels are called within the exome coordinates.

Next, using `LocHap` (Sengupta et al., 2016) we find mutation pair positions, the number of alleles and number of reads mapped to them. `LocHap` searches for multiple SNVs that are scaffolded by the same pair-end reads, that is, they can be recorded on one paired end read. We refer to such sets of multiple SNV's as local haplotypes (LH). When more than two genotypes are exhibited by an LH, it is called a LH variant (LHV). Using individual BAM files and the combined VCF file, `LocHap` generates four individual output file in HCF format (Sengupta et al., 2016). An HCF file contains LHV segments with two or three SNV positions. In this analysis, we are only interested in mutation pair, and therefore filter out all the LHV segments consisting of more than two SNV locations. We restrict our analysis to copy number neutral regions. To further improve data quality, we drop all LHVs where two SNVs are

very close to each other (within, say, 50 bps) or close to any type of structural variants such as indels. We also remove those LHVs where either of the SNVs is mapped with strand bias by most reads, or either of the SNVs is mapped towards the end of the most aligned reads. Finally, we only consider mutation pairs that have strong evidence of heterogeneity. Since LHVs exhibit > 2 genotypes in the short reads, by definition they are somatic mutations.

At the end of this process, 69 mutation pairs are left and we record the read data from HCF files for the analysis. In addition, in the hope of utilizing more information from the data, we randomly choose 69 un-paired SNVs and include them in the analysis. Since in practice, tumor samples often include contamination with normal cells, we incorporate inference for tumor purity as described in Section 2.5.2. We run MCMC simulation for $30,000$ iterations, discarding the first $10,000$ iterations as initial burn-in and keeping every 10th MCMC sample. We set the hyperparameter exactly as in the simulation study of Section 2.5.2.

Results. The posterior distribution $p_b(C \mid \boldsymbol{n}'')$ (Figure 2.10(d)) reports $p_b(C \mid \boldsymbol{n}'') = 0.24,\ 0.31,\ 0.17$ and 0.12 for $C = 1,\ 2,\ 3$ and 4, respectively, and then quickly drops below 0.1, with posterior mode $\hat{C} = 2$. This means, excluding the effect of normal cell contamination, the tumor samples have two subclones. Figure 2.10(a, b) show the estimated subclone matrix $\hat{\boldsymbol{Z}}^P$ and $\hat{\boldsymbol{Z}}^S$ corresponding to mutation pairs and SNVs, respectively. The first column of $\hat{\boldsymbol{Z}}^P$ and $\hat{\boldsymbol{Z}}^S$ represents the normal subclone. The rows for both matrices are



(a) $\hat{\boldsymbol{Z}}^P$ (b) $\hat{\boldsymbol{Z}}^S$ (c) $\hat{\boldsymbol{w}}$

(d) $p_b(C \mid \boldsymbol{n}'')$ (e) Histogram of $(\hat{p}_{tkg} - \bar{p}_{tkg})$

Figure 2.10: Lung cancer. Posterior inference under PairClone.

reordered for a better display. Figure 2.10(c) shows the estimated subclone proportions $\hat{\boldsymbol{w}}$ for the four samples. The second column of $\hat{\boldsymbol{w}}$ represents the proportions of normal subclones in the four samples. The small values indicate high purity of the tumor samples. The similar proportions across the four samples reflect the spatial proximity of the samples. Furthermore, excluding a few exceptions that might be due to model mis-fitting, the subclones form a simple phylogenetic tree: $* \to 1 \to 2$. Subclones 1 and 2 share a large portion of common mutations, while subclone 2 has some private mutations that are

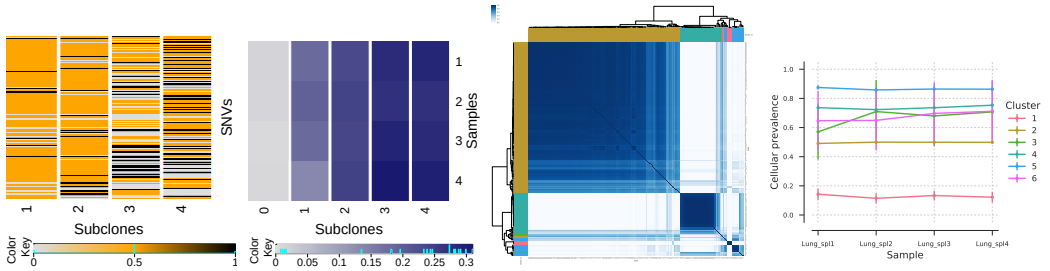
missing in subclone 1.

For informal model checking we inspect a histogram of realized residuals (Figure 2.10(e)). To define residuals, we calculate estimated multinomial probabilities $\{\hat{p}_{tkg}\}$ according to $\hat{\boldsymbol{Z}}$, $\hat{\boldsymbol{w}}$ and empirical values of $\{v_{tk1}, v_{tk2}, v_{tk3}\}$. Let $\bar{p}_{tkg} = n_{tkg}/N_{tk}$. The figure plots the residuals $(\hat{p}_{tkg} - \bar{p}_{tkg})$. The resulting histogram of residuals is centered around zero with little mass beyond ± 0.04, indicating a good model fit.

2.6.2 Using SNVs only

For comparison, we also run BayClone and PyClone on the same dataset. Using the log pseudo marginal likelihood (LPML), BayClone reports $\hat{C} = 4$ subclones. The estimated subclone matrix in BayClone's format is shown in Figure 2.11(a), with the rows reordered in the same way as in Figure 2.10(a, b). In light of the earlier simulation results we believe that the inference under PairClone is more reliable. Figure 2.11(b) shows the estimated subclone proportions under BayClone. Figure 2.11(c) shows the estimated clustering of the SNV loci under PyClone (the color coding along the axes). PyClone identifies 6 different clusters. The largest cluster (shown in brown) corresponds to loci that have heterozygous variants in both subclones 1 and 2, the second-largest cluster (shown in blueish green) corresponds to loci that have homozygous wild types in subclone 1 and homozygous variants in subclone 2, and the other smaller clusters represent other less common combinations. The clusters match with clustering of rows of $\hat{\boldsymbol{Z}}^P$ and $\hat{\boldsymbol{Z}}^S$. PyClone does not immediately give

inference on subclones, but combing clusters with similar cellular prevalence across samples one is able to conjecture subclones. In this sense, PyClone gives similar result compared with PairClone. Finally, Figure 2.11(d) displays PyClone's estimated cellular prevalences of clusters across different samples. The estimated subclone proportions and cellular prevalences across the four samples remain very similar also under the BayClone and PyClone output, which strengthens our inference that the four samples possess the same subclonal profile, each with two subclones.

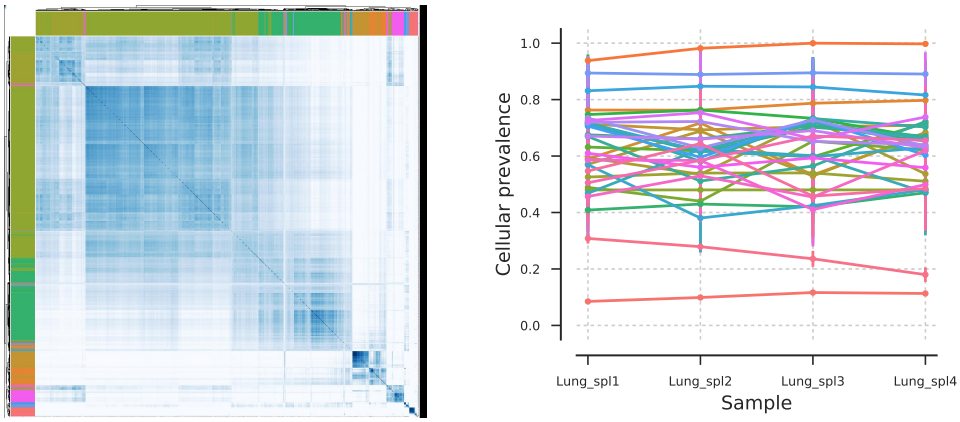


(a) $\hat{\boldsymbol{Z}}_{\text{BC}}$ (b) $\hat{\boldsymbol{w}}_{\text{BC}}$ (c) Clustering matrix (d) Cellular prevalence

Figure 2.11: Lung cancer. Posterior inference under BayClone (a, b) and PyClone (c, d).

For another comparison, we run PyClone with a much larger number of SNVs ($S = 1800$, which include the 69 pairs and 69 SNVs we ran analysis before) to evaluate the information gain by using additional marginal counts. The results are summarized in Figure 2.12, with panel (a) showing the estimated clustering of the 1800 SNVs. PyClone reports 34 clusters. The two largest clusters (olive and green clusters) in Panel (a) match with the two largest clus-

ters (brown and bluish green clusters) in Figure 2.11(c) and also corroborate the two subclones inferred by PairClone. In addition, PyClone infers lots of noisy tiny clusters using 1800 SNVs, which we argue model only noise. In summary, this comparison shows the additional marginal counts do not noticeably improve inference on tumor heterogeneity, and modeling mutation pairs is a reasonable way to extract useful information from the data.



(a) Clustering matrix (b) Cellular prevalence

Figure 2.12: Lung cancer. Posterior inference under PyClone using 1800 SNVs. PyClone inferred 34 clusters with two major clusters (olive and green) and many small noisy clusters (other colors).

2.7 Discussion

We can significantly enrich our understanding of cancer development by using high throughput NGS data to infer co-existence of subpopulations which are genetically different across tumors and within a single tumor (inter and intra tumor heterogeneity, respectively). In this paper, we have presented

a novel feature allocation model for reconstructing such subclonal structure using mutation pair data. Proposed inference explicitly models overlapping mutation pairs. We have shown that more accurate inference can be obtained using mutation pairs data compared to using only marginal counts for single SNVs. Short reads mapped to mutation pairs can provide direct evidence for heterogeneity in the tumor samples. In this way the proposed approach is more reliable than methods for subclonal reconstruction that rely on marginal variant allele fractions only.

The proposed model is easily extended for data where an LH segment consists of more than two SNVs. We can easily accommodate n-tuples instead of pairs of SNVs by increasing the number of categorical values (Q) that the entries in the $\boldsymbol{Z}$ matrix can take. There are several more interesting directions of extending the current model. For example, one could account for the potential phylogenetic relationship among subclones (i.e the columns in the $\boldsymbol{Z}$ matrix). Such extensions would enable one to infer mutational timing and allow the reconstruction of tumor evolutionary histories.

The proposed model characterizes tumor heterogeneity with SNV data. Other genetic variants including CNV could also provide important information about tumor heterogeneity. However, incorporating such data increases the complexity in modeling. Like other existing methods we chose to focus on the SNV data, keeping the development of models for other genetic variants as a research topic of it own. A future direction is to develop computationally efficient and reliable approaches to incorporating other genetic variants,

presumably by utilizing other available data in tumor heterogeneity.

Lastly, we focus on statistical inference using bulk sequencing data on tumor samples. Alternatively, biologists can apply single-cell sequencing on each tumor cell and study its genome one by one. This is a gold standard that can examine tumor heterogeneity at the single-cell level. However, single-cell sequencing is still expensive and cannot scale up. Also, many bioinformatics and statistical challenges are unmet in analyzing single-cell sequencing data.

Chapter 3

A Bayesian Treed Feature Allocation Model for Tumor Subclone Phylogeny Reconstruction Using Mutation Pairs

We present a latent feature allocation model to reconstruct a tumor phylogenetic tree and corresponding tumor heterogeneity. Similar to most current methods for inference on tumor heterogeneity, we consider data from next-generation sequencing. Unlike most methods that use information in short reads mapped to single nucleotide variants (SNVs), we consider subclone reconstruction using pairs of two proximal SNVs that can be mapped by the same short reads. A key part of the inference model is a phylogenetic tree prior that is used to construct a dependent prior on tumor cell subpopulations. The use of the tree structure in the prior greatly strengthens inference. Only subclones that conform with an a priori plausible phylogenetic tree are assigned non-negligible probability. The proposed Bayesian framework implies posterior distributions on the number of subclones, their genotypes, cellular proportions, and the phylogenetic tree spanned by the inferred subclones. The proposed method is validated against different sets of simulated and real-world data using single and multiple tumor samples. An open source software package is available at `http://www.compgenome.org/pairclonetree`.

3.1 Introduction

Tumor cells emerge from a Darwinian-like selection among multiple competing subpopulations of cells (Nowell, 1976, Bonavia et al., 2011, Marusyk et al., 2012). During tumorigenesis, through sequential clonal expansion and selection cells acquire distinct mutations. This process leads to genetically divergent subpopulations of cells, also known as subclones (Navin et al., 2010, Gerlinger et al., 2012, Nik-Zainal et al., 2012, Bignell et al., 2010, Bozic et al., 2010, Raphael et al., 2014). Reconstructing the subclones and their evolutionary relationship could help investigators to identify driver mutations that emerge early in the development or during the progression period. Such results provide insight about targeted therapies (Aparicio and Caldas, 2013, Papaemmanuil et al., 2011, Varela et al., 2011, Stephens et al., 2012).

A recent surge of genetic sequencing data makes it possible to investigate tumor subclonal architecture in detail (Oesper et al., 2013, Strino et al., 2013, Fischer et al., 2014, Miller et al., 2014, Roth et al., 2014, Jiao et al., 2014, Deshwar et al., 2015, Zare et al., 2014, Sengupta et al., 2015, Marass et al., 2017). We will discuss details of some (Marass et al., 2017, Jiao et al., 2014, Deshwar et al., 2015) later in Section 3.4, after we have introduced the required notation. Latest developments of next generation sequencing (NGS) technology enabled researchers to develop a variety of techniques that are broadly known as subclonal reconstruction. One of the aims is to deconvolute observed genomic data from a tumor into constituent signals corresponding to various subclones and to reconstruct their relationship in a phylogeny. In

most methods the reconstruction is based on short reads that are mapped to single nucleotide variants (SNVs) (few methods also consider somatic copy number aberrations, SCNA). SNV-based subclone calling methods utilize variant allele fractions (VAFs), that is, the fractions of alleles (or short reads) at each locus that carry mutations. Since humans are diploid, the VAFs of short reads for a homogeneous cell population should be 0, 0.5 or 1.0 for any locus in copy number neutral (copy number = 2) regions and after adjusting for tumor purity. VAFs different from 0, 0.5 or 1.0 are therefore evidence for heterogeneity. Based on this idea, existing SNV-based subclone calling methods either cluster mutations (Miller et al., 2014, Roth et al., 2014, Jiao et al., 2014, Deshwar et al., 2015), or use latent feature allocation methods to infer the subclone genotypes and their proportions (Zare et al., 2014, Sengupta et al., 2015, Marass et al., 2017). All are based on observed VAFs.

3.1.1 Main Idea

We assume that the available data are from T ($T \geq 1$) samples from a single patient and the main inference goal is intra-tumor heterogeneity. We present a novel approach to reconstruct tumor subclones and their corresponding phylogenetic tree based on mutation pairs. Here a mutation pair refers to a pair of proximal SNVs on the genomes that can be simultaneously mapped by the same paired-end short reads, with one SNV on each end. In other words, mutation pairs can be phased by short reads. See Fig. 3.1 for an illustration. Short reads mapped to only one of the SNV loci are treated as

partially missing paired-end reads and are not excluded from our approach. Specifically, marginal SNV reads can be included in our analysis. See Section 3.2.1 for more details. The idea of working with phased mutation pairs was introduced in Chapter 2. We build on it and develop a novel and entirely different inference approach by explicitly modeling the underlying phylogenetic relationship. That is, we model tumor heterogeneity based on a representation of a phylogenetic tree of tumor cell subpopulations. A prior probability model on such phylogenetic trees induces a dependent prior on the mutation profiles of latent tumor cell subpopulations. In particular, the phylogenetic tree of tumor cell subpopulations is included as a random quantity in the Bayesian model. Currently, we only consider mutation pairs in copy neutral region i.e. copy number two. The proposed inference aims to reconstruct (i) subclones defined by the haplotypes across all the mutation pairs, (ii) cellular proportion of each subclone, and (iii) a phylogenetic tree spanned by the subclones.

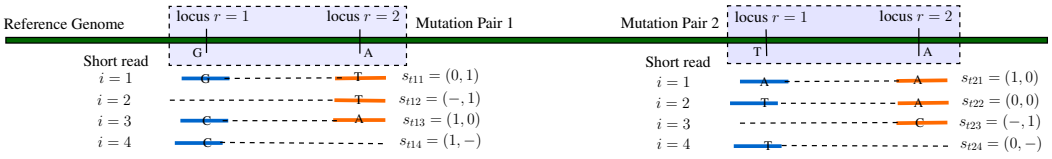


Figure 3.1: Short reads data from mutation pairs using NGS. Here s_{tki} denotes the i-th read for the k-th mutation pair in sample t. Each s_{tki} is a 2-dimensional vector which corresponds to the two proximal SNVs in the mutation pair, and each component of the vector takes values 0, 1 or – representing wild type, variant or missing genotype, respectively.

Consider an NGS data set with K mutation pairs shared across all T ($T \geq 1$) samples. We assume that the samples are composed of C homogeneous

subclones. The number of subclones C is unknown and becomes one of the model parameters. We use a $K \times C$ matrix $\boldsymbol{Z}$ to represent the subclones, in which each column of $\boldsymbol{Z}$ represents a subclone and each row represents a mutation pair. That is, the (kc) element $\boldsymbol{z}_{kc}$ of the matrix corresponds to subclone c and mutation pair k. Each $\boldsymbol{z}_{kc}$ is itself again a matrix. It is a 2×2 matrix that represents the genotypes of the two alleles of the mutation pair. See Fig. 3.2(b). An important step in the model construction is that the columns (subclones) of $\boldsymbol{Z}$ form a phylogenetic tree $\mathcal{T}$. The tree encodes the parent-child relationship across the subclones. A detailed construction of the tree and a prior probability model of $\mathcal{T}$ and $\boldsymbol{Z}$ are introduced later. Lastly, we denote $\boldsymbol{w}_t = (w_{t1}, \ldots, w_{tC})$ the cellular proportions of the subclones in sample t where $0 < w_{tc} < 1$ for all c and $\sum_{c=0}^{C} w_{tc} = 1$.

Using NGS data we infer $\mathcal{T}$, C, $\boldsymbol{Z}$ and $\boldsymbol{w}$ based on a simple idea that variant reads can only arise from subclones with variant alleles consistent with an underlying phylogeny. We develop a treed *latent feature allocation model* (LFAM) to implement this reconstruction. Mutation pairs are the objects of the LFAM, and subclones are the latent features chosen by the mutation pairs (in contrast to the phylogenetic Indian Buffet Process (Miller et al., 2012) which builds a tree structure on objects, rather than features). Note that subclone reconstruction based on LFAM allows overlapping mutations across subclones and therefore does not require the infinite sites assumption (Nik-Zainal et al., 2012). This is different from many existing cluster-based models in the literature. While LFAM attempts to directly infer genotypes of

all subclonal genomes, cluster-based models first infer SNV clusters based on VAFs and then reconstruct subclonal genotypes based on the clusters.

Advantage of using mutation pairs. Mutation pairs contain phasing information that improves the accuracy of subclone reconstruction. If two nucleotides reside on the same short read, we know that they must appear in the same DNA strand in a subclone. For example, consider a scenario with one mutation pair and two subclones. Suppose the reference genome allele is (A, G) for that mutation pair, with the notion that A and G are phased by the same DNA strand. Suppose the two subclones have diploid genomes at the two loci and the genotypes for both DNA strands are ((C, G), (A, T)) for subclone $c = 1$, and ((C, T), (A, G)) for $c = 2$. Since in NGS short reads are generated from a single DNA strand, short reads could be any of the four haplotypes (C, G), (A, T), (C, T) or (A, G) for this mutation pair. If indeed relative large counts of short reads with each haplotype are observed, one can reliably infer that there are heterogeneous cell subpopulations in the tumor sample. In contrast, if we ignore the phasing information and only consider the (marginal) VAFs for each SNV, then the observed VAFs for both SNVs are 0.5, which could be heterogeneous mutations from a single cell population. In this paper, we leverage the power of using mutation pairs over single SNVs to incorporate partial phasing information in our model. We assume that mutation pairs and their mapped short reads counts have been obtained using a bioinformatics pipeline, such as `LocHap` (Sengupta et al., 2016). Our aim

is to use short reads mapping data on mutation pairs to reconstruct tumor subclones and their phylogeny.

Difference from traditional phylogenetic tree. Phylogenetic trees are usually used to approximate perfect phylogeny for a fixed number of haplotypes (Gusfield, 1991, Bafna et al., 2003, Pe'er et al., 2004). Most methods lack assessment of tree uncertainties and report a single tree estimate. Also, methods based on SNVs put the observed mutation profile of SNV at the leaf nodes. This is natural if the splits in the tree create subpopulations that acquire or do not acquire a new mutation (or set of mutations). In contrast, we define a tree with all descendant nodes differing from the parent node by some mutations. That is, all node, including interior nodes, correspond to tumor cell subpopulations. See details below. For clarification we note that the prior structure in our model is different from the phylogenetic Indian Buffet Process (pIBP) (Miller et al., 2012), which models phylogeny of the objects rather than the features.

3.1.2 Representation of Subclones

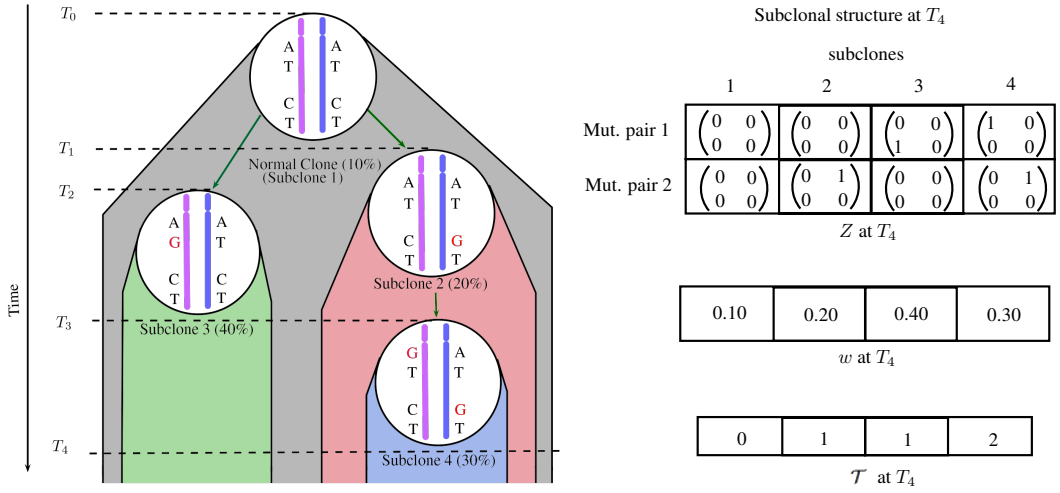
Fig. 3.2 presents a stylized example of temporal evolution of a tumor, starting from time T_0 and evolving until time T_4 with the normal clone (subclone $c = 1$) and three tumor subclones ($c = 2, 3, 4$). Each tumor subclone is marked by two mutation pairs with distinct somatic mutation profiles. In

Fig. 3.2 the true phylogenetic tree is plotted connecting the stylized subclones. The true population frequencies of the subclones are marked in parentheses. In panel (b) subclone genomes, their population frequencies and the phylogenetic relationship are represented by $\boldsymbol{Z}$, $\boldsymbol{w}$, and $\mathcal{T}$. The entries of $\mathcal{T}$ report for each subclone the index of the parent subclone (with $\mathcal{T}_1 = 0$ for the root clone $c = 1$).

Suppose K mutation pairs with C subclones are present. The subclone phylogeny can be visualized with a rooted tree with C nodes. We use a C-dimensional *parent* vector $\mathcal{T}$ to encode the parent-child relationship of a tree, where $\mathcal{T}_c = \mathcal{T}[c] = j$ means that subclone j is the parent of subclone c. The parent vector uniquely defines the topology of a rooted tree. We assume that the tumor evolutionary process always starts from the normal clone, indexed by $c = 1$. The normal clone does not have a parent, and we denote it by $\mathcal{T}_1 = 0$. For example, the parent vector representation of the subclone phylogeny in Fig. 3.2 is $\mathcal{T} = (0, 1, 1, 2)$.

We use the $K \times C$ matrix $\boldsymbol{Z}$ to represent the subclone genotypes. Each column of $\boldsymbol{Z}$ defines a subclone, and each row of $\boldsymbol{Z}$ corresponds to a mutation pair. The entry $\boldsymbol{z}_{kc}$ records the genotypes for mutation pair k in subclone c. Since each subclone has two alleles $j = 1, 2$, and each mutation pair has two loci $r = 1, 2$, the entry $\boldsymbol{z}_{kc}$ is itself a 2×2 matrix, i.e. $\boldsymbol{z}_{kc} = (\boldsymbol{z}_{kcj}, j = 1, 2)$ and $\boldsymbol{z}_{kcj} = (z_{kcjr}, r = 1, 2)$,

$$\boldsymbol{z}_{kc} = (\boldsymbol{z}_{kc1}, \boldsymbol{z}_{kc2}) = \left[\begin{pmatrix} z_{kc11} \\ z_{kc12} \end{pmatrix} \begin{pmatrix} z_{kc21} \\ z_{kc22} \end{pmatrix} \right]$$



(a) (b)

Figure 3.2: Schematic of subclonal evolution and subclone structure. Panel (a) shows the evolution of subclones over time. Panel (b) shows the subclonal structure at T_4 with genotypes $\boldsymbol{Z}$, cellular proportions $\boldsymbol{w}$ and parent vector $\mathcal{T}$. For each mutation pair k and subclone c, the entry $\boldsymbol{z}_{kc}$ of $\boldsymbol{Z}$ is a 2×2 matrix corresponding to the arrangement in the figure in panel (a), that is, with alleles in the two columns, and SNVs in the rows.

where $\begin{pmatrix} z_{kc11} \\ z_{kc12} \end{pmatrix}$ and $\begin{pmatrix} z_{kc21} \\ z_{kc22} \end{pmatrix}$ represent mutation pairs of allele 1 and allele 2, respectively. Theoretically, each z_{kcjr} can be any one of the four nucleotide sequences, A, C, G, T. However, at a single locus, the probability of having more than two sequences is negligible since it would require the same locus to be mutated twice throughout the life span of the tumor, which is extremely unlikely. Therefore, we assume z_{kcjr} can only take two possible values, with $z_{kcjr} = 1$ (or 0) indicating that the corresponding locus has a mutation (or does not have a mutation) compared to the reference genome, respectively. For example, in Fig. 3.2, we have $K = 2$ mutation pairs and $C = 4$ subclones. For

mutation pair $k = 2$ in subclone $c = 4$, the allele $j = 1$ harbors no mutation, while the allele $j = 2$ has a mutation at the first locus $r = 1$, which translates to $\boldsymbol{z}_{24} = (00, 10)$ (writing 00 as a shorthand for $(0, 0)^T$, etc.). Altogether, $\boldsymbol{z}_{kc}$ can take $2^4 = 16$ possible values $\boldsymbol{z}_{kc} \in \{(00, 00), (00, 01), \ldots, (11, 11)\}$. Since we do not have phasing information across mutation pairs, the $\boldsymbol{z}_{kc}$ values having mirrored columns lead to exactly the same data likelihood and thus are indistinguishable. Therefore, we reduce the number of possible values of $\boldsymbol{z}_{kc}$ to $Q = 10$. We list them below for further reference:
$\boldsymbol{z}^{(1)} = (00, 00)$, $\boldsymbol{z}^{(2)} = (00, 01)$, $\boldsymbol{z}^{(3)} = (00, 10)$, $\boldsymbol{z}^{(4)} = (00, 11)$, $\boldsymbol{z}^{(5)} = (01, 01)$, $\boldsymbol{z}^{(6)} = (01, 10)$, $\boldsymbol{z}^{(7)} = (10, 10)$, $\boldsymbol{z}^{(8)} = (01, 11)$, $\boldsymbol{z}^{(9)} = (10, 11)$ and $\boldsymbol{z}^{(10)} = (11, 11)$.
We assume that the normal subclone has no mutation, $\boldsymbol{z}_{k1} = \boldsymbol{z}^{(1)}$ for all k, indicating all mutations are somatic. In addition to these C true subclones, we introduce a background subclone, indexed as $c = 0$ and without biological meaning, to account for experimental noise and tiny subclones that are not detectable given the sequencing depth. We assume that the background subclone is a random mixture of all possible genotypes. See more discussion in Section 3.2.1.

Finally, we introduce notation for mixing proportions. Suppose T tissue samples are dissected from the same patient. We assume that the samples are admixtures of C subclones, each sample with a different set of mixing proportions (population frequencies). We use a $T \times (C+1)$ matrix $\boldsymbol{w}$ to record the proportions, where w_{tc} represents the population frequencies of subclone c

in sample t, $0 < w_{tc} < 1$ and $\sum_{c=0}^{C} w_{tc} = 1$. The proportions w_{t1} denotes the proportion of normal cells contamination in sample t.

The rest of the paper is organized as follows: Section 3.2 and Section 3.3 describe the latent feature allocation model and posterior inference, respectively. Section 3.4 presents two simulation studies. Section 3.5 reports analysis results for an actual experiment. We conclude with a discussion in Section 3.6.

3.2 The PairCloneTree Model

3.2.1 Sampling Model

Let $\boldsymbol{N}$ be a $T \times K$ matrix with N_{tk} representing read depth for mutation pair k in sample t. It records the number of times any locus of the mutation pair is covered by sequencing reads (see Fig. 3.1). Let $\boldsymbol{s}_{tki} = (s_{tkir}, r = 1, 2)$ be a specific short read where $r = 1, 2$ index the two loci in a mutation pair, $i = 1, 2, \ldots, N_{tk}$. We use $s_{tkir} = 1$ (or 0) to denote a variant (reference) sequence at the read, compared to the reference genome. An important feature of the data is that read i may not overlap with locus r. We use $s_{tkir} = -$ to represent the missing sequence on the read. Reads that do not overlap with either of the two loci are not included in the model as they do not contribute any information about the mutation pair. In summary, $\boldsymbol{s}_{tki}$ can take $G = 8$ possible values,

$$\boldsymbol{s}_{tki} \in \{\boldsymbol{s}^{(1)}, \ldots, \boldsymbol{s}^{(8)}\} = \{00, 01, 10, 11, -0, -1, 0-, 1-\}.$$

Among all N_{tk} reads, let $n_{tkg} = \sum_i I\left(\boldsymbol{s}_{tki} = \boldsymbol{s}^{(g)}\right)$ be the number of short reads having genotype $\boldsymbol{s}^{(g)}$. As illustrated in Fig. 3.1 out of total 4 reads ($N_{t1} = 4$), we have $n_{t12} = 1, n_{t13} = 1, n_{t16} = 1$ and $n_{t18} = 1$.

We assume a multinomial sampling model for the observed read counts

$$(n_{tk1}, \ldots, n_{tk8}) \mid N_{tk} \sim \text{Mn}(N_{tk}; p_{tk1}, \ldots, p_{tk8}),$$

where p_{tkg} is the probability of observing a short read $\boldsymbol{s}_{tki}$ with genotype $\boldsymbol{s}^{(g)}$. Later we link p_{tkg} with the underlying subclone structures.

If desired, it is straightforward to incorporate data for marginal SNV reads in the model. These reads can be treated as, without loss of generality, right missing reads, i.e. $s_{tki2} = -$. In this case, $n_{tk1} = \ldots = n_{tk6} = 0$, and the multinomial sampling model reduces to a binomial model. The addition of marginal SNV counts does not typically improve inference. See more details in Chapter 2.

Construction of p_{tkg}. For a short read $\boldsymbol{s}_{tki}$, depending on whether it covers both loci or only one locus, we consider three cases: (i) a read covers both loci, taking values $\boldsymbol{s}_{tki} \in \{\boldsymbol{s}^{(1)}, \ldots, \boldsymbol{s}^{(4)}\}$ (complete read); (ii) a read covers the second locus, taking values $\boldsymbol{s}_{tki} \in \{\boldsymbol{s}^{(5)}, \boldsymbol{s}^{(6)}\}$ (left missing read); and (iii) a read covers the first locus, taking values $\boldsymbol{s}_{tki} \in \{\boldsymbol{s}^{(7)}, \boldsymbol{s}^{(8)}\}$ (right missing read). Let $v_{tk1}, v_{tk2}, v_{tk3}$ denote the probabilities of observing a short read satisfying cases (i), (ii) and (iii), respectively. Conditional on cases (i), (ii) or (iii), let $\tilde{p}_{tkg}$ be the conditional probability of observing $\boldsymbol{s}_{tki} = \boldsymbol{s}^{(g)}$. We have $p_{tkg} =$

$v_{tk1}\,\tilde{p}_{tkg}, g = 1, \ldots, 4$, $p_{tkg} = v_{tk2}\,\tilde{p}_{tkg}, g = 5, 6$, and $p_{tkg} = v_{tk3}\,\tilde{p}_{tkg}, g = 7, 8$. We assume non-informative missingness and do not make inference on v's, so they remain constants in the likelihood.

We express $\tilde{p}_{tkg}$ in terms of $\boldsymbol{Z}$ and $\boldsymbol{w}$ based on the following generative model. Consider a sample t. To generate a short read, we first select a subclone c with probability w_{tc}. Next we select with probability 0.5 one of the two alleles $j = 1, 2$. Finally, we record the read $\boldsymbol{s}^{(g)}$, $g = 1, 2, 3$ or 4, corresponding to the chosen allele $\boldsymbol{z}_{kcj} = (z_{kcj1}, z_{kcj2})$. In the case of left (or right) missing locus we observe $\boldsymbol{s}^{(g)}$, $g = 5$ or 6 (or $g = 7$ or 8), corresponding to the observed locus of the chosen allele. Reflecting these three generative steps, we denote the probability of observing a short read $\boldsymbol{s}^{(g)}$ from subclone c that bears sequence $\boldsymbol{z}_{kcj}$ by

$$A(\boldsymbol{s}^{(g)}, \boldsymbol{z}_{kc}) = \sum_{j=1}^{2} 0.5 \times I(s_1^{(g)} = z_{kcj1})\, I(s_2^{(g)} = z_{kcj2}), \tag{3.1}$$

with the understanding that $I(- = z_{kcjr}) \equiv 1$ for missing reads. Implicit in (3.1) is the restriction $A(\boldsymbol{s}^{(g)}, \boldsymbol{z}_{kc}) \in \{0, 0.5, 1\}$, depending on the arguments.

Finally, using the conditional probabilities $A(\cdot)$ we obtain the marginal probability of observing a short read $\boldsymbol{s}^{(g)}$ from the tumor sample t with C subclones with cellular proportions $\{w_{tc}\}$ as

$$\tilde{p}_{tkg} = \sum_{c=1}^{C} w_{tc}\, A(\boldsymbol{s}^{(g)}, \boldsymbol{z}_{kc}) + w_{t0}\, \rho_g. \tag{3.2}$$

The first term in Eq. 3.2 states that the probability of observing a short read with genotype $\boldsymbol{s}^{(g)}$ is a weighted sum of the A's across all the subclones. Here

$w_{t0}\rho_g$ stands for the probability of observing $\boldsymbol{s}^{(g)}$ due to random noise. It can be thought of as a background subclone with weight w_{t0}, which is a random mixture of four genotypes 00, 01, 10 and 11 with proportions ρ_g. We assume the random noise does not differ across different mutation pairs, thus ρ_g does not have an index k. Note that $\rho_1 + \ldots + \rho_4 = \rho_5 + \rho_6 = \rho_7 + \rho_8 = 1$. Again, the background subclone (denoted by $c = 0$) has no biological meaning and is only used to account for noise and artifacts in the NGS data (sequencing errors, mapping errors, etc.).

3.2.2 Prior Model

We construct a hierarchical prior model, starting with $p(C)$, then a prior on the tree for a given number of nodes, $p(\mathcal{T} \mid C)$, and finally a prior on the subclonal genotypes given the phylogenetic tree $\mathcal{T}$.

Prior for C and $\mathcal{T}$. We assume a geometric prior for the number of subclones, $p(C) = (1-\alpha)^{C-1}\alpha$, $C \in \{1, 2, 3, \ldots\}$. Conditional on C, the prior on the tree, $p(\mathcal{T} \mid C)$ is as in Chipman et al. (1998). For a tree with C nodes, we let

$$p(\mathcal{T} \mid C) \propto \prod_{c=1}^{C} (1 + \eta_c)^{-\beta},$$

where η_c is the depth of node c, or the number of generations between node c and the normal subclone 1. The prior penalizes deeper trees and thus favors parsimonious representation of subclonal structure.

Prior for $\boldsymbol{Z}$. The subclone genotype matrix $\boldsymbol{Z}$ can be thought of as a feature allocation for categorical matrices. The mutation pairs are the objects, and the subclones are the latent features chosen by the objects. Each feature has 10 categories corresponding to the $Q = 10$ different genotypes. Conditioning on $\mathcal{T}$ the subclone genotype matrix needs to introduce dependence across features to reflect the assumed phylogeny. We construct a prior for $\boldsymbol{Z}$ based on the following generative model. We start from a normal subclone denoted by $\boldsymbol{z}_{\cdot 1} = \mathbf{0}$. Now consider a subclone $c > 1$ and defined by $\boldsymbol{z}_{\cdot c}$. The subclone preserves all mutations from its parent $\boldsymbol{z}_{\cdot \mathcal{T}_c}$, but also gains a Poisson number of new mutations. We assume the new mutations randomly happen at the unmutated loci of the parent subclone. A formal description of prior of $\boldsymbol{Z}$ follows.

For a subclone c, let $\ell_{kc} = \sum_{j,r} z_{kcjr}$ denote the number of mutations in mutation pair k, and let $\mathcal{L}_c = \{k : \ell_{kc} < 4\}$ denote the mutation pairs in subclone c that have less than four mutations. Let $m_{kc} = \ell_{kc} - \ell_{k\mathcal{T}_c}$ denote the number of new mutations that mutation pair k gains compared to its parent, and let $m_{\cdot c} = \sum_k m_{kc}$. We assume (i) The child subclone should acquire at least one additional mutation compared with its parent (otherwise subclone c would be identical to its parent $\mathcal{T}_c$). (ii) If the parent has already acquired all four mutations for a given k, then the child can not gain any more new mutation. That is, if $\ell_{k\mathcal{T}_c} = 4$, then $m_{kc} = 0$. (iii) Each mutation pair can gain at most one additional mutation in each generation, $m_{kc} \in \{0, 1\}$. Based on these assumptions, given a parent subclone $\boldsymbol{z}_{\cdot \mathcal{T}_c}$, we construct a child

subclone $\boldsymbol{z}_{\cdot c}$ as follows. Let $\mathcal{M}_c = \{k : m_{kc} = 1\}$ be the set of mutation pairs in subclone c where new mutations are gained. Let Choose$(\mathcal{L}, m)$ denote a uniformly chosen subset of $\mathcal{L}$ of size m, and let $X \sim$ Trunc-Pois$(\lambda; [a, b])$ represent a Poisson distribution with mean λ, truncated to $a \leq X \leq b$. We assume

$$\begin{aligned} m_{\cdot c} \mid \boldsymbol{z}_{\cdot \mathcal{T}_c}, \mathcal{T}, C &\sim \text{Trunc-Pois}(\lambda; [1, |\mathcal{L}_{\mathcal{T}_c}|]), \\ \mathcal{M}_c \mid m_{\cdot c}, \boldsymbol{z}_{\cdot \mathcal{T}_c}, \mathcal{T}, C &\sim \text{Choose}(\mathcal{L}_{\mathcal{T}_c}, m_{\cdot c}). \end{aligned} \tag{3.3}$$

The lower bound and upper bound of the truncated Poisson reflect assumptions (i) and (ii) respectively. Also, Eq. 3.3 implicitly captures assumption (iii).

Next, for a mutation pair that gains one new mutation, we assume the new mutation randomly arises in any of the unmutated loci in the parent subclone. Let $\mathcal{Z}_{kc} = \{(j, r) : z_{kcjr} = 0\}$, and let Unif$(A)$ denote a uniform distribution over the set A. We first choose

$$(j^*, r^*) \mid \boldsymbol{z}_{\cdot \mathcal{T}_c}, \mathcal{T}, C \sim \text{Unif}(\mathcal{Z}_{k\mathcal{T}_c}),$$

and then set $z_{kcj^*r^*} = 1$. So we have

$$p(\boldsymbol{Z} \mid \mathcal{T}, C) \propto \prod_{c=2}^{C} \text{Trunc-Pois}(m_{\cdot c}; [1, |\mathcal{L}_{\mathcal{T}_c}|]) . \frac{1}{\begin{pmatrix} |\mathcal{L}_{\mathcal{T}_c}| \\ m_{\cdot c} \end{pmatrix}} . \prod_{k \in \mathcal{M}_c} \frac{1}{|\mathcal{Z}_{k\mathcal{T}_c}|}.$$

Prior for $\boldsymbol{w}$ and $\boldsymbol{\rho}$. We design $p(\boldsymbol{w})$ in such a manner that we could put an informative prior for w_{t1} if a reliable estimate for tumor purity is available based on some prior bioinformatics pipeline (e.g. Van Loo et al. (2010), Carter

et al. (2012)). Recall that $c = 1$ is the normal subclone, i.e., w_{t1} is the normal subclone proportion, and that $\sum_{c=0,c\neq 1}^{C} w_{tc} + w_{t1} = 1$. We assume a Beta-Dirichlet prior on $\boldsymbol{w}$ such that,

$$w_{t1} \sim \text{Be}(a_p, b_p); \quad \text{and} \quad \frac{w_{tc}}{(1 - w_{t1})} \sim \text{Dir}(d_0, d, \cdots, d),$$

where $c = 0, 2, 3, \cdots, C$. We set $d_0 << d$ as w_{t0} is only a correction term to account for background noise and model mis-specification term.

The model is completed with a prior for $\boldsymbol{\rho} = \{\rho_g\}$. We consider complete read, left missing read and right missing read separately, and assume

$$\rho_{g_1} \sim \text{Dir}(d_1, \ldots, d_1); \quad \rho_{g_2} \sim \text{Dir}(2d_1, 2d_1); \quad \rho_{g_3} \sim \text{Dir}(2d_1, 2d_1),$$

where $g_1 = \{1, 2, 3, 4\}$, $g_2 = \{5, 6\}$ and $g_3 = \{7, 8\}$.

3.3 Posterior Inference

Let $\boldsymbol{x} = (\boldsymbol{Z}, \boldsymbol{w}, \boldsymbol{\rho})$ denote the unknown parameters except for the number of subclones C and the tree $\mathcal{T}$. Markov chain Monte Carlo (MCMC) simulation from the posterior $p(\boldsymbol{x} \mid \boldsymbol{n}, \mathcal{T}, C)$ is used to implement posterior inference. Gibbs sampling transition probabilities are used to update $\boldsymbol{Z}$, and Metropolis-Hastings transition probabilities are used to update $\boldsymbol{w}$ and $\boldsymbol{\rho}$. For example, we update $\boldsymbol{Z}$ by row with

$$p(\boldsymbol{z}_{k\cdot} \mid \boldsymbol{z}_{-k\cdot}, \ldots) \propto \prod_{t=1}^{T} \prod_{g=1}^{G} \left[\sum_{c=1}^{C} w_{tc}\, A(\boldsymbol{h}_g, \boldsymbol{z}_{kc}) + w_{t0}\, \rho_g \right]^{n_{tkg}} \cdot p(\boldsymbol{z}_{k\cdot} \mid \boldsymbol{z}_{-k\cdot}, \mathcal{T}, C),$$

where $\boldsymbol{z}_{k\cdot}$ is a row of $\boldsymbol{Z}$ satisfying the phylogeny $\mathcal{T}$.

Since the posterior distribution $p(\boldsymbol{x} \mid \boldsymbol{n}, \mathcal{T}, C)$ is expected to be highly multi-modal, we utilize parallel tempering (Geyer, 1991) to improve the mixing of the chain. Specifically, we use OpenMP parallel computing API (Dagum and Menon, 1998) in C++, to implement a scalable parallel tempering algorithm.

Updating C and $\mathcal{T}$. In general, posterior MCMC on tree structures can be very challenging to implement (Chipman et al., 1998, Denison et al., 1998). However, the problem here is manageable since plausible numbers for C constrain $\mathcal{T}$ to moderately small trees. We assume that the number of nodes is *a priori* restricted to $C_{\min} \leq C \leq C_{\max}$. Conditional on the number of subclones C, the number of possible tree topologies is finite. Let $\mathfrak{T}$ denote the (discrete) sample space of $(\mathcal{T}, C)$. Updating the values of $(\mathcal{T}, C)$ involves trans-dimensional MCMC. At each iteration, we propose new values for $(\mathcal{T}, C)$ from a uniform proposal, $q(\tilde{\mathcal{T}}, \tilde{C} \mid \mathcal{T}, C) \sim \text{Unif}(\mathfrak{T})$.

In order to search the space $\mathfrak{T}$ for the number of subclones and trees that best explain the observed data, we follow a similar approach as in Lee et al. (2015) and Chapter 2 (motivated by fractional Bayes' factor in O'Hagan (1995)) that splits the data into a training set and a test set. Recall that $\boldsymbol{n}$ represents the read counts data. We split $\boldsymbol{n}$ into a training set $\boldsymbol{n}'$ with $n'_{tkg} = bn_{tkg}$, and a test set $\boldsymbol{n}''$ with $n''_{tkg} = (1-b)n_{tkg}$. Let $p_b(\boldsymbol{x} \mid \mathcal{T}, C) = p(\boldsymbol{x} \mid \boldsymbol{n}', \mathcal{T}, C)$ be the posterior evaluated on the training set only. We use p_b in two instances. First, p_b is used as an informative prior instead of the

original prior $p((\boldsymbol{x} \mid \mathcal{T}, C)$, and second, p_b is used as a proposal distribution for $\tilde{\boldsymbol{x}}$, $q(\tilde{\boldsymbol{x}} \mid \tilde{\mathcal{T}}, \tilde{C}) = p_b(\tilde{\boldsymbol{x}} \mid \tilde{\mathcal{T}}, \tilde{C})$. Finally, the acceptance probability of proposal $(\tilde{\mathcal{T}}, \tilde{C}, \tilde{\boldsymbol{x}})$ is evaluated on the test set. Importantly, in the acceptance probability the (intractable) normalization constant of p_b cancels out, making this approach computationally feasible.

$$p_{\text{acc}}(\mathcal{T}, C, \boldsymbol{x}, \tilde{\mathcal{T}}, \tilde{C}, \tilde{\boldsymbol{x}}) = 1 \wedge \frac{p(\boldsymbol{n}'' \mid \tilde{\boldsymbol{x}}, \tilde{\mathcal{T}}, \tilde{C})}{p(\boldsymbol{n}'' \mid \boldsymbol{x}, \mathcal{T}, C)} \cdot \frac{p(\tilde{\mathcal{T}}, \tilde{C}) \cancel{p_b(\tilde{\boldsymbol{x}} \mid \tilde{\mathcal{T}}, \tilde{C})}}{p(\mathcal{T}, C) \cancel{p_b(\boldsymbol{x} \mid \mathcal{T}, C)}} \cdot \frac{q(\mathcal{T}, C \mid \tilde{\mathcal{T}}, \tilde{C}) \cancel{q(\boldsymbol{x} \mid \mathcal{T}, C)}}{q(\tilde{\mathcal{T}}, \tilde{C} \mid \mathcal{T}, C) \cancel{q(\tilde{\boldsymbol{x}} \mid \tilde{\mathcal{T}}, \tilde{C})}}.$$

Here we use p_b as an informative proposal distribution for $\tilde{\boldsymbol{x}}$ to achieve a better mixing Markov chain Monte Carlo simulation with reasonable acceptance probabilities. Without the use of an informative proposal, the proposed new tree is almost always rejected because the multinomial likelihood with the large sample size is very peaked. Under the modified prior $p_b(\cdot)$, the resulting conditional posterior on $\boldsymbol{x}$ remains entirely unchanged, $p_b(\boldsymbol{x} \mid \mathcal{T}, C, \boldsymbol{n}) = p(\boldsymbol{x} \mid \mathcal{T}, C, \boldsymbol{n})$ (Appendix A.2).

The described uniform tree proposal is in contrast to usual search algorithms for trees that generate proposals from neighboring trees. The advantage of this kind of proposal is to ensure a reasonable acceptance probability. But such algorithms have an important drawback that they quickly gravitate towards a local mode and then get stuck. A possible approach to addressing this problem is to repeatedly restart the algorithm from different starting trees. See Chipman et al. (1998) for more details. Our uniform tree proposal

combined with the data splitting scheme is another way to mitigate this challenge, efficiently searching the tree space while keeping a reasonable acceptance probability.

Point estimates for parameters. All posterior inference is contained in the posterior distribution for $\boldsymbol{x}, C$ and $\mathcal{T}$. For example, the marginal posterior distribution of C and $\mathcal{T}$ gives updates posterior probabilities for all possible values of C and $\mathcal{T}$. It is still useful to report point estimates. We use the posterior modes $(\hat{C}, \hat{\mathcal{T}})$ as point estimates for $(C, \mathcal{T})$, and conditional on $\hat{C}$ and $\hat{\mathcal{T}}$, we use the maximum a posteriori (MAP) estimator as an estimation for the other parameters. The MAP is approximated as the MCMC sample with highest posterior probability. Let $\{\boldsymbol{x}^{(l)}, l = 1, \ldots, L\}$ be a set of MCMC samples of $\boldsymbol{x}$, and

$$\hat{l} = \underset{l \in \{1,\ldots,L\}}{\arg\max}\ p(\boldsymbol{n} \mid \boldsymbol{x}^{(l)}, \hat{\mathcal{T}}, \hat{C})\, p(\boldsymbol{x}^{(l)} \mid \hat{\mathcal{T}}, \hat{C}).$$

We report point estimates as $\hat{\boldsymbol{Z}} = \boldsymbol{Z}^{(\hat{l})}$, $\hat{\boldsymbol{w}} = \boldsymbol{w}^{(\hat{l})}$ and $\hat{\boldsymbol{\rho}} = \boldsymbol{\rho}^{(\hat{l})}$.

3.4 Simulation Studies

We present two simulation studies to assess the proposed approach. We simulate single sample and multi-sample data with different read depths to test the performance of our model in different scenarios. In both simulation studies, we generate hypothetical read count data for $K = 100$ mutation pairs, which is a typical number of mutation pairs in a tumor sample. However, if needed,

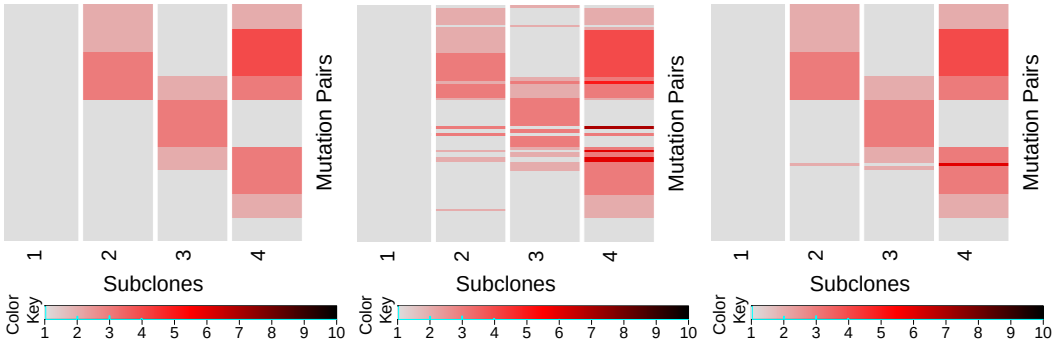
a much larger number of SNVs could be included in the model, with the only limiting concern being computational efficiency, which remains a challenge for all current methods.

3.4.1 Simulation 1

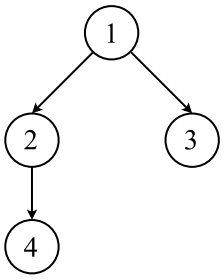
In the first simulation study, we consider $T = 1$ sample, which is the case for most real-world tumor cases due to the challenge in obtaining multiple samples from a patient. However, this does not rule out meaningful inference. As we will show, with good read depth, the simulation truth can still be recovered. Note that the relevant sample size is not the number of tissue samples, but closer to the number of reads, which is large even for $T = 1$.

We consider $K = 100$ mutation pairs and assume a simulation truth with $C = 4$ latent subclones. Fig. 3.3(a) and (d) show the true underlying subclonal genotypes and phylogeny, respectively. We use a heatmap to show the subclone matrix $\boldsymbol{Z}$, where colors from light gray to red to black are used to represent genotypes $\boldsymbol{z}^{(1)}$ to $\boldsymbol{z}^{(10)}$. The subclone weights are simulated from $\text{Dir}(0.01, \sigma(15, 10, 8, 5))$, where $\sigma(15, 10, 8, 5)$ stands for a random permutation of the four numbers. For the single sample in this simulation we get $\boldsymbol{w} =$ (0.000, 0.135, 0.169, 0.470, 0.226). The noise factor $\boldsymbol{\rho}$ is generated from its prior with $d_1 = 1$. In order to mimic a typical rate of observing left (or right) missing reads, we set $v_{tk2} = v_{tk3} = 0.25$, for $k = 1, \ldots, 50$, and $v_{tk2} = v_{tk3} = 0.3$, for $k = 51, \ldots, 100$. For the read depth N_{tk}, we consider two scenarios. In the first scenario, we consider 500x depth and generate $N_{tk} \sim$

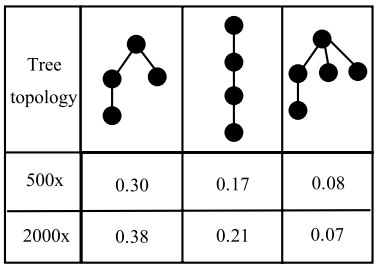
Discrete-Unif($[400, 600]$); in the second scenario, we consider 2000x depth and generate $N_{tk} \sim$ Discrete-Unif($[1900, 2100]$). While these read depth values are impossible from existing whole-genome sequencing technology, they are available from whole-exome or targeted sequencing experiments.



(a) $\boldsymbol{Z}$ (b) $\hat{\boldsymbol{Z}}_{500x}$ (c) $\hat{\boldsymbol{Z}}_{2000x}$



(d) True phylogeny



(e) Posterior prob. of tree topology

Figure 3.3: Simulation 1. Simulation truth $\boldsymbol{Z}$ (a) and phylogeny (d), and posterior inference under PairCloneTree (b, c, e).

We fit the model with the following hyperparameters: $\alpha = 0.5$, $\beta = 0.5$, $d = 0.5$, $d_0 = 0.03$, $d_1 = 1$, where the values of α and β imply mild penalty for deep and bushy trees (Chipman et al., 1998), and other hyperparameters are

generic non-informative choices. We set $a_p = d, b_p = d_0 + (C-1)d$ for given C as a non-informative prior choice and set $\lambda = 2K/C$ to express our prior belief that about half of the mutations occur uniformly at each generation. We use $C_{\min} = 2$ and $C_{\max} = 5$ as the range of C, since the vast majority of the methods in the literature show that even though a tumor sample could possess thousands to millions of SNVs, the number of inferred subclones usually is in the low single digit. Empirically, we choose the training set fraction as $b = 0.95$, as it performs well in all simulation studies. We run a total of 8000 MCMC simulations. Discarding the first 3000 draws as initial burn-in, we have a Monte Carlo sample with 5000 posterior draws.

Posterior inference with 500x read depth is summarized in Fig. 3.3(b, e). Fig. 3.3(e) shows the top three tree topologies and corresponding posterior probabilities. The posterior mode recovers the true phylogeny. Fig. 3.3(b) shows the estimated genotypes with 500x read depth, conditional on the posterior modes $(\hat{C}, \hat{\mathcal{T}})$. Some mismatches are due to the single sample and limited read depth. The estimated subclone proportions are $\hat{\boldsymbol{w}} = (0.000, 0.073, 0.171, 0.517, 0.239)$, which agrees with the truth.

Posterior inference with 2000x read depth is summarized in Fig. 3.3(c, e). The posterior mode recovers the true phylogeny. Fig. 3.3(c) shows the estimated genotypes. The simulation shows how larger read depths improve posterior accuracy and improve the power of recovering the latent structure. In particular, this shows that even with a single sample, with reasonable read depth, the truth can still be recovered. The estimated subclone proportions

are $\hat{\boldsymbol{w}} = (0.000, 0.127, 0.168, 0.477, 0.228)$.

3.4.2 Comparison with Cloe and PhyloWGS

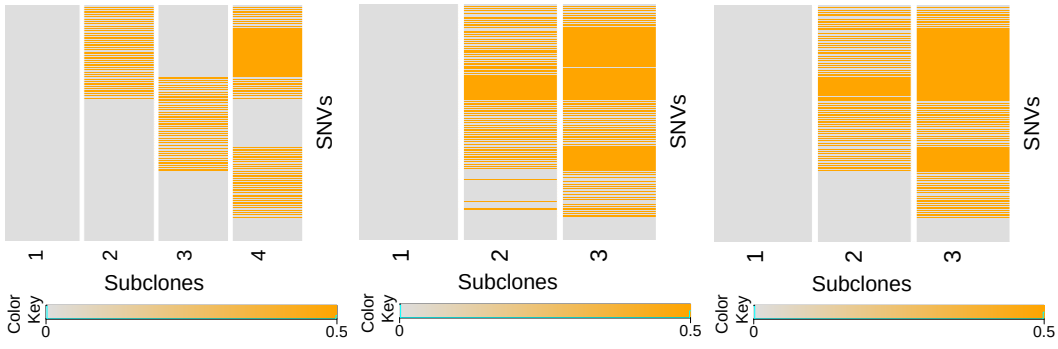
There is no other subclone calling method based on paired-end read data that also infers phylogeny. We therefore compare with other similar model-based approaches. In particular, we use Cloe (Marass et al., 2017) and PhyloWGS (Jiao et al., 2014, Deshwar et al., 2015) for inference with the same simulated data. These two methods also use highly structured Bayesian nonparametric priors and MCMC simulations for posterior inference. Both methods take mutant read counts and read depths for SNVs as input. Therefore, we discard the phasing information in mutation pairs and only record marginal counts for SNVs as the input. The simulation truth in Cloe and PhyloWGS's format is shown in Fig. 3.4(a). The orange color means a heterozygous mutation at the corresponding SNV locus.

Cloe infers clonal genotypes and phylogeny based on a similar feature allocation model. We run Cloe with the default hyperparameters for the same number of 8000 iterations with the first 3000 draws as initial burn-in. After that we carry out model selection for C with $2 \leq C \leq 5$. For the 500x read depth data, based on MAP estimate, Cloe reports 3 subclones with phylogeny $1 \rightarrow 2 \rightarrow 3$, and the subclone genotypes are shown in Fig. 3.4(b) with subclone proportions $\hat{\boldsymbol{w}}^{\text{Cloe}} = (0.569, 0.218, 0.213)$. For the 2000x read depth data, Cloe infers 2 subclones (genotypes not shown).

PhyloWGS, on the other hand, infers clusters of mutations and phy-

logeny. One can then make phylogenetic analysis to conjecture subclones and genotypes. Let $\tilde{\phi}_i$ denote the fraction of cells with a variant allele at locus i. The $\tilde{\phi}_i$'s are latent quantities related to the observed VAF for each SNV. PhyloWGS infers the phylogeny by clustering SNVs with matching $\tilde{\phi}_i$'s under a tree-structured prior for the unique values ϕ_j. In particular, they use the tree-structured stick breaking process (TSSB) (Adams et al., 2010). The TSSB implicitly defines a prior on the formation of subclones, including the prior on C and the number of novel loci that arise in each subclone. In contrast, PairCloneTree explicitly defines these model features, allowing easier prior control on C and $\mathcal{M}_c$. We run PhyloWGS with the default hyperparameters and 2500 iterations with a burn-in of 1000 samples. We only consider loci with VAF > 0 as the other loci do not provide information for PhyloWGS clustering. We then report cluster sizes and phylogeny based on MAP estimate. For the 500x read depth data, PhyloWGS reports 3 subclones with phylogeny $0 \rightarrow 1(77, 0.429) \rightarrow 2(53, 0.218)$, where 0 refers to the normal subclone, and the numbers in the brackets refer to the cluster sizes and cellular prevalences. The conjectured subclone genotypes are shown in Fig. 3.4(c), with subclone proportions $\hat{\boldsymbol{w}}^{\text{PWGS}} = (0.571,\ 0.211,\ 0.218)$. For the 2000x read depth data, PhyloWGS reports 3 subclones with phylogeny $0 \rightarrow 1(80, 0.431) \rightarrow 2(50, 0.227)$ (genotypes not shown).

Inferences under Cloe and PhyloWGS do not entirely recover the truth. The reason is probably that the common mutations of subclones 2 and 4 ($\mathcal{M}_2$ with a cellular prevalence of $0.169 + 0.226$) have a similar cellular prevalence



(a) $\boldsymbol{Z}^{\text{Cloe}}$ (b) $\hat{\boldsymbol{Z}}^{\text{Cloe}}_{500\text{x}}$ (c) $\hat{\boldsymbol{Z}}^{\text{PWGS}}_{500\text{x}}$

Figure 3.4: Simulation 1. Simulation truth $\boldsymbol{Z}^{\text{Cloe}}$ (a), and posterior inference under Cloe (b) and PhyloWGS (c).

with the mutations of subclone 3 ($\mathcal{M}_3$ with a cellular prevalence of 0.470). Here we abuse the notation slightly and let $\mathcal{M}_c$ denote the new mutations that subclone c gained. Therefore, Cloe infers that $\mathcal{M}_2$ and $\mathcal{M}_3$ belong to the same subclone ($\mathcal{M}_2^{\text{Cloe}} \approx \mathcal{M}_2 \cup \mathcal{M}_3$ and $\mathcal{M}_3^{\text{Cloe}} \approx \mathcal{M}_4$). Similarly, PhyloWGS clusters $\mathcal{M}_2$ and $\mathcal{M}_3$ together. Using more informative mutation pairs data, PairCloneTree is able to identify that $\mathcal{M}_2$ and $\mathcal{M}_3$ belong to different subclones. The comparisons support the argument in Section 3.1 that the inclusion of phasing information from the paired-end read data increases statistical power in recovering the underlying structure. Note that PairCloneTree is based on a different sampling model and has a very different representation of $\boldsymbol{Z}$. Therefore, there is no obvious way to quantify the three model's performance under the same scale.

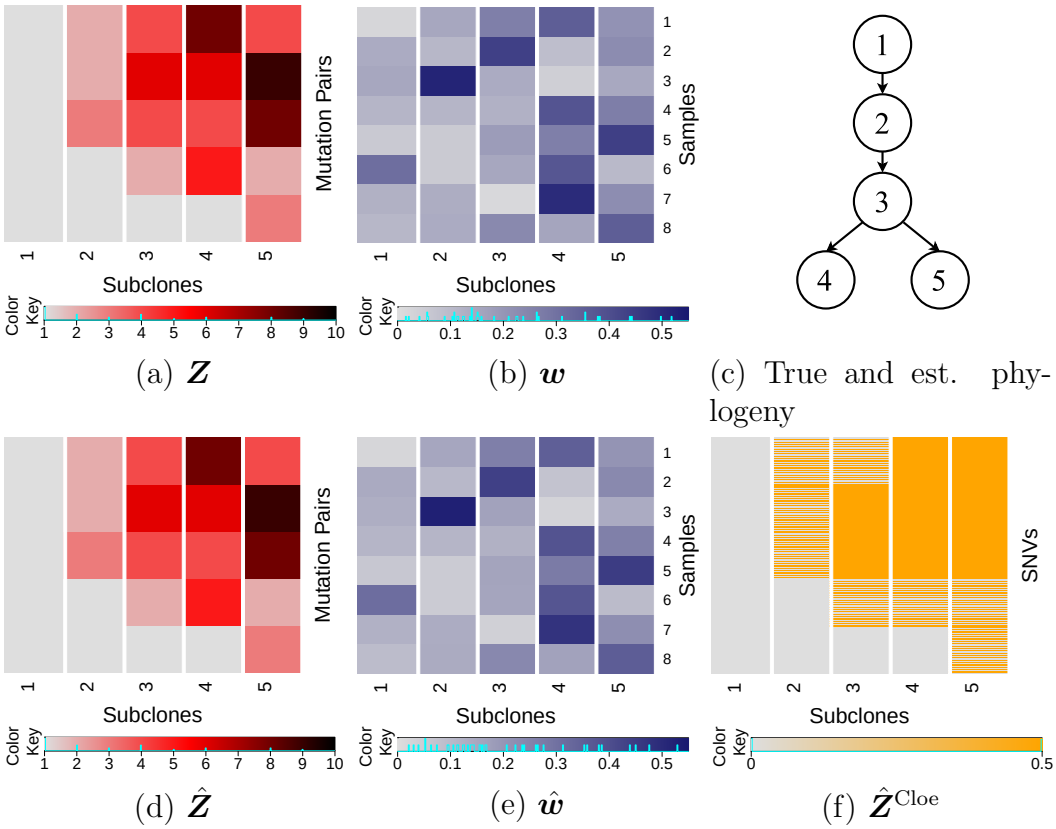
3.4.3 Simulation 2

In the second simulation, we evaluate the performance of the proposed approach on multiple samples. We still consider $K = 100$ mutation pairs, but with a more complicated subclone structure, $C = 5$. We generate hypothetical data for $T = 8$ samples. The subclone proportions in each sample t are generated from $\boldsymbol{w}_t \sim \text{Dir}(0.01, \sigma(25, 15, 10, 8, 5))$. Fig. 3.5(a, b, c) show simulation truth $\boldsymbol{Z}$, $\boldsymbol{w}$ and the phylogeny, respectively. We show $\boldsymbol{w}$ in a heatmap with light gray to deep blue scale. A darker blue color indicates higher abundance of a subclone in a sample, while a lighter gray color indicates lower abundance. The proportions of the background subclone w_{t0}'s are not shown as they only take tiny values, $w_{t0} < 10^{-3}$. The average sequence depth for the eight samples was about 500x.

The hyperparameters are set to be the same as in simulation 1. We run the same number of MCMC iterations.

The true phylogeny is recovered with 100% posterior probability (Fig. 3.5(c)). Fig. 3.5 (d, e) show the estimated genotypes $\hat{\boldsymbol{Z}}$ and subclone proportions $\hat{\boldsymbol{w}}$. The truth is exactly recovered. The simulation shows that with more information from eight samples inference becomes quite reliable.

For comparison we again run Cloe and PhyloWGS on this data. Cloe correctly infers the number of subclones, and the estimated subclone genotypes match the truth, shown in Fig. 3.5 (f). However, Cloe infers the phylogeny as $1 \to 2 \to 3 \to 4 \to 5$. On the other hand, PhyloWGS infers the phylogeny



(a) $\boldsymbol{Z}$ (b) $\boldsymbol{w}$ (c) True and est. phylogeny

(d) $\hat{\boldsymbol{Z}}$ (e) $\hat{\boldsymbol{w}}$ (f) $\hat{\boldsymbol{Z}}^{\text{Cloe}}$

Figure 3.5: Simulation 2. Simulation truth $\boldsymbol{Z}$ (a), $\boldsymbol{w}$ (b) and phylogeny (c), and posterior inference under PairCloneTree (c, d, e) and Cloe (f).

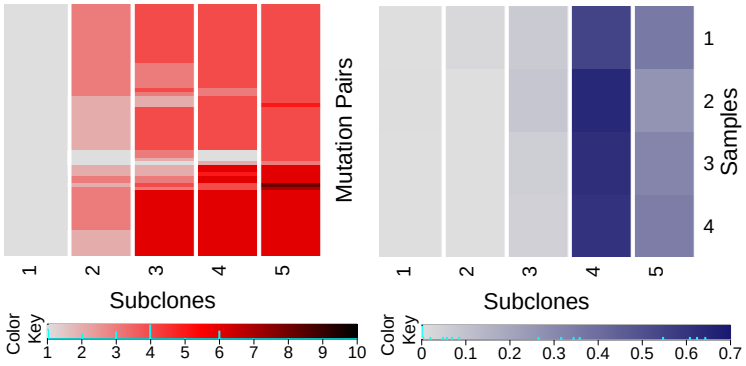
as $0 \to 1 \begin{matrix} \to 2 \\ \to 3 \end{matrix}$ (details not shown). Both methods approximate but still miss some detail in the simulation truth.

3.5 Lung Cancer Data

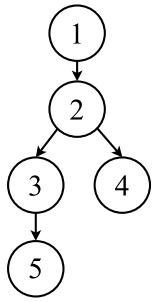
We use whole-exome sequencing (WES) data generated from four ($T = 4$) surgically dissected tumor samples taken from a single patient diagnosed

with lung adenocarcinoma. DNA is extracted from all four samples and exome library is sequenced on an Illumina HiSeq 2000 platform in paired-end fashion. Each of the read is 100 base-pair long and coverage is 200x-400x. We use BWA (Li and Durbin, 2009) and GATK's UniformGenotyper (McKenna et al., 2010) for mapping and variant calling, respectively. In order to find mutation pair location along with their genotypes with number of reads supporting them, we use a bioinformatics tool called `LocHap` (Sengupta et al., 2016). This tool searches for two or three SNVs that are scaffolded by the same reads. When the scaffolded SNVs, known as local haplotypes, exhibits more than two haplotypes, it is known as local haplotype variant (LHV). Using the individual BAM and VCF files `LocHap` finds a few hundreds LHVs on average in a WES sample. We select LHVs with two SNVs as we are interested in mutation pairs only. On those LHVs, we run the bioinformatics filters suggested by `LocHap` to keep the mutation pairs with high calling quality. We focus our analysis in copy number neutral regions. In the end, we get 69 mutation pairs for the sample and record the read count data from `LocHap`'s output.

We use the same hyperparameters and MCMC setting as in the simulations. Fig. 3.6 (d) shows some of the the posterior probabilities of the subclone phylogeny. The posterior mode is shown in Fig. 3.6 (c) with $C = 5$ subclones. Fig. 3.6 (a, b) show the estimated subclone genotypes $\hat{\boldsymbol{Z}}$ and cellular proportions $\hat{\boldsymbol{w}}$, respectively ($\hat{w}_{t0} < 4 \times 10^{-3}$ and are not shown). The rows for $\hat{\boldsymbol{Z}}$ are reordered for better display. The cellular proportions of the subclones show strong similarity across the 4 samples, indicating homogeneity



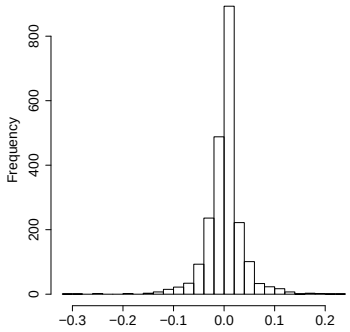
(a) $\hat{\boldsymbol{Z}}$ (b) $\hat{\boldsymbol{w}}$



(c) Estimated phylogeny

Tree topology			
posterior probability	0.79	0.19	0.01

(d) Posterior prob. of tree topology



(e) Histogram of $(\hat{p}_{tkg} - \bar{p}_{tkg})$

Figure 3.6: Posterior inference with PairCloneTree for lung cancer data set.

of the samples. This is expected as the samples are dissected from proximal sites. Subclone 1, which is the normal subclone, takes a small proportion in all 4 samples, indicating high purity of the tumor samples. Subclones 2 and 3 are also included in only small proportions. They have almost vanished in the samples. However, as parents of subclones 4 and 5, respectively, they are important for the reconstruction of the subclone phylogeny. Subclones 4 and 5 are the two main subclones. They share a large proportion of common muta-

tions, but each one has some private mutations, consistent with the estimated tree. Finally, Fig. 3.6 (e) shows a histogram of residuals, where we calculate empirical values $\bar{p}_{tkg} = n_{tkg}/N_{tk}$ and plot the difference $(\hat{p}_{tkg} - \bar{p}_{tkg})$. The residuals are centered at zero with little variation, indicating a good model fit.

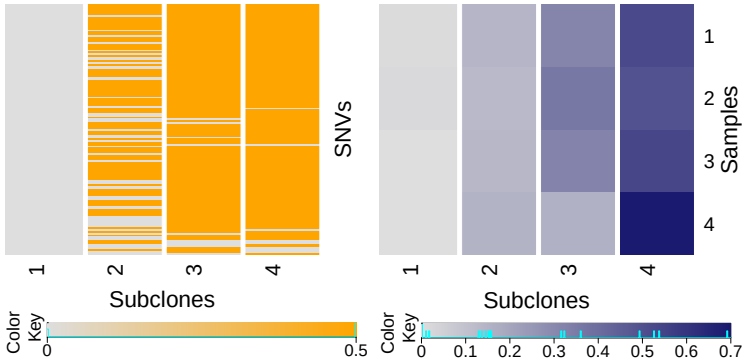
For comparison, we run Cloe and PhyloWGS on the same data set with default hyperparameters. Cloe infers four subclones with phylogeny $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$. Fig. 3.7 (a, b) show the estimated genotypes $\hat{\boldsymbol{Z}}^{\text{Cloe}}$ and cellular proportions $\hat{\boldsymbol{w}}^{\text{Cloe}}$, respectively. PhyloWGS estimates 6 clusters (and a cluster 0 for normal subclone) of the SNVs with phylogeny

$$0 \rightarrow 1 \rightarrow 2 \begin{array}{cccc} \rightarrow & 3 & \rightarrow & 4 \\ \rightarrow & 5 & \rightarrow & 6 \end{array}.$$

Fig. 3.7 (c) summarizes the cluster sizes and cellular prevalences. In light of the earlier simulation studies we believe that the inference under PairCloneTree is more reliable. Cloe and PhyloWGS outputs confirm that the four samples have similar proportions of all the subclones, indicating little inter-sample heterogeneity. Also, Cloe and PhyloWGS infer very small normal cell proportion, which corroborates PairCloneTree's finding that the tumor samples have high purity.

3.6 Discussion

In this work, using a treed LFAM we infer subclonal genotypes structure for mutation pairs, their cellular proportions and the phylogenetic relationship among subclones. This is the first attempt to generate a subclonal phylogenetic



(a) $\hat{\boldsymbol{Z}}^{\text{Cloe}}$ (b) $\hat{\boldsymbol{w}}^{\text{Cloe}}$

Clusters (sizes) \ Samples	1	2	3	4
0 (0)	1.0	1.0	1.0	1.0
1 (93)	0.977	0.990	0.989	0.980
2 (32)	0.854	0.816	0.832	0.852
3 (6)	0.532	0.431	0.473	0.596
4 (3)	0.433	0.408	0.445	0.386
5 (3)	0.252	0.180	0.229	0.152
6 (1)	0.112	0.108	0.056	0.127

(c) Cellular prevalence (PhyloWGS)

Figure 3.7: Posterior inference with Cloe (a, b) and PhyloWGS (c) for lung cancer data set.

structure using mutation pair data. We show that more accurate inference can be obtained using mutation pairs data compared to using only marginal counts for single SNVs. The model can be easily extended to incorporate more than two SNVs. Another way of extending the model is to encode mutation times inside the length of the edges of phylogenetic tree.

The major motivation for accurate estimation of heterogeneity in tumor is personalized medicine. The next step towards this goal is to use varying

estimates of subclonal proportions across patients to drive adaptive treatment allocation.

Currently the heterogeneity is measured mostly with SNV and CNA data. However, structural variants (SVs) like deletion, duplication, inversion, translocation and other large genome rearrangement arguably provide more accurate (Fan et al., 2014) VAF estimation, which is the key input for characterizing the heterogeneity. Therefore incorporation of SVs into the current model could significantly improve the outcome of tumor heterogeneity analysis. Recently, in Brocks et al. (2014) the authors attempted to explain the intratumor heterogeneity in DNA methylation and copy-number pattern by a unified evolutionary process. So the current genome centric definition of tumor heterogeneity could be extended by incorporation of methylation, DNA mutation, and RNA expression data in an integromics model.

Finally in the era of big data it is important to factor computation into the research effort, and build efficient computational models that could handle millions of SNVs. Linear response variational Bayes (Giordano et al., 2015) or MAD-Bayes (Broderick et al., 2013, Xu et al., 2015) methods could be considered as alternative computational strategies to tackle the problem.

Chapter 4

A Nonparametric Bayesian Approach to Dropout in Longitudinal Studies with Auxiliary Covariates

We develop a nonparametric Bayesian approach to missing outcome data in longitudinal studies in the presence of auxiliary covariates. In the presence of auxiliary covariates, we consider a joint model for the full data response, missingness and auxiliary covariates. We include auxiliary covariates to "move" the missingness "closer" to missing at random (MAR). In particular, we specify a nonparametric Bayesian model for the observed data via Gaussian process priors and Bayesian additive regression trees. These model specifications allow us to capture non-linear and non-additive effects, in contrast to existing parametric methods. We then separately specify the conditional distribution of the missing data response given the observed data response, missingness and auxiliary covariates (i.e. the extrapolation distribution) using identifying restrictions. We introduce meaningful sensitivity parameters that allow for a simple sensitivity analysis. Informative priors on those sensitivity parameters can be elicited from subject-matter experts. We use Monte Carlo integration to compute the full data estimands. Performance of our approach is assessed using simulated datasets. Our methodology is motivated by, and

applied to, data from a clinical trial on treatments for schizophrenia.

4.1 Introduction

In longitudinal clinical studies, the research objective is often to make inference on a subject's full data response conditional on covariates that are of primary interest; for example, to calculate the treatment effect of a test drug at the end of a study. The vector of responses for a research subject is often incomplete due to dropout. Dropout is typically non-ignorable (Rubin, 1976, Daniels and Hogan, 2008) and in such cases the joint distribution of the full data response and missingness needs to be modeled. In addition to the covariates that are of primary interest, we would often have access to some *auxiliary covariates* (often collected at baseline) that are not desired in the model for the primary research question. Such variables can often provide information about the missing responses and missing data mechanism. For example, missing at random (MAR) (Rubin, 1976) might only hold conditionally on auxiliary covariates (Daniels and Hogan, 2008). In this setting, auxiliary covariates should be incorporated in the joint model as well, but we should proceed with inference unconditional on these auxiliary covariates.

The full data distribution can be factored into the observed data distribution and the extrapolation distribution (Daniels and Hogan, 2008). The observed data distribution can be identified by the observed data, while the extrapolation distribution cannot. Identifying the extrapolation distribution relies on untestable assumptions such as parametric models for the full data

distribution or identifying restrictions (Linero and Daniels, 2017). Such assumptions can be indexed by unidentified parameters called *sensitivity parameters* (Daniels and Hogan, 2008). The observed data do not provide any information to estimate the sensitivity parameters. Under the Bayesian paradigm, informative priors can be elicited from subject-matter experts and be placed on those sensitivity parameters. Finally, it is desirable to conduct a *sensitivity analysis* (Daniels and Hogan, 2008, National Research Council, 2011) to assess the sensitivity of inferences to such assumptions. The inclusion of auxiliary covariates can ideally reduce the extent of sensitivity analysis that is needed for drawing accurate inferences.

In this paper, we propose a Bayesian nonparametric model for the joint distribution of the full data response, missingness and auxiliary covariates. We use identifying restrictions to identify the extrapolation distribution and introduce sensitivity parameters that are meaningful to subject-matter experts and allow for a simple sensitivity analysis.

4.1.1 Missing Data in Longitudinal Studies

Literature about longitudinal missing data with non-ignorable dropout can be mainly divided into two categories: likelihood-based and moment-based (semiparametric). Likelihood-based approaches include selection models (e.g. Heckman, 1979, Diggle and Kenward, 1994, Molenberghs et al., 1997), pattern mixture models (e.g. Little, 1993, 1994, Hogan and Laird, 1997) and shared-parameter models (e.g. Wu and Carroll, 1988, Follmann and Wu, 1995, Pulk-

stenis et al., 1998, Henderson et al., 2000). These three types of models differ from how the joint distribution of the response and missingness is factorized. Likelihood-based approaches often make strong parametric model assumptions to identify the full data distribution. For a comprehensive review see, for example, Daniels and Hogan (2008) or Little and Rubin (2014). Moment-based approaches, on the other hand, typically specify a semiparametric model for the marginal distribution of the response, and a semiparametric or parametric model for the missingness conditional on the response. Moment-based approaches are in general more robust to model misspecification since they make minimal distributional assumptions. See, for example, Robins et al. (1995), Rotnitzky et al. (1998), Scharfstein et al. (1999), Tsiatis (2007), Tsiatis et al. (2011).

There are several recent papers under the likelihood-based paradigm that are relevant to our approach, such as Wang et al. (2010), Linero and Daniels (2015), Linero (2017), Linero and Daniels (2017). These papers specify Bayesian nonparametric models for the observed data distribution, and thus have similar robustness to semiparametric approaches. However, existing approaches do not utilize information from auxiliary covariates. In the presence of auxiliary covariates, Daniels et al. (2014) model longitudinal binary responses using a parametric model under ignorable missingness. Our goal is to develop a nonparametric Bayesian approach to longitudinal missing data with non-ignorable dropout that also allows for incorporating auxiliary covariates. As mentioned earlier, the reason to include auxiliary covariates is that

we anticipate it will make the missingness "closer" to MAR.

4.1.2 Notation and Terminology

We introduce some notation and terminology as follows. Consider the responses for a subject i at J time points. Let $\boldsymbol{Y}_i = (Y_{i1}, \ldots, Y_{iJ})$ be the vector of longitudinal outcomes that was planned to be collected, $\bar{\boldsymbol{Y}}_{ij} = (Y_{i1}, \ldots, Y_{ij})$ be the history of outcomes through the first j times, and $\tilde{\boldsymbol{Y}}_{ij} = (Y_{i,j+1}, \ldots, Y_{iJ})$ be the future outcomes after time j. Let S_i denote the dropout time or dropout pattern, which is defined as the last time a subject's response is recorded, i.e. $S_i = \max\{j : Y_{ij} \text{ is observed}\}$. Missingness is called *monotone* if Y_{ij} is observed for all $j \leq S_i$, and missingness is called *intermittent* if Y_{ij} is missing for some $j < S_i$. For monotone missingness, S_i captures all the information about missingness. In the following discussion, we will concern ourselves with monotone missingness. Dropout is called *random* (Diggle and Kenward, 1994) if the dropout process only depends on the observed responses, i.e. the missing data are MAR; dropout is called *informative* if the dropout process also depends on the unobserved responses, i.e. the missing data are missing not at random (MNAR). We denote by $\boldsymbol{X}_i$ the covariates that are of primary interest, and $\boldsymbol{V}_i = (V_{i1}, \ldots, V_{iQ})$ the Q auxiliary covariates that are not of primary interest. Those auxiliary covariates should be related to the outcome and missingness. The observed data for subject i is $(\bar{\boldsymbol{Y}}_{iS_i}, S_i, \boldsymbol{V}_i, \boldsymbol{X}_i)$, and the full data is $(\boldsymbol{Y}_i, S_i, \boldsymbol{V}_i, \boldsymbol{X}_i)$. In general, we are interested in expectation of the form $E[t(\boldsymbol{Y}_i) \mid \boldsymbol{X}_i]$, where t denotes some functional of $\boldsymbol{Y}_i$. Finally, denote by

$p(\boldsymbol{y}, s, \boldsymbol{v} \mid \boldsymbol{x}, \boldsymbol{\omega})$ the joint model for the full data response, missingness and auxiliary covariates conditional on the covariates that are of primary interest, where $\boldsymbol{\omega}$ represents the parameter vector.

4.1.3 The Schizophrenia Clinical Trial

Our work is motivated by a multicenter, randomized, double-blind clinical trial on treatments for schizophrenia. For this clinical trial, the longitudinal outcomes are the positive and negative syndrome scale (PANSS) scores (Kay et al., 1987). The outcomes are collected at $J = 6$ time points corresponding to baseline, day 4 after baseline, and weeks 1, 2, 3 and 4 after baseline. The possible dropout patterns are $S_i = 2, 3, 4, 5, 6$. The covariate of primary interest is treatment, with $X_i = 1, 2$ or 3 corresponding to test drug, active control or placebo, respectively. In addition, we have access to $Q = 7$ auxiliary covariates including age, onset (of schizophrenia) age, height, weight, country, sex and education level.

The dataset consists of $N = 204$ subjects, with 45 subjects for the active control arm, 78 subjects for the placebo arm, and 81 subjects for the test drug arm. Figure 4.1 shows the individual trajectories and mean responses over time for the three treatment arms. The dropout rates are 33.3%, 20.0% and 25.6% for the test drug, active control and placebo arms, respectively. Table 4.1 shows the detailed dropout rates for each dropout pattern. Subjects drop out for a variety of reasons. Some reasons including adverse events (e.g. occurrence of side effects), pregnancy and protocol violation are thought to

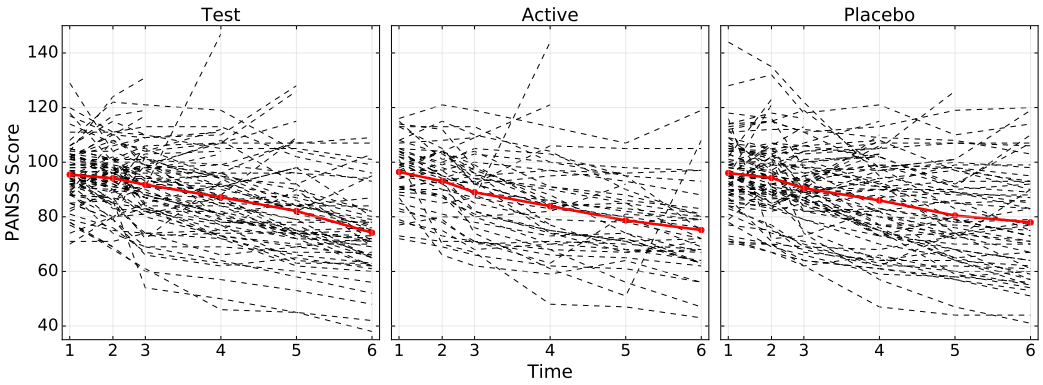


Figure 4.1: Trajectories of individual responses (dashed black lines) and mean responses (thick red lines) over time for the active control, placebo and test drug arms.

be random dropouts, while the other reasons such as disease progression, lack of efficacy, physician decision and withdraw by patient are thought to be informative dropouts. It is ideal to treat those reasons differently while making inference. Table 4.1 shows the informative dropout rates for each dropout pattern. For the test, active and placebo arms, the percentages of informative dropouts among all dropouts are 88.9%, 77.8% and 100.0%, respectively. The dataset has a few intermittent missing outcomes (1 for the test drug arm, 1 for the active control arm, and 2 for the placebo arm). We focus our study on monotone missingness and assume partial ignorability (Harel and Schafer, 2009) for the few intermittent missing outcomes.

The goal of this study is to estimate the change from baseline treatment effect,

$$r_x = \mathrm{E}[Y_{i6} - Y_{i1} \mid X_i = x].$$

	$S_i = 2$	$S_i = 3$	$S_i = 4$	$S_i = 5$	Overall
Test	4.9 (3.7)	12.3 (9.9)	8.6 (8.6)	7.4 (7.4)	33.3 (29.6)
Active	2.2 (2.2)	4.4 (2.2)	8.9 (6.7)	4.4 (4.4)	20.0 (15.6)
Placebo	3.8 (3.8)	5.1 (5.1)	11.5 (11.5)	5.1 (5.1)	25.6 (25.6)

Table 4.1: Dropout rates (%) for different dropout patterns in the three treatment arms, with informative dropout rates in parentheses.

In particular, the treatment effect improvements over placebo, i.e. $r_1 - r_2$ and $r_3 - r_2$, are of interest. This dataset was previously analyzed in Linero and Daniels (2015), which took a Bayesian nonparametric approach as well, but did not utilize information from the auxiliary covariates.

4.1.4 Overview

We stratify the model by treatment, and suppress the treatment variable x to simplify notation hereafter. The extrapolation factorization (Daniels and Hogan, 2008) is

$$p(\boldsymbol{y}, s, \boldsymbol{v} \mid \boldsymbol{\omega}) = p(\tilde{\boldsymbol{y}}_s \mid \bar{\boldsymbol{y}}_s, s, \boldsymbol{v}, \boldsymbol{\omega}_E) p(\bar{\boldsymbol{y}}_s, s, \boldsymbol{v} \mid \boldsymbol{\omega}_O),$$

where the extrapolation distribution, $p(\tilde{\boldsymbol{y}}_s \mid \bar{\boldsymbol{y}}_s, s, \boldsymbol{v}, \boldsymbol{\omega}_E)$, is not identified by the data in the absence of uncheckable assumptions or constraints on the parameter space. The observed data distribution $p(\bar{\boldsymbol{y}}_s, s, \boldsymbol{v} \mid \boldsymbol{\omega}_O)$ is identified and can be estimated nonparametrically. We factorize the observed data distribution based on pattern-mixture modeling (Little, 1993),

$$p(\bar{\boldsymbol{y}}_s, s, \boldsymbol{v} \mid \boldsymbol{\omega}_O) = p(\bar{\boldsymbol{y}}_s \mid s, \boldsymbol{v}, \boldsymbol{\pi}) p(s \mid \boldsymbol{v}, \boldsymbol{f}) p(\boldsymbol{v} \mid \boldsymbol{\eta}), \tag{4.1}$$

where we assume distinct parameters $\boldsymbol{\omega}_O = (\boldsymbol{\pi}, \boldsymbol{f}, \boldsymbol{\eta})$ parametrizing the response model, the missingness and the distribution of the auxiliary covariates, respectively.

The remainder of this article is structured as follows. In Section 4.2 we specify Bayesian (nonparametric) models for (4.1). In Section 4.3, we use identifying restrictions to identify the extrapolation distribution. In Section 4.4, we describe our posterior inference and computation approaches. In Section 4.5, we present simulation studies to validate our model and compare with results using other methods. In Section 4.6, we apply our method to a clinical trial on treatments for schizophrenia. We conclude with a discussion in Section 4.7.

4.2 Probability Model for the Observed Data

The model specification (4.1) brings two challenges:

1. For the models $p(\bar{\boldsymbol{y}}_s \mid s, \boldsymbol{v}, \boldsymbol{\pi})$ and $p(s \mid \boldsymbol{v}, \boldsymbol{f})$, it is unclear how the auxiliary covariates are related to the responses and dropout patterns. For example, the auxiliary covariates contain height and weight, which might not have a linear and additive effect on the responses. For example, the responses might have a linear relationship with the body mass index, which is calculated by weight/height^2.

2. For the model $p(\bar{\boldsymbol{y}}_s \mid s, \boldsymbol{v}, \boldsymbol{\pi})$, the observed patterns are sparse. For example, the dropout pattern $S_i = 2$ for the active control arm has only 1

observation.

To mitigate challenge 1, we specify nonparametric models for $p(\bar{\boldsymbol{y}}_s \mid s, \boldsymbol{v}, \boldsymbol{\pi})$ and $p(s \mid \boldsymbol{v}, \boldsymbol{f})$ via Gaussian process priors and Bayesian additive regression trees. Such models are highly flexible and robust to model misspecification. To address challenge 2, we utilize informative priors such as autoregressive (AR) and conditional autoregressive (CAR) priors to share information across neighboring patterns. Detailed model specifications are as follows.

4.2.1 Model for the Observed Data Responses Conditional on Pattern and Auxiliary Covariates

We define the model for observed data responses conditional on drop out time and auxiliary covariates, i.e. $p(\bar{\boldsymbol{y}}_s \mid s, \boldsymbol{v}, \boldsymbol{\pi})$, as follows. The distribution $p(\bar{\boldsymbol{y}}_s \mid s, \boldsymbol{v}, \boldsymbol{\pi})$ can be factorized as

$$p_s\left(\bar{\boldsymbol{y}}_s \mid \boldsymbol{v}, \boldsymbol{\pi}\right) = p_s(y_s \mid \bar{\boldsymbol{y}}_{s-1}, \boldsymbol{v}, \boldsymbol{\pi}) \cdots p_s(y_2 \mid \bar{\boldsymbol{y}}_1, \boldsymbol{v}, \boldsymbol{\pi}) p_s(y_1 \mid \boldsymbol{v}, \boldsymbol{\pi}), \tag{4.2}$$

where the subscript s corresponds to conditioning on dropping out pattern $S = s$.

We assume

$$\left(Y_j \mid \bar{\boldsymbol{Y}}_{j-1} = \bar{\boldsymbol{y}}_{j-1}, S = s, \boldsymbol{V} = \boldsymbol{v}, \boldsymbol{\pi}\right) = \begin{cases} a_0(\boldsymbol{v}, j, s) + \varepsilon_{js}, & j = 1; \\ a(\boldsymbol{v}, j, s) + \bar{\boldsymbol{y}}_{j-1}^T \boldsymbol{\Phi}_{js} + \varepsilon_{js}, & j \geq 2; \end{cases} \tag{4.3}$$

where $j = 1, \ldots, s;\ s = 2, \ldots, J$. Here a_0 and a are stochastic processes indexed by $\mathcal{U} = \mathcal{V} \times \mathcal{J}$, where $\mathcal{V}$ is the state space of $\boldsymbol{v}$ and $\mathcal{J} \subset \{1, \ldots, J\}^2$ is

the state space of (j, s). Furthermore, $\mathbf{\Phi}_{js}$ is the vector of lag coefficients for each time/pattern, and ε_{js}'s are Gaussian errors,

$$\varepsilon_{js} \sim N(0, \sigma_{js}^2).$$

We place Gaussian process priors (Rasmussen and Williams, 2006) on a and a_0,

$$\begin{aligned} a_0(\boldsymbol{v}, j, s) &\sim \mathcal{GP}\left[\mu_0(\boldsymbol{v}, j, s),\, C_0(\boldsymbol{v}, j, s; \boldsymbol{v}', j', s')\right]; \\ a(\boldsymbol{v}, j, s) &\sim \mathcal{GP}\left[\mu(\boldsymbol{v}, j, s),\, C(\boldsymbol{v}, j, s; \boldsymbol{v}', j', s')\right], \end{aligned}$$

with mean functions μ_0, μ: $\mathcal{U} \to \mathbb{R}$ and covariance functions C_0, C: $\mathcal{U} \times \mathcal{U} \to \mathbb{R}^+$. Specifically,

$$\begin{aligned} \mu_0(\boldsymbol{v}, j, s) &= \boldsymbol{v}^T \boldsymbol{\beta}_{0s} + b_{js}; \\ \mu(\boldsymbol{v}, j, s) &= \boldsymbol{v}^T \boldsymbol{\beta}_s + b_{js}, \end{aligned} \tag{4.4}$$

and

$$\begin{aligned} C_0(\boldsymbol{v}, j, s; \boldsymbol{v}', j', s') &= \kappa_0^2\, D_0(\boldsymbol{v}, j, s; \boldsymbol{v}', j', s') + \tilde{\kappa}_0^2\, I(\boldsymbol{v}, j, s; \boldsymbol{v}', j', s'); \\ C(\boldsymbol{v}, j, s; \boldsymbol{v}', j', s') &= \kappa^2\, D(\boldsymbol{v}, j, s; \boldsymbol{v}', j', s') + \tilde{\kappa}^2\, I(\boldsymbol{v}, j, s; \boldsymbol{v}', j', s'). \end{aligned} \tag{4.5}$$

We use two different stochastic processes a_0 and a for $j = 1$ and $j \geq 2$. The reason is that for $j = 1$, a_0 represent the mean initial response with no past; for $j \geq 2$, a represents the mean at subsequent thus with a measured past. In the mean functions (4.4), $\boldsymbol{\beta}_{0s}$ and $\boldsymbol{\beta}_s$ are the vectors of regression coefficients of the auxiliary covariates, and b_{js} is the time/pattern specific

intercepts. In the covariance functions (4.5), $D_0(a;b)$ and $D(a;b)$ are the exponential distances between a and b, defined by

$$D_0(\boldsymbol{v}, j, s; \boldsymbol{v}', j', s') = \exp\left[- \frac{\|\tilde{\boldsymbol{v}} - \tilde{\boldsymbol{v}}'\|_2^2}{2h_{v0}^2} - \frac{|\tilde{j} - \tilde{j}'|}{h_{j0}} - \frac{|\tilde{s} - \tilde{s}'|}{h_{s0}} \right],$$
$$D(\boldsymbol{v}, j, s; \boldsymbol{v}', j', s') = \exp\left[- \frac{\|\tilde{\boldsymbol{v}} - \tilde{\boldsymbol{v}}'\|_2^2}{2h_{v}^2} - \frac{|\tilde{j} - \tilde{j}'|}{h_{j}} - \frac{|\tilde{s} - \tilde{s}'|}{h_{s}} \right].$$

Here κ_0^2, h_{v0}, h_{j0}, h_{s0}, $\tilde{\kappa}_0^2$, κ^2, h_v, h_j, h_s, $\tilde{\kappa}^2$ are the hyperparameters. The values $\tilde{\boldsymbol{v}}$, $\tilde{j}$ and $\tilde{s}$ are standardized values for $\boldsymbol{v}$, j and s,

$$\tilde{v}_{iq} = \frac{v_{iq} - \text{mean}(v_{\cdot q})}{\text{sd}(v_{\cdot q})},$$
$$\tilde{j}_i = \frac{j_i - \min(j_\cdot)}{\max(j_\cdot) - \min(j_\cdot)}, \quad \text{and} \quad \tilde{s}_i = \frac{s_i - \min(s_\cdot)}{\max(s_\cdot) - \min(s_\cdot)}.$$

For categorical covariates, the distance between $\boldsymbol{v}$ and $\boldsymbol{v}'$ is calculated by counting the number of different values. In addition, in (4.5), $I(a;b)$ is the Kronecker delta function that takes the value 1 if $a = b$ and 0 otherwise. The function $I(a;b)$ is used to introduce a small nugget for the diagonal covariances, which overcomes near-singularity of the covariance matrices and improves numerical stability. The Gaussian processes flexibly model the relationship between auxiliary covariates and response and accounts for possibly non-linear and non-additive effects.

For the noise variance σ_{js}^2, we assume an inverse Gamma shrinkage prior,

$$\sigma_{js}^2 \mid g_\sigma \overset{iid}{\sim} \text{IG}(\lambda_\sigma, \lambda_\sigma g_\sigma), \quad j = 1, \ldots, s, \ s = 2, \ldots, J,$$

with $\mathrm{E}(1/\sigma_{js}^2) = 1/g_\sigma$ and $\mathrm{Var}(1/\sigma_{js}^2) = 1/\lambda_\sigma g_\sigma^2$. We put hyper-priors on λ_σ and g_σ,

$$\lambda_\sigma - 2 \sim \mathrm{IG}(1,1),$$
$$g_\sigma \sim \mathrm{Gamma}(1,1).$$

Next, we consider the parameters in the mean functions (4.4). We allow the regression coefficients of the auxiliary covariates to vary by pattern. However, it is typical to have sparse patterns. As a result, we consider an informative prior that assumes regression coefficients for neighboring patterns to be similar. In particular, we specify AR(1) type priors on $\boldsymbol{\beta}_{0s}$ and $\boldsymbol{\beta}_s$. For $\boldsymbol{\beta}_s$, we assume

$$\boldsymbol{\beta} \sim N\left[X_\beta \tilde{\boldsymbol{\beta}}, \sigma_\beta^2 \Sigma_\beta(\rho)\right],$$

where

$$\boldsymbol{\beta} = \begin{pmatrix} \boldsymbol{\beta}_2 \\ \boldsymbol{\beta}_3 \\ \vdots \\ \boldsymbol{\beta}_J \end{pmatrix}, \qquad X_\beta = \begin{pmatrix} I \\ I \\ \vdots \\ I \end{pmatrix},$$

and

$$\Sigma_\beta(\rho) = \frac{1}{1-\rho^2} \begin{pmatrix} I & \rho I & \cdots & \rho^{J-2} I \\ \rho I & I & \cdots & \rho^{J-3} I \\ \vdots & \vdots & & \vdots \\ \rho^{J-2} I & \rho^{J-3} I & \cdots & I \end{pmatrix}.$$

The prior on $\boldsymbol{\beta}$ introduces three unknown hyperparameters $\tilde{\boldsymbol{\beta}}$, σ_β^2 and ρ. We specify diffuse normal, inverse Gamma and uniform priors, respectively,

$$\tilde{\boldsymbol{\beta}} \sim N(\mathbf{0}, \delta_\beta^2 I), \quad \sigma_\beta^2 \sim \mathrm{IG}(\lambda_1^\beta, \lambda_2^\beta), \quad \rho \sim \mathrm{Unif}(0,1).$$

Similarly, for $\boldsymbol{\beta}_{0s}$,

$$\boldsymbol{\beta}_0 \sim N\left[X_\beta \tilde{\boldsymbol{\beta}}_0, \sigma^2_{\beta_0}\Sigma_\beta(\rho_0)\right], \quad \text{with hyper-priors}$$
$$\tilde{\boldsymbol{\beta}}_0 \sim N(\mathbf{0}, \delta^2_{\beta_0} I), \quad \sigma^2_{\beta_0} \sim \text{IG}(\lambda_1^{\beta_0}, \lambda_2^{\beta_0}), \quad \rho_0 \sim \text{Unif}(0,1).$$

The time/pattern specific intercepts are given conditional autoregressive (CAR) type priors (Banerjee et al., 2014, De Oliveira, 2012) as we expect them to be similar for neighboring patterns/times. Let $\boldsymbol{b}_0 = (b_{12}, b_{13}, \ldots, b_{1J})$ and $\boldsymbol{b} = (b_{22}; b_{23}, b_{33}; \ldots; b_{2J}, \ldots, b_{JJ})$. The potential neighbors of b_{js} are $\{b_{j-1,s}, b_{j+1,s}, b_{j,s-1}, b_{j,s+1}\}$. Denote by $\mathcal{N}^b_{js} = \{(j', s') : b_{j's'} \text{ is neighbor of } b_{js}\}$ and $N^b_{js} = |\mathcal{N}^b_{js}|$ which is the number of neighbors of b_{js}. The CAR type prior assigns conditional priors on b_{js} given its neighbors, and under several regularity conditions the conditionals indicate a joint distribution. In particular, we assume

$$b_{js} \mid b_{-js} \sim N\left(\tilde{b} + \sum_{j's' \in \mathcal{N}^b_{js}} \frac{\gamma_b}{N^b_{js}}\left(b_{j's'} - \tilde{b}\right), \frac{\sigma^2_b}{N^b_{js}}\right),$$

which induces a joint prior on $\boldsymbol{b}$ of the form

$$\boldsymbol{b} \sim N\left(\mathbf{1}\tilde{b}, \sigma^2_b(I - \gamma_b W_b)^{-1}\mathcal{N}_b\right),$$

where

$$(W_b)_{jsj's'} = \begin{cases} 1/N^b_{js}, & \text{if } (j,s) \text{ and } (j',s') \text{ are neighbors;} \\ 0, & \text{otherwise,} \end{cases}$$

$\mathcal{N}_b = \text{diag}(1/N^b_{js})$, $\tilde{b}$ is a mean parameter for $\boldsymbol{b}$, σ^2_b is a variance parameter and γ_b is a spatial dependence parameter. Let $\left(e^b_1\right)^{-1}$ and $\left(e^b_2\right)^{-1}$ denote the max

and min eigenvalues of W_b. To guarantee that $I - \gamma_b W_b$ is positive definite, γ_b is required to belong to (e_2^b, e_1^b). Furthermore, it is not unreasonable to assume the spatial correlation is positive, i.e. $0 < \gamma_b < e_1^b$. We put hyper-priors on $\tilde{b}$, σ_b^2 and γ_b,

$$\tilde{b} \sim N(0, \delta_b^2), \quad \sigma_b^2 \sim \text{IG}(\lambda_1^b, \lambda_2^b), \quad \gamma_b \sim \text{Unif}(0, e_1^b).$$

Similarly, for $\boldsymbol{b}_0$, we assume

$$\begin{aligned} &\boldsymbol{b}_0 \sim N\left(\mathbf{1}\tilde{b}_0, \sigma_{b_0}^2 (I - \gamma_{b_0} W_{b_0})^{-1} \mathcal{N}_{b_0}\right); \quad \text{with hyper-priors} \\ &\tilde{b}_0 \sim N(0, \delta_{b_0}^2), \quad \sigma_{b_0}^2 \sim \text{IG}(\lambda_1^b, \lambda_2^b), \quad \gamma_{b_0} \sim \text{Unif}(0, e_1^{b_0}). \end{aligned}$$

We then consider the parameters in the covariance functions (4.5). We put inverse Gamma priors on κ_0^2 and κ^2,

$$\kappa_0^2 \sim \text{IG}(\lambda_1^{\kappa_0}, \lambda_2^{\kappa_0}), \quad \kappa^2 \sim \text{IG}(\lambda_1^{\kappa}, \lambda_2^{\kappa}).$$

For simplicity, we fix the length scales h_{v0}, h_{j0}, h_{s0}, h_v, h_j and h_s. For example, in practice, we set $h_{v0}^2 = h_v^2 = Q$ to introduce moderate correlation between the responses of two subjects with similar $\boldsymbol{V}$'s; we set $h_{j0} = h_j = 5$, $h_{s0} = h_s = 5$ to introduce strong correlation between the responses of one subject measured at two different time points. We also fix $\tilde{\kappa}_0^2$ and $\tilde{\kappa}^2$ at small values, e.g. $\tilde{\kappa}_0^2 = \tilde{\kappa}^2 = 0.01$.

We complete the model with a prior for the lag coefficients. For each time/pattern, we break $\boldsymbol{\Phi}_{js}$ into three parts: $\boldsymbol{\Phi}_{js} = (\boldsymbol{\phi}_{3js}, \phi_{2js}, \phi_{1js})$, ϕ_{1js}, ϕ_{2js} and $\boldsymbol{\phi}_{3js}$ correspond to lag-1 response, lag-2 response and higher-order lag

responses, respectively. We put a CAR type prior on $\boldsymbol{\phi}_1 = \{\phi_{1js}\}$, similar to the priors on b_{js},

$$\boldsymbol{\phi}_1 \sim N\left(\mathbf{1}\tilde{\phi}_1, \sigma^2_{\phi_1}(I - \gamma_{\phi_1} W_{\phi_1})^{-1}\mathcal{N}_{\phi_1}\right); \quad \text{with hyper-priors}$$
$$\tilde{\phi}_1 \sim N(1, \delta^2_{\phi_1}), \quad \sigma^2_{\phi_1} \sim \text{IG}(\lambda_1^{\phi_1}, \lambda_2^{\phi_1}), \quad \text{and } \gamma_{\phi_1} \sim \text{Unif}(0, e_1^{\phi_1}).$$

For ϕ_{2js} and $\boldsymbol{\phi}_{3js}$ we simply put normal priors with more prior mass around 0 to indicate the prior belief that higher-order lags have less impact on current response. Specifically,

$$\phi_{2js} \sim N(0, \sigma^2_{\phi_2}), \qquad \sigma^2_{\phi_2} \sim \text{IG}(\lambda_1^{\phi_2}, \lambda_2^{\phi_2});$$
$$\boldsymbol{\phi}_{3js} \sim N(0, \sigma^2_{\phi_3} I), \quad \sigma^2_{\phi_3} \sim \text{IG}(\lambda_1^{\phi_3}, \lambda_2^{\phi_3}),$$

with $\lambda_1^{\phi_2} > \lambda_2^{\phi_2}$ and $\lambda_1^{\phi_3} > \lambda_2^{\phi_3}$.

4.2.2 Model for the Pattern Conditional on Auxiliary Covariates

We model the hazard of dropout at time j with Bayesian additive regression trees (BART) (Chipman et al., 2010),

$$p(S = j \mid S \geq j, \boldsymbol{v}, \boldsymbol{f}) = F_N(f_j(\boldsymbol{v})),$$

where F_N denotes the standard normal cdf (probit link), and $f_j(\boldsymbol{v})$ is the sum of tree models from BART. The BART model captures complex relationships between auxiliary covariates and dropout including interactions and nonlinearities. We use the default priors for $f_j(\cdot)$ given in Chipman et al. (2010).

4.2.3 Model for the Auxiliary Covariates

We use a Bayesian bootstrap (Rubin, 1981) prior for the distribution for $\boldsymbol{v}$. Suppose $\boldsymbol{v}$ can only take the N discrete values that we observed, $\boldsymbol{V} \in \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N\}$. The probability for each is

$$p(\boldsymbol{V} = \boldsymbol{v}_i \mid \boldsymbol{\eta}) = \eta_i, \tag{4.6}$$

where $\sum_{i=1}^{N} \eta_i = 1$. We place a Dirichlet distribution prior on $\boldsymbol{\eta}$,

$$(\eta_1, \ldots, \eta_N) \sim \text{Dir}(d_\eta, \ldots, d_\eta).$$

4.3 The Extrapolation distribution

The extrapolation distribution for our setting can be sequentially factorized as

$$p_s(\tilde{\boldsymbol{y}}_s \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v}, \boldsymbol{\omega}_E) = p_s(y_{s+1} \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v}, \boldsymbol{\omega}_E) \cdot p_s(y_{s+2} \mid \bar{\boldsymbol{y}}_{s+1}, \boldsymbol{v}, \boldsymbol{\omega}_E) \cdots p_s(y_J \mid \bar{\boldsymbol{y}}_{J-1}, \boldsymbol{v}, \boldsymbol{\omega}_E). \tag{4.7}$$

The extrapolation distribution is not identified by the observed data. To identify the extrapolation distribution, we use identifying restrictions that express the extrapolation distribution as a function of the observed data distribution; see Linero and Daniels (2017) for a comprehensive discussion. For example, missing at random (MAR) (Rubin, 1976) is a joint identifying restriction that completely identifies the extrapolation distribution. It is shown in Molenberghs et al. (1998) that MAR is equivalent to the available case missing

value (ACMV) restriction in the pattern mixture model framework. The same statement is true when conditional on $\boldsymbol{V}$, in which case MAR is referred to as auxiliary variable MAR (A-MAR) (Daniels and Hogan, 2008). ACMV sets

$$p_k(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v}, \boldsymbol{\omega}_E) = p_{\geq j}(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v}, \boldsymbol{\pi}),$$

for $k < j$ and $2 \leq j < J$, where the subscript $\geq j$ corresponds to conditioning on $S \geq j$.

When the missingness is not at random, a partial identifying restriction (Linero and Daniels, 2017) is the missing non-future dependence (NFD) assumption (Kenward et al., 2003). NFD states that the probability of dropout at time j depends only on $\bar{\boldsymbol{y}}_{j+1}$. Similarly, when conditional on $\boldsymbol{V}$, auxiliary variable NFD (A-NFD) sets

$$p(S = j \mid \bar{\boldsymbol{y}}_J, \boldsymbol{v}, \boldsymbol{\omega}) = p(S = j \mid \bar{\boldsymbol{y}}_{j+1}, \boldsymbol{v}, \boldsymbol{\omega}).$$

Within the pattern-mixture framework, NFD is equivalent to the non-future missing value (NFMV) restriction (Kenward et al., 2003). Under A-NFD, we have

$$p_k(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v}, \boldsymbol{\omega}_E) = p_{\geq j-1}(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v}, \boldsymbol{\pi}), \tag{4.8}$$

for $k < j-1$ and $2 < j \leq J$. NFMV leaves one conditional distribution per incomplete pattern unidentified: $p_s(y_{s+1} \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v})$. To identify $p_s(y_{s+1} \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v})$, we assume a location shift τ_{s+1} (Daniels and Hogan, 2000),

$$\left[Y_{s+1} \mid \bar{\boldsymbol{Y}}_s, S = s, \boldsymbol{V}, \boldsymbol{\omega}\right] \stackrel{\mathrm{d}}{=} \left[Y_{s+1} + \tau_{s+1} \mid \bar{\boldsymbol{Y}}_s, S \geq s+1, \boldsymbol{V}, \boldsymbol{\omega}\right], \tag{4.9}$$

where $\stackrel{\mathrm{d}}{=}$ denotes equality in distribution, and τ_{s+1} measures the deviation of the unidentified distribution $p_s(y_{s+1} \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v})$ from ACMV. In particular, ACMV holds when $\tau_{s+1} = 0$; τ_{s+1} is a *sensitivity parameter* (Daniels and Hogan, 2008). To help calibrate the magnitude of τ_{s+1}, we set

$$\left[\tau_{s+1} \mid \bar{Y}_s = \bar{\boldsymbol{y}}_s, \boldsymbol{V} = \boldsymbol{v}\right] = \tilde{\tau} \cdot \Delta_{s+1}(\bar{\boldsymbol{y}}_s, \boldsymbol{v}), \tag{4.10}$$

where $\Delta_{s+1}(\bar{\boldsymbol{y}}_s, \boldsymbol{v})$ is the standard deviation of $p_s(y_{s+1} \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v})$ under ACMV, and $\tilde{\tau}$ represents the number of standard deviations that $p_s(y_{s+1} \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v})$ is deviated from ACMV. Importantly, note that, based on the calibration, for a fixed $\tilde{\tau}$ we would have a smaller Δ using auxiliary covariates and thus a smaller deviation from ACMV, in comparison to unconditional on $\boldsymbol{V}$.

4.4 Posterior Inference and Computation

4.4.1 Posterior Sampling for Observed Data Model Parameters

We use a Markov chain Monte Carlo (MCMC) algorithm to draw samples from the posterior $\boldsymbol{w}_O^{(l)} \overset{iid}{\sim} p(\boldsymbol{w}_O \mid \{\bar{\boldsymbol{y}}_{is_i}, s_i, \boldsymbol{v}_i\}_{i=1}^N)$, $l = 1, \ldots, L$. Note that we use distinct parameters $\boldsymbol{\pi}, \boldsymbol{f}, \boldsymbol{\eta}$ for $p(\bar{\boldsymbol{y}}_s \mid s, \boldsymbol{v}, \boldsymbol{\pi})$, $p(s \mid \boldsymbol{v}, \boldsymbol{f})$ and $p(\boldsymbol{v} \mid \boldsymbol{\eta})$, and the parameters are also a priori independent, $p(\boldsymbol{\pi}, \boldsymbol{f}, \boldsymbol{\eta}) = p(\boldsymbol{\pi})p(\boldsymbol{f})p(\boldsymbol{\eta})$. Therefore, the posterior distribution of $\boldsymbol{w}_O$ can be factored as

$$p\left(\boldsymbol{w}_O \mid \{\bar{\boldsymbol{y}}_{is_i}, s_i, \boldsymbol{v}_i\}_{i=1}^N\right) = p\left(\boldsymbol{\pi} \mid \{\bar{\boldsymbol{y}}_{is_i}, s_i, \boldsymbol{v}_i\}_{i=1}^N\right) p\left(\boldsymbol{f} \mid \{s_i, \boldsymbol{v}_i\}_{i=1}^N\right) p\left(\boldsymbol{\eta} \mid \{\boldsymbol{v}_i\}_{i=1}^N\right),$$

and posterior simulation can be conducted independently for $\boldsymbol{\pi}$, $\boldsymbol{f}$ and $\boldsymbol{\eta}$. Gibbs transition probabilities are used to update $\boldsymbol{\pi}$ (details in Appendix B.1),

the R package `BayesTree` (Chipman and McCulloch, 2016) is used to update $\boldsymbol{f}$, and $\boldsymbol{\eta}$ is updated by directly sampling from its posterior $\boldsymbol{\eta} \mid \{\boldsymbol{v}_i\}_{i=1}^N \sim \text{Dir}(1+d_\eta, \ldots, 1+d_\eta)$.

4.4.2 Computation of Expectation of Functionals of Full-data Responses

Our interest lies in the expectation of functionals of $\boldsymbol{y}$, given by

$$\begin{aligned} E[t(\boldsymbol{y})] &= \int_{\boldsymbol{y}} t(\boldsymbol{y}) p(\boldsymbol{y}) d\boldsymbol{y} \\ &= \int_{\boldsymbol{y}} t(\boldsymbol{y}) \left[\sum_s \int_v p_s(\tilde{\boldsymbol{y}}_s \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v}) p_s(\bar{\boldsymbol{y}}_s \mid \boldsymbol{v}) p(s \mid \boldsymbol{v}) p(\boldsymbol{v}) d\boldsymbol{v} \right] d\boldsymbol{y}. \end{aligned} \quad (4.11)$$

Once we have obtained posterior samples $\{\boldsymbol{w}_O^{(l)}, l = 1, \ldots, L\}$, the expression (4.11) can be computed by Monte Carlo integration. Since the desired functionals are functionals of $\boldsymbol{y}$, computing (4.11) involves sampling pseudo-data based on the posterior samples. We note that this is an application of G-computation (Robins, 1986, Scharfstein et al., 2014, Linero and Daniels, 2015) within the Bayesian paradigm (see Algorithm 1).

Algorithm 1 G-computation

1: **for** l in $1, \ldots, L$ **do**
2: **for** m in $1, \ldots, M$ **do**
3: 1. Draw $\boldsymbol{V}^* = \boldsymbol{v}^* \sim p(\boldsymbol{v}^* \mid \boldsymbol{\eta}^{(l)})$
4: 2. Draw $S^* = s^* \sim p(s^* \mid \boldsymbol{v}^*, \boldsymbol{f}^{(l)})$
5: 3. Draw $\bar{\boldsymbol{Y}}_s^* = \bar{\boldsymbol{y}}_s^* \sim p(\bar{\boldsymbol{y}}_s^* \mid s^*, \boldsymbol{v}^*, \boldsymbol{\pi}^{(l)})$
6: 4. Draw $\tilde{\boldsymbol{Y}}_s^* = \tilde{\boldsymbol{y}}_s^* \sim p(\tilde{\boldsymbol{y}}_s^* \mid \bar{\boldsymbol{y}}_s^*, s^*, \boldsymbol{v}^*, \boldsymbol{\omega}_E^{(l)})$
7: 5. Set $\boldsymbol{Y}^{*(m,l)} = (\bar{\boldsymbol{Y}}_s^*, \tilde{\boldsymbol{Y}}_s^*)$
8: **end for**
9: **end for**
10: **return** $(1/ML) \cdot \sum_{m,l} t\left[\boldsymbol{Y}^{*(m,l)}\right]$

In detail, for step 1, we draw $\boldsymbol{V}^* = \boldsymbol{v}_i$ with probability $p(\boldsymbol{V} = \boldsymbol{v}_i \mid \boldsymbol{\eta}^{(l)}) = \eta_i^{(l)}$. For step 2, we draw S^* by sequentially sampling from $R \sim$ Bernoulli$[p(S^* = j \mid S^* \geq j, \boldsymbol{v})]$. If $R = 1$, take $S^* = j$; otherwise proceed with $p(S^* = j+1 \mid S^* \geq j+1, \boldsymbol{v})$, $j = 2, \ldots, J$. For step 3, we first draw $y_1^* \sim N\left(a_0(\boldsymbol{v}^*, 1, s^*), \sigma_{1s^*}^2\right)$ and then sequentially draw $y_j^* \sim N\left(a(\boldsymbol{v}^*, j, s^*) + \bar{\boldsymbol{y}}_{j-1}^T \boldsymbol{\Phi}_{js^*}, \sigma_{js^*}^2\right)$, $j = 2, \ldots, s^*$ as in (4.2), where $a_0(\boldsymbol{v}^*, 1, s^*)$ and $a(\boldsymbol{v}^*, j, s^*)$ are generated by G-P prediction rule (Rasmussen and Williams, 2006). For step 4, we sequentially draw y_j^* for $j = s^* + 1, \ldots, J$ as in (4.7) from the unidentified distributions, now identified using identifying restrictions. When the ACMV restriction is specified, step 4 involves sampling from a distribution of $p_{\geq j}(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v})$, where

$$p_{\geq j}(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v}) = \sum_{k=j}^{J} \alpha_{kj}(\bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v})\, p_k(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v}), \tag{4.12}$$

and

$$\alpha_{kj}(\bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v}) = p(S = k \mid \bar{\boldsymbol{y}}_{j-1}, S \geq j, \boldsymbol{v}) \\ = \frac{p(\bar{\boldsymbol{y}}_{j-1} \mid S = k, \boldsymbol{v}) p(S = k \mid S \geq j, \boldsymbol{v})}{\sum_{k=j}^{J} p(\bar{\boldsymbol{y}}_{j-1} \mid S = k, \boldsymbol{v}) p(S = k \mid S \geq j, \boldsymbol{v})}.$$

The distribution in (4.12) is a mixture distribution over patterns. We sample from (4.12) by first drawing $K = k$ with probability α_{kj}, $k = j, \ldots, J$, then drawing a sample from $p_k(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v})$. When the NFMV restriction is specified, step 4 also involves sampling from a distribution $p_{\geq j-1}(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v})$, where

$$p_{\geq j-1}(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v}) = \alpha_{j-1,j-1}(\bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v})\, p_{j-1}(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v}) + \\ [1 - \alpha_{j-1,j-1}(\bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v})] p_{\geq j}(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v}).$$

Sampling from $p_{\geq j-1}(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v})$ is done by first sampling $Y_j^* \sim p_{\geq j}(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v})$ as in (4.12). Then draw $R \sim \text{Bernoulli}[\alpha_{j-1,j-1}]$. If $R = 1$, apply the location shift (4.9), otherwise, retain Y_j^*. See Appendix B.2 for more details of steps 3 and 4.

4.5 Simulation Studies

We conduct several simulation studies similar to the data example to assess the operating characteristic of our proposed model. We simulate responses for $J = 6$ time points and fit our model to estimate the change from baseline treatment effect, i.e. $\text{E}[Y_J - Y_1]$. We take $\kappa_0^2 \sim \text{IG}(10, 1)$ and $\kappa^2 \sim \text{IG}(10, 1)$ to shrink the nonparametric model towards a simple linear regression model,

and set the other prior and hyperprior parameters at standard noninformative choices. See Appendix B.3 for exact values. For comparison, we consider two alternatives: (1) a parametric model that consists of a linear regression model for $p_s(y_j \mid \bar{\boldsymbol{y}}_{j-1}, \boldsymbol{v})$, a sequential logit model for $p(s \mid \boldsymbol{v})$, and a Bayesian bootstrap model for $p(\boldsymbol{v})$, as in Equations (4.15), (4.14) and (4.6), respectively; (2) a parametric model without $\boldsymbol{V}$ that consists of a linear regression model for $p_s(y_j \mid \bar{\boldsymbol{y}}_{j-1})$ and a Bayesian bootstrap model for $p(s)$. We use noninformative priors for the two parametric models. For each simulation scenario below, we generate 500 datasets with $N = 200$ subjects per dataset.

4.5.1 Performance Under MAR

We first evaluate the performance of our model under the ACMV restriction (MAR). Since this restriction completely identifies the extrapolation distribution, this simulation study validates the appropriateness of our observed data model specification. We consider the following three simulation scenarios.

Scenario 1. We test the performance of our approach when the data are generated from a simple linear pattern-mixture model to assess loss of efficiency from using an unnecessary complex modeling approach. For each subject, we first simulate $Q = 4$ auxiliary covariates from a multivariate normal distribution

$$\boldsymbol{V} \overset{iid}{\sim} N(\boldsymbol{0}, \Sigma_{vv}). \tag{4.13}$$

We then generate dropout time using a sequential logit model

$$\text{logit}\, P(S = s \mid S \geq s, \boldsymbol{V}) = \zeta_s + \boldsymbol{V}^T \boldsymbol{\xi}_s. \tag{4.14}$$

Next, we generate $\bar{\boldsymbol{Y}}_s$ from

$$\begin{aligned} &\left(Y_j \mid \bar{\boldsymbol{Y}}_{j-1}, S = s, \boldsymbol{V}\right) \sim N\left(\mu_{js}(\bar{\boldsymbol{Y}}_{j-1}, \boldsymbol{V}), \sigma_{js}^2\right), \quad \text{for } j = 1, \ldots, s \\ &\text{where} \quad \mu_{js}(\bar{\boldsymbol{Y}}_{j-1}, \boldsymbol{V}) = \begin{cases} \boldsymbol{V}^T \boldsymbol{\beta}_{0s} + b_{js} & \text{if } j = 1 \\ \boldsymbol{V}^T \boldsymbol{\beta}_s + b_{js} + \bar{\boldsymbol{Y}}_{j-1}^T \boldsymbol{\Phi}_{js} & \text{if } j \geq 2 \end{cases} \end{aligned} \tag{4.15}$$

Finally, the distribution of $\tilde{\boldsymbol{Y}}_s$ is specified under the ACMV restriction (for calculating the simulation truth of the mean estimate).

The parameters in (4.13), (4.14) and (4.15) are chosen by fitting the model to the test drug arm of the schizophrenia clinical trial (after standardizing the responses and the auxiliary covariates with mean 0 and standard deviation 1). See Appendix B.3 for details.

Scenario 2. We consider a scenario where the covariates and the responses have more complicated structures in order to test the performance of our model when linearity does not hold. For simplicity, for each subject, we simulate $Q = 3$ auxiliary covariates from $\boldsymbol{V} \overset{iid}{\sim} N(\boldsymbol{0}, \Sigma_{vv})$. The responses and drop out times are generated in the same way as in scenario 1, but we include interactions and nonlinearities by replacing $\boldsymbol{V}$ in Equations (4.14) and (4.15) with $\tilde{\boldsymbol{V}} = (V_1, V_2, V_3, V_1 \times V_2, V_1 \times V_3, V_2 \times V_3, V_1^2, V_2^2, V_3^3)$. The regression coefficients $\boldsymbol{\xi}_s$, $\boldsymbol{\beta}_{0s}$ and $\boldsymbol{\beta}_s$ change accordingly. See Appendix B.3 for further details.

Scenario 3. We consider a scenario with a very different structure from our model formulation. In particular, we consider a lag-1 selection model with a mixture model for the joint distribution of $\boldsymbol{Y}$ and $\boldsymbol{V}$. We generate

$$\begin{aligned} &K \sim \text{Categorical}(\boldsymbol{\pi}), \\ &\Omega^{(K)} \sim \mathcal{W}^{-1}\left((\nu - J - Q - 1)\Omega_0^{(K)}, \nu\right), \\ &\begin{pmatrix} \boldsymbol{Y} \\ \boldsymbol{V} \end{pmatrix} \mid K \sim N\left[\boldsymbol{\mu}^{(K)}, \Omega^{(K)}\right], \\ &\text{logit}\, P(S = s \mid S \geq s, \boldsymbol{Y}, \boldsymbol{V}) = \zeta_s + \ell_s Y_s + \boldsymbol{V}^T \boldsymbol{\xi}_s, \end{aligned} \tag{4.16}$$

where $\mathcal{W}^{-1}\left((\nu - \dim(\Omega_0) - 1)\Omega_0, \nu\right)$ is an inverse-Wishart distribution with precision parameter ν and mean Ω_0. See Linero and Daniels (2015) for further details on this type of model. Formulating a joint distribution as in (4.16) allows us to impose complicated relationships between $\boldsymbol{Y}$ and $\boldsymbol{V}$ (Müller et al., 1996). We consider $Q = 3$ auxiliary covariates and 5 mixture components. We assume $\boldsymbol{\mu}^{(K)}$ and $\Omega_0^{(K)}$ correspond to a linear model of $(\boldsymbol{Y} \mid \boldsymbol{V})$ and have the form

$$\boldsymbol{\mu}^{(K)} = \begin{pmatrix} \boldsymbol{\mu}_y^{(K)} \\ \mathbf{0} \end{pmatrix}, \quad \Omega_0^{(K)} = \begin{pmatrix} \Sigma_{yy}^{(K)} & \Sigma_{yv}^{(K)} \\ \Sigma_{vy}^{(K)} & \Sigma_{vv} \end{pmatrix}.$$

In particular, we generate $\boldsymbol{\mu}^{(K)}$ and $\Omega_0^{(K)}$ according to Linero and Daniels (2015) by fitting the mixture model to the active control arm of the schizophrenia clinical trial. See Appendix B.3 for further details.

The simulation results are summarized in Table 4.2. For scenario 1, the true data generating model is the linear regression model with $\boldsymbol{V}$. The three models have similar performance in terms of MSE. The 95% credible

Model	Bias	CI width	CI coverage	MSE
		Scenario 1		
NP	-0.010(0.004)	0.294(0.002)	0.910(0.012)	0.008(0.000)
LM	-0.005(0.004)	0.379(0.001)	0.969(0.007)	0.008(0.000)
No $\boldsymbol{V}$	0.004(0.004)	0.385(0.002)	0.969(0.007)	0.008(0.000)
		Scenario 2		
NP	0.038(0.010)	0.915(0.004)	0.933(0.011)	0.058(0.004)
LM	0.197(0.010)	0.954(0.004)	0.855(0.015)	0.097(0.005)
No $\boldsymbol{V}$	0.289(0.010)	1.009(0.004)	0.794(0.017)	0.138(0.007)
		Scenario 3		
NP	0.001(0.007)	0.668(0.002)	0.953(0.009)	0.028(0.002)
LM	0.008(0.007)	0.705(0.002)	0.968(0.008)	0.028(0.002)
No $\boldsymbol{V}$	0.026(0.007)	0.707(0.002)	0.964(0.008)	0.028(0.002)

Table 4.2: Summary of simulation results under MAR. Values shown are posterior means, with Monte Carlo standard errors in parentheses. NP, LM and No $\boldsymbol{V}$ represent the proposed nonparametric model, the linear regression model with auxiliary covariates and the linear regression model without auxiliary covariates, respectively. Coverage of 95% credible intervals.

interval of the nonparametric model has a frequentist coverage rate less than 95% due to the prior information, i.e., the Gaussian process priors and the AR/CAR priors, being quite strong and the sample size ($N = 200$) being relatively small. Therefore, the Bayesian credible interval is unlikely to have the expected frequentist coverage. The linear regression model ignoring $\boldsymbol{V}$ does not perform worse than the one including $\boldsymbol{V}$. The reason is probably that the (linear) effects of different $\boldsymbol{V}$'s on $t(\boldsymbol{Y})$ cancel out in the integration (4.11). For scenario 2, the true data generating model does not match any of the three models used for inference. The nonparametric model significantly outperforms the parametric linear regression models in all aspects. The result

suggests that when the model is misspecified, the nonparametric model has much more robust performance. We also note that when $\boldsymbol{Y}$ and S do not have a linear relationship with $\boldsymbol{V}$, ignoring $\boldsymbol{V}$ results in more significant bias than including $\boldsymbol{V}$ (even mistakenly). For scenario 3, the true data generating model is a mixture of linear regression models but has a different parameterization with the three models used for inference. The three models again have similar performance. For a pattern-mixture model, the marginal distribution of the responses $\boldsymbol{Y}$ is a mixture distribution over patterns, which explains the good performance of the three models. For all the three scenarios, the nonparametric model always gives narrower credible intervals and has lower bias, in particular versus the model without auxiliary covariates.

In summary, the nonparametric approach loses little when the corresponding parametric model holds, and it significantly outperforms the other approaches when the model used for inference is misspecified. The simulation results suggest that the nonparametric approach is more favorable compared with the parametric approaches and accommodates complex mean models.

4.5.2 Performance Under MNAR

To assess the sensitivity of our model to untestable assumptions for the extrapolation distribution, we fit our model to simulated data under an NFD restriction (4.8). We consider simulation scenarios 2 and 3 as in Section 4.5.1, where the simulation truth is still generated under MAR. We complete our model with a location shift (Equations (4.9) and (4.10)). Recall that the

sensitivity parameter $\tilde{\tau}$ measures the deviation of our model from MAR, and the simulation truth corresponds to $\tilde{\tau} = 0$. The sensitivity parameter $\tilde{\tau}$ is given four different priors: Unif$(-0.75, 0.25)$, Unif$(-0.5, 0.5)$, Unif$(-0.25, 0.75)$, Unif$(0, 1)$. All the four priors contain the simulation truth. Compared to fixing the value of $\tilde{\tau}$, using a uniform prior conveys uncertainty about the identifying restriction. For example, using a point mass prior $\tilde{\tau} = 0$ implies MAR with no uncertainty, while using a prior such that $\mathrm{E}[\tilde{\tau}] = 0$ and $\mathrm{Var}[\tilde{\tau}] > 0$ implies MAR with uncertainty.

The simulation results are summarized in Table 4.3. When the sensitivity parameter $\tilde{\tau}$ is centered at the correct value 0, the nonparametric model significantly outperforms the parametric linear regression models under scenario 2 and performs as well as the parametric linear regression models under scenario 3. Comparing with the simulation results under MAR (Table 4.2), the use of a uniform prior for $\tilde{\tau}$ induces more uncertainty on inference according to the wider credible intervals. We also note that, when $\tilde{\tau}$ is not centered at 0, the models using $\boldsymbol{V}$ still perform better than the model not using $\boldsymbol{V}$. This is due to the calibration of the location shift (Equations (4.9) and (4.10)). For the same $\tilde{\tau}$ we would have a smaller deviation from ACMV using $\boldsymbol{V}$ compared to not using $\boldsymbol{V}$. This property makes the missingness "closer" to MAR and reduces the extent of sensitivity analysis with the inclusion of $\boldsymbol{V}$.

Model	$E(\tilde{\tau})$	Bias	CI width	CI coverage	MSE
			Scenario 2		
NP	-0.25	-0.046(0.010)	0.979(0.004)	0.968(0.008)	0.051(0.003)
	0	0.039(0.010)	0.994(0.004)	0.966(0.008)	0.052(0.003)
	0.25	0.129(0.010)	1.016(0.004)	0.942(0.010)	0.068(0.004)
	0.5	0.224(0.010)	1.036(0.004)	0.874(0.014)	0.103(0.006)
LM	-0.25	0.081(0.010)	1.056(0.004)	0.966(0.008)	0.056(0.004)
	0	0.189(0.010)	1.075(0.004)	0.923(0.012)	0.086(0.005)
	0.25	0.301(0.010)	1.101(0.005)	0.841(0.016)	0.142(0.007)
	0.5	0.418(0.010)	1.132(0.005)	0.700(0.020)	0.228(0.009)
No $\boldsymbol{V}$	-0.25	0.155(0.009)	1.135(0.005)	0.961(0.008)	0.069(0.004)
	0	0.281(0.009)	1.161(0.005)	0.889(0.014)	0.125(0.006)
	0.25	0.411(0.009)	1.191(0.005)	0.771(0.018)	0.216(0.008)
	0.5	0.546(0.010)	1.223(0.005)	0.583(0.021)	0.348(0.011)
			Scenario 3		
NP	-0.25	-0.049(0.007)	0.691(0.002)	0.944(0.010)	0.032(0.002)
	0	-0.008(0.007)	0.695(0.002)	0.972(0.007)	0.030(0.002)
	0.25	0.033(0.008)	0.696(0.002)	0.963(0.008)	0.032(0.002)
	0.5	0.076(0.008)	0.703(0.002)	0.929(0.011)	0.037(0.002)
LM	-0.25	-0.042(0.007)	0.725(0.002)	0.961(0.008)	0.031(0.002)
	0	-0.001(0.007)	0.728(0.002)	0.980(0.006)	0.030(0.002)
	0.25	0.042(0.008)	0.734(0.002)	0.972(0.007)	0.032(0.002)
	0.5	0.085(0.008)	0.741(0.002)	0.948(0.010)	0.038(0.002)
No $\boldsymbol{V}$	-0.25	-0.047(0.007)	0.751(0.002)	0.972(0.007)	0.031(0.002)
	0	0.015(0.007)	0.761(0.002)	0.987(0.005)	0.029(0.002)
	0.25	0.079(0.007)	0.768(0.002)	0.966(0.008)	0.036(0.002)
	0.5	0.144(0.008)	0.783(0.003)	0.909(0.012)	0.052(0.003)

Table 4.3: Summary of simulation results under MNAR. Values shown are posterior means, with Monte Carlo standard errors in parentheses. NP, LM and No $\boldsymbol{V}$ represent the proposed nonparametric model, the linear regression model with auxiliary covariates and the linear regression model without auxiliary covariates, respectively. Coverage of 95% credible intervals. The values of $E(\tilde{\tau})$, -0.25, 0, 0.25 and 0.5, correspond to prior specifications Unif$(-0.75, 0.25)$, Unif$(-0.5, 0.5)$, Unif$(-0.25, 0.75)$ and Unif$(0, 1)$, respectively.

4.6 Application to the Schizophrenia Clinical Trial

We implement inference under the proposed model for data from the schizophrenia clinical trial described in Section 4.1.3. Recall the quantity of interest is the change from baseline treatment effect, $r_x = E[Y_{i6} - Y_{i1} \mid X_i = x]$, where $x = 1, 2$ or 3 correspond to treatments under active control, placebo or test drug, respectively. We are particularly interested in the treatment effect improvements over placebo, i.e. $r_1 - r_2$ and $r_3 - r_2$.

4.6.1 Comparison to Alternatives and Assessment of Model Fit

We first compare the fit among the proposed model and alternatives. We consider a linear regression model with auxiliary covariates and a linear regression model without auxiliary covariates, as we have used in the simulation studies. We use the log-pseudo marginal likelihood (LPML) as the model selection criteria. LPML is defined by

$$\text{LPML} = \frac{1}{N} \sum_{i=1}^{N} \log(\text{CPO}_i).$$

Here CPO_i is the conditional predictive ordinate (Geisser and Eddy, 1979) for observation i,

$$\text{CPO}_i = p\left(\bar{\boldsymbol{Y}}_{iS_i}, S_i, V_i \mid \{\bar{\boldsymbol{Y}}_{i'S_{i'}}, S_{i'}, V_{i'}\}_{i'=1, i' \neq i}^{N}\right).$$

LPML can be straightforwardly estimated using posterior samples $\{\boldsymbol{\omega}_O^{(l)}, l = 1, \ldots, L\}$ (Gelfand and Dey, 1994). A model with higher LPML is more favorable compared to models with lower LPMLs. We fit the three models to the

data and calculate the LPML by taking the summation of the LPML under each treatment arm. The results are summarized in Table 4.4. The proposed nonparametric model had the largest LPML over the linear regression models with and without auxiliary covariates. This is not surprising in light of the earlier simulation results. We also compare inferences on treatment effect improvements over placebo under the MAR assumption using the three models. The results are summarized in Table 4.4.

Model	LPML	$r_1 - r_3$	$r_2 - r_3$
NP	-31.97	0.90(-5.56, 7.63)	-7.09(-15.75, 0.98)
LM	-32.61	-1.11(-7.96, 5.73)	-6.99(-14.61, 0.54)
No V	-32.71	-1.86(-8.53, 4.86)	-8.10(-15.44, -0.98)

Table 4.4: Comparison of LPML (the second column) and inference results under MAR (the third and fourth columns). NP, LM and No V represent the proposed model, linear regression model with auxiliary covariates and linear regression model without auxiliary covariates, respectively. For the inference results under MAR, values shown are posterior means, with 95% credible intervals in parentheses.

Next, we assess the "absolute" goodness of fit of the proposed model. We estimate the cumulative dropout rates and observed-data means at each time point and under each treatment using the proposed model. That is,

$$p(S \leq j \mid x) = \int p(S \leq j \mid \boldsymbol{v}, x) p(\boldsymbol{v} \mid x) d\boldsymbol{v}, \quad \text{and}$$
$$\mathrm{E}(Y_j \mid S \geq j, x) = \int \mathrm{E}(Y_j \mid S \geq j, \boldsymbol{v}, x) p(\boldsymbol{v} \mid S \geq j, x) d\boldsymbol{v}.$$

We then compare those estimates with results obtained from the empirical distribution (not using information from auxiliary covariates). The comparison

is shown in Figure 4.2. Despite some small differences that are due to the use of information from auxiliary covariates and prior information, there is no evidence for lack of fit.

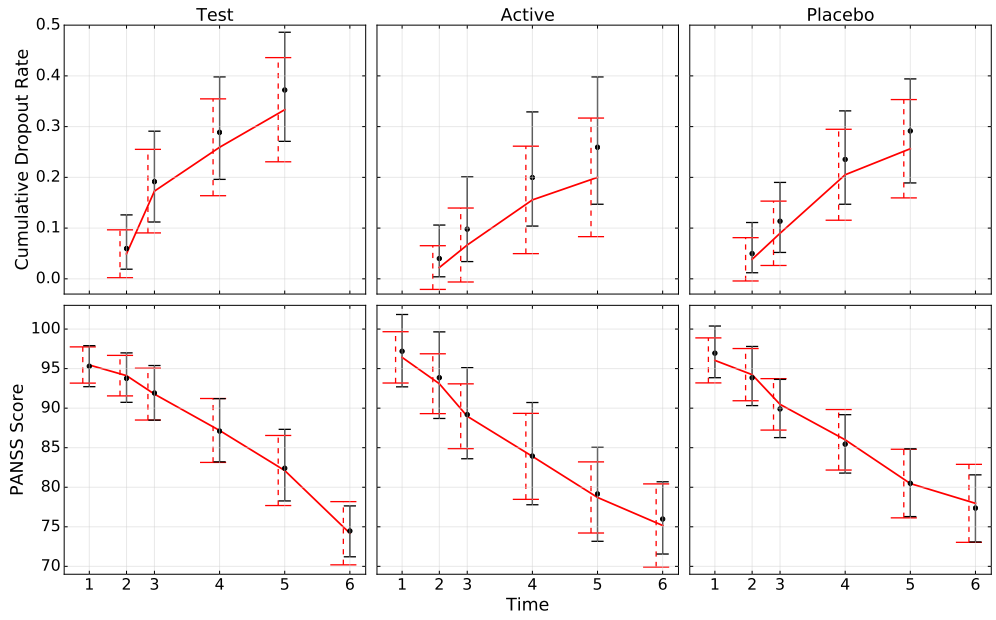


Figure 4.2: Cumulative dropout rates (top) and observed-data means (bottom) over time obtained from the model versus the ones obtained from the empirical distribution. The solid red line represents the empirical values, black dots represent the posterior means, red dashed error bars represent frequentist 95% confidence intervals, and black solid error bars represent the model's 95% credible intervals.

4.6.2 Inference

A large portion of subjects dropout for reasons that suggest the missing data are MNAR (see Section 4.1.3). To identify the extrapolation distribution, we make the NFD assumption (4.8). Recall that the NFD assumption leaves

one conditional distribution per incomplete pattern unidentified: $p_s(y_{s+1} \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v}, x)$. To better identify $p_s(y_{s+1} \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v}, x)$, rather than simply assuming a location shift (4.9), we make use of information regarding the type of dropout. Let $Z_i = 1$ or 0 denote subject i drops out for informative or noninformative reasons, respectively. We model Z conditional on observed data responses, pattern, auxiliary covariates and treatment with a logistic regression,

$$\text{logit} P(Z = 1 \mid \bar{\boldsymbol{Y}}_s, S = s, \boldsymbol{V}, X = x) = \zeta_{sx} + \bar{\boldsymbol{Y}}_s^T \boldsymbol{\ell}_{sx} + \boldsymbol{V}^T \boldsymbol{\xi}_{sx}.$$

An alternative nonparametric choice for modeling $P(Z \mid \bar{\boldsymbol{Y}}_S, S, \boldsymbol{V}, X)$ is BART. Since the sample size of Z (i.e. number of dropout subjects for each pattern and each treatment) is small, we find that the simpler logistic regression model suffices here.

The indicator Z is used to help identify $p_s(y_{s+1} \mid \bar{\boldsymbol{y}}_s, \boldsymbol{v}, x)$. We assume

$$\begin{aligned}
&\left[Y_{s+1} \mid \bar{\boldsymbol{Y}}_s, S = s, \boldsymbol{V}, X, \boldsymbol{\omega}\right] \stackrel{\text{d}}{=} \\
&\quad P(Z = 1 \mid \bar{\boldsymbol{Y}}_s, S = s, \boldsymbol{V}, X) \cdot \left[Y_{s+1} + \tau_{s+1} \mid \bar{\boldsymbol{Y}}_s, S \geq s+1, \boldsymbol{V}, X, \boldsymbol{\omega}\right] + \\
&\qquad P(Z = 0 \mid \bar{\boldsymbol{Y}}_s, S = s, \boldsymbol{V}, X) \cdot \left[Y_{s+1} \mid \bar{\boldsymbol{Y}}_s, S \geq s+1, \boldsymbol{V}, X, \boldsymbol{\omega}\right], \quad (4.17)
\end{aligned}$$

which is a mixture of an ACMV assumption and a location shift. The idea is that, if a subject drops out for a reason associated with MAR, we impute the next missing value under ACMV; otherwise, we impute the next missing value by applying a location shift. The sensitivity parameter τ_{s+1} is interpretable to subject-matter experts. Suppose two hypothetical subjects A and B have the same auxiliary covariates and histories up to time s, and suppose subject

B drops out for an informative reason at time s while subject A remains on study. Then, the response of subject B at time $(s+1)$ is stochastically identical to the response of subject A at time $(s+1)$ after applying the location shift τ_{s+1}. As the prior for τ_{s+1}, we assume $\tau_{s+1} \geq 0$ as we expect subject B would have a higher PANSS score at time $(s+1)$ than subject A. The magnitude of τ_{s+1} is calibrated as in Equation (4.10),

$$\left[\tau_{s+1} \mid \bar{Y}_s = \bar{\boldsymbol{y}}_s, \boldsymbol{V} = \boldsymbol{v}, X = x\right] = \tilde{\tau}_x \cdot \Delta_{s+1,x}(\bar{\boldsymbol{y}}_s, \boldsymbol{v}). \tag{4.18}$$

We assume a uniform prior on $\tilde{\tau}_x$, $\tilde{\tau}_x \sim \text{Unif}(0,1)$, as it is thought unlikely that the deviation from ACMV would exceed a standard deviation (Linero and Daniels, 2015).

Figure 4.3 summarizes change from baseline treatment effect improvements of the test drug and active drug over placebo. We implement inference under both the MAR and the mixture of MAR/MNAR (Equations (4.17) and (4.18)) assumptions. For the test drug arm, the treatment effect improvement $r_1 - r_3$ has posterior mean 0.90 and 95% credible interval $(-5.56, 7.63)$ under MAR, and posterior mean 0.91 and 95% credible interval $(-6.47, 8.54)$ under MNAR. There is no evidence that the test drug has better performance than placebo. The MNAR assumption has little effect on inference on $r_1 - r_3$, since the test and placebo arms have similar informative dropout rates. For the active drug arm, the treatment effect improvement $r_2 - r_3$ has posterior mean -7.09 and 95% credible interval $(-15.75, 0.98)$ under MAR, and posterior mean -7.61 and 95% credible interval $(-16.49, 0.28)$ under MNAR. There

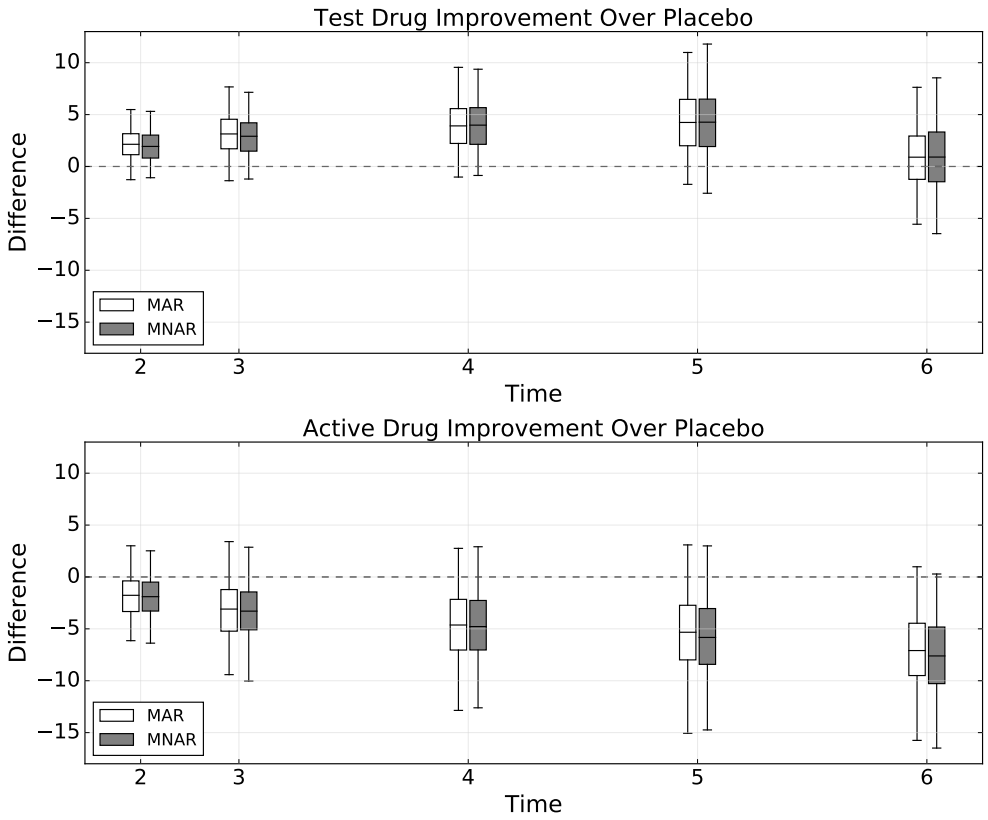


Figure 4.3: Change from baseline treatment effect improvements of the test drug (top) and active drug (bottom) over placebo over time. Smaller values indicate more improvement compared to placebo. The dividing line within the boxes represents the posterior mean, the bottom and top of the boxes are the first and third quartiles, and the ends of the whiskers show the 0.025 and 0.975 quantiles.

appears to be some evidence that the active drug has better treatment effect than placebo, especially under MNAR. The MNAR assumption makes the difference between the active and placebo arms more significant, since the active arm has a smaller informative dropout rate.

4.6.3 Sensitivity Analysis

To assess the sensitivity of inferences on treatment effect improvements ($r_1 - r_3$ and $r_2 - r_3$) to the informative priors on the sensitivity parameters ($\tilde{\tau}_1$, $\tilde{\tau}_2$ and $\tilde{\tau}_3$), we consider a set of point-mass priors for each $\tilde{\tau}_x$ along the $[0, 1]$ grid. Figure 4.4 summarizes how inferences on $r_1 - r_3$ and $r_2 - r_3$ change for different choices of $\tilde{\tau}_1$, $\tilde{\tau}_2$ and $\tilde{\tau}_3$. Considering a significance level of 0.05. The sensitivity analysis corroborates our conclusion that there is no evidence that the test drug has better performance than placebo. For all the choices of $\tilde{\tau}_1$ and $\tilde{\tau}_3$, the posterior probability of $r_1 - r_3 < 0$ does not reach the 0.95 cutoff. On the other hand, the sensitivity analysis shows that there is some evidence that the active drug is superior than placebo. For all the combinations of $\tilde{\tau}_2$ and $\tilde{\tau}_3$, the posterior probability of $r_2 - r_3 < 0$ is greater than 0.84. For most favorable values of $\tilde{\tau}_2$ and $\tilde{\tau}_3$, the posterior probability of $r_2 - r_3 < 0$ is greater than 0.95, although it only occurs when $\tilde{\tau}_2$ is substantially smaller than $\tilde{\tau}_3$. In summary, for all the choices of $\tilde{\tau}_x$, we do not reach substantially different results, which improves our confidence on the previous conclusions. Finally, inferences on $r_1 - r_3$ and $r_2 - r_3$ under the uniform prior $\tilde{\tau}_x \sim \text{Unif}(0, 1)$ are roughly the same as inferences under the point-mass prior $\tilde{\tau}_x = 0.5$, which means using the uniform prior does not induce much more uncertainty in this scenario.

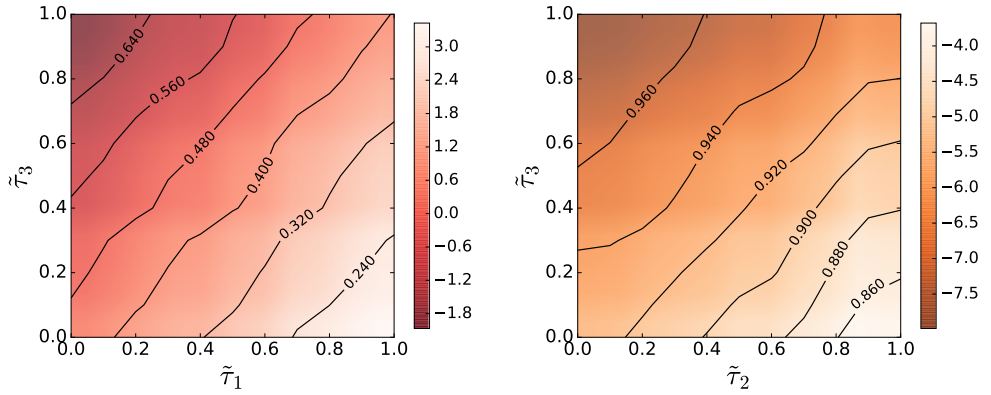


Figure 4.4: Contour plots showing inferences on treatment effect improvements $r_1 - r_3$ (left) and $r_2 - r_3$ (right) for different choices of the sensitivity parameters along the $[0, 1]$ grid. The colors represent posterior means of $r_x - r_3$, where a deeper color indicates more improvement compared to placebo. The black lines show posterior probabilities of $r_x - r_3 < 0$.

4.7 Discussion

In this work, we have developed a nonparametric Bayesian approach to monotone missing data with non-ignorable missingness in the presence of auxiliary covariates. Under the extrapolation factorization, we flexibly model the observed data distribution and specify the extrapolation distribution using identifying restrictions. We have shown the inclusion of auxiliary covariates in the model could in general improve the accuracy of inferences and reduce the extent of sensitivity analysis. We have also shown more accurate inferences can be obtained by using the proposed nonparametric Bayesian approach compared to using more restrictive parametric approaches.

In the model for the observed responses conditional on past responses,

pattern and auxiliary covariates, we have assumed the effect of past responses on current response is linear (Equation (4.3) case $j \geq 2$). To make the model more flexible, we could include past responses in the index set of the stochastic process $a(\cdot)$, i.e. to model $\left(Y_j \mid \bar{\boldsymbol{Y}}_{j-1}, S, \boldsymbol{V}, \boldsymbol{\pi}\right) = a(\bar{\boldsymbol{Y}}_{j-1}, \boldsymbol{V}, j, S) + \varepsilon_{js}$. This way the model could account for possible nonlinearity and interactions in the past responses. However, this modeling approach is complicated by the fact that for different time j, the dimension of past responses $\bar{\boldsymbol{Y}}_{j-1}$ is different, so we leave it as an extension of this work. A possible compromise could be including only lag-1 response in $a(\cdot)$.

The computation complexity of the Gaussian process is cubic in the number of data points. The problem is manageable in our application since the schizophrenia clinical trial dataset only contains 204 subjects. When a much larger number of subjects is considered, several methods have been proposed to tackle the computational bottleneck of the GP (see Banerjee et al., 2008, 2013, Hensman et al., 2013, Datta et al., 2016). To identify the extrapolation distribution under NFD, we assume a location shift. Alternatively, we can consider exponential tilting (Rotnitzky et al., 1998, Birmingham et al., 2003).

A possible extension of our work is to consider continuous time dropout. The Gaussian process is naturally suitable for the continuous case. Another extension would be more flexible incorporation of auxiliary covariates beyond the mean. Another possible future direction is to extend our method to non-monotone missing data without imposing the partial ignorability assumption. In the setting of binary outcomes, our method can be extended by using a

probit link.

Chapter 5

Future Work

In the preceding chapters, we have developed nonparametric Bayesian models for biomedical data analysis. In particular, we have presented a novel feature allocation model for tumor subclone reconstruction using mutation pair data, a treed feature allocation model for tumor subclone phylogeny reconstruction using mutation pair data, and a nonparametric Bayesian approach to monotone missing data with auxiliary covariates in longitudinal studies. We have shown how inferences under our proposed models compare favorably with inferences under existing methods, and significantly outperform inferences under existing methods in certain scenarios.

As we have mentioned in the discussion section in each chapter, there are several directions for future works. For the tumor heterogeneity problem, the proposed models can be extended for data where a local haplotype segment consists of more than two SNVs. We can accommodate n-tuples instead of pairs of SNVs by increasing the number of categorical values (Q) that the entries of $\boldsymbol{Z}$ can take. The current model measures tumor heterogeneity with single nucleotide variants (SNVs) data in copy number neutral regions. We have discussed possibilities to incorporate copy number variants (CNVs)

and have specific research plans to formally incorporate CNVs into the model and software. Also, structural variants (SVs) such as deletion, duplication, inversion, translocation and other large genome rearrangement provide more information for characterizing tumor heterogeneity. Utilizing information from SVs is another direction of characterizing tumor heterogeneity. The major motivation of tumor subclone reconstruction is application to precision medicine. The reconstructed tumor subclones can be used as basis for adaptive Bayesian clinical trial design. Finally, we plan to develop computation efficient algorithms to handle large numbers (e.g. millions) of SNVs.

For the missing data problem, our model focuses on monotone missing data with discrete time dropout. A possible extension of this work is to consider continuous time dropout. The Gaussian process is suitable for this case. Another possible future direction is to extend our method to non-monotone missing data without imposing the partial ignorability assumption. In the setting of binary outcomes, our method can be extended by using a probit link. For large numbers of subjects, existing methods (Banerjee et al., 2008, 2013, Hensman et al., 2013, Datta et al., 2016) that tackle the computational bottleneck of the Gaussian process can be employed in our application.

Appendices

Appendix A

Appendix for Chapter 2

A.1 MCMC Implementation Details

We first introduce θ_{tc} as an unscaled abundance level of subclone c in sample t. Assume $\theta_{t0} \sim \text{Gamma}(d_0, 1)$ and $\theta_{tc} \mid C \sim \text{Gamma}(d, 1)$. Let $w_{tc} = \theta_{tc} / \sum_{c'=0}^{C} \theta_{tc'}$, then $\boldsymbol{w}_t \sim \text{Dirichlet}(d_0, d, \ldots, d)$. We make inference on $\boldsymbol{\theta}$ instead of $\boldsymbol{w}$ as the value of $\boldsymbol{\theta}$ is not restricted in a C-simplex. Similarly, we introduce ρ_g^* as an unscaled version of ρ_g. We let $\rho_g^* \sim \text{Gamma}(d_1, 1)$ and $\rho_g = \rho_g^* / \sum_{g'=1}^{4} \rho_{g'}^*$ for $g = 1, \ldots, 4$, $\rho_g^* \sim \text{Gamma}(2d_1, 1)$ and $\rho_g = \rho_g^* / \sum_{g'=5}^{6} \rho_{g'}^*$ for $g = 5, 6$, and $\rho_g^* \sim \text{Gamma}(2d_1, 1)$ and $\rho_g = \rho_g^* / \sum_{g'=7}^{8} \rho_{g'}^*$ for $g = 7, 8$.

Conditional on C, the posterior distribution for the other parameters is given by

$$
\begin{aligned}
p(\boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{\rho}^* \mid \boldsymbol{n}, C) \propto & \prod_{t=1}^{T} \prod_{k=1}^{K} \prod_{g=1}^{G} \tilde{p}_{tkg}^{n_{tkg}} \times \prod_{c=1}^{C} \prod_{q=1}^{Q} \pi_{cq}^{m_{cq}} \times \\
& \prod_{c=1}^{C} \left[\pi_{c1}^{1-1} (1 - \pi_{c1})^{\alpha/C - 1} \cdot \prod_{q=2}^{Q} \tilde{\pi}_{cq}^{\beta - 1} \right] \times \\
& \prod_{t=1}^{T} \left[\theta_{t0}^{d_0 - 1} e^{-\theta_{t0}} \prod_{c=1}^{C} \left(\theta_{tc}^{d-1} e^{-\theta_{tc}} \right) \right] \times \\
& \prod_{g=1}^{4} \left(\rho_g^{*d_1 - 1} e^{-\rho_g^*} \right) \cdot \prod_{g=5}^{8} \left(\rho_g^{*2d_1 - 1} e^{-\rho_g^*} \right).
\end{aligned}
$$

where $m_{cq} = \sum_{k=1}^{K} I(\boldsymbol{z}_{kc} = \boldsymbol{z}^{(q)})$ counts the number of mutation pairs in subclone c having genotype $\boldsymbol{z}^{(q)}$.

Updating $\boldsymbol{Z}$. We update $\boldsymbol{Z}$ by sampling each $\boldsymbol{z}_{kc}$ from:

$$p(\boldsymbol{z}_{kc} = \boldsymbol{z}^{(q)} \mid \ldots) \propto \prod_{t=1}^{T}\prod_{g=1}^{G}\left[\sum_{c'=1,c'\neq c}^{C} w_{tc'}\, A(\boldsymbol{h}_g, \boldsymbol{z}_{kc'}) + w_{tc}\, A(\boldsymbol{h}_g, \boldsymbol{z}^{(q)}) + w_{t0}\,\rho_g\right]^{n_{tkg}} \cdot \pi_{cq}$$

Updating $\boldsymbol{\pi}$. The posterior distribution for $\boldsymbol{\pi}$ is

$$\begin{aligned} p(\boldsymbol{\pi} \mid \ldots) &\propto \prod_{c=1}^{C}\left[\left(\prod_{q=1}^{Q}\pi_{cq}^{m_{cq}}\right)\cdot \pi_{c1}^{1-1}(1-\pi_{c1})^{\alpha/C-1}\cdot\prod_{q=2}^{Q}\tilde{\pi}_{cq}^{\beta-1}\right] \\ &= \prod_{c=1}^{C}\left[\pi_{c1}^{m_{c1}+1-1}(1-\pi_{c1})^{K-m_{c1}+\alpha/C-1}\cdot\prod_{q=2}^{Q}\tilde{\pi}_{cq}^{m_{cq}+\beta-1}\right]. \end{aligned}$$

For each $c = 1, \ldots, C$, we update $\boldsymbol{\pi}_c$ by sampling from

$$\begin{aligned} \pi_{c1} \mid \ldots &\sim \text{Beta}(m_{c1}+1, K - m_{c1} + \alpha/C), \\ (\tilde{\pi}_{c2}, \ldots, \tilde{\pi}_{cQ}) \mid \ldots &\sim \text{Dirichlet}(m_{c2}+\beta, \ldots, m_{cQ}+\beta), \end{aligned}$$

and transforming by $(\pi_{c2}, \ldots, \pi_{cQ}) = (1-\pi_{c1})\cdot(\tilde{\pi}_{c2}, \ldots, \tilde{\pi}_{cQ})$.

Updating $\boldsymbol{\theta}$. We update each θ_{tc} sequentially. For $c = 1, \ldots, C$,

$$p(\theta_{tc} \mid \ldots) \propto \prod_{k=1}^{K}\prod_{g=1}^{G}\left[\sum_{c=1}^{C} w_{tc}\, A(\boldsymbol{h}_g, \boldsymbol{z}_{kc}) + w_{t0}\,\rho_g\right]^{n_{tkg}} \cdot \theta_{tc}^{d-1} e^{-\theta_{tc}}.$$

A Metropolis-Hastings transition probability is used to update θ_{tc}. At each iteration, we propose a new $\tilde{\theta}_{tc}$ (on the log scale) by $\log(\tilde{\theta}_{tc}) \sim N(\log \theta_{tc}, 0.2)$, and evaluate the acceptance probability by

$$p_{\text{acc}}(\theta_{tc}, \tilde{\theta}_{tc}) = 1 \wedge \left[\Big(p(\tilde{\theta}_{tc} \mid \ldots)\, p(\theta_{tc} \mid \tilde{\theta}_{tc}) \Big) \Big/ \Big(p(\theta_{tc} \mid \ldots)\, p(\tilde{\theta}_{tc} \mid \theta_{tc}) \Big) \right].$$

The term $p(\theta_{tc} \mid \tilde{\theta}_{tc})/p(\tilde{\theta}_{tc} \mid \theta_{tc}) = \tilde{\theta}_{tc}/\theta_{tc}$ takes into account the Jacobian of the log transformation. For $c = 0$, the only difference is to substitute d with d_0.

Updating $\boldsymbol{\rho}^*$. We update each ρ_g^* sequentially. For $g = 1, \ldots, 4$,

$$p(\rho_g^* \mid \ldots) \propto \prod_{t=1}^{T} \prod_{k=1}^{K} \prod_{g=1}^{G} \left[\sum_{c=1}^{C} w_{tc}\, A(\boldsymbol{h}_g, \boldsymbol{z}_{kc}) + w_{t0}\, \rho_g \right]^{n_{tkg}} \cdot \rho_g^{*d_1 - 1} e^{-\rho_g^*}.$$

A Metropolis-Hastings transition probability is used to update ρ_g^*. At each iteration, we propose a new $\tilde{\rho}_g^*$ (on the log scale) by $\log(\tilde{\rho}_g^*) \sim N(\log \rho_g^*, 0.1)$, and evaluate the acceptance probability by

$$p_{\text{acc}}(\rho_g^*, \tilde{\rho}_g^*) = 1 \wedge \left[\big(p(\tilde{\rho}_g^* \mid \ldots)\, p(\rho_g^* \mid \tilde{\rho}_g^*) \big) \,/\, \big(p(\rho_g^* \mid \ldots)\, p(\tilde{\rho}_g^* \mid \rho_g^*) \big) \right].$$

The term $p(\rho_g^* \mid \tilde{\rho}_g^*)/p(\tilde{\rho}_g^* \mid \rho_g^*) = \tilde{\rho}_g^*/\rho_g^*$ takes into account the Jacobian of the log transformation. For $g = 4, \ldots, 8$, the only difference is to substitute d_1 with $2d_1$.

Parallel tempering. Parallel tempering (PT) is a MCMC technique first proposed by Geyer (1991). A good review can be found in Liu (2008). PT

is suitable for sampling from a multi-modal state space. It helps the MCMC chain to move freely among local modes which is desired in our application, and to create a better mixing Markov chain.

Algorithm 2 Parallel Tempering

1: Draw initial state $(\boldsymbol{x}_1^{(0)}, \ldots, \boldsymbol{x}_I^{(0)})$ from appropriate distributions
2: **for** l in $1, \ldots, L$ **do**
3: Draw $u \sim \text{Uniform}(0, 1)$
4: **if** $u \leq u_0$ **then**
5: Conduct the parallel step: update every $\boldsymbol{x}_i^{(l)}$ to $\boldsymbol{x}_i^{(l+1)}$ via respective MCMC scheme
6: **else**
7: Conduct the swapping step: draw $i \sim$ Discrete-Uniform$(1, \ldots, I-1)$, propose a swap between $\boldsymbol{x}_i^{(l)}$ and $\boldsymbol{x}_{i+1}^{(l)}$, accept the swap with probability

$$\min\left\{1, \frac{\pi_i(\boldsymbol{x}_{i+1}^{(l)})\pi_{i+1}(\boldsymbol{x}_i^{(l)})}{\pi_i(\boldsymbol{x}_i^{(l)})\pi_{i+1}(\boldsymbol{x}_{i+1}^{(l)})}\right\}$$

8: **end if**
9: **end for**

To sample from the target distribution $\pi(\boldsymbol{x})$, we consider a family of distributions $\Pi = \{\pi_i, i = 1, \ldots, I\}$, where $\pi_i(\boldsymbol{x}) \propto \pi(\boldsymbol{x})^{1/\Delta_i}$. Without loss of generality, let $\Delta_I = 1$ and $\pi_I(\boldsymbol{x}) = \pi(\boldsymbol{x})$. Denote by $\mathcal{X}_i$ the state space of $\pi_i(\boldsymbol{x})$. The PT scheme is illustrated in Algorithm 2.

In our application, we find by simulation that PT works well with $I = 10$ temperatures and $\{\Delta_1, \ldots, \Delta_{10}\} = \{4.5, 3.2, 2.5, 2, 1.7, 1.5, 1.35, 1.2, 1.1, 1\}$. We therefore use this parameter setting for all the simulation studies as well as the lung cancer dataset.

A.2 Updating C

For updating C, we split the data into a training set $\boldsymbol{n}'$, and a test set $\boldsymbol{n}''$ with $n'_{tkg} = bn_{tkg}$ and $n''_{tkg} = (1-b)n_{tkg}$. Let $p_b(\boldsymbol{x} \mid C) = p(\boldsymbol{x} \mid \boldsymbol{n}', C)$ denote the posterior of $\boldsymbol{x}$ conditional on C evaluated on the training set only. We use p_b in two occasions. First, we replace the original prior $p(\boldsymbol{x} \mid C)$ by $p_b(\boldsymbol{x} \mid C)$, and second, we use p_b as a proposal distribution of $\tilde{\boldsymbol{x}}$ as $q(\tilde{\boldsymbol{x}} \mid \tilde{C}) = p_b(\tilde{\boldsymbol{x}} \mid \tilde{C})$. We show that the use of the training sample posterior as proposal and modified prior in equation (4) (original manuscript) implies an approximation in the reported marginal posterior for C, but leaves the conditional posterior for all other parameters (given C) unchanged.

We evaluate the acceptance probability of $\tilde{C}$ on the test data by

$$\begin{aligned} p_{\text{acc}}(C, \boldsymbol{x}, \tilde{C}, \tilde{\boldsymbol{x}}) &= 1 \wedge \frac{p(\boldsymbol{n}'' \mid \tilde{\boldsymbol{x}}, \tilde{C})}{p(\boldsymbol{n}'' \mid \boldsymbol{x}, C)} \cdot \frac{p(\tilde{C})p(\tilde{\boldsymbol{x}} \mid \boldsymbol{n}', \tilde{C})}{p(C)p(\boldsymbol{x} \mid \boldsymbol{n}', C)} \cdot \frac{q(C \mid \tilde{C})q(\boldsymbol{x} \mid C)}{q(\tilde{C} \mid C)q(\tilde{\boldsymbol{x}} \mid \tilde{C})} \\ &= 1 \wedge \frac{p(\boldsymbol{n}'' \mid \tilde{\boldsymbol{x}}, \tilde{C})}{p(\boldsymbol{n}'' \mid \boldsymbol{x}, C)} \cdot \frac{p(\tilde{C})}{p(C)}. \end{aligned}$$

Under the model $p_b(\cdot)$ with the modified prior, the implied conditional posterior on $\boldsymbol{x}$ satisfies

$$\begin{aligned} p_b(\boldsymbol{x} \mid C, \boldsymbol{n}) &= \frac{p_b(\boldsymbol{x} \mid C)p(\boldsymbol{n}'' \mid \boldsymbol{x}, C)}{\int p_b(\boldsymbol{x} \mid C)p(\boldsymbol{n}'' \mid \boldsymbol{x}, C)d\boldsymbol{x}} \\ &= \frac{p(\boldsymbol{x} \mid C)p(\boldsymbol{n}' \mid \boldsymbol{x}, C)p(\boldsymbol{n}'' \mid \boldsymbol{x}, C)}{\int p(\boldsymbol{x} \mid C)p(\boldsymbol{n}' \mid \boldsymbol{x}, C)p(\boldsymbol{n}'' \mid \boldsymbol{x}, C)d\boldsymbol{x}} \\ &= p(\boldsymbol{x} \mid C, \boldsymbol{n}), \end{aligned}$$

which indicates the conditional posterior of $\boldsymbol{x}$ remains entirely unchanged. The implied marginal posterior on C is $p_b(C \mid \boldsymbol{n}'') \propto p(C)\, p_b(\boldsymbol{n}'' \mid C)$, with

the likelihood on the test data evaluated as $p_b(\boldsymbol{n}'' \mid C) = \int p(\boldsymbol{n}'' \mid \boldsymbol{x}, C)\, p_b(\boldsymbol{x} \mid C) d\boldsymbol{x}$. The use of the prior $p_b(\tilde{\boldsymbol{x}} \mid \tilde{C})$ is similar to the construction of the fractional Bayes factor (FBF) (O'Hagan, 1995). Let $\boldsymbol{u} = \{\boldsymbol{\pi}, \boldsymbol{w}, \boldsymbol{\rho}\}$ denote the parameters other than $\boldsymbol{Z}$ and let $\boldsymbol{u}^*$ denote the maximum likelihood estimate for $\boldsymbol{u}$. We follow O'Hagan (1995) to show that inference on C is as if we were making use of only a fraction $(1-b)$ of the data, with a dimension penalty. In short,

$$p_b(C \mid \boldsymbol{n}'') \propto p(C) p(\boldsymbol{n} \mid \boldsymbol{u}^*, C)^{1-b}\, b^{\,p_C/2},$$

approximately, where $\boldsymbol{u}^*$ is the maximum likelihood estimate of $\boldsymbol{u}$, and p_C is the number of unconstrained parameters in $\boldsymbol{u}$. To obtain this approximation, consider the marginal sampling model under $p_b(\cdot)$, after marginalizing with respect to $\boldsymbol{x}$:

$$\begin{aligned} p_b(\boldsymbol{n}'' \mid C) &= \int p(\boldsymbol{n}'' \mid \boldsymbol{x}, C) p_b(\boldsymbol{x} \mid C) d\boldsymbol{x} \\ &= \int p(\boldsymbol{n}'' \mid \boldsymbol{x}, C)\, \frac{p(\boldsymbol{n}' \mid \boldsymbol{x}, C) p(\boldsymbol{x} \mid C)}{\int p(\boldsymbol{n}' \mid \boldsymbol{x}, C) p(\boldsymbol{x} \mid C) d\boldsymbol{x}}\, d\boldsymbol{x} \\ &= \frac{\int p(\boldsymbol{n} \mid \boldsymbol{x}, C) p(\boldsymbol{x} \mid C) d\boldsymbol{x}}{\int p(\boldsymbol{n}' \mid \boldsymbol{x}, C) p(\boldsymbol{x} \mid C) d\boldsymbol{x}}. \end{aligned}$$

Here we substituted the training sample posterior as (new) prior $p_b(\boldsymbol{x} \mid C)$. The integration includes a marginalization with respect to the discrete $\boldsymbol{Z}$,

$$\begin{aligned} \int p(\boldsymbol{n} \mid \boldsymbol{x}, C) p(\boldsymbol{x} \mid C) d\boldsymbol{x} &= \int \sum_{\boldsymbol{Z}} p(\boldsymbol{n} \mid \boldsymbol{Z}, \boldsymbol{u}, C) p(\boldsymbol{Z} \mid \boldsymbol{u}, C) p(\boldsymbol{u} \mid C) d\boldsymbol{u} \\ &= \int p(\boldsymbol{n} \mid \boldsymbol{u}, C) p(\boldsymbol{u} \mid C) d\boldsymbol{u}, \end{aligned}$$

For the remaining real valued parameters $\boldsymbol{u}$ we use an appropriate one-to-one transformation (e.g. logit transformation) $\boldsymbol{u} \mapsto \tilde{\boldsymbol{u}}$, such that $\tilde{\boldsymbol{u}}$ is unconstrained. To simplify notation we continue to refer to the transformed parameter as $\boldsymbol{u}$ only. Next, under the binomial sampling model $p(\boldsymbol{n}' \mid \boldsymbol{x}, C) \propto p(\boldsymbol{n} \mid \boldsymbol{x}, C)^b$, leading to

$$\begin{aligned} p_b(\boldsymbol{n}'' \mid C) &= \frac{\int p(\boldsymbol{n} \mid \boldsymbol{u}, C) p(\boldsymbol{u} \mid C) d\boldsymbol{u}}{\int p(\boldsymbol{n}' \mid \boldsymbol{u}, C) p(\boldsymbol{u} \mid C) d\boldsymbol{u}} \\ &= \underbrace{\frac{\left[\prod_{t,k} N_{tk}! / (n_{tk1}! \cdots n_{tkG}!)\right]^b}{\prod_{t,k} (bN_{tk})! / \left[(bn_{tk1})! \cdots (bn_{tkG})!\right]}}_{m(\boldsymbol{n})} \cdot \underbrace{\frac{\int p(\boldsymbol{n} \mid \boldsymbol{u}, C) p(\boldsymbol{u} \mid C) d\boldsymbol{u}}{\int p(\boldsymbol{n} \mid \boldsymbol{u}, C)^b p(\boldsymbol{u} \mid C) d\boldsymbol{u}}}_{h_b(\boldsymbol{n} \mid C)}, \end{aligned}$$

Let $m(\boldsymbol{n})$ and $h_b(\boldsymbol{n} \mid C)$ denote the two factors. The first, $m(\boldsymbol{n})$, is a constant term. And the second factor, $h_b(\boldsymbol{n} \mid C)$, has exactly the same form as equation (12) in O'Hagan (1995), who shows

$$h_b(\boldsymbol{n} \mid C) \approx p(\boldsymbol{n} \mid \boldsymbol{u}^*, C)^{1-b} b^{\,p_C/2}$$

Let $N = \sum_{t,k} N_{tk}$. The argument of Gelfand and Dey (1994) (case (e)) suggests that the error in this approximation is of order $O(1/N^2)$ (note that Gelfand and Dey use expansion around the M.A.P. while O'Hagan uses expansions around the M.L.E.). This establishes the stated approximation of the posterior $p_b(C \mid \boldsymbol{n}'') \approx k \cdot p(C) p(\boldsymbol{n} \mid \boldsymbol{u}^*, C)^{1-b}\, b^{\,p_C/2}$, approximately.

A.3 Calibration of b

The construction of an informative prior $p_b(\boldsymbol{x} \mid C) \equiv p(\boldsymbol{x} \mid \boldsymbol{n}', C)$ based on a training sample $\boldsymbol{n}'$ is similar to the use of a training sample in the construction of the fractional Bayes factor (FBF) of O'Hagan (1995). However, there is an important difference. In the FBF construction the aim is to replace a noninformative prior in the evaluation of a Bayes factor. A minimally informative prior p_b with small b suffices. In contrast, here $p_b(\boldsymbol{x} \mid C)$ is (also) used as proposal distribution in the trans-dimensional MCMC. The aim is to construct a good proposal that fits the data well and thus leads to good acceptance probabilities and a well mixing Markov chain. With the highly informative multinomial likelihood we find that we need a large training sample, that is, large b. In Appendix A.2 we show that the effect of using p_b is that $p(C \mid \boldsymbol{n})$ is approximated by

$$p_b(C \mid \boldsymbol{n}'') \propto p(C) p(\boldsymbol{n} \mid \boldsymbol{u}^*, C)^{1-b} b^{\,p_C/2},$$

where $\boldsymbol{u} = \{\boldsymbol{\pi}, \boldsymbol{w}, \boldsymbol{\rho}\}$ are the parameters other than $\boldsymbol{Z}$, $\boldsymbol{u}^*$ is the maximum likelihood estimate of $\boldsymbol{u}$, and p_C is the number of unconstrained parameters in $\boldsymbol{u}$. Importantly, however, inference on other parameters, $p(\boldsymbol{x} \mid C, \boldsymbol{n})$, remains entirely unchanged.

We therefore recommend to focus on inference for C when calibrating b. Carrying out simulation studies with single and multi-sample data, we find that the simulation truth for C is best recovered with a test sample size $(1-b)\sum_{t=1}^{T}\sum_{k=1}^{K} N_{tk} \approx 160/T$, where N_{tk} is the total number of short reads

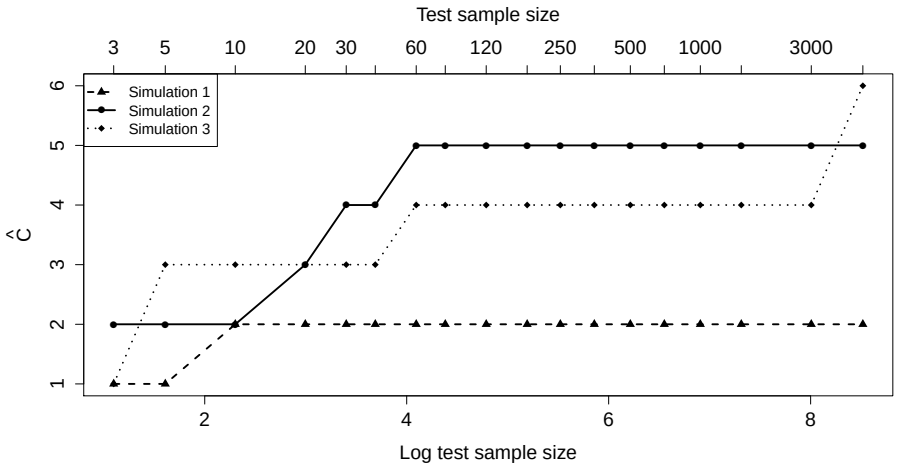


Figure A.1: Path plot of $\hat{C}$ with different test sample sizes for three simulations. The true number of subclones are 2, 4, and 3 for simulations 1, 2, and 3, respectively.

mapped to mutation pair k in sample t. For example, Figure A.1 plots the posterior mode of C against test sample sizes for simulated data in three simulations. For multi-sample data we find (empirically, by simulation) that the test sample size can be reduced, at a rate linear in T. In summary we recommend to set b to achieve a test sample size around $160/T$. Following these guidelines, in our implementation in the previous section, we used values $b = 0.992$ for simulation 1, $b = 0.9998$ for simulation 2, and $b = 0.999911$ for simulation 3.

A.4 Validation of the MCMC scheme

Validation of the correctness of the sampler. We first use a scheme to validate the correctness of our MCMC sampler in the style of Geweke (2004).

The joint density of the parameters and observed data can be written as $p(\boldsymbol{x}, \boldsymbol{n}) = p(\boldsymbol{x})p(\boldsymbol{n} \mid \boldsymbol{x})$. Let g be any function $g : \mathcal{X} \times \mathcal{N} \to \mathbb{R}$ satisfying $\mathrm{Var}[g(\boldsymbol{x}, \boldsymbol{n})] < \infty$, where $\mathcal{X}$ and $\mathcal{N}$ represent sample spaces of $\boldsymbol{x}$ and $\boldsymbol{n}$, respectively. Denote by $\bar{g} = E[g(\boldsymbol{x}, \boldsymbol{n})]$, which can be evaluated by independent Monte Carlo simulation from the joint distribution, or in some cases might be known exactly as prior mean of functions of parameters only. Alternatively, the same mean can be estimated by a different Markov chain Monte Carlo scheme for the joint distribution, constructed by an initial draw $\boldsymbol{x}^{(0)} \sim p(\boldsymbol{x})$, followed by $\boldsymbol{n}^{(l)} \sim p(\boldsymbol{n} \mid \boldsymbol{x}^{(l-1)})$, $\boldsymbol{x}^{(l)} \sim q(\boldsymbol{x} \mid \boldsymbol{x}^{(l-1)}, \boldsymbol{n}^{(l)})$, and $g^{(l)} = g(\boldsymbol{n}^{(l)}, \boldsymbol{x}^{(l)})$, for $l = 1, \ldots, L$. Under certain conditions, $\{\boldsymbol{x}^{(l)}, \boldsymbol{n}^{(l)}\}$ is ergodic with unique invariant kernel $p(\boldsymbol{x}, \boldsymbol{n})$. If the simulator is error-free, one should have

$$(\bar{g}^{(L)} - \bar{g}) / \left[L^{-1}\hat{S}_g(0)\right]^{1/2} \xrightarrow{d} N(0, 1), \tag{A.1}$$

where $\hat{S}_g(0)$ is consistent spectral density estimate for $\{g^{(l)}, l = 1, \ldots, L\}$. In our application, we take $g(\boldsymbol{x}, \boldsymbol{n}) = w_{tc}$ and p_{tkg}. We set the number of samples $T = 4$, and the number of mutation pairs $K = 80$. Since our inference on C is not a standard MCMC, we fix $C = 3$ here and only consider $\boldsymbol{x} = \{\boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{w}, \boldsymbol{\rho}\}$. Table A.1 shows the statistic (A.1) for five randomly selected w_{tc} and p_{tkg}. The recorded z-scores show no evidence for errors in the simulator.

Convergence diagnostic. Next, we present some convergence diagnostics of our MCMC chain, including trace plots, autocorrelation plots, and test statistics described in Geweke (1991). Those convergence diagnostics are based on the posterior distribution of parameters $p(\boldsymbol{x} \mid \boldsymbol{n}) \propto p(\boldsymbol{x})p(\boldsymbol{n} \mid \boldsymbol{x})$. Let g

Test statistic	z-score	p-value
w_{12}	-0.4736149	0.6357745
w_{43}	-1.441169	0.149537
$p_{1,23,3}$	0.9413715	0.3465145
$p_{3,60,7}$	1.388424	0.1650079
$p_{2,13,2}$	-0.6051894	0.5450532

Table A.1: Geweke's statistics and the corresponding z-scores and p-values.

be any function $g : \mathcal{X} \to \mathbb{R}$, and $g^{(l)} = g(\boldsymbol{x}^{(l)})$ where $\{\boldsymbol{x}^{(l)}, l = 1, \ldots, L\}$ are samples from the posterior. Let

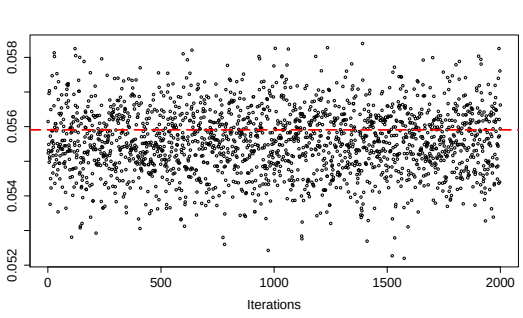
$$\bar{g}_L^A = L_A^{-1} \sum_{l=1}^{L_A} g^{(l)}, \qquad \bar{g}_L^B = L_B^{-1} \sum_{l=l^*}^{L} g^{(l)} \quad (l^* = L - L_B + 1),$$

and let $\hat{S}_g^A(0)$ and $\hat{S}_g^B(0)$ denote consistent spectral density estimates for $\{g^{(l)}, l = 1, \ldots, L_A\}$ and $\{g^{(l)}, l = l^*, \ldots, L\}$, respectively. If the ratios L_A/L and L_B/L are fixed, with $(L_A + L_B)/L < 1$, then as $L \to \infty$,

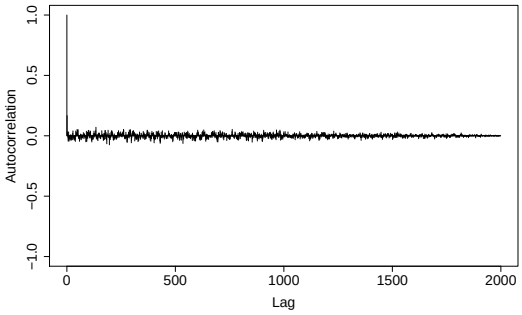
$$(\bar{g}_L^A - \bar{g}_L^B) / \left[L_A^{-1} \hat{S}_g^A(0) + L_B^{-1} \hat{S}_g^B(0) \right]^{1/2} \xrightarrow{d} N(0, 1).$$

In our application, a reasonable choice of g is $g(\boldsymbol{x}) = p_{tkg}(\boldsymbol{Z}, \boldsymbol{w}, \boldsymbol{\rho})$. We use simulation 2 as an example, and show some plots and Geweke's statistics for some randomly chosen p_{tkg}. Figure A.2(a, c) shows the trace plot for p_{tkg}, with the red dashed line denoting the true value. The posterior samples are centered around the true value and symmetrically distributed. Figure A.2(b, d) shows the autocorrelation plot for p_{tkg}. The autocorrelations between MCMC draws are small, indicating good mixing of the chain. Table A.2 shows the Geweke's statistics for five randomly selected p_{tkg}. The p-values for

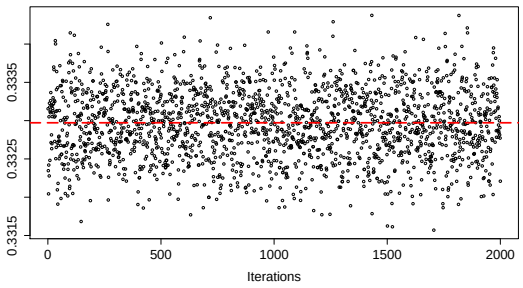
them are all greater than 0.05, representing those statistics pass the Geweke's diagnostic, and there is no strong evidence that the chain does not converge.



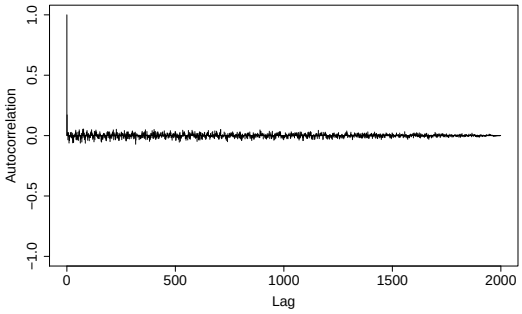
(a) Trace plot of $p_{1,5,2}$



(b) Autocorrelation plot of $p_{1,5,2}$



(c) Trace plot of $p_{3,68,7}$



(d) Autocorrelation plot of $p_{3,68,7}$

Figure A.2: Convergence check for Simulation 2.

Test statistic	z-score	p-value
$p_{1,5,2}$	0.1748906	0.8611656
$p_{3,68,7}$	-0.02609703	0.9791799
$p_{4,25,5}$	0.4454738	0.6559774
$p_{2,96,4}$	-1.341994	0.179598
$p_{1,66,1}$	-0.2727737	0.7850272

Table A.2: Convergence check for Simulation 2.

Appendix B

Appendix for Chapter 4

B.1 MCMC Implementation Details

We introduce some notation as follows. First considering the responses. Denote by N_s the number of subjects having dropout pattern s, $s = 2, \ldots, J$. Let $\boldsymbol{y}_{js}$ denote the subjects' responses at time j in pattern s, and $\bar{Y}_{js}$ denote the subjects' histories through the first j times in pattern s, i.e.

$$\begin{aligned}\boldsymbol{y}_{js} &= (y_{1js}, y_{2js}, \ldots, y_{N_s,j,s})^T, \\ \bar{Y}_{js} &= (\boldsymbol{y}_{1s}, \boldsymbol{y}_{2s}, \ldots, \boldsymbol{y}_{js})\,.\end{aligned}$$

Let $\boldsymbol{y}_{\text{vec0}}$ denote the initial responses (with no past) for all subjects, and $\boldsymbol{y}_{\text{vec}}$ denote the subsequent responses (with measured pasts) for all subjects,

$$\begin{aligned}\boldsymbol{y}_{\text{vec0}} &= \left(\boldsymbol{y}_{12}^T, \boldsymbol{y}_{13}^T, \ldots, \boldsymbol{y}_{1J}^T\right)^T \\ \boldsymbol{y}_{\text{vec}} &= \left(\boldsymbol{y}_{22}^T, \boldsymbol{y}_{23}^T, \boldsymbol{y}_{33}^T, \ldots, \boldsymbol{y}_{2J}^T, \ldots, \boldsymbol{y}_{JJ}^T\right)^T.\end{aligned}$$

We then consider the means and covariate matrices for the responses. Let $\boldsymbol{a}_{js}$ denote the vector of random variables (we abuse notation slightly, let $\boldsymbol{a}_{js}$ include $\bar{Y}_{j-1,s}\boldsymbol{\Phi}_{js}$ when $j \geq 2$, to simplify notation),

$$\boldsymbol{a}_{js} = \begin{cases} \left(a_0(\boldsymbol{v}_{1s}, j, s), \ldots, a_0(\boldsymbol{v}_{N_s,s}, j, s)\right)^T, & \text{if } j = 1; \\ \left(a(\boldsymbol{v}_{1s}, j, s), \ldots, a(\boldsymbol{v}_{N_s,s}, j, s)\right)^T + \bar{Y}_{j-1,s}\boldsymbol{\Phi}_{js}, & \text{if } j \geq 2. \end{cases}$$

The vector $\boldsymbol{a}_{js}$ is the mean of $\boldsymbol{y}_{js}$. Let $\boldsymbol{a}_0$ and $\boldsymbol{a}$ denote the vector of random variables,

$$\begin{aligned}\boldsymbol{a}_0 &= \left(\boldsymbol{a}_{12}^T, \boldsymbol{a}_{13}^T, \ldots, \boldsymbol{a}_{1J}^T\right)^T \\ \boldsymbol{a} &= \left(\boldsymbol{a}_{22}^T, \boldsymbol{a}_{23}^T, \boldsymbol{a}_{33}^T, \ldots, \boldsymbol{a}_{2J}^T, \ldots, \boldsymbol{a}_{JJ}^T\right)^T.\end{aligned}$$

Denote by

$$\begin{aligned}\Sigma_{y_0} &= \operatorname{diag}\left(\sigma_{12}^2 I_{N_2}, \ldots, \sigma_{1J}^2 I_{N_J}\right), \\ \Sigma_y &= \operatorname{diag}\left(\sigma_{22}^2 I_{N_2}, \sigma_{23}^2 I_{N_3}, \sigma_{33}^2 I_{N_3}, \ldots, \sigma_{2J}^2 I_{N_J}, \ldots, \sigma_{JJ}^2 I_{N_J}\right).\end{aligned}$$

Thus, the likelihoods for the initial responses and subsequent responses are

$$\begin{aligned}\boldsymbol{y}_{\text{vec0}} \mid \boldsymbol{a}_0, \Sigma_{y_0} &\sim N(\boldsymbol{a}_0, \Sigma_{y_0}), \\ \boldsymbol{y}_{\text{vec}} \mid \boldsymbol{a}, \Sigma_y &\sim N(\boldsymbol{a}, \Sigma_y).\end{aligned}$$

Next, we consider the priors for $\boldsymbol{a}_0$ and $\boldsymbol{a}$. Denote by

$$\begin{aligned}\boldsymbol{\theta}_0 &= (\boldsymbol{\beta}_0, \boldsymbol{b}_0), \\ \boldsymbol{\theta} &= (\boldsymbol{\beta}, \boldsymbol{b}, \boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_3).\end{aligned}$$

Let $\boldsymbol{D}_0$ and $\boldsymbol{D}$ denote the exponential distance matrices for $\boldsymbol{a}_0$ and $\boldsymbol{a}$,

$$\begin{aligned}[\boldsymbol{D}_0]_{ijs,i'j's'} &= D_0(\boldsymbol{v}_{is}, j, s; \boldsymbol{v}_{i's}, j', s'), \\ [\boldsymbol{D}]_{ijs,i'j's'} &= D(\boldsymbol{v}_{is}, j, s; \boldsymbol{v}_{i's}, j', s').\end{aligned}$$

We have

$$\begin{aligned}\boldsymbol{a}_0 \mid \boldsymbol{\theta}_0, \kappa_0^2 &\sim N(X_{\theta_0}\boldsymbol{\theta}_0, \kappa_0^2 \boldsymbol{D}_0 + \tilde{\kappa}_0^2 I), \\ \boldsymbol{a} \mid \boldsymbol{\theta}, \kappa^2 &\sim N(X_\theta \boldsymbol{\theta}, \kappa^2 \boldsymbol{D} + \tilde{\kappa}^2 I),\end{aligned}$$

where X_{θ_0} and X_θ are the design matrices corresponding to Equation (4.4).

Denote by $\boldsymbol{C}_0 = \kappa_0^2 \boldsymbol{D}_0 + \tilde{\kappa}_0^2 I$ and $\boldsymbol{C} = \kappa^2 \boldsymbol{D} + \tilde{\kappa}^2 I$. Integrating out $\boldsymbol{a}_0$ and $\boldsymbol{a}$, the (marginal) likelihoods become

$$\begin{aligned} \boldsymbol{y}_{\text{vec}0} \mid \boldsymbol{\theta}_0, \Sigma_{y_0}, \kappa_0^2 &\sim N(X_{\theta_0}\boldsymbol{\theta}_0, \Sigma_{y_0} + \boldsymbol{C}_0), \\ \boldsymbol{y}_{\text{vec}} \mid \boldsymbol{\theta}, \Sigma_y, \kappa^2 &\sim N(X_\theta \boldsymbol{\theta}, \Sigma_y + \boldsymbol{C}). \end{aligned}$$

Update $\boldsymbol{a}_0$ and $\boldsymbol{a}$. It is not unusual to integrate out $\boldsymbol{a}_0$ and $\boldsymbol{a}$ for posterior inference on Gaussian process. However, we find that including $\boldsymbol{a}_0$ and $\boldsymbol{a}$ in the posterior inference would improve the mixing of the Markov chain. Therefore, we update $\boldsymbol{a}_0$ and $\boldsymbol{a}$ at each iteration.

1. The likelihood and prior for $\boldsymbol{a}_0$ are

$$\begin{aligned} \boldsymbol{y}_{\text{vec}0} \mid \boldsymbol{a}_0, \Sigma_{y_0} &\sim N(\boldsymbol{a}_0, \Sigma_{y_0}), \\ \boldsymbol{a}_0 \mid \boldsymbol{\theta}_0, \kappa_0^2 &\sim N(X_{\theta_0}\boldsymbol{\theta}_0, C_0), \end{aligned}$$

which lead to the posterior

$$\begin{aligned} \boldsymbol{a}_0 \mid \boldsymbol{\theta}_0, \kappa_0^2, \Sigma_{y_0}, \boldsymbol{y}_{\text{vec}0} &\sim N(\boldsymbol{a}_0^*, \Sigma_{a_0}^*), \text{ where} \\ \Sigma_{a_0}^* &= [C_0^{-1} + \Sigma_{y_0}^{-1}]^{-1}, \\ \boldsymbol{a}_0^* &= \Sigma_{a_0}^* [C_0^{-1} X_{\theta_0}\boldsymbol{\theta}_0 + \Sigma_{y_0}^{-1} \boldsymbol{y}_{\text{vec}0}]. \end{aligned}$$

2. The likelihood and prior for $\boldsymbol{a}$ are

$$\begin{aligned} \boldsymbol{y}_{\text{vec}} \mid \boldsymbol{a}, \Sigma_y &\sim N(\boldsymbol{a}, \Sigma_y), \\ \boldsymbol{a} \mid \boldsymbol{\theta}, \kappa^2 &\sim N(X_\theta \boldsymbol{\theta}, C), \end{aligned}$$

which lead to the posterior

$$\begin{aligned}\boldsymbol{a} \mid \boldsymbol{\theta}, \kappa^2, \Sigma_y, \boldsymbol{y}_{\text{vec}} &\sim N(\boldsymbol{a}^*, \Sigma_a^*), \text{ where} \\ \Sigma_a^* &= [C^{-1} + \Sigma_y^{-1}]^{-1}, \\ \boldsymbol{a}^* &= \Sigma_a^*[C^{-1} X_\theta \boldsymbol{\theta} + \Sigma_y^{-1} \boldsymbol{y}_{\text{vec}}].\end{aligned}$$

Update κ_0^2 and κ^2. 1. The likelihood and prior for κ_0^2 are

$$\begin{aligned}\boldsymbol{a}_0 \mid \boldsymbol{\theta}_0, \kappa_0^2 &\sim N(X_{\theta_0}\boldsymbol{\theta}_0, \kappa_0^2 \boldsymbol{D}_0 + \tilde{\kappa}_0^2 I), \\ \kappa_0^2 &\sim \text{IG}(\lambda_1^{\kappa_0}, \lambda_2^{\kappa_0}).\end{aligned}$$

The posterior for κ_0^2 is

$$p(\kappa_0^2 \mid \boldsymbol{\theta}_0, \boldsymbol{a}_0) \propto p_N(\boldsymbol{a}_0 \mid X_{\theta_0}\boldsymbol{\theta}_0, \kappa_0^2 \boldsymbol{D}_0 + \tilde{\kappa}_0^2 I) \cdot p_{\text{IG}}(\kappa_0^2 \mid \lambda_1^{\kappa_0}, \lambda_2^{\kappa_0}),$$

where $p_N(\boldsymbol{x} \mid \boldsymbol{\mu}, \Sigma)$ represents (multivariate) normal density at $\boldsymbol{x}$ with mean $\boldsymbol{\mu}$ and covariance matrix Σ, and $p_{\text{IG}}(x \mid a, b)$ represents inverse gamma density at x with shape parameter a and rate parameter b. We use Metropolis-Hastings step to update κ_0^2.

2. The likelihood and prior for κ^2 are

$$\begin{aligned}\boldsymbol{a} \mid \boldsymbol{\theta}, \kappa^2 &\sim N(X_\theta \boldsymbol{\theta}, \kappa^2 \boldsymbol{D} + \tilde{\kappa}^2 I), \\ \kappa^2 &\sim \text{IG}(\lambda_1^{\kappa}, \lambda_2^{\kappa}).\end{aligned}$$

The posterior for κ^2 is

$$p(\kappa^2 \mid \boldsymbol{\theta}, \boldsymbol{a}) \propto p_N(\boldsymbol{a} \mid X_\theta \boldsymbol{\theta}, \kappa^2 \boldsymbol{D} + \tilde{\kappa}^2 I) \cdot p_{\text{IG}}(\kappa^2 \mid \lambda_1^{\kappa}, \lambda_2^{\kappa}).$$

We use Metropolis-Hastings step to update κ^2.

Update Σ_{y_0} and Σ_y. The likelihood and prior for σ_{js}^2 are

$$\boldsymbol{y}_{js} \mid \boldsymbol{a}_{js}, \sigma_{js}^2 \sim N(\boldsymbol{a}_{js}, \sigma_{js}^2 I),$$
$$\sigma_{js}^2 \mid \lambda_\sigma, g_\sigma \sim \mathrm{IG}(\lambda_\sigma, \lambda_\sigma g_\sigma).$$

The posterior for σ_{js}^2 is

$$\sigma_{js}^2 \mid \lambda_\sigma, g_\sigma, \boldsymbol{a}_{js}, \boldsymbol{y}_{js} \sim \mathrm{IG}\left(\lambda_\sigma + \frac{N_s}{2},\ \lambda_\sigma g_\sigma + \frac{RSS_{js}}{2}\right),$$

where $RSS_{js} = \|\boldsymbol{y}_{js} - \boldsymbol{a}_{js}\|_2^2$.

There are two hyperparameters related to σ_{js}^2: λ_σ and g_σ. Their conditional posteriors are

$$p(\lambda_\sigma \mid \{\sigma_{js}^2\}, g_\sigma) \propto \frac{(g_\sigma \lambda_\sigma)^{(2+J)(J-1)\lambda_\sigma/2}}{\Gamma(\lambda_\sigma)^{(2+J)(J-1)/2}} \prod_{j,s} \left(\sigma_{js}^2\right)^{-(\lambda_\sigma - 1)} \cdot \exp\left(-\sum_{j,s} \frac{g_\sigma}{\sigma_{js}^2}\lambda_\sigma\right) \exp\left(-\frac{1}{\lambda_\sigma - 2}\right),$$

and

$$g_\sigma \mid \{\sigma_{js}^2\}, \lambda_\sigma \sim \mathrm{Gamma}\left(\frac{(2+J)(J-1)}{2}\lambda_\sigma + 1,\ \sum_{j,s} \frac{\lambda_\sigma}{\sigma_{js}^2} + 1\right).$$

We use Metropolis-Hastings step to update λ_σ.

Update $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}$. We integrate out $\boldsymbol{a}_0$ and $\boldsymbol{a}$ to update $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}$. The likelihoods become

$$\boldsymbol{y}_{\mathrm{vec}0} \mid \boldsymbol{\theta}_0, \Sigma_{y_0}, \kappa_0^2 \sim N(X_{\theta_0}\boldsymbol{\theta}_0, \Sigma_{y_0} + C_0),$$
$$\boldsymbol{y}_{\mathrm{vec}} \mid \boldsymbol{\theta}, \Sigma_y, \kappa^2 \sim N(X_\theta \boldsymbol{\theta}, \Sigma_y + C).$$

1. For $\boldsymbol{\theta}_0$, the prior is

$$\boldsymbol{\theta}_0 \mid \tilde{\boldsymbol{\beta}}_0, \sigma^2_{\beta_0}, \rho_0, \tilde{b}_0, \sigma^2_{b_0}, \gamma_{b_0} \sim N(\tilde{\boldsymbol{\theta}}_0, \Sigma_{\theta_0}),$$

where $\tilde{\boldsymbol{\theta}}_0 = (X_\beta \tilde{\boldsymbol{\beta}}_0, \mathbf{1}\tilde{b}_0)$, and

$$\Sigma_\theta = \operatorname{diag}\big(\sigma^2_{\beta_0}\Sigma_\beta(\rho_0), \sigma^2_{b_0}(I - \gamma_{b_0}W_{b_0})^{-1}\mathcal{N}_{b_0}\big).$$

Thus, the posterior of $\boldsymbol{\theta}_0$ is

$$\begin{aligned}
&\boldsymbol{\theta}_0 \mid \boldsymbol{y}_{\text{vec}0}, \ldots \sim N(\boldsymbol{\theta}^*_0, \Sigma^*_{\theta_0}), \quad \text{where} \\
&\Sigma^*_{\theta_0} = \left[\Sigma^{-1}_{\theta_0} + X^T_{\theta_0}(\Sigma_{y_0} + C_0)^{-1}X_{\theta_0}\right]^{-1}, \\
&\boldsymbol{\theta}^*_0 = \Sigma^*_{\theta_0}\left[\Sigma^{-1}_{\theta_0}\tilde{\boldsymbol{\theta}}_0 + X^T_{\theta_0}(\Sigma_{y_0} + C_0)^{-1}\boldsymbol{y}_{\text{vec}0}\right].
\end{aligned}$$

2. For $\boldsymbol{\theta}$, the prior is

$$\boldsymbol{\theta} \mid \tilde{\boldsymbol{\beta}}, \sigma^2_\beta, \rho, \tilde{b}, \sigma^2_b, \gamma_b, \tilde{\phi}_1, \sigma^2_{\phi_1}, \gamma_{\phi_1}, \sigma^2_{\phi_2}, \sigma^2_{\phi_3} \sim N(\tilde{\boldsymbol{\theta}}, \Sigma_\theta),$$

where $\tilde{\boldsymbol{\theta}} = (X_\beta\tilde{\boldsymbol{\beta}}, \mathbf{1}\tilde{b}, \mathbf{1}\tilde{\phi}_1, \mathbf{0}, \mathbf{0})$, and

$$\begin{aligned}
\Sigma_\theta = \operatorname{diag}\big(&\sigma^2_\beta\Sigma_\beta(\rho), \sigma^2_b(I - \gamma_b W_b)^{-1}\mathcal{N}_b, \\
&\sigma^2_{\phi_1}(I - \gamma_{\phi_1}W_{\phi_1})^{-1}\mathcal{N}_{\phi_1}, \sigma^2_{\phi_2}I, \sigma^2_{\phi_3}I\big).
\end{aligned}$$

Thus, the posterior of $\boldsymbol{\theta}$ is

$$\begin{aligned}
&\boldsymbol{\theta} \mid \boldsymbol{y}_{\text{vec}}, \ldots \sim N(\boldsymbol{\theta}^*, \Sigma^*_\theta), \quad \text{where} \\
&\Sigma^*_\theta = \left[\Sigma^{-1}_\theta + X^T_\theta(\Sigma_y + C)^{-1}X_\theta\right]^{-1}, \\
&\boldsymbol{\theta}^* = \Sigma^*_\theta\left[\Sigma^{-1}_\theta\tilde{\boldsymbol{\theta}} + X^T_\theta(\Sigma_y + C)^{-1}\boldsymbol{y}_{\text{vec}}\right].
\end{aligned}$$

Hyperparameters related to $\boldsymbol{\beta}$ and $\boldsymbol{\beta}_0$. There are three hyperparameters related to $\boldsymbol{\beta}$: $\tilde{\boldsymbol{\beta}}$, σ_β^2 and ρ. The conditional posteriors are as follows.

1. Conditional posterior of $\tilde{\boldsymbol{\beta}}$:

$$\begin{aligned}\tilde{\boldsymbol{\beta}} \mid \boldsymbol{\beta}, \sigma_\beta^2, \rho \sim N(\tilde{\boldsymbol{\beta}}^*, \Sigma_{\tilde{\boldsymbol{\beta}}}^*), \quad \text{where} \\ \Sigma_{\tilde{\boldsymbol{\beta}}}^* = \left[\frac{1}{\delta_\beta^2} I + \frac{1}{\sigma_\beta^2} X_\beta' \Sigma_\beta(\rho)^{-1} X_\beta\right]^{-1}, \\ \tilde{\boldsymbol{\beta}}^* = \Sigma_{\tilde{\boldsymbol{\beta}}}^* \left[\frac{1}{\sigma_\beta^2} X_\beta' \Sigma_\beta(\rho)^{-1} \boldsymbol{\beta}\right].\end{aligned}$$

2. Conditional posterior of σ_β^2:

$$\sigma_\beta^2 \mid \boldsymbol{\beta}, \tilde{\boldsymbol{\beta}}, \rho \sim \text{IG}\left[\lambda_1^\beta + \frac{(J-1)Q}{2}, \lambda_2^\beta + \frac{1}{2}(\boldsymbol{\beta} - X_\beta \tilde{\boldsymbol{\beta}})' \Sigma_\beta(\rho)^{-1} (\boldsymbol{\beta} - X_\beta \tilde{\boldsymbol{\beta}})\right].$$

3. Conditional posterior of ρ:

$$\begin{aligned}& p(\rho \mid \boldsymbol{\beta}, \tilde{\boldsymbol{\beta}}, \sigma_\beta^2) \\ &\propto \det[\sigma_\beta^{-2} \Sigma_\beta(\rho)^{-1}]^{1/2} \exp\left[-\frac{1}{2\sigma_\beta^2} (\boldsymbol{\beta} - X_\beta \tilde{\boldsymbol{\beta}})' \Sigma_\beta(\rho)^{-1} (\boldsymbol{\beta} - X_\beta \tilde{\boldsymbol{\beta}})\right] \\ &\propto (1-\rho^2)^{Q/2} \exp\left[-\frac{1}{2\sigma_\beta^2}\left(\rho^2 R_{\beta 1} - 2\rho R_{\beta 2}\right)\right],\end{aligned}$$

where

$$R_{\beta 1} = \sum_{s=3}^{J-1} \|\boldsymbol{\beta}_s - \tilde{\boldsymbol{\beta}}\|_2^2, \qquad R_{\beta 2} = \sum_{s=3}^{J} (\boldsymbol{\beta}_s - \tilde{\boldsymbol{\beta}})'(\boldsymbol{\beta}_{s-1} - \tilde{\boldsymbol{\beta}}).$$

We use the following properties to derive the conditional posterior of ρ. The

inverse and determinant of $\Sigma_\beta(\rho)$ are

$$\Sigma_\beta(\rho)^{-1} = \begin{pmatrix} I & -\rho I & & & & \\ -\rho I & (1+\rho^2)I & -\rho I & & & \\ & -\rho I & (1+\rho^2)I & -\rho I & & \\ & & -\rho I & \ddots & \ddots & \\ & & & \ddots & (1+\rho^2)I & -\rho I \\ & & & & -\rho I & I \end{pmatrix},$$

and $\det[\Sigma_\beta(\rho)^{-1}] = (1-\rho^2)^Q$, respectively. To update $\tilde{\boldsymbol{\beta}}$ and $\sigma^2_{\tilde\beta}$, we use regular Gibbs steps. To update ρ, given $\{\boldsymbol{\beta}, \tilde{\boldsymbol{\beta}}, \sigma^2_{\tilde\beta}\}$ we can easily evaluate its posterior on the $[0,1]$ grid, and sample from it.

Similarly, there are three hyperparameters related to $\boldsymbol{\beta}_0$: $\tilde{\boldsymbol{\beta}}_0$, $\sigma^2_{\tilde\beta_0}$ and ρ_0. Their conditional posteriors have exactly the same form as those for $\tilde{\boldsymbol{\beta}}$, $\sigma^2_{\tilde\beta}$ and ρ.

Hyperparameters related to $\boldsymbol{b}$ and $\boldsymbol{b}_0$. There are three hyperparameters related to $\boldsymbol{b}$: $\tilde{b}$, σ^2_b and γ_b. The conditional posteriors are as follows.

1. Conditional posterior of $\tilde{b}$:

$$\begin{aligned} \tilde{b} \mid \boldsymbol{b}, \sigma^2_b, \gamma_b \sim N(\tilde{b}^*, \delta^{*2}_{\tilde b}), \quad \text{where} \\ \delta^{*2}_{\tilde b} = \left[\frac{1}{\delta^2_b} + \frac{1}{\sigma^2_b}\mathbf{1}^T\mathcal{N}_b^{-1}(I-\gamma_b W_b)\mathbf{1}\right]^{-1}, \\ \tilde{b}^* = \delta^{*2}_{\tilde b}\left[\frac{1}{\sigma^2_b}\mathbf{1}^T\mathcal{N}_b^{-1}(I-\gamma_b W_b)\boldsymbol{b}\right]. \end{aligned}$$

2. Conditional posterior of σ^2_b:

$$\sigma^2_b \mid \boldsymbol{b}, \tilde{b}, \gamma_b \sim \text{IG}\left[\lambda^b_1 + \frac{\dim(\boldsymbol{b})}{2}, \lambda^b_2 + \frac{1}{2}(\boldsymbol{b}-\mathbf{1}\tilde{b})'\mathcal{N}_b^{-1}(I-\gamma_b W_b)(\boldsymbol{b}-\mathbf{1}\tilde{b})\right].$$

3. Conditional posterior of γ_b:

$$p(\gamma_b \mid \boldsymbol{b}, \tilde{b}, \sigma_b^2) \propto \det(I - \gamma_b W_b)^{1/2} \cdot \exp\left[\gamma_b \cdot \frac{1}{2\sigma_b^2}(\boldsymbol{b} - \mathbf{1}\tilde{b})' \mathcal{N}_b^{-1} W_b (\boldsymbol{b} - \mathbf{1}\tilde{b})\right].$$

To update $\tilde{b}$ and σ_b^2, we use regular Gibbs steps. To update γ_b, given $\{\boldsymbol{b}, \tilde{b}, \sigma_b^2\}$ we can easily evaluate its posterior on the $[0, 1]$ grid, and sample from it. To facilitate computation, we can calculate $\det(I - \gamma_b W_b)^{1/2}$ on the $[0, 1]$ grid, save the values and use it at each iteration.

Similarly, there are three hyperparameters related to $\boldsymbol{b}_0$: $\tilde{b}_0$, $\sigma_{b_0}^2$ and γ_{b_0}. Their conditional posteriors have exactly the same form as those for $\tilde{b}$, σ_b^2 and γ_b.

Hyperparameters related to $\boldsymbol{\phi}_1$. There are three hyperparameters related to $\boldsymbol{\phi}_1$: $\tilde{\phi}_1$, $\sigma_{\phi_1}^2$ and γ_{ϕ_1}. The conditional posteriors are as follows.

1. Conditional posterior of $\tilde{\phi}_1$:

$$\tilde{\phi}_1 \mid \boldsymbol{\phi}_1, \sigma_{\phi_1}^2, \gamma_\phi \sim N(\tilde{\phi}^*, \delta_{\tilde{\phi}}^{*2}), \quad \text{where}$$

$$\delta_{\tilde{\phi}}^{*2} = \left[\frac{1}{\delta_{\phi_1}^2} + \frac{1}{\sigma_{\phi_1}^2}\mathbf{1}'\mathcal{N}_{\phi_1}^{-1}(I - \gamma_{\phi_1} W_{\phi_1})\mathbf{1}\right]^{-1},$$

$$\tilde{\phi}^* = \delta_{\tilde{\phi}}^{*2}\left[\frac{1}{\delta_{\phi_1}^2} \cdot 1 + \frac{1}{\sigma_{\phi_1}^2}\mathbf{1}'\mathcal{N}_{\phi_1}^{-1}(I - \gamma_{\phi_1} W_{\phi_1})\boldsymbol{\phi}_1\right].$$

2. Conditional posterior of $\sigma^2_{\phi_1}$:

$$\sigma^2_{\phi_1} \mid \boldsymbol{\phi}_1, \tilde{\phi}_1, \gamma_{\phi_1} \sim \text{IG}\Bigg[\lambda_1^{\phi_1} + \frac{\dim(\boldsymbol{\phi}_1)}{2}, \\ \lambda_2^{\phi_1} + \frac{1}{2}(\boldsymbol{\phi}_1 - \mathbf{1}\tilde{\phi}_1)'\mathcal{N}_{\phi_1}^{-1}(I - \gamma_{\phi_1}W_{\phi_1})(\boldsymbol{\phi}_1 - \mathbf{1}\tilde{\phi}_1)\Bigg].$$

3. Conditional posterior of γ_{ϕ_1}:

$$p(\gamma_{\phi_1} \mid \boldsymbol{\phi}_1, \tilde{\phi}_1, \sigma^2_{\phi_1}) \propto \det(I - \gamma_{\phi_1}W_{\phi_1})^{1/2} \cdot \\ \exp\left[\gamma_{\phi_1} \cdot \frac{1}{2\sigma^2_{\phi_1}}(\boldsymbol{\phi}_1 - \mathbf{1}\tilde{\phi}_1)'\mathcal{N}_{\phi_1}^{-1}W_{\phi_1}(\boldsymbol{\phi}_1 - \mathbf{1}\tilde{\phi}_1)\right].$$

Hyperparameters related to $\boldsymbol{\phi}_2$ and $\boldsymbol{\phi}_3$. There is one hyperparameter related to $\boldsymbol{\phi}_2$: $\sigma^2_{\phi_2}$. The conditional posterior is

$$\sigma^2_{\phi_2} \mid \boldsymbol{\phi}_2 \sim \text{IG}\left[\lambda_1^{\phi_2} + \frac{1}{2}\dim(\boldsymbol{\phi}_2),\ \lambda_2^{\phi_2} + \frac{1}{2}\boldsymbol{\phi}_2^T\boldsymbol{\phi}_2\right].$$

Similarly, there is one hyperparameter related to $\boldsymbol{\phi}_3$: $\sigma^2_{\phi_3}$. The conditional posterior is

$$\sigma^2_{\phi_3} \mid \boldsymbol{\phi}_3 \sim \text{IG}\left[\lambda_1^{\phi_3} + \frac{1}{2}\dim(\boldsymbol{\phi}_3),\ \lambda_2^{\phi_3} + \frac{1}{2}\boldsymbol{\phi}_3^T\boldsymbol{\phi}_3\right].$$

Update intermittent missing responses. The focus of our method is dealing with monotone missing data. Sometimes there are (typically few) intermittent missing responses, and we impute it under the partial ignorability assumption (Harel and Schafer, 2009). Suppose y_{ijs} is missing. Its conditional distribution is

$$p\left(y_{ijs} \mid y_{-ijs}, \boldsymbol{\pi}\right) \propto p\left(\boldsymbol{y}_{\text{vec0}}, \boldsymbol{y}_{\text{vec}} \mid \boldsymbol{\pi}\right),$$

We use a Metropolis-Hastings step to update y_{ijs}. We use a symmetric normal proposal distribution, $y_{ijs}^{\text{pro}} \sim N\left(y_{ijs}^{\text{cur}},\ 0.5 \times \text{sd}(\boldsymbol{y}_{\text{vec0}}, \boldsymbol{y}_{\text{vec}})\right)$.

B.2 G-computation Implementation Details

We describe in detail how to draw the pseudo responses using Gaussian process prediction rule, i.e. steps 3 and 4 in Algorithm 1. We generally use a superscript $*$ to denote the pseudo subject and response.

Observed response. To draw a vector of pseudo observed responses $\bar{\boldsymbol{Y}}_s^* = \bar{\boldsymbol{y}}_s^*$ from $p(\bar{\boldsymbol{y}}_s^* \mid s^*, \boldsymbol{v}^*, \boldsymbol{\pi})$, we do the following.

1. Draw y_1^* from $p(y_1^* \mid s^*, \boldsymbol{v}^*, \boldsymbol{\pi})$. Consider the joint distribution of $a_{1s}^* = a_0(\boldsymbol{v}^*, 1, s^*)$ and the training data points $\boldsymbol{y}_{\text{vec0}}$,

$$\begin{pmatrix} \boldsymbol{y}_{\text{vec0}} \\ a_{1s}^* \end{pmatrix} \sim N\left[\begin{pmatrix} X_{\theta_0}\boldsymbol{\theta}_0 \\ \mu_{1s}^* \end{pmatrix}, \begin{pmatrix} \Sigma_{y0} + C_0 & C_{1s*} \\ C_{1s*}^T & C_{1s**} \end{pmatrix}\right],$$

where

$$\begin{aligned} \mu_{1s}^* &= \mu_0(\boldsymbol{v}^*, 1, s^*), \\ C_{1s*} &= C_0(V, \boldsymbol{j}_{\text{vec0}}, \boldsymbol{s}_{\text{vec0}}; \boldsymbol{v}^*, 1, s^*), \\ C_{1s**} &= C_0(\boldsymbol{v}^*, 1, s^*; \boldsymbol{v}^*, 1, s^*), \end{aligned}$$

$\boldsymbol{j}_{\text{vec0}}$ and $\boldsymbol{s}_{\text{vec0}}$ are vectors of times and patterns corresponding to responses $\boldsymbol{y}_{\text{vec0}}$. The predictive distribution for a_{1s}^* is thus

$$\begin{aligned} a_{1s}^* \mid \boldsymbol{y}_{\text{vec0}}, \boldsymbol{\pi} \sim N\big[&\mu_{1s}^* + C_{1s*}^T(\Sigma_{y0} + C_0)^{-1}(\boldsymbol{y}_{\text{vec0}} - X_{\theta_0}\boldsymbol{\theta}_0), \\ &C_{1s**} - C_{1s*}^T(\Sigma_{y0} + C_0)^{-1}C_{1s*}\big], \end{aligned}$$

and we can draw

$$y_1^* \sim N(a_{1s}^*, \sigma_{1s^*}^2).$$

2. Draw y_j^* from $p(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, s^*, \boldsymbol{v}^*, \boldsymbol{\pi})$, $(1 < j \leq s^*)$. The joint distribution of $a_{js}^* = a(\boldsymbol{v}^*, j, s^*) + \bar{\boldsymbol{y}}_{j-1}^{*T} \boldsymbol{\Phi}_{js^*}$ and the training data points $\boldsymbol{y}_{\text{vec}}$ is

$$\begin{pmatrix} \boldsymbol{y}_{\text{vec}} \\ a_{js}^* \end{pmatrix} \sim N\left[\begin{pmatrix} X_\theta \boldsymbol{\theta} \\ \mu_{js}^* + \bar{\boldsymbol{y}}_{j-1}^{*T} \boldsymbol{\Phi}_{js^*} \end{pmatrix}, \begin{pmatrix} \Sigma_y + C & C_{js*} \\ C_{js*}^T & C_{js**} \end{pmatrix} \right],$$

where

$$\begin{aligned} \mu_{js}^* &= \mu(\boldsymbol{v}^*, j, s^*), \\ C_{js*} &= C(V, \boldsymbol{j}_{\text{vec}}, \boldsymbol{s}_{\text{vec}}; \boldsymbol{v}^*, j, s^*), \\ C_{js**} &= C(\boldsymbol{v}^*, j, s^*; \boldsymbol{v}^*, j, s^*), \end{aligned}$$

$\boldsymbol{j}_{\text{vec}}$ and $\boldsymbol{s}_{\text{vec}}$ are vectors of times and patterns corresponding to responses $\boldsymbol{y}_{\text{vec}}$. The predictive distribution for a_{js}^* is thus

$$\begin{aligned} a_{js}^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{y}_{\text{vec}}, \boldsymbol{\pi} \sim N\big[& \mu_{js}^* + \bar{\boldsymbol{y}}_{j-1}^{*T} \boldsymbol{\Phi}_{js^*} + C_{js*}^T (\Sigma_y + C)^{-1} (\boldsymbol{y}_{\text{vec}} - X_\theta \boldsymbol{\theta}), \\ & C_{js**} - C_{js*}^T (\Sigma_y + C)^{-1} C_{js*} \big], \end{aligned}$$

and we can draw

$$y_j^* \sim N(a_{js}^*, \sigma_{js^*}^2).$$

Missing response. To draw a pseudo response $Y_j^* = y_j^*$ from the extrapolation distribution $p(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, s^*, \boldsymbol{v}^*, \boldsymbol{\omega})$ $(j > s^*)$, do the following.

(I) Under MAR,

$$p(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, S = s^*, \boldsymbol{\omega}) = p(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, S \geq j, \boldsymbol{\omega})$$
$$= \sum_{k=j}^{J} \alpha_{kj} p(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, S = k, \boldsymbol{\omega}), \qquad \text{(B.1)}$$

where

$$\alpha_{kj} = \alpha_{kj}(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*) = p(S = k \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, S \geq j)$$
$$= \frac{p(\bar{\boldsymbol{y}}_{j-1}^* \mid \boldsymbol{v}^*, S = k)\, p(S = k \mid \boldsymbol{v}^*, S \geq j)}{\sum_{k=j}^{J} p(\bar{\boldsymbol{y}}_{j-1}^* \mid \boldsymbol{v}^*, S = k)\, p(S = k \mid \boldsymbol{v}^*, S \geq j)}, \quad k = j, \ldots, J$$

The above expression can be calculated by

$$p(\bar{\boldsymbol{y}}_{j-1}^* \mid \boldsymbol{v}^*, S = k) = p_k(y_1^* \mid \boldsymbol{v}^*) \cdot \prod_{l=2}^{j-1} p_k(y_l^* \mid \bar{\boldsymbol{y}}_{l-1}^*, \boldsymbol{v}^*)$$

where

$$p_k(y_1^* \mid \boldsymbol{v}^*) = p_N\big[y_1^* \mid \mu_{1k}^* + C_{1k*}^T(\Sigma_{y0} + C_0)^{-1}(\boldsymbol{y}_{\text{vec0}} - X_{\theta_0}\boldsymbol{\theta}_0),$$
$$C_{1k**} - C_{1k*}^T(\Sigma_{y0} + C_0)^{-1}C_{1k*}\big],$$
$$p_k(y_l^* \mid \bar{\boldsymbol{y}}_{l-1}^*, \boldsymbol{v}^*) = p_N\big[y_l^* \mid \mu_{lk}^* + \bar{\boldsymbol{y}}_{l-1}^{*T}\boldsymbol{\Phi}_{lk} + C_{lk*}^T(\Sigma_y + C)^{-1}\cdot$$
$$(\boldsymbol{y}_{\text{vec}} - X_\theta\boldsymbol{\theta}), C_{lk**} - C_{lk*}'(C_{lk} + \sigma_{lk}^2 I)^{-1}C_{lk*}\big],$$

and

$$p(S = k \mid \boldsymbol{v}^*, S \geq j)$$
$$= p(S = k \mid \boldsymbol{v}^*, S \geq k) \cdot \prod_{l=j}^{k-1} p(S \geq l+1 \mid \boldsymbol{v}^*, S \geq l)$$
$$= p(S = k \mid \boldsymbol{v}^*, S \geq k) \cdot \prod_{l=j}^{k-1} \left[1 - p(S = l \mid \boldsymbol{v}^*, S \geq l)\right].$$

To sample from (B.1), after calculating $(\alpha_{jj}, \ldots, \alpha_{Jj})$, we can draw $K = k$ with probability α_{kj}, and sample $Y_j^* = y_j^*$ from $p_k(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, \boldsymbol{\omega})$.

(II) Under NFD.

(II-1) For $j = s^* + 1$,

$$[Y_j \mid \bar{\boldsymbol{Y}}_{j-1}, S = j-1, \boldsymbol{V}, \boldsymbol{\omega}] \stackrel{\mathrm{d}}{=} \left[Y_j + \tau_j \mid \bar{\boldsymbol{Y}}_{j-1}, S \geq j, \boldsymbol{V}, \boldsymbol{\omega}\right].$$

We first sample from $p_{\geq j}(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, \boldsymbol{\omega})$. Then, we apply the location shift (4.9) with

$$\tau_j(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*) = \tilde{\tau} \cdot \Delta_j(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*),$$

where $\Delta_j(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*)$ is chosen to be the standard deviation of $p_{j-1}(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, \boldsymbol{\omega})$ under MAR, i.e. $p_{\geq j}(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, \boldsymbol{\omega})$. We have

$$p_{\geq j}(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, \boldsymbol{\omega}) = \sum_{k=j}^{J} \alpha_{kj} N(\tilde{\mu}_{jk}, \tilde{\sigma}_{jk}^2).$$

The standard deviation of this normal mixture is given by

$$\Delta_j(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*) = \sqrt{\sum_{k=j}^{J} \alpha_{kj} \tilde{\sigma}_{jk}^2 + \sum_{k=j}^{J} \alpha_{kj} \tilde{\mu}_{jk}^2 - \left(\sum_{k=j}^{J} \alpha_{kj} \tilde{\mu}_{jk}\right)^2}.$$

(II-2) For $j > s^* + 1$,

$$\begin{aligned} p(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, S = s^*, \boldsymbol{\omega}) &= p(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, S \geq j-1, \boldsymbol{\omega}) \\ &= \sum_{k=j-1}^{J} \alpha_{k,j-1} p(y_j^* \mid \bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*, S = k, \boldsymbol{\omega}), \end{aligned} \tag{B.2}$$

where

$$\begin{aligned}\alpha_{k,j-1} &= \alpha_{k,j-1}(\bar{\boldsymbol{y}}^*_{j-1}, \boldsymbol{v}^*) = p(S = k \mid \bar{\boldsymbol{y}}^*_{j-1}, \boldsymbol{v}^*, S \geq j-1) \\ &= \frac{p(\bar{\boldsymbol{y}}^*_{j-1} \mid \boldsymbol{v}^*, S = k)\, p(S = k \mid \boldsymbol{v}^*, S \geq j-1)}{\sum_{k=j-1}^{J} p(\bar{\boldsymbol{y}}^*_{j-1} \mid \boldsymbol{v}^*, S = k)\, p(S = k \mid \boldsymbol{v}^*, S \geq j-1)}, \quad k = j-1, \ldots, J.\end{aligned}$$

To sample from (B.2), after calculating $(\alpha_{j-1,j-1}, \ldots, \alpha_{J,j-1})$, we can draw $K = k$ with probability $\alpha_{k,j-1}$.

(II-2a) If $k = j-1$, draw again $K' = k'$ with probability $\alpha_{k',j-1}/(1-\alpha_{j-1,j-1})$ for $k' = j, \ldots, J$. Then, sample $Y^*_j = y^*_j$ from $p_{k'}(y^*_j \mid \bar{\boldsymbol{y}}^*_{j-1}, \boldsymbol{v}^*, \boldsymbol{\omega})$, and apply the location shift (4.9).

(II-2b) If $k \in \{j, \ldots, J\}$, sample $Y^*_j = y^*_j$ from $p_k(y^*_j \mid \bar{\boldsymbol{y}}^*_{j-1}, \boldsymbol{v}^*, \boldsymbol{\omega})$.

The steps for sampling the pseudo response $\boldsymbol{Y}^* = \boldsymbol{y}^*$ from $p(\boldsymbol{y}^* \mid s^*, \boldsymbol{v}^*, \boldsymbol{\omega})$ are summarized in Algorithm 3, where

$$\begin{aligned}\tilde{\mu}_{1s} &= \mu^*_{1s} + C^T_{1s*}(\Sigma_{y0} + C_0)^{-1}(\boldsymbol{y}_{\text{vec}0} - X_{\theta_0}\boldsymbol{\theta}_0), \\ \tilde{\sigma}^2_{js} &= C_{1s**} - C^T_{1s*}(\Sigma_{y0} + C_0)^{-1}C_{1s*} + \sigma^2_{1s^*},\end{aligned}$$

and

$$\begin{aligned}\tilde{\mu}_{js} &= \mu^*_{js} + \bar{\boldsymbol{y}}^{*T}_{j-1}\boldsymbol{\Phi}_{js} + C^T_{js*}(\Sigma_y + C)^{-1}(\boldsymbol{y}_{\text{vec}} - X_\theta\boldsymbol{\theta}), \\ \tilde{\sigma}^2_{js} &= C_{js**} - C^T_{js*}(\Sigma_y + C)^{-1}C_{js*} + \sigma^2_{js^*},\end{aligned}$$

for $j = 2, \ldots, J$.

Algorithm 3 Draw $\boldsymbol{Y}^* = \boldsymbol{y}^*$ from $p(\boldsymbol{y}^* \mid s^*, \boldsymbol{v}^*, \boldsymbol{\omega})$

Draw $Y_1^* = y_1^* \sim N(\tilde{\mu}_{1s}, \tilde{\sigma}_{1s}^2)$
for j in $2, \ldots, s^*$ **do**
    Draw $Y_j^* = y_j^* \sim N(\tilde{\mu}_{js}, \tilde{\sigma}_{js}^2)$
end for
if MAR **then**
    for j in $s^* + 1, \ldots, J$ **do**
        Calculate $\boldsymbol{\alpha}_j(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*) = (\alpha_{jj}, \ldots, \alpha_{Jj})$
        Draw $K = k \sim \text{Categorical}[(j, \ldots, J); \boldsymbol{\alpha}_j]$
        Draw $y_j^* \sim N(\tilde{\mu}_{jk}, \tilde{\sigma}_{jk}^2)$
    end for
else if NFD **then**
    Set $j = s^* + 1$
    Calculate $\boldsymbol{\alpha}_j(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*) = (\alpha_{jj}, \ldots, \alpha_{Jj})$
    Draw $K = k \sim \text{Categorical}[(j, \ldots, J); \boldsymbol{\alpha}_j]$
    Calculate $\tau_j(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*) = \tilde{\tau} \cdot \Delta_j(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*)$
    Draw $y_j^* \sim N(\tilde{\mu}_{jk} + \tau_j, \tilde{\sigma}_{jk}^2)$
    for j in $s^* + 2, \ldots, J$ **do**
        Calculate $\boldsymbol{\alpha}_{j-1}(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*) = (\alpha_{j-1,j-1}, \ldots, \alpha_{J,j-1})$
        Draw $K = k \sim \text{Categorical}[(j-1, \ldots, J); \boldsymbol{\alpha}_{j-1}]$
        if $k = j - 1$ **then**
            Calculate $\boldsymbol{\alpha}'_j = (\alpha_{j,j-1}, \ldots, \alpha_{J,j-1})/(1 - \alpha_{j-1,j-1})$
            Draw $K' = k' \sim \text{Categorical}[(j, \ldots, J); \boldsymbol{\alpha}'_j]$
            Calculate $\tau_j(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*) = \tilde{\tau} \cdot \Delta_j(\bar{\boldsymbol{y}}_{j-1}^*, \boldsymbol{v}^*)$
            Draw $y_j^* \sim N(\tilde{\mu}_{jk'} + \tau_j, \tilde{\sigma}_{jk'}^2)$
        else
            Draw $y_j^* \sim N(\tilde{\mu}_{jk}, \tilde{\sigma}_{jk}^2)$.
        end if
    end for
end if

B.3 Simulation Details

Prior and hyperprior parameters. We set the prior and hyperprior parameters at standard noninformative choices. Table B.1 shows the exact values.

$\delta^2_{\beta_0}$	30	$\delta^2_{b_0}$	30	$\delta^2_{\phi_1}$	30	$\lambda_1^{\phi_3}$	100
$\lambda_1^{\beta_0}$	1	$\lambda_1^{b_0}$	1	$\lambda_1^{\phi_1}$	1	$\lambda_2^{\phi_3}$	1
$\lambda_2^{\beta_0}$	1	$\lambda_2^{b_0}$	1	$\lambda_2^{\phi_1}$	1	d_η	0.1
δ^2_{β}	30	δ^2_{b}	30	$\lambda_1^{\phi_2}$	30		
λ_1^{β}	1	λ_1^{b}	1	$\lambda_2^{\phi_2}$	1		
λ_2^{β}	1	λ_2^{b}	1				

Table B.1: Choices of prior and hyperprior parameters in the observed data model. These parameters are used for simulations and real data analysis.

Scenario 1. The covariance matrix for generating $\boldsymbol{V}$ is

$$\Sigma_{vv} = \begin{pmatrix} 1.0 & 0.52 & -0.22 & 0.07 \\ 0.52 & 1.0 & -0.23 & -0.02 \\ -0.22 & -0.23 & 1.0 & 0.45 \\ 0.07 & -0.02 & 0.45 & 1.0 \end{pmatrix},$$

which is the correlation matrix of the subjects' numerical auxiliary covariates from the schizophrenia clinical trial dataset.

The parameters for generating S are

$$\boldsymbol{\zeta} = (-4.346, -2.193, -2.606, -2.678)^T,$$

where ζ_s corresponds to the $(s-1)$-th element $(s = 2, \ldots, 5)$, and

$$\boldsymbol{\xi} = \begin{pmatrix} -1.057 & 0.328 & -0.121 & 0.273 \\ -0.826 & 0.128 & 0.525 & -0.781 \\ -0.487 & 0.479 & 0.534 & -0.480 \\ 0.642 & 0.129 & 0.448 & 0.122 \end{pmatrix},$$

where $\boldsymbol{\xi}_s$ corresponds to the $(s-1)$-th column $(s = 2, \ldots, 5)$. These parameters come from fitting the sequential logistic regression model to the test drug arm of the schizophrenia clinical trial dataset and taking posterior mean of each parameter.

The parameters for generating $\bar{\boldsymbol{Y}}_S$ are

$$\{\sigma_{js}^2\} = \begin{pmatrix} 0.232 & 0.221 & & & & \\ 0.365 & 0.243 & 0.196 & & & \\ 0.403 & 0.222 & 0.228 & 0.941 & & \\ 0.438 & 0.228 & 0.225 & 0.213 & 0.284 & \\ 0.335 & 0.192 & 0.265 & 0.140 & 0.167 & 0.160 \end{pmatrix},$$

where σ_{js}^2 corresponds to the element in the $(s-1)$-th row and j-th column;

$$(\boldsymbol{b}_0, \boldsymbol{b}) = \begin{pmatrix} 0.069 & -0.191 & & & & \\ 0.507 & 0.219 & 0.302 & & & \\ 0.393 & 0.060 & -0.022 & 0.399 & & \\ 0.798 & 0.048 & -0.051 & 0.051 & 0.362 & \\ 0.384 & -0.107 & -0.250 & -0.367 & -0.250 & -0.321 \end{pmatrix},$$

where b_{js} corresponds to the element in the $(s-1)$-th row and j-th column;

$$\boldsymbol{\beta}_0 = \begin{pmatrix} -0.046 & 0.174 & -0.005 & 0.024 & 0.230 \\ -0.200 & -0.099 & -0.124 & -0.451 & -0.163 \\ -0.315 & -0.191 & -0.104 & 0.140 & 0.032 \\ -0.053 & 0.065 & 0.003 & -0.044 & -0.092 \end{pmatrix},$$

where $\boldsymbol{\beta}_{0s}$ corresponds to the $(s-1)$-th column;

$$\boldsymbol{\beta} = \begin{pmatrix} -0.080 & -0.117 & -0.118 & 0.010 & 0.066 \\ -0.044 & -0.113 & 0.023 & -0.035 & -0.030 \\ -0.109 & -0.020 & -0.014 & -0.022 & 0.056 \\ 0.170 & 0.127 & 0.166 & -0.060 & 0.002 \end{pmatrix},$$

where $\boldsymbol{\beta}_s$ corresponds to the $(s-1)$-th column;

$$(\boldsymbol{\phi}_1) = \begin{pmatrix} 1.078 & & & & \\ 1.088 & 0.938 & & & \\ 0.830 & 0.893 & 0.830 & & \\ 0.637 & 0.877 & 0.907 & 1.065 & \\ 0.881 & 0.871 & 0.842 & 0.929 & 0.943 \end{pmatrix},$$

where ϕ_{1js} corresponds to the element in the $(s-1)$-th row and $(j-1)$-th column;

$$(\boldsymbol{\phi}_2) = \begin{pmatrix} -0.045 & & & \\ 0.040 & -0.025 & & \\ 0.021 & 0.022 & 0.035 & \\ 0.089 & 0.129 & 0.019 & -0.020 \end{pmatrix},$$

where ϕ_{2js} corresponds to the element in the $(s-2)$-th row and $(j-2)$-th column; and

$$(\boldsymbol{\phi}_3) = \begin{pmatrix} 0.011 & & \\ 0.037 & \begin{pmatrix} 0.074 \\ 0.037 \end{pmatrix} & \\ 0.078 & \begin{pmatrix} -0.027 \\ -0.086 \end{pmatrix} & \begin{pmatrix} 0.021 \\ 0.010 \\ -0.009 \end{pmatrix} \end{pmatrix},$$

where $\boldsymbol{\phi}_{3js}$ corresponds to the element in the $(s-3)$-th row and $(j-3)$-th column. These parameters come from fitting the linear regression model to the test drug arm of the schizophrenia clinical trial dataset and taking posterior mean of each parameter.

Scenario 2. We use the same choices of $\boldsymbol{b}_0$, $\boldsymbol{b}$, $\boldsymbol{\phi}_1$, $\boldsymbol{\phi}_2$ and $\boldsymbol{\phi}_3$ as in Scenario 1. We set

$$\Sigma_{vv} = \begin{pmatrix} 1.0 & 0.52 & -0.22 \\ 0.52 & 1.0 & -0.23 \\ -0.22 & -0.23 & 1.0 \end{pmatrix},$$

i.e. the upper left 3×3 submatrix of Σ_{vv} in Scenario 1. We change $\{\sigma_{js}^2\}$, $\boldsymbol{\zeta}$, $\boldsymbol{\xi}$, $\boldsymbol{\beta}_0$ and $\boldsymbol{\beta}$ to

$$\{\sigma_{js}^2\} = \begin{pmatrix} 0.155 & 0.101 & & & & \\ 0.217 & 0.133 & 0.112 & & & \\ 0.099 & 0.082 & 0.101 & 0.115 & & \\ 0.141 & 0.127 & 0.169 & 0.132 & 0.107 & \\ 0.106 & 0.119 & 0.095 & 0.081 & 0.266 & 0.174 \end{pmatrix},$$

where σ_{js}^2 corresponds to the element in the $(s-1)$-th row and j-th column;

$$\boldsymbol{\zeta} = (-3.0, -2.1, -1.6, -1.3)^T,$$

where ζ_s corresponds to the $(s-1)$-th element $(s = 2, \ldots, 5)$, and

$$\boldsymbol{\xi} = \begin{pmatrix} -1.057 & 0.328 & -0.121 & 0.273 \\ -0.826 & 0.128 & 0.525 & -0.781 \\ -0.487 & 0.479 & 0.534 & -0.480 \\ -0.528 & 0.164 & -0.061 & 0.136 \\ -0.413 & 0.064 & 0.263 & -0.390 \\ -0.244 & 0.239 & 0.267 & -0.240 \\ 0.321 & 0.064 & 0.224 & 0.061 \\ -0.528 & 0.164 & -0.061 & 0.136 \\ -0.413 & 0.064 & 0.263 & -0.390 \end{pmatrix},$$

where $\boldsymbol{\xi}_s$ corresponds to the $(s-1)$-th column $(s = 2, \ldots, 5)$.

$$\boldsymbol{\beta}_0 = \begin{pmatrix} -0.530 & -0.508 & -0.561 & -0.507 & -0.525 \\ -0.366 & -0.377 & -0.421 & -0.417 & -0.386 \\ 0.351 & 0.309 & 0.323 & 0.318 & 0.346 \\ 0.283 & 0.291 & 0.282 & 0.277 & 0.275 \\ -0.316 & -0.321 & -0.319 & -0.319 & -0.316 \\ 0.288 & 0.285 & 0.293 & 0.288 & 0.289 \\ 0.033 & 0.030 & 0.033 & 0.020 & 0.033 \\ -0.083 & -0.087 & -0.094 & -0.082 & -0.092 \\ 0.124 & 0.125 & 0.115 & 0.120 & 0.116 \end{pmatrix},$$

where $\boldsymbol{\beta}_{0s}$ corresponds to the $(s-1)$-th column;

$$\boldsymbol{\beta} = \begin{pmatrix} -0.395 & -0.387 & -0.427 & -0.434 & -0.443 \\ 0.320 & 0.337 & 0.339 & 0.317 & 0.338 \\ 0.331 & 0.349 & 0.400 & 0.385 & 0.356 \\ 0.317 & 0.315 & 0.309 & 0.313 & 0.310 \\ 0.354 & 0.355 & 0.342 & 0.354 & 0.349 \\ -0.301 & -0.299 & -0.303 & -0.306 & -0.306 \\ -0.082 & -0.082 & -0.073 & -0.068 & -0.079 \\ -0.077 & -0.088 & -0.082 & -0.085 & -0.081 \\ -0.129 & -0.126 & -0.130 & -0.133 & -0.128 \end{pmatrix},$$

where $\boldsymbol{\beta}_s$ corresponds to the $(s-1)$-th column.

Scenario 3. The parameter for generating K is

$$\boldsymbol{\pi} = (0.119, 0.579, 0.001, 0.115, 0.186),$$

which is taken from Linero and Daniels (2015) by fitting the mixture model to the active control arm of the schizophrenia clinical trial dataset.

The parameters for the joint distribution of $\boldsymbol{Y}$ and $\boldsymbol{V}$ are specified and generated as follows. Within mixture component k, the joint distribution of $\boldsymbol{Y}$ and $\boldsymbol{V}$ is

$$\begin{pmatrix} \boldsymbol{Y} \\ \boldsymbol{V} \end{pmatrix} \mid K = k \sim N\left[\boldsymbol{\mu}^{(k)}, \Omega^{(k)}\right],$$

where

$$\begin{aligned} \boldsymbol{\mu}^{(k)} &= \begin{pmatrix} \boldsymbol{\mu}_y^{(k)} \\ \mathbf{0} \end{pmatrix}, \\ \Omega^{(k)} &\sim \mathcal{W}^{-1}\left((\nu - J - Q - 1)\Omega_0^{(k)}, \nu\right), \\ \Omega_0^{(k)} &= \begin{pmatrix} \Sigma_{yy}^{(k)} & \Sigma_{yv}^{(k)} \\ \Sigma_{vy}^{(k)} & \Sigma_{vv} \end{pmatrix}. \end{aligned}$$

Here $\boldsymbol{\mu}_y^{(k)}$ and $\Omega_0^{(k)}$ correspond to a linear model of $(\boldsymbol{Y} \mid \boldsymbol{V})$, where

$$\begin{aligned}
\boldsymbol{V} \mid K = k &\sim N(\mathbf{0}, \Sigma_{vv}), \\
Y_1 \mid \boldsymbol{V}, K = k &\sim N\left(b_1^{(k)} + \boldsymbol{V}^T \boldsymbol{\beta}_0^{(k)},\ \sigma_1^{2(k)}\right), \\
Y_j \mid \bar{\boldsymbol{Y}}_{j-1}, \boldsymbol{V}, K = k &\sim N\left(b_j^{(k)} + \boldsymbol{V}^T \boldsymbol{\beta}^{(k)} + \phi_j^{(k)} Y_{j-1},\ \sigma_j^{2(k)}\right), \quad j = 2, \ldots, J.
\end{aligned}$$

Let $\boldsymbol{b}^{(k)} = (b_1^{(k)}, \ldots, b_J^{(k)})^T$, $B^{(k)} = (\boldsymbol{\beta}_0^{(k)}, \boldsymbol{\beta}^{(k)}, \ldots, \boldsymbol{\beta}^{(k)})$, $\Sigma_0^{(k)} = \mathrm{diag}(\sigma_1^{2(k)}, \ldots, \sigma_J^{2(k)})$,

$$\Phi^{(k)} = \begin{pmatrix}
0 & 0 & 0 & \cdots & 0 \\
\phi_2^{(k)} & 0 & 0 & \cdots & 0 \\
0 & \phi_3^{(k)} & 0 & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & \phi_J^{(k)} & 0
\end{pmatrix},$$

and $\Psi^{(k)} = \left(I - \Phi^{(k)}\right)^{-1}$. We have

$$\begin{aligned}
\boldsymbol{\mu}_y^{(k)} &= \Psi^{(k)} \boldsymbol{b}^{(k)}, \\
\Sigma_{yy}^{(k)} &= \Psi^{(k)} B^{(k)T} \Sigma_{vv} B^{(k)} \Psi^{(k)T} + \Psi^{(k)} \Sigma_0^{(k)} \Psi^{(k)T}, \\
\Sigma_{yv}^{(k)} &= \Psi^{(k)} B^{(k)T} \Sigma_{vv}.
\end{aligned}$$

We use the same Σ_{vv} as in Scenario 2. The parameters $\{\boldsymbol{\mu}_y^{(k)}\}$ and $\Sigma_0^{(k)}$ are taken from Linero and Daniels (2015) (after standardization), which are generated by fitting the model to the active control arm of the schizophrenia clinical trial dataset. In particular,

$$\{\boldsymbol{\mu}_y^{(k)}\} = \begin{pmatrix}
0.715 & 0.559 & -0.649 & -0.085 & 0.677 \\
0.581 & 0.406 & -1.368 & -0.207 & 0.799 \\
0.329 & 0.175 & -1.404 & -0.851 & 0.944 \\
0.319 & -0.217 & -1.650 & -1.181 & 1.276 \\
0.889 & -0.473 & -1.765 & -1.363 & 0.483 \\
-0.664 & -0.593 & -3.195 & -1.562 & 1.081
\end{pmatrix},$$

where $\boldsymbol{\mu}_y^{(k)}$ corresponds to the k-th column. Then, we add the effects of auxiliary covariates by randomly generating $B^{(k)}$ and $\Phi^{(k)}$ (values not shown). Based on $B^{(k)}$, $\Phi^{(k)}$, Σ_{vv} and $\Sigma_0^{(k)}$ we calculate $\Omega_0^{(k)}$. Finally, we generate $\Omega^{(k)} \sim \mathcal{W}^{-1}\left((\nu - J - Q - 1)\Omega_0^{(k)}, \nu\right)$ and get

$$\Omega^{(1)} = \left(\begin{array}{cccccc|ccc} 0.9 & 1.3 & 1.7 & 1.9 & 2.3 & 2.6 & -1.0 & -0.4 & 0.4 \\ 1.3 & 2.2 & 2.9 & 3.4 & 4.2 & 4.9 & -1.6 & -0.4 & 0.9 \\ 1.7 & 2.9 & 4.1 & 4.8 & 5.9 & 7.0 & -2.1 & -0.4 & 1.4 \\ 1.9 & 3.4 & 4.8 & 5.8 & 7.1 & 8.3 & -2.4 & -0.3 & 1.7 \\ 2.3 & 4.2 & 5.9 & 7.1 & 8.8 & 10.4 & -3.0 & -0.4 & 2.2 \\ 2.6 & 4.9 & 7.0 & 8.3 & 10.4 & 12.2 & -3.5 & -0.4 & 2.6 \\ \hline -1.0 & -1.6 & -2.1 & -2.4 & -3.0 & -3.5 & 1.7 & 0.5 & -0.2 \\ -0.4 & -0.4 & -0.4 & -0.3 & -0.4 & -0.4 & 0.5 & 0.7 & -0.1 \\ 0.4 & 0.9 & 1.4 & 1.7 & 2.2 & 2.6 & -0.2 & -0.1 & 1.2 \end{array}\right),$$

$$\Omega^{(2)} = \left(\begin{array}{cccccc|ccc} 0.2 & 0.3 & 0.3 & 0.4 & 0.5 & 0.6 & -0.2 & -0.3 & 0.3 \\ 0.3 & 0.6 & 0.8 & 1.0 & 1.3 & 1.6 & -0.2 & -0.3 & 0.7 \\ 0.3 & 0.8 & 1.2 & 1.5 & 1.9 & 2.4 & -0.3 & -0.2 & 1.0 \\ 0.4 & 1.0 & 1.5 & 2.0 & 2.5 & 3.1 & -0.4 & -0.1 & 1.2 \\ 0.5 & 1.3 & 1.9 & 2.5 & 3.2 & 4.0 & -0.4 & -0.2 & 1.6 \\ 0.6 & 1.6 & 2.4 & 3.1 & 4.0 & 5.0 & -0.4 & -0.2 & 2.1 \\ \hline -0.2 & -0.2 & -0.3 & -0.4 & -0.4 & -0.4 & 0.5 & 0.1 & 0.1 \\ -0.3 & -0.3 & -0.2 & -0.1 & -0.2 & -0.2 & 0.1 & 0.9 & -0.4 \\ 0.3 & 0.7 & 1.0 & 1.2 & 1.6 & 2.1 & 0.1 & -0.4 & 1.2 \end{array}\right),$$

$$\Omega^{(3)} = \left(\begin{array}{cccccc|ccc} 1.2 & 1.3 & 1.3 & 1.3 & 1.5 & 1.6 & -0.8 & -0.8 & 0.4 \\ 1.3 & 1.5 & 1.6 & 1.7 & 1.9 & 2.1 & -0.9 & -0.7 & 0.6 \\ 1.3 & 1.6 & 1.7 & 1.9 & 2.2 & 2.4 & -0.9 & -0.5 & 0.7 \\ 1.3 & 1.7 & 1.9 & 2.1 & 2.4 & 2.7 & -0.9 & -0.4 & 0.8 \\ 1.5 & 1.9 & 2.2 & 2.4 & 2.9 & 3.3 & -1.1 & -0.4 & 0.9 \\ 1.6 & 2.1 & 2.4 & 2.7 & 3.3 & 3.7 & -1.2 & -0.3 & 1.1 \\ \hline -0.8 & -0.9 & -0.9 & -0.9 & -1.1 & -1.2 & 0.8 & 0.5 & -0.1 \\ -0.8 & -0.7 & -0.5 & -0.4 & -0.4 & -0.3 & 0.5 & 0.9 & -0.1 \\ 0.4 & 0.6 & 0.7 & 0.8 & 0.9 & 1.1 & -0.1 & -0.1 & 0.6 \end{array}\right),$$

$$\Omega^{(4)} = \left(\begin{array}{cccccc|ccc} 1.0 & 1.3 & 1.5 & 1.7 & 2.0 & 2.2 & -0.9 & -0.7 & 0.5 \\ 1.3 & 2.0 & 2.4 & 2.7 & 3.2 & 3.6 & -1.4 & -0.7 & 0.6 \\ 1.5 & 2.4 & 2.9 & 3.4 & 4.0 & 4.5 & -1.7 & -0.7 & 0.8 \\ 1.7 & 2.7 & 3.4 & 4.0 & 4.7 & 5.4 & -2.0 & -0.7 & 0.9 \\ 2.0 & 3.2 & 4.0 & 4.7 & 5.6 & 6.3 & -2.3 & -0.7 & 1.0 \\ 2.2 & 3.6 & 4.5 & 5.4 & 6.3 & 7.3 & -2.6 & -0.7 & 1.2 \\ \hline -0.9 & -1.4 & -1.7 & -2.0 & -2.3 & -2.6 & 1.3 & 0.5 & -0.1 \\ -0.7 & -0.7 & -0.7 & -0.7 & -0.7 & -0.7 & 0.5 & 0.9 & -0.2 \\ 0.5 & 0.6 & 0.8 & 0.9 & 1.0 & 1.2 & -0.1 & -0.2 & 0.7 \end{array}\right),$$

$$\Omega^{(5)} = \left(\begin{array}{cccccc|ccc} 0.8 & 1.0 & 1.3 & 1.5 & 1.7 & 2.0 & -0.8 & -0.4 & 0.5 \\ 1.0 & 1.7 & 2.2 & 2.7 & 3.2 & 3.8 & -1.4 & -0.3 & 0.7 \\ 1.3 & 2.2 & 2.9 & 3.7 & 4.3 & 5.1 & -1.8 & -0.2 & 1.0 \\ 1.5 & 2.7 & 3.7 & 4.8 & 5.7 & 6.7 & -2.2 & -0.1 & 1.2 \\ 1.7 & 3.2 & 4.3 & 5.7 & 6.8 & 8.0 & -2.6 & -0.1 & 1.4 \\ 2.0 & 3.8 & 5.1 & 6.7 & 8.0 & 9.5 & -3.1 & -0.0 & 1.7 \\ \hline -0.8 & -1.4 & -1.8 & -2.2 & -2.6 & -3.1 & 1.4 & 0.3 & -0.3 \\ -0.4 & -0.3 & -0.2 & -0.1 & -0.1 & -0.0 & 0.3 & 0.5 & -0.1 \\ 0.5 & 0.7 & 1.0 & 1.2 & 1.4 & 1.7 & -0.3 & -0.1 & 0.7 \end{array}\right),$$

The parameters for generating S are

$$\boldsymbol{\zeta} = (-2.61, -2.75, -2.08, -1.52)^T,$$

where ζ_s corresponds to the $(s-1)$-th element $(s = 2, \ldots, 5)$,

$$\boldsymbol{\ell} = (-0.96, 0.66, 0.78, 0.54)^T,$$

where ℓ_s corresponds to the $(s-1)$-th element $(s = 2, \ldots, 5)$, and

$$\boldsymbol{\xi} = \begin{pmatrix} -1.057 & 0.328 & -0.121 & 0.273 \\ -0.826 & 0.128 & 0.525 & -0.781 \\ -0.487 & 0.479 & 0.534 & -0.480 \end{pmatrix},$$

where $\boldsymbol{\xi}_s$ corresponds to the $(s-1)$-th column $(s = 2, \ldots, 5)$. The parameters are chosen to mimic the dropout rate of the real data.

Bibliography

Adams, R. P., Z. Ghahramani, and M. I. Jordan (2010). Tree-structured stick breaking for hierarchical data. In *Advances in neural information processing systems*, pp. 19–27.

Aldous, D. J. (1985). Exchangeability and related topics. In *École d'Été de Probabilités de Saint-Flour XIII1983*, pp. 1–198. Springer.

Almendro, V., A. Marusyk, and K. Polyak (2013). Cellular heterogeneity and molecular evolution in cancer. *Annual Review of Pathology: Mechanisms of Disease 8*, 277–302.

Antoniak, C. E. (1974). Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The annals of statistics*, 1152–1174.

Aparicio, S. and C. Caldas (2013). The implications of clonal genome evolution for cancer medicine. *New England journal of medicine 368*(9), 842–851.

Bafna, V., D. Gusfield, G. Lancia, and S. Yooseph (2003). Haplotyping as perfect phylogeny: A direct approach. *Journal of Computational Biology 10*(3-4), 323–340.

Banerjee, A., D. B. Dunson, and S. T. Tokdar (2013). Efficient Gaussian process regression for large datasets. *Biometrika 100*(1), 75.

Banerjee, S., B. P. Carlin, and A. E. Gelfand (2014). *Hierarchical Modeling and Analysis for Spatial Data*. CRC Press.

Banerjee, S., A. E. Gelfand, A. O. Finley, and H. Sang (2008). Gaussian predictive process models for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 70*(4), 825–848.

Bignell, G. R., C. D. Greenman, H. Davies, A. P. Butler, S. Edkins, J. M. Andrews, G. Buck, L. Chen, D. Beare, C. Latimer, et al. (2010). Signatures of mutation and selection in the cancer genome. *Nature 463*(7283), 893–898.

Birmingham, J., A. Rotnitzky, and G. M. Fitzmaurice (2003). Pattern–mixture and selection models for analysing longitudinal data with monotone missing patterns. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 65*(1), 275–297.

Blackwell, D. (1973). Discreteness of Ferguson selections. *The Annals of Statistics 1*(2), 356–358.

Blackwell, D. and J. B. MacQueen (1973). Ferguson distributions via Pólya urn schemes. *The Annals of Statistics*, 353–355.

Blei, D. M. and M. I. Jordan (2006). Variational inference for Dirichlet process mixtures. *Bayesian analysis 1*(1), 121–143.

Bonavia, R., W. K. Cavenee, F. B. Furnari, et al. (2011). Heterogeneity maintenance in glioblastoma: A social network. *Cancer research 71*(12), 4055–4060.

Bozic, I., T. Antal, H. Ohtsuki, H. Carter, D. Kim, S. Chen, R. Karchin, K. W. Kinzler, B. Vogelstein, and M. A. Nowak (2010). Accumulation of driver and passenger mutations during tumor progression. *Proceedings of the National Academy of Sciences 107*(43), 18545–18550.

Brocks, D., Y. Assenov, S. Minner, O. Bogatyrova, R. Simon, C. Koop, C. Oakes, M. Zucknick, D. B. Lipka, J. Weischenfeldt, et al. (2014). Intratumor DNA methylation heterogeneity reflects clonal evolution in aggressive prostate cancer. *Cell reports 8*(3), 798–806.

Broderick, T., M. I. Jordan, and J. Pitman (2013). Cluster and feature modeling from combinatorial stochastic processes. *Statistical Science 28*(3), 289–312.

Broderick, T., B. Kulis, and M. Jordan (2013). MAD-Bayes: MAP-based asymptotic derivations from Bayes. In *Proceedings of The 30th International Conference on Machine Learning*, pp. 226–234.

Broderick, T., J. Pitman, and M. I. Jordan (2013). Feature allocations, probability functions, and paintboxes. *Bayesian Analysis 8*(4), 801–836.

Brooks, S., A. Gelman, G. Jones, and X.-L. Meng (2011). *Handbook of Markov Chain Monte Carlo*. CRC Press.

Carter, S. L., K. Cibulskis, E. Helman, A. McKenna, H. Shen, T. Zack, P. W. Laird, R. C. Onofrio, W. Winckler, B. A. Weir, et al. (2012). Absolute

quantification of somatic DNA alterations in human cancer. *Nature biotechnology 30*(5), 413–421.

Chipman, H. and R. McCulloch (2016). *BayesTree: Bayesian Additive Regression Trees*. R package version 0.3-1.4.

Chipman, H. A., E. I. George, and R. E. McCulloch (1998). Bayesian CART model search. *Journal of the American Statistical Association 93*(443), 935–948.

Chipman, H. A., E. I. George, and R. E. McCulloch (2010). BART: Bayesian additive regression trees. *The Annals of Applied Statistics 4*(1), 266–298.

Choi, T. and M. J. Schervish (2007). On posterior consistency in nonparametric regression problems. *Journal of Multivariate Analysis 98*(10), 1969–1987.

Christensen, R. (2011). *Plane Answers to Complex Questions: The Theory of Linear Models*. Springer Science & Business Media.

Church, D. M., V. A. Schneider, T. Graves, K. Auger, F. Cunningham, N. Bouk, H.-C. Chen, R. Agarwala, W. M. McLaren, G. R. Ritchie, et al. (2011). Modernizing reference genome assemblies. *PLoS Biol 9*(7), e1001091.

Dagum, L. and R. Menon (1998). OpenMP: An industry standard api for shared-memory programming. *Computational Science & Engineering, IEEE 5*(1), 46–55.

Daniels, M., C. Wang, and B. Marcus (2014). Fully Bayesian inference under ignorable missingness in the presence of auxiliary covariates. *Biometrics 70*(1), 62–72.

Daniels, M. J. and J. W. Hogan (2000). Reparameterizing the pattern mixture model for sensitivity analyses under informative dropout. *Biometrics 56*(4), 1241–1248.

Daniels, M. J. and J. W. Hogan (2008). *Missing Data in Longitudinal Studies: Strategies for Bayesian Modeling and Sensitivity Analysis*. CRC Press.

Datta, A., S. Banerjee, A. O. Finley, and A. E. Gelfand (2016). Hierarchical nearest-neighbor Gaussian process models for large geostatistical datasets. *Journal of the American Statistical Association 111*(514), 800–812.

De Blasi, P., S. Favaro, A. Lijoi, R. H. Mena, I. Prünster, and M. Ruggiero (2015). Are Gibbs-type priors the most natural generalization of the Dirichlet process? *IEEE transactions on pattern analysis and machine intelligence 37*(2), 212–229.

De Finetti, B. (1931). Funzione caratteristica di un fenomeno aleatorio. *Atti della R. Academia Nazionale dei Lincei, Serie 6 4*, 251–299.

De Finetti, B. (1974). *Theory of Probability*. John Wiley & Sons, New York.

De Oliveira, V. (2012). Bayesian analysis of conditional autoregressive models. *Annals of the Institute of Statistical Mathematics 64*(1), 107–133.

Denison, D. G. T., B. K. Mallick, and A. F. M. Smith (1998). A Bayesian CART algorithm. *Biometrika 85*(2), pp. 363–377.

Deshwar, A. G., S. Vembu, C. K. Yung, G. H. Jang, L. Stein, and Q. Morris (2015). PhyloWGS: Reconstructing subclonal composition and evolution from whole-genome sequencing of tumors. *Genome Biology 16*(1), 35.

Dexter, D. L., H. M. Kowalski, B. A. Blazar, Z. Fligiel, R. Vogel, and G. H. Heppner (1978). Heterogeneity of tumor cells from a single mouse mammary tumor. *Cancer Research 38*(10), 3174–3181.

Dey, D. K., S. K. Ghosh, and B. K. Mallick (2000). *Generalized Linear Models: A Bayesian Perspective*. CRC Press.

Diggle, P. and M. G. Kenward (1994). Informative drop-out in longitudinal data analysis. *Applied statistics*, 49–93.

Dobson, A. J. and A. Barnett (2008). *An Introduction to Generalized Linear Models*. CRC press.

Doshi, F., K. Miller, J. V. Gael, and Y. W. Teh (2009). Variational inference for the Indian buffet process. In *International Conference on Artificial Intelligence and Statistics*, pp. 137–144.

Escobar, M. D. and M. West (1995). Bayesian density estimation and inference using mixtures. *Journal of the american statistical association 90*(430), 577–588.

Fan, X., W. Zhou, Z. Chong, L. Nakhleh, and K. Chen (2014). Towards accurate characterization of clonal heterogeneity based on structural variation. *BMC bioinformatics 15*(1), 1.

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *The annals of statistics*, 209–230.

Fischer, A., I. Vázquez-García, C. J. Illingworth, and V. Mustonen (2014). High-definition reconstruction of clonal composition in cancer. *Cell reports 7*(5), 1740–1752.

Follmann, D. and M. Wu (1995). An approximate generalized linear model with random effects for informative missing data. *Biometrics*, 151–168.

Geisser, S. and W. F. Eddy (1979). A predictive approach to model selection. *Journal of the American Statistical Association 74*(365), 153–160.

Gelfand, A. E. and D. K. Dey (1994). Bayesian model choice: Asymptotics and exact calculations. *Journal of the Royal Statistical Society. Series B (Methodological)*, 501–514.

Gelman, A., J. B. Carlin, H. S. Stern, and D. B. Rubin (2014). *Bayesian Data Analysis*, Volume 2. Chapman & Hall/CRC Boca Raton, FL, USA.

Gerlinger, M., A. Rowan, S. Horswell, J. Larkin, D. Endesfelder, E. Gronroos, P. Martinez, N. Matthews, A. Stewart, P. Tarpey, I. Varela, B. Phillimore, S. Begum, N. McDonald, A. Butler, D. Jones, K. Raine, C. Latimer, C. Santos, M. Nohadani, A. Eklund, B. Spencer-Dene, G. Clark, L. Pickering,

G. Stamp, M. Gore, Z. Szallasi, J. Downward, P. Futreal, and C. Swanton (2012). Intratumor heterogeneity and branched evolution revealed by multiregion sequencing. *N Engl J Med. 366*(10), 883–892.

Gershman, S. J. and D. M. Blei (2012). A tutorial on Bayesian nonparametric models. *Journal of Mathematical Psychology 56*(1), 1–12.

Geweke, J. (1991). Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. In *Federal Reserve Bank of Minneapolis Research Department Staff Report 148*.

Geweke, J. (2004). Getting it right: Joint distribution tests of posterior simulators. *Journal of the American Statistical Association 99*(467), 799–804.

Geyer, C. J. (1991). Markov chain Monte Carlo maximum likelihood. In *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, pp. 156–163. Interface Foundation of North America.

Ghosal, S. and A. Roy (2006). Posterior consistency of Gaussian process prior for nonparametric binary regression. *The Annals of Statistics*, 2413–2429.

Ghosh, J. and R. Ramamoorthi (2003). *Bayesian Nonparametrics*. Springer Series in Statistics. Springer.

Giordano, R. J., T. Broderick, and M. I. Jordan (2015). Linear response methods for accurate covariance estimates from mean field variational Bayes. In *Advances in Neural Information Processing Systems*, pp. 1441–1449.

Gramacy, R. B. and H. K. H. Lee (2008). Bayesian treed Gaussian process models with an application to computer modeling. *Journal of the American Statistical Association 103*(483), 1119–1130.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika 82*(4), 711–732.

Green, P. J. and S. Richardson (2001). Modelling heterogeneity with and without the Dirichlet process. *Scandinavian journal of statistics 28*(2), 355–375.

Griffiths, T. L. and Z. Ghahramani (2006). Infinite latent feature models and the Indian buffet process. In *Advances in Neural Information Processing Systems 18*.

Griffiths, T. L. and Z. Ghahramani (2011). The Indian buffet process: An introduction and review. *Journal of Machine Learning Research 12*, 1185–1224.

Gusfield, D. (1991). Efficient algorithms for inferring evolutionary trees. *Networks 21*(1), 19–28.

Harel, O. and J. L. Schafer (2009). Partial and latent ignorability in missing-data problems. *Biometrika 96*(1), 37–50.

Heckman, J. (1979). Sample selection bias as a specification error. *Econometrica 47*(1), 153–162.

Henderson, R., P. Diggle, and A. Dobson (2000). Joint modelling of longitudinal measurements and event time data. *Biostatistics 1*(4), 465–480.

Hensman, J., N. Fusi, and N. D. Lawrence (2013). Gaussian processes for big data. *arXiv preprint arXiv:1309.6835*.

Hewitt, E. and L. J. Savage (1955). Symmetric measures on cartesian products. *Transactions of the American Mathematical Society 80*(2), 470–501.

Hjort, N. L. (1990). Nonparametric Bayes estimators based on beta processes in models for life history data. *The Annals of Statistics*, 1259–1294.

Hjort, N. L., C. Holmes, P. Müller, and S. G. Walker (2010). *Bayesian Nonparametrics*, Volume 28. Cambridge University Press.

Hoff, P. D. (2009). *A First Course in Bayesian Statistical Methods*. Springer Science & Business Media.

Hogan, J. W. and N. M. Laird (1997). Mixture models for the joint distribution of repeated measures and event times. *Statistics in medicine 16*(3), 239–257.

Ishwaran, H. and L. F. James (2001). Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association 96*(453), 161–173.

Jain, S. and R. M. Neal (2004). A split-merge Markov chain Monte Carlo procedure for the Dirichlet process mixture model. *Journal of Computational and Graphical Statistics 13*(1), 158–182.

Jiao, W., S. Vembu, A. G. Deshwar, L. Stein, and Q. Morris (2014). Inferring clonal evolution of tumors from single nucleotide somatic mutations. *BMC bioinformatics 15*(1), 35.

Kay, S. R., A. Flszbein, and L. A. Opfer (1987). The positive and negative syndrome scale (panss) for schizophrenia. *Schizophrenia bulletin 13*(2), 261.

Kenward, M. G., G. Molenberghs, and H. Thijs (2003). Pattern-mixture models with proper time dependence. *Biometrika 90*(1), 53–71.

Kim, Y., L. James, and R. Weissbach (2012). Bayesian analysis of multistate event history data: Beta-Dirichlet process prior. *Biometrika 99*(1), 127–140.

Kimura, M. (1969). The number of heterozygous nucleotide sites maintained in a finite population due to steady flux of mutations. *Genetics 61*(4), 893.

Knowles, D. and Z. Ghahramani (2011). Nonparametric Bayesian sparse factor models with application to gene expression modeling. *The Annals of Applied Statistics*, 1534–1552.

Lee, J., P. Mueller, S. Sengupta, K. Gulukota, and Y. Ji (2016). Bayesian inference for intratumour heterogeneity in mutations and copy number variation. *Journal of the Royal Statistical Society: Series C 65*(4), 547–563.

Lee, J., P. Müller, K. Gulukota, Y. Ji, et al. (2015). A Bayesian feature allocation model for tumor heterogeneity. *The Annals of Applied Statistics 9*(2), 621–639.

Li, H. (2014). Toward better understanding of artifacts in variant calling from high-coverage samples. *Bioinformatics 30*(20), 2843–2851.

Li, H. and R. Durbin (2009). Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics 25*(14), 1754–1760.

Lijoi, A. and I. Prünster (2010). Models beyond the Dirichlet process. *Bayesian nonparametrics 28*, 80.

Linero, A. R. (2017+). Bayesian nonparametric analysis of longitudinal studies in the presence of informative missingness. *Biometrika (to appear)*.

Linero, A. R. and M. J. Daniels (2015). A flexible Bayesian approach to monotone missing data in longitudinal studies with nonignorable missingness with application to an acute schizophrenia clinical trial. *Journal of the American Statistical Association 110*(509), 45–55.

Linero, A. R. and M. J. Daniels (2017+). A general Bayesian nonparametric approach for missing outcome data. *Statistical Science (submitted)*.

Little, R. J. (1993). Pattern-mixture models for multivariate incomplete data. *Journal of the American Statistical Association 88*(421), 125–134.

Little, R. J. (1994). A class of pattern-mixture models for normal incomplete data. *Biometrika*, 471–483.

Little, R. J. and D. B. Rubin (2014). *Statistical Analysis with Missing Data*. John Wiley & Sons.

Liu, J. S. (2008). *Monte Carlo Strategies in Scientific Computing*. Springer Science & Business Media.

Lo, A. Y. et al. (1984). On a class of Bayesian nonparametric estimates: I. density estimates. *The Annals of Statistics 12*(1), 351–357.

MacEachern, S. N. (2000). Dependent Dirichlet processes.

MacEachern, S. N. and P. Müller (1998). Estimating mixture of Dirichlet process models. *Journal of Computational and Graphical Statistics 7*(2), 223–238.

Marass, F., F. Mouliere, K. Yuan, N. Rosenfeld, and F. Markowetz (2017). A phylogenetic latent feature model for clonal deconvolution. *The Annals of Applied Statistics 10*(4), 2377–2404.

Mardis, E. R. (2008). Next-generation DNA sequencing methods. *Annu. Rev. Genomics Hum. Genet. 9*, 387–402.

Marjanovic, N. D., R. A. Weinberg, and C. L. Chaffer (2013). Cell plasticity and heterogeneity in cancer. *Clinical Chemistry 59*(1), 168–179.

Marusyk, A., V. Almendro, and K. Polyak (2012). Intra-tumour heterogeneity: A looking glass for cancer? *Nature Reviews Cancer 12*(5), 323–334.

McKenna, A., M. Hanna, E. Banks, A. Sivachenko, K. Cibulskis, A. Kernytsky, K. Garimella, D. Altshuler, S. Gabriel, M. Daly, et al. (2010). The genome

analysis toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Research 20*(9), 1297–1303.

Miller, C. A., B. S. White, N. D. Dees, M. Griffith, J. S. Welch, O. L. Griffith, R. Vij, M. H. Tomasson, T. A. Graubert, M. J. Walter, et al. (2014). SciClone: Inferring clonal architecture and tracking the spatial and temporal patterns of tumor evolution. *PLoS Computational Biology 10*(8), e1003665.

Miller, K. T., T. Griffiths, and M. I. Jordan (2012). The phylogenetic Indian buffet process: A non-exchangeable nonparametric prior for latent features. *arXiv preprint arXiv:1206.3279*.

Molenberghs, G., M. G. Kenward, and E. Lesaffre (1997). The analysis of longitudinal ordinal data with nonrandom drop-out. *Biometrika 84*(1), 33–44.

Molenberghs, G., B. Michiels, M. G. Kenward, and P. J. Diggle (1998). Monotone missing data and pattern-mixture models. *Statistica Neerlandica 52*(2), 153–161.

Müller, P., A. Erkanli, and M. West (1996). Bayesian curve fitting using multivariate normal mixtures. *Biometrika 83*(1), 67–79.

Müller, P., F. Quintana, and G. Rosner (2004). A method for combining inference across related nonparametric Bayesian models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 66*(3), 735–749.

Müller, P. and F. A. Quintana (2004). Nonparametric Bayesian data analysis. *Statistical science 19*(1), 95–110.

Müller, P., F. A. Quintana, A. Jara, and T. Hanson (2015). *Bayesian Nonparametric Data Analysis*. Springer.

National Research Council (2011). *The Prevention and Treatment of Missing Data in Clinical Trials*. National Academies Press.

Navin, N., A. Krasnitz, L. Rodgers, K. Cook, J. Meth, J. Kendall, M. Riggs, Y. Eberling, J. Troge, V. Grubor, et al. (2010). Inferring tumor progression from genomic heterogeneity. *Genome research 20*(1), 68–80.

Neal, R. M. (1992). Bayesian mixture modeling. In *Maximum Entropy and Bayesian Methods*, pp. 197–211. Springer.

Neal, R. M. (1995). *Bayesian Learning for Neural Networks*. Ph. D. thesis, University of Toronto.

Neal, R. M. (1998). Regression and classification using Gaussian process priors. In *Bayesian Statistics 6*, pp. 475–501. Oxford University Press.

Neal, R. M. (2000). Markov chain sampling methods for Dirichlet process mixture models. *Journal of computational and graphical statistics 9*(2), 249–265.

Nik-Zainal, S., L. B. Alexandrov, D. C. Wedge, P. Van Loo, C. D. Greenman, K. Raine, D. Jones, J. Hinton, J. Marshall, L. A. Stebbings, et al. (2012).

Mutational processes molding the genomes of 21 breast cancers. *Cell 149*(5), 979–993.

Nik-Zainal, S., P. Van Loo, D. C. Wedge, L. B. Alexandrov, C. D. Greenman, K. W. Lau, K. Raine, D. Jones, J. Marshall, M. Ramakrishna, et al. (2012). The life history of 21 breast cancers. *Cell 149*(5), 994–1007.

Nowell, P. C. (1976). The clonal evolution of tumor cell populations. *Science 194*(4260), 23–28.

Oesper, L., A. Mahmoody, and B. J. Raphael (2013). THetA: Inferring intra-tumor heterogeneity from high-throughput DNA sequencing data. *Genome Biol 14*(7), R80.

O'Hagan, A. (1978). Curve fitting and optimal design for prediction. *Journal of the Royal Statistical Society. Series B (Methodological) 40*(1), 1–42.

O'Hagan, A. (1995). Fractional Bayes factor for model comparison. *Journal of the Royal Statistical Society. Series B (Methodological) 57*(1), 99 – 138.

Orbanz, P. (2014, May). Lecture notes on Bayesian nonparametrics (`http://stat.columbia.edu/~porbanz/papers/porbanz_BNP_draft.pdf`).

Orbanz, P. and Y. W. Teh (2011). Bayesian nonparametric models. In *Encyclopedia of Machine Learning*, pp. 81–89. Springer.

Paisley, J. W., A. K. Zaas, C. W. Woods, G. S. Ginsburg, and L. Carin (2010). A stick-breaking construction of the beta process. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 847–854.

Papaemmanuil, E., M. Cazzola, J. Boultwood, L. Malcovati, P. Vyas, D. Bowen, A. Pellagatti, J. Wainscoat, E. Hellstrom-Lindberg, C. Gambacorti-Passerini, et al. (2011). Somatic SF3B1 mutation in myelodysplasia with ring sideroblasts. *New England Journal of Medicine 365*(15), 1384–1395.

Parmigiani, G., E. S. Garrett, R. Anbazhagan, and E. Gabrielson (2002). A statistical framework for expression-based molecular classification in cancer. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 64*(4), 717–736.

Pe'er, I., T. Pupko, R. Shamir, and R. Sharan (2004). Incomplete directed perfect phylogeny. *SIAM Journal on Computing 33*(3), 590–607.

Pitman, J. and M. Yor (1997). The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *The Annals of Probability*, 855–900.

Polyak, K. (2011). Heterogeneity in breast cancer. *The Journal of Clinical Investigation 121*(10), 3786.

Pulkstenis, E. P., T. R. Ten Have, and J. R. Landis (1998). Model for the analysis of binary longitudinal pain data subject to informative dropout through remedication. *Journal of the American Statistical Association 93*(442), 438–450.

Raphael, B. J., J. R. Dobson, L. Oesper, and F. Vandin (2014). Identifying

driver mutations in sequenced cancer genomes: Computational approaches to enable precision medicine. *Genome medicine 6*(1), 1.

Rasmussen, C. (2000). The infinite Gaussian mixture model. In *Advances in Neural Information Processing Systems*, pp. 554–560. MIT Press.

Rasmussen, C. E. and C. K. I. Williams (2006). *Gaussian Processes for Machine Learning*. MIT Press.

Robins, J. (1986). A new approach to causal inference in mortality studies with a sustained exposure periodapplication to control of the healthy worker survivor effect. *Mathematical Modelling 7*(9-12), 1393–1512.

Robins, J. M., A. Rotnitzky, and L. P. Zhao (1995). Analysis of semiparametric regression models for repeated outcomes in the presence of missing data. *Journal of the American Statistical Association 90*(429), 106–121.

Rodriguez, A., D. B. Dunson, and A. E. Gelfand (2008). The nested Dirichlet process. *Journal of the American Statistical Association 103*(483), 1131–1154.

Roth, A., J. Khattra, D. Yap, A. Wan, E. Laks, J. Biele, G. Ha, S. Aparicio, A. Bouchard-Côté, and S. P. Shah (2014). PyClone: Statistical inference of clonal population structure in cancer. *Nature methods 11(4)*, 396–398.

Rotnitzky, A., J. M. Robins, and D. O. Scharfstein (1998). Semiparametric regression for repeated outcomes with nonignorable nonresponse. *Journal of the american statistical association 93*(444), 1321–1339.

Roweis, S. and Z. Ghahramani (1999). A unifying review of linear Gaussian models. *Neural computation 11*(2), 305–345.

Rubin, D. B. (1976). Inference and missing data. *Biometrika 63*(3), 581–592.

Rubin, D. B. (1981). The Bayesian bootstrap. *The annals of statistics 9*(1), 130–134.

Scharfstein, D., A. McDermott, W. Olson, and F. Wiegand (2014). Global sensitivity analysis for repeated measures studies with informative dropout: A fully parametric approach. *Statistics in Biopharmaceutical Research 6*(4), 338–348.

Scharfstein, D. O., A. Rotnitzky, and J. M. Robins (1999). Adjusting for nonignorable drop-out using semiparametric nonresponse models. *Journal of the American Statistical Association 94*(448), 1096–1120.

Sengupta, S. (2013). *Two Models Involving Bayesian Nonparametric Techniques*. Ph. D. thesis, University of Florida.

Sengupta, S., K. Gulukota, Y. Zhu, C. Ober, K. Naughton, W. Wentworth-Sheilds, and Y. Ji (2016). Ultra-fast local-haplotype variant calling using paired-end DNA-sequencing data reveals somatic mosaicism in tumor and normal blood samples. *Nucleic acids research 44*(3), e25–e25.

Sengupta, S., J. Wang, J. Lee, P. Müller, K. Gulukota, A. Banerjee, and Y. Ji (2015). BayClone: Bayesian nonparametric inference of tumor subclones

using NGS data. In *Proceedings of The Pacific Symposium on Biocomputing (PSB)*, Volume 20, pp. 467–478.

Sethuraman, J. (1994). A constructive definition of Dirichlet priors. *Statistica sinica*, 639–650.

Shackleton, M., E. Quintana, E. R. Fearon, and S. J. Morrison (2009). Heterogeneity in cancer: Cancer stem cells versus clonal evolution. *Cell 138*(5), 822–829.

Stephens, P. J., P. S. Tarpey, H. Davies, P. Van Loo, C. Greenman, D. C. Wedge, S. Nik-Zainal, S. Martin, I. Varela, G. R. Bignell, et al. (2012). The landscape of cancer genes and mutational processes in breast cancer. *Nature 486*(7403), 400–404.

Stingl, J. and C. Caldas (2007). Molecular heterogeneity of breast carcinomas and the cancer stem cell hypothesis. *Nature Reviews Cancer 7*(10), 791–799.

Strino, F., F. Parisi, M. Micsinai, and Y. Kluger (2013). TrAp: A tree approach for fingerprinting subclonal tumor composition. *Nucleic acids research 41*(17), e165–e165.

Teh, Y. W. (2011). Dirichlet process. In *Encyclopedia of machine learning*, pp. 280–287. Springer.

Teh, Y. W. and D. Gorur (2009). Indian buffet processes with power-law behavior. In *Advances in neural information processing systems*, pp. 1838–1846.

Teh, Y. W., D. Görür, and Z. Ghahramani (2007). Stick-breaking construction for the Indian buffet process. In *International Conference on Artificial Intelligence and Statistics*, pp. 556–563.

Teh, Y. W., M. I. Jordan, M. J. Beal, and D. M. Blei (2006). Hierarchical Dirichlet processes. *Journal of the American Statistical Association 101*(476), 1566–1581.

Thibaux, R. and M. I. Jordan (2007). Hierarchical beta processes and the Indian buffet process. In *International Conference on Artificial Intelligence and Statistics*, pp. 564–571.

Tipping, M. E. and C. M. Bishop (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 61*(3), 611–622.

Tsiatis, A. (2007). *Semiparametric Theory and Missing Data*. Springer Science & Business Media.

Tsiatis, A. A., M. Davidian, and W. Cao (2011). Improved doubly robust estimation when data are monotonely coarsened, with application to longitudinal studies with dropout. *Biometrics 67*(2), 536–545.

Van Loo, P., S. H. Nordgard, O. C. Lingjærde, H. G. Russnes, I. H. Rye, W. Sun, V. J. Weigman, P. Marynen, A. Zetterberg, B. Naume, et al. (2010). Allele-specific copy number analysis of tumors. *Proceedings of the National Academy of Sciences 107*(39), 16910–16915.

Varela, I., P. Tarpey, K. Raine, D. Huang, C. K. Ong, P. Stephens, H. Davies, D. Jones, M.-L. Lin, J. Teague, et al. (2011). Exome sequencing identifies frequent mutation of the SWI/SNF complex gene PBRM1 in renal carcinoma. *Nature 469*(7331), 539–542.

Walker, S. G., P. Damien, P. W. Laud, and A. F. Smith (1999). Bayesian nonparametric inference for random distributions and related functions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 61*(3), 485–527.

Wang, C., M. Daniels, D. O. Scharfstein, and S. Land (2010). A Bayesian shrinkage model for incomplete longitudinal binary data with application to the breast cancer prevention trial. *Journal of the American Statistical Association 105*(492), 1333–1346.

West, M., P. Müller, and M. Escobar (1994). Hierarchical priors and mixture models, with application in regression and density estimation. In *Aspects of Uncertainty*, pp. 363–386. Wiley.

Williamson, S., P. Orbanz, and Z. Ghahramani (2010). Dependent Indian buffet processes. In *International Conference on Artificial Intelligence and Statistics*, Volume 9, pp. 924–931.

Wu, M. C. and R. J. Carroll (1988). Estimation and comparison of changes in the presence of informative right censoring by modeling the censoring process. *Biometrics*, 175–188.

Xu, Y., P. Müller, Y. Yuan, K. Gulukota, and Y. Ji (2015). MAD Bayes for tumor heterogeneity – feature allocation with exponential family sampling. *Journal of the American Statistical Association 110*(510), 503–514.

Zare, H., J. Wang, A. Hu, K. Weber, J. Smith, D. Nickerson, C. Song, D. Witten, C. A. Blau, and W. S. Noble (2014). Inferring clonal composition from multiple sections of a breast cancer. *PLoS Computational Biology 10*(7), e1003703.