

Article

# Learning Data Heterogeneity with Dirichlet Diffusion Trees

Shuning Huo and Hongxiao Zhu \* 

Department of Statistics, Virginia Tech, Blacksburg, VA 24061, USA

\* Correspondence: hongxiao@vt.edu

## Abstract

Characterizing complex heterogeneous structures in high-dimensional data remains a significant challenge. Traditional approaches often rely on summary statistics such as histograms, skewness, or kurtosis, which—despite their simplicity—are insufficient for capturing nuanced patterns of heterogeneity. Motivated by a brain tumor study, we consider data in the form of point clouds, where each observation consists of a variable number of points. Our goal is to detect differences in the heterogeneity structures across distinct groups of observations. To this end, we employ the Dirichlet Diffusion Tree (DDT) to characterize the latent heterogeneity structure of each observation. We further extend the DDT framework by introducing a regression component that links covariates to the hyperparameters of the latent trees. We develop a Markov chain Monte Carlo algorithm for posterior inference, which alternatively updates the latent tree structures and the regression coefficients. The effectiveness of our proposed method is evaluated by a simulation study and a real-world application in brain tumor imaging.

**Keywords:** Dirichlet diffusion tree; data heterogeneity; latent tree models

**MSC:** 62F15

## 1. Introduction

Rapid technological advancements have resulted in the generation of vast amounts of high-dimensional data across diverse domains. Common examples include high-throughput genetic markers, spectral profiles, medical imaging data, and a wide range of other digital measurements. The availability of such data provides a foundation for gaining deeper insights into the underlying mechanisms of complex phenomena. However, simply acquiring more data does not guarantee efficient or accurate knowledge discovery. One major challenge lies in the lack of appropriate methods to understand complex heterogeneous structures—the presence of diverse and dissimilar characteristics between or within samples of a dataset. Such data heterogeneity may be caused by spatial-temporal variations, multiple data sources, sub-populations, nested experimental design, or other unknown latent factors.

As a motivating example, we consider an application involving brain tumor imaging. The dataset comprises 2D slices of brain magnetic resonance imaging (MRI) measurements from 63 patients diagnosed with Glioblastoma Multiforme (GBM), a common and aggressive malignant brain tumor. Patients with GBM usually have a poor prognosis and a short survival time if they remain untreated. By treatment, the median survival time for patients with GBM is 12–15 months [1]. Figure 1 presents two examples based on T2-weighted MRI scans with segmented tumor regions: one from a patient with a longer survival time



Academic Editor: Xuejun Wang

Received: 30 June 2025

Revised: 3 August 2025

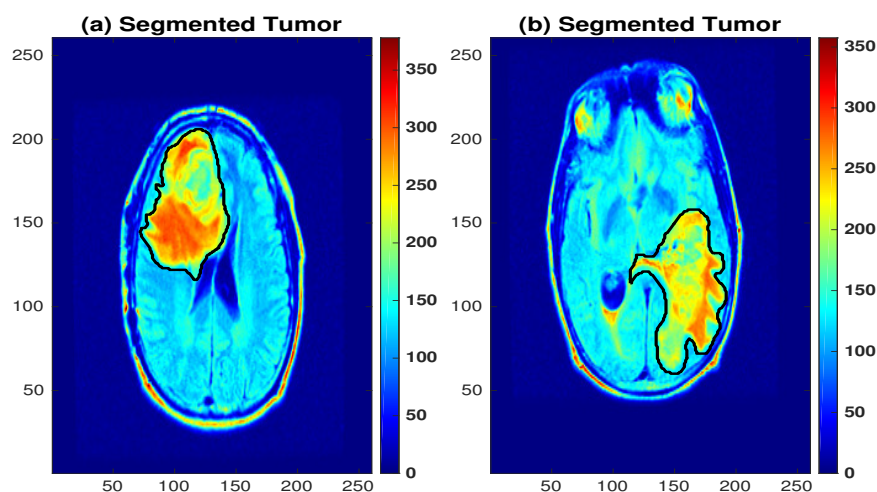
Accepted: 8 August 2025

Published: 11 August 2025

**Citation:** Huo, S.; Zhu, H. Learning Data Heterogeneity with Dirichlet Diffusion Trees. *Mathematics* **2025**, *13*, 2568. <https://doi.org/10.3390/math13162568>

**Copyright:** © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

(a) and the other from a patient with a shorter survival time (b). In addition to tumor imaging data, demographic variables such as age, gender, and survival time, as well as a few genomic variables were also available. As shown in Figure 1, the brain tumor images exhibit spatially heterogeneous pixel intensities. This heterogeneity may reflect underlying pathological changes in the tumor tissue, be associated with genetic variations, and potentially serve as a biomarker for prediction of disease status or prognosis. To understand such heterogeneity, statistical analysis should address two key questions: How to characterize the heterogeneity pattern in pixel intensity? And how do we learn the underlying linkage between the heterogeneity pattern with other variables?



**Figure 1.** Examples of 2D slices of T2-weighted FLAIR MRIs of two patients diagnosed with GBM, with segmented tumor regions outlined in black. (a) Image from a long-survival patient (survival time > 12 months). (b) Image from a short-survival patient (survival time  $\leq$  12 months).

To address these questions, a variety of methods have been proposed. Many existing studies rely on distributional statistics—such as skewness, kurtosis, or percentiles of pixel intensities—to quantify tumor heterogeneity [2]. Others employ texture analysis techniques, such as the co-occurrence matrix [3], or use custom-defined spatial distance measures [4] to capture heterogeneity. Although these approaches are relatively simple and easy to implement, they are inherently limited: they focus primarily on the summary statistics of the data and fail to capture more complex heterogeneity. More recently, a growing number of studies have focused on characterizing tumor heterogeneity and exploring its associations with clinical and genomic variables. For instance, Bharath et al. [5] proposed a geometric framework for analyzing tumor shapes and incorporated the shape features alongside other variables to predict survival time. Yang et al. [6] proposed a quantile regression approach to model subject-specific pixel intensity distributions and examined differences across groups of samples. A recent review by Poursaeed et al. [7] offers a comprehensive overview of state-of-the-art methods for predicting glioblastoma survival using diverse types of input data. Despite these advances, there remains a lack of reliable tools for characterizing the tumor heterogeneity and understanding their associations with survival time, genetic markers, and other demographic factors.

In many scenarios, heterogeneity between variables can be characterized using non-parametric tree procedures. The hierarchical nature of trees allows relationships between variables to be represented in a flexible framework, leading to meaningful interpretations in various scientific applications. Examples include phylogenetic trees [8] for biological evolution, hierarchical clustering [9,10], and decision trees. In Bayesian statistics, tree-based methods have also received a great deal of attention. Markov chain Monte Carlo (MCMC) has been adopted to generate random tree samples in posterior inference. These

methods have been shown to be effective in accommodating complex, hierarchical data structures. For example, random-walk MCMC algorithms have been used extensively to model complex biological evolution processes in Bayesian phylogenetic inference [11–13]. In the Bayesian regression tree framework, Chipman et al. [14] developed the Bayesian classification and regression tree (BCART), which established the foundation for later development. Gramacy and Lee [15] introduced a treed Gaussian process to model nonstationary data with application to response surfaces. Such tree models rely on partitioning the feature space hierarchically to achieve regression or classification. Besides the regression model, Aldous [16] developed parametric probability models for trees to capture topological features and branch length information. Such models establish a basis for modeling heterogeneous data with nonparametric tree priors. Neal [17] proposed a DDT prior, a top-down stochastic process to generate a rooted binary tree. In this work, Neal [17] has shown the effectiveness of the DDT prior in density estimation. Later on, Knowles and Ghahramani [18] extended the Dirichlet Diffusion Tree to embrace more flexibility by allowing multiple children for each node. Compared to DDT, dealing with trees with an arbitrary number of children requires significant changes in the probability model and the computation. In recent years, nonparametric hierarchical tree models have shown effectiveness in various unsupervised learning tasks [19,20]. Despite the progress on using trees to model complex data structures, some critical questions remain unanswered. For example, how to use latent trees to characterize data heterogeneity for a group of samples, and how to associate latent trees with covariates in a regression setup.

Motivated by the brain tumor application and the challenges of modeling data heterogeneity using latent trees, we propose a Bayesian latent tree model to characterize heterogeneity patterns in tumor pixel intensities and to investigate their associations with covariates of interest. We represent the pixel intensities of each tumor image as a set of point clouds, where each observation consists of a variable number of points. We assume that each observation is governed by a latent hierarchical tree structure, which may reflect heterogeneous tumor cell patterns arising from factors such as differing evolution stages or etiologies. Our objective is to identify and compare these latent structures across groups of observations. To be more specific, we adopt Bayesian DDT to model the latent hierarchical tree structure, and propose a regression framework by introducing covariates to the hyper-parameters of the latent trees. The latent tree structures describe the heterogeneity patterns in data, and regression coefficients can be used to detect differences across groups. To perform posterior inference, we propose an MCMC algorithm to alternatively update the latent tree structures and the regression coefficients.

Compared to existing approaches on modeling data heterogeneity, our proposed methodology offers several distinctive advantages: (1) Unlike the ad hoc approaches that depend on density estimation, our latent tree model offers a more flexible framework that can capture hidden hierarchical structures in observed data. (2) Posterior samples of the latent trees can be used to summarize the heterogeneity structure of each observation. (3) By introducing covariates, our proposed model can be used to discover associations between data heterogeneity and other variables of interest, or test differences on data heterogeneity across groups of observations. (4) While sampling latent trees can be computationally expensive, our proposed MCMC algorithm can be performed partially in parallel by using multicore computers, which leads to improved computation scalability. We demonstrate the performance of the proposed method by a simulation study and a real-data application by using brain Glioblastoma Multiforme (GBM) images. In the GBM data analysis, we focus on characterizing the heterogeneity in pixel intensities of the brain tumor images. We also investigate the differences in heterogeneity across two groups of patients: short-survival and long-survival patients.

The remainder of this paper is organized as follows. Section 2.1 provides a brief review of DDT. In Section 2.2, we introduce the proposed Bayesian DDT model, followed by details on posterior inference and the algorithm in Section 2.3. Section 3 presents a simulation study to evaluate the performance of the proposed model. In Section 4, we apply the proposed method to the brain tumor application. Section 5 provides concluding remarks and highlights the novel contribution of the proposed work. Finally, Section 6 discusses related issues and potential directions for future work.

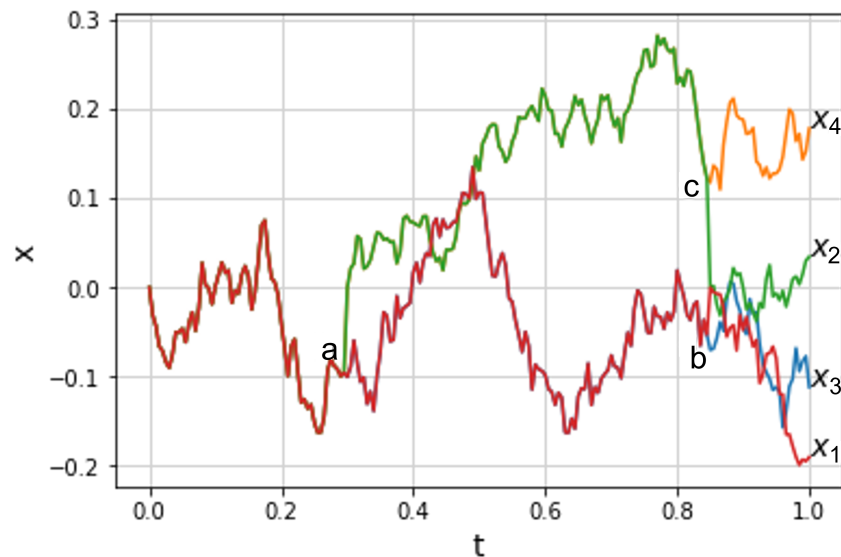
## 2. Methods: Bayesian Dirichlet Diffusion Trees for Data Heterogeneity

### 2.1. An Overview of Dirichlet Diffusion Trees

Complex, high dimensional data—such as curves and images—can be represented in various ways. One effective approach is to use tree structures to capture latent hierarchical relationships within the data. The strength of tree-based representations lies in their flexibility to model hierarchy. In this paper, we focus on binary trees, which offers a favorable balance between interpretability and computational efficiency, and facilitates flexible estimation and uncertainty quantification. Priors over binary trees are often defined through stochastic generative processes. For example, Kingman’s coalescent [21] is a probabilistic model that generates infinite binary trees by randomly merging pairs of lineages backward in time. Another prominent example is the DDT, introduced by Neal [17], which defines a top-down generative process for constructing rooted binary tree via random splits. In a DDT, the tree evolves through a diffusion process governed by a divergence function, and the leaves of the tree correspond to a set of data points. Extensions of the DDT framework have been proposed by Knowles and Ghahramani [18] to allow nodes to have an arbitrary number of children rather than being strictly binary. In this paper, we use DDT to model the latent hierarchical structure underlying a set of point clouds.

We illustrate the generation process of a DDT using an example. Figure 2 shows the sample paths of a DDT for generating  $N = 4$  data points with leaf values  $x_1$ ,  $x_2$ ,  $x_3$ , and  $x_4$ . There are four paths, each terminating at a leaf node. The DDT generation process constructs these paths sequentially. The first path starts at time  $t = 0$  and evolves according to a Brownian motion with variance  $\sigma^2$ , reaching a leaf node at  $t = 1$  with value  $x_1$ . Specifically, denoting the value of the first path at time  $t$  by  $x_1(t)$ , its value at time  $t + dt$  is given by  $x_1(t + dt) = x_1(t) + N(0, \sigma^2 dt)$ . Indeed, one can show that  $x_1(t) \sim \mathcal{N}(0, \sigma^2 t)$ . The second path also begins at  $t = 0$  and initially follows the trajectory of  $x_1$ . At time  $t_a$ , it diverges from the path of  $x_1$ , forming an internal node with divergence time  $t = t_a$  and value  $x_a$ . The divergence time,  $t_a$ , is governed by a divergence function  $a(t)$ , for example,  $a(t) = c/(1 - t)$ . Specifically, the probability for the second path to diverge from the first path within an infinitesimal time period  $dt$  is given by  $a(t)dt$ . After diverging, the second path evolves independently from the first, following a Brownian motion until  $t = 1$  and terminating at a leaf node with value  $x_2$ . The third path follows the trajectories of the previous two paths up to time  $t_a$ , at which point it must choose to follow either the left subtree or right subtree. This decision is determined by the branching probability—the proportion of how many times each branch has been previously traversed. In this example, the probabilities that the third path follows the left subtree and right subtree are both  $1/2$ , since each branch has been previously traversed once. The third path chooses to follow the first path—the path of  $x_1$ —after  $t_a$ . Then, at time point  $t_b$ , it diverges with the first path, reaching the leaf node with value  $x_3$  at  $t = 1$ . Similarly, the fourth path traces the trajectories of the previous three paths up to time  $t_a$ . At that point, it must choose between the left subtree (with probability  $1/3$ ) or right subtree (with probability  $2/3$ ). It ultimately follows the path of  $x_2$ , diverging at  $t_c$  and arriving at the leaf node with value  $x_4$  at  $t = 1$ . In general, a subsequent path follows the path of the previous data points until divergence.

Given an infinitesimal time  $dt$ , the probability of divergence is  $a(t)dt/n$ , where  $n$  denotes the number of data points that have previously been through the path. If divergence has not happened at an internal node, it has to choose whether to follow either the left subtree or the right subtree based on the branching probability, that is, the proportion of how many times each branch has been traversed before. Additionally, Neal [22] proved that the probability distribution generated by the DDT is exchangeable; therefore, the order in which the paths are generated does not affect the overall probability. Therefore, we can also describe the paths in Figure 2 following the order of the leaf node indices  $x_1, x_2, x_3, x_4$  for paths one to four, respectively.



**Figure 2.** A sample of  $N = 4$  data points generated from the Dirichlet Diffusion Tree.

In the generative process, several parameters of the DDT influence the branching behavior of the latent trees, including  $c$  in the divergence function and  $\sigma^2$  in Brownian motion. The parameter  $c$  governs the timing of divergence events, while  $\sigma^2$  establishes diffusion variance. In this paper, we consider the divergence function  $a(t) = c/(1-t)$ , where  $c > 0$ . Since  $a(t)$  determines how likely a diverge will occur, it influences the latent structure of the tree. This latent tree structure, though unobserved, determines the spatial distribution of the leaves, much like the submerged portion of an iceberg. It therefore provides a comprehensive characterization of the data heterogeneity. With the choice of  $a(t) = c/(1-t)$ , larger values of  $c$  (e.g.,  $c > 1$ ) tend to produce more homogeneous structures, leading to weaker dependence among data points. Smaller values (e.g.,  $c < 1$ ) typically result in more heterogeneous patterns, with the emergence of sub-clusters or local clusters [22].

The DDT can be applied to real datasets to estimate latent trees based on a set of observed data points, treating the observed data as leaf nodes and the tree structure as a latent variable. Bayesian inference provides an effective framework for estimating both the latent tree structures and the parameters of the DDT. Applications have been implemented in the context of density estimation and clustering [17]. To estimate the parameters of a DDT, it is often necessary to calculate the likelihood of a latent tree given the observed data. In general, the probability of a specific latent tree can be factorized into two components—a tree factor and a data factor. The tree factor takes into account the probability of obtaining the given hierarchical structure and the associated internal divergence times. The data factor accounts for the probability of obtaining the latent internal node locations as well as the observed leaf nodes. In what follows, we outline the foundations for calculating

the probability of a tree based on the sample illustrated in Figure 2. Further details can be found in Neal [22]. First, the probability that a path does not diverge between two time points  $s < t$  along a branch previously traversed by  $n$  data points is given by

$$Pr(\text{not diverging}) = \exp\left\{\frac{A(s) - A(t)}{n}\right\},$$

where  $A(t) = \int_0^t a(u)du$  denotes the cumulative divergence function. In particular, if the divergence function takes the form  $a(t) = c/(1 - t)$ , then the cumulative divergence is  $A(t) = -c \log(1 - t)$ . Second, if a new path is following a path traversed by  $n$  previous paths, it will diverge from this path within an infinitesimal interval  $dt$  with probability  $a(t)dt/n$ . Finally, at an existing divergence point, the probability that a path chooses a particular branch is proportional to the number of data points that have previously followed that branch. Given the above facts, the probability of the tree factor for obtaining the hierarchical structure in Figure 2 can be calculated by

$$\exp\{-A(t_c)\}a(t_c) \exp\left\{-\frac{A(t_b)}{2}\right\}\frac{a(t_b)}{2} \exp\left\{-\frac{A(t_b)}{3}\right\}\frac{1}{3} \exp\{A(t_b) - A(t_a)\}a(t_a).$$

Given the branching structure and divergence time, one can further calculate the data factor by

$$\begin{aligned} &\phi(x_b; 0, \sigma^2 t_b)\phi(x_c; x_b, \sigma^2(t_c - t_b))\phi(x_a; x_b, \sigma^2(t_a - t_b))\phi(x_4; x_c, \sigma^2(1 - t_c)) \\ &\cdot \phi(x_3; x_c, \sigma^2(1 - t_c))\phi(x_1; x_a, \sigma^2(1 - t_a))\phi(x_2; x_a, \sigma^2(1 - t_a)), \end{aligned}$$

where  $\phi(x; \mu, \sigma^2)$  denotes the probability density function for a normal distribution with mean  $\mu$  and variance  $\sigma^2$ .

### 2.2. A Bayesian Latent Tree Model for Characterizing Data Heterogeneity

Let  $\{\mathbf{y}_1, \dots, \mathbf{y}_n\}$  denote independent observations from  $n$  subjects, where  $\mathbf{y}_i$  represents the  $i$ th observation. We assume that each observation consists of a point cloud, specifically,  $\mathbf{y}_i = \{y_{i1}, y_{i2}, \dots, y_{in_i}\}$  with  $y_{ij} \in \mathbb{R}^k$  for  $i = 1, 2, \dots, n$ . In addition to the point cloud data, we also observe a covariate for each subject, denoted by  $\{z_1, \dots, z_n\}$ , which may represent attributes such as age, gender, or disease status. We further assume that the elements within each point cloud  $\mathbf{y}_i$  are exchangeable, meaning their order does not affect the analysis. These elements constitute the leaves of a latent tree structure. The topological structure of the latent tree—including its branches and internal nodes—captures the underlying evolutionary process or the latent hierarchical organization of the data. We model the joint distribution of the latent tree and its leaves (i.e., the observed data points) using a DDT process.

Let  $\mathcal{T}_i = \{\mathbf{t}_i, \mathbf{x}_i\}$  denote the latent tree structure associated with the observation  $\mathbf{y}_i$ , where  $\mathbf{t}_i$  represents the divergence times and  $\mathbf{x}_i$  denotes the values at the intermediate nodes. The elements of the observed point cloud  $\mathbf{y}_i$  correspond to the leaves of the tree. Therefore,  $\mathcal{T}_i$  and  $\mathbf{y}_i$  together form a complete tree structure. We model the joint distribution of the latent trees  $\{\mathcal{T}_i, \mathbf{y}_i\}$  using DDTs and incorporate the scalar covariate  $z_i$  into the model parameters. The proposed model is summarized as follows:

$$\begin{aligned} \{\mathbf{y}_i, \mathcal{T}_i\} &\sim DDT(\sigma^2, \alpha_{z_i}), \\ \alpha_{z_i}(t) &= \exp(c_0 + c_1 z_i)/(1 - t), \quad t \in [0, 1], \\ \sigma^2 &\sim IG(a, b), \\ (c_0, c_1)^T &\sim \mathcal{MVN}(\mathbf{0}, \sigma_1^2 \mathbf{I}). \end{aligned} \tag{1}$$

Here,  $z_i$  denotes a real-valued covariate. Denote  $\mathbf{c} = (c_0, c_1)^T$  and  $\mathbf{z} = (z_1, \dots, z_n)^T$ . In (1), we introduce an exponential transformation of the linear term  $c_0 + c_1 z_i$  to link the covariate to the divergence function. In the above model, we have placed a multivariate normal prior for the regression coefficients  $\mathbf{c}$  and an inverse-gamma prior on the divergence variance parameter  $\sigma^2$ , characterized by a shape parameter  $a$  and a rate parameter  $b$ . In this model,  $\sigma_1^2$ ,  $a$ , and  $b$  are treated as fixed hyperparameters. We often set the values of  $\sigma_1^2$ ,  $a$ , and  $b$  to induce vague priors, following a similar strategy used by Neal [17]. In many situations, to improve mixing and enhance computational stability, we may also fix the value for  $\sigma^2$  and omit the inverse-gamma prior from the model. Further discussion on how to determine  $\sigma^2$  is provided in Section 6. Denote  $\Omega$  the set of parameters in DDT, then  $\Omega = \{\sigma, c_0, c_1\}$ . The joint distribution of the full tree and the DDT parameters can be written as

$$p(\mathbf{y}_i, \mathcal{T}_i, \Omega) \propto p(\mathbf{y}_i, \mathcal{T}_i \mid \Omega) p(\Omega), \tag{2}$$

where  $p(\Omega)$  denotes the prior distribution for the DDT parameters.

In order to perform inference based on the joint posterior distribution in (2), calculating the likelihood of the full tree is critical. We achieve this by considering the geometric structure of the tree following the notation of Knowles et al. [23]. Let  $\mathcal{S}_i$  denote the set of segments (also known as branches) in a full tree. Let  $[ab]$  denote a contiguous segment in  $\mathcal{S}_i$  that connects nodes  $a$  with  $b$ . Let  $x_a$  and  $x_b$  denote the node locations, and let  $t_a$  and  $t_b$  denote the divergence time at nodes  $a$  and  $b$ , respectively. Denote the number of terminal nodes (i.e., the leaf nodes) under node  $a$  by  $n(a)$ . Furthermore, denote the number of leaf nodes under the left and right subtree of node  $a$  by  $l(a)$  and  $r(a)$ , respectively. Therefore, we have  $l(a) + r(a) = n(a)$ . As reviewed in Section 2.1, the likelihood of a tree can be decomposed into two components: a tree factor and a data factor. The tree factor comprises two elements—the tree branching process and the divergence factor. Consider the segment  $[ab]$  as an example. It is evident that  $n(b)$  nodes will pass through node  $b$ , and among them, one will be the first to split at this node. By the exchangeability property, we may assume without loss of generality that the second path (among all the paths traversing  $[ab]$ ) diverges at node  $b$ . Since  $[ab]$  is a contingent segment, no subsequent paths will diverge during  $[ab]$ . Once the split occurs, the subsequent paths must choose to follow either the left or the right subtree. Under this setup, the probability that no divergence happens on the segment  $[ab]$  and a split happens at node  $b$  can be calculated by

$$p_{nd}(t_b \mid t_a) = \alpha_{z_i}(t_b) \prod_{j=1}^{n(b)-1} \exp[(A(t_a) - A(t_b))/j] = \alpha_{z_i}(t_b) \exp[(A(t_a) - A(t_b))H_{n(b)-1}], \tag{3}$$

where  $H_n = \sum_{i=1}^n 1/i$  denotes the harmonic number at  $n$ . Note that  $b$  in (3) denotes an internal node. All segments in the tree will contribute to the divergence probability by the above format. Collecting all elements related to the internal node  $b$  in  $\mathcal{T}_i$ , i.e.,  $[ab]$  and any segment starting with node  $b$ , the divergence factor for  $t_b$  under our choice of divergence function  $a_{z_i}(t) = \exp(c_0 + c_1 z_i)/(1 - t)$  is

$$\alpha_{z_i}(t_b) \exp\{A(t_b)(H_{l(b)-1} + H_{r(b)-1} - H_{n(b)-1})\} \propto (1 - t_b)^{\exp(c_0 + c_1 z_i) J_{l(b), r(b)} - 1}, \tag{4}$$

where  $J_{l,r} = H_{l+r-1} - H_{l-1} - H_{r-1}$ . To compute the branching probability in the tree factor, consider all paths passing through node  $b$ . Except for the first and second paths, each remaining path must choose to follow either the left or the right subtree. Ultimately, these paths will form  $l(b)$  leaf nodes in the left subtree and  $r(b)$  in the right subtree. This branching probability can be expressed as:

$$p([ab]) = \frac{(l(b) - 1)!(r(b) - 1)!}{(n(b) - 1)!}.$$

To better understand the above formula, one can consider a special case where all  $l(b) > 1$  paths choose the left branch first, followed by all  $r(b) > 1$  paths choosing the right branch. Furthermore, moving from  $a$  to  $b$  constitutes a data factor:

$$p(x_b | x_a, t_a, t_b) = \mathcal{N}(x_b | x_a, \sigma^2(t_b - t_a)).$$

Based on the above calculations, the full joint probability of  $\mathbf{y}_i$  and  $\mathcal{T}_i$  is given by the product over all segments (branches) in  $\mathcal{S}_i$ , and takes the following form:

$$p(\mathbf{y}_i, \mathcal{T}_i) = \prod_{[ab] \in \mathcal{S}_i} p_{nd}(t_b | t_a) p([ab]) p(x_b | x_a, t_a, t_b).$$

### 2.3. Posterior Sampling using Markov Chain Monte Carlo

Based on the Bayesian latent tree introduced in Section 2.2, we propose an MCMC sampling scheme to obtain posterior samples of the model parameters. In particular, we aim to estimate both the topology of the latent trees—including the divergence time and internal node states—and the parameters of DDT. Let  $\{\mathcal{T}_i\}_{i=1}^n$  denote the latent trees and  $\{\mathbf{y}_i\}_{i=1}^n$  the random observations. The joint distribution is given by:

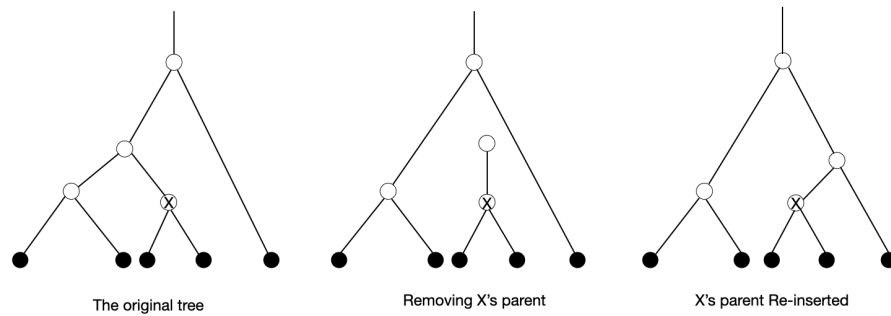
$$\begin{aligned} p(\{\mathbf{y}_i\}_{i=1}^n, \{\mathcal{T}_i\}_{i=1}^n, \Omega) &\propto \prod_{i=1}^n p(\mathbf{y}_i, \mathcal{T}_i | \Omega) p(\Omega), \\ &\propto \prod_{i=1}^n \prod_{[ab] \in \mathcal{S}_i} p_{nd}(t_b | t_a) p(x_b | x_a, t_a, t_b) p([ab]) p(\Omega), \\ &\propto \prod_{i=1}^n \prod_{[ab] \in \mathcal{S}_i} (1 - t_b)^{\alpha_{z_i}(t_b) J_{l(b), r(b)} - 1} \times p(x_b | x_a, t_a, t_b) p([ab]) p(\Omega). \end{aligned}$$

Therefore,

$$\begin{aligned} p(\mathbf{c} | \{\mathbf{y}_i\}_{i=1}^n, \{\mathcal{T}_i\}_{i=1}^n) &\propto \prod_{i=1}^n \prod_{[ab] \in \mathcal{S}_i} (1 - t_b)^{\alpha_{z_i}(t_b) J_{l(b), r(b)} - 1} p(\mathbf{c}), \tag{5} \\ p(\sigma^2 | \{\mathbf{y}_i\}_{i=1}^n, \{\mathcal{T}_i\}_{i=1}^n) &\propto \prod_{i=1}^n \prod_{[ab] \in \mathcal{S}_i} p(x_b | x_a, t_a, t_b) p(\sigma^2). \end{aligned}$$

Based on the above results, we can design an MCMC procedure to sample both the tree structures and the DDT parameters. First, we introduce the procedures to update the latent tree structures. For simplicity, for the step of updating  $\mathcal{T}_i$ , we omit the index  $i$  from  $\mathcal{T}_i$  and  $\mathbf{y}_i$ . To sample the latent tree for each observation, we adopt a Metropolis–Hastings sampler, following Neal [22]. Since each latent tree contains the hierarchical structure of the nodes and the divergence times of the internal nodes, we need to deal with both during updates. We illustrate the procedure of proposing a new tree in the Metropolis–Hastings step in Figure 3. Specifically, in each MCMC iteration, we first randomly select a node—either a leaf or an internal node—identify its parent node, and detach the parent node from the current tree. Let  $x$  denote the randomly selected node,  $p_x$  its parent node, and  $t$  the divergence time of  $p_x$ . We then propose a new divergence time  $t'$  for the removed node following the data generation process of DDT described in Section 2.1. Note that to maintain the rest of the tree structures, we need to set the constraint that  $t' < t$ . Therefore,

we need to repeat the process until the proposed divergence time is earlier than its child node’s divergence time.



**Figure 3.** The procedure for proposing a new tree in the Metropolis–Hastings sampler adopted from Neal [22]. Vertical axis denotes divergence time and horizontal axis denotes location of nodes. Black nodes are leaves, which correspond to random observations  $y_i$  in our model. The internal node marked with “x” indicates the randomly selected node that is updated during the Metropolis–Hastings step.

Denoting the transition probability from tree  $\mathcal{T}$  to  $\mathcal{T}'$  by  $q(\mathcal{T}' | \mathcal{T})$ , the acceptance ratio can be calculated by:

$$\alpha(\mathcal{T}' | \mathcal{T}) = \min \left\{ 1, \frac{p(\mathbf{y}, \mathcal{T}' | \Omega)q(\mathcal{T} | \mathcal{T}')}{p(\mathbf{y}, \mathcal{T} | \Omega)q(\mathcal{T}' | \mathcal{T})} \right\}. \tag{6}$$

In (6),  $q(\mathcal{T}' | \mathcal{T})$  is the probability of the first path in  $\mathcal{T}'$  that goes through the node  $x$ . Similarly,  $q(\mathcal{T} | \mathcal{T}')$  corresponds to the probability of the first path in  $\mathcal{T}$  that goes through the node  $x$ . Furthermore,  $p(\mathbf{y}, \mathcal{T}' | \Omega)$  can be calculated by the full tree likelihood after updating the tree, while  $p(\mathbf{y}, \mathcal{T} | \Omega)$  denotes the original full tree likelihood. Specifically, this likelihood ratio can be further simplified to the changes in likelihood—the data factor and the tree factor. Change in the data factor can be performed by integrating over the locations of the missing node. Furthermore, change in the tree factor is achieved by calculating changes in the divergence factor and the hierarchical structure of all the affected nodes (from root to the subtree rooted at  $p_x$  headed by node  $x$ ) induced by detachment or attachment of the node  $p_x$ . In practice, we sample the tree structures by traversing all internal nodes and implement the above procedure.

We now turn to update of the DDT parameters  $c_0$  and  $c_1$  in  $\Omega$ . These parameters are crucial for capturing the association between data heterogeneity and a covariate of interest. The linear component  $c_0 + c_1z_i$  as a whole governs the divergence rate through the exponential transformation  $\exp(c_0 + c_1z_i)$ . Smaller values of  $\exp(c_0 + c_1z_i)$  typically indicate greater heterogeneity in the data—i.e., the presence of local clusters or sub-clusters—whereas larger values tend to reflect more homogeneous data [22]. We propose a Metropolis–Hasting sampler with a random walk proposal to generate posterior samples of  $c_0$  and  $c_1$ . The proposed value is denoted by  $\mathbf{c}' = (c'_0, c'_1)^T$ . Conditional on the current values of  $\{\mathcal{T}_i\}_{i=1}^n$  and  $\sigma^2$ , the acceptance probability of  $\mathbf{c}$  can be computed as follows:

$$\alpha(\mathbf{c}' | \mathbf{c}) = \min \left\{ 1, \frac{p(\mathbf{c}' | \{\mathbf{y}_i\}_{i=1}^n, \{\mathcal{T}'_i\}_{i=1}^n)}{p(\mathbf{c} | \{\mathbf{y}_i\}_{i=1}^n, \{\mathcal{T}'_i\}_{i=1}^n)} \right\}. \tag{7}$$

In (7), the numerator and denominator of the factor term can be calculated by following (5). As a special case, when there are no covariates  $\{z_i, i = 1, \dots, n\}$  in the model, the divergence function simplifies to  $\alpha_{z_i}(t) = c_0/(1 - t)$ . In this setting, we can derive a conjugate

prior for  $c_0$  by assuming a Gamma prior with shape parameter  $\alpha_c$  and rate parameter  $\beta_c$ . The corresponding conditional posterior of  $c_0$  is then given by

$$\begin{aligned}
 p(c_0 \mid \{\mathbf{y}_i\}_{i=1}^n, \{\mathcal{T}_i\}_{i=1}^n) &\propto \prod_{i=1}^n \prod_{[ab] \in \mathcal{S}_i} (1 - t_b)^{c_0 J_{l(b), r(b)} - 1} p(c_0), \\
 &\propto \left\{ \prod_{i=1}^n \prod_{[ab] \in \mathcal{S}_i} (1 - t_b)^{c_0 J_{l(b), r(b)} - 1} \times c_0^{\alpha_c - 1} \exp^{-\beta_c c_0} \right\}, \\
 &\propto \left\{ \prod_{i=1}^n \prod_{[ab] \in \mathcal{S}_i} \exp((c_0 J_{l(b), r(b)} - 1) \log(1 - t_b)) \right\} \times c_0^{\alpha_c - 1} \exp^{-\beta_c c_0}, \\
 &\propto c_0^{\alpha_c - 1} \exp^{-c_0(\beta_c + \sum_{i=1}^n \sum_{[ab] \in \mathcal{S}_i} \log(1 - t_b))}.
 \end{aligned}$$

Therefore,  $c_0 \sim \text{Gamma}(\alpha_c, \beta_c + \sum_{i=1}^n \sum_{[ab] \in \mathcal{S}_i} \log(1 - t_b))$ .

Given the latent tree structure of each observation—specifically, the intermediate nodes and the divergence times—it is convenient to sample the diffusion variance parameter  $\sigma$  by an inverse gamma distribution, i.e.,

$$\sigma^2 \sim \text{IG}\left(a + \frac{\sum_{i=1}^n N_{\mathcal{S}_i}}{2}, b + \sum_{i=1}^n \sum_{[ab] \in \mathcal{S}_i} \frac{(x_b - x_a)^2}{2(t_b - t_a)}\right), \tag{8}$$

where  $N_{\mathcal{S}_i}$  denotes the total number of segments belonging to the tree  $\mathcal{T}_i$ .

The detailed steps of our proposed MCMC sampler are summarized in Algorithm 1. To improve computation efficiency, when applying Algorithm 1, we employ parallel computing to update the latent trees for different observations independently.

---

**Algorithm 1:** The MCMC sampler for the latent regression tree model

---

- 1 Initialize  $\{\mathcal{T}_i\}_{i=1}^n, \Omega$ ; Denote  $B$  the number of posterior iterations to perform.
  - 2 **for**  $b$  from 1 to  $B$  **do**
  - 3     **for**  $i$  from 1 to  $n$  **do**
  - 4         Sample  $\mathcal{T}_i$  given  $\mathbf{y}_i$  and  $\Omega$ ;
  - 5         Propose a candidate tree through the following steps: Randomly select a node  $x \in \mathcal{T}_i$ . Denote the parent node of  $x$  by  $p_x$ , and denote the divergence time of  $x$  by  $t$ ;
  - 6         Detach the subtree rooted at  $p_x$  from  $\mathcal{T}_i$ ;
  - 7         Generate  $t' \in (0, 1)$  by DDT until  $t' < t$ ;
  - 8         Attach the subtree rooted at  $p_x$  at the new divergence time  $t'$  in  $\mathcal{T}_i$ . We treat the resulting tree as the candidate, denoted by  $\mathcal{T}'_i$ ;
  - 9         Calculate  $\alpha(\mathcal{T}'_i \mid \mathcal{T}_i)$  following (6);
  - 10     **end**
  - 11     Given  $\{\mathcal{T}_i\}$  and  $\sigma$ , sample  $\mathbf{c}'$  by the Metropolis–Hasting sampler described in (7);
  - 12     Given  $\{\mathcal{T}_i\}$  and  $\mathbf{c}$ , sample  $\sigma^2$  following (8);
  - 13 **end**
- 

### 3. Simulation Study

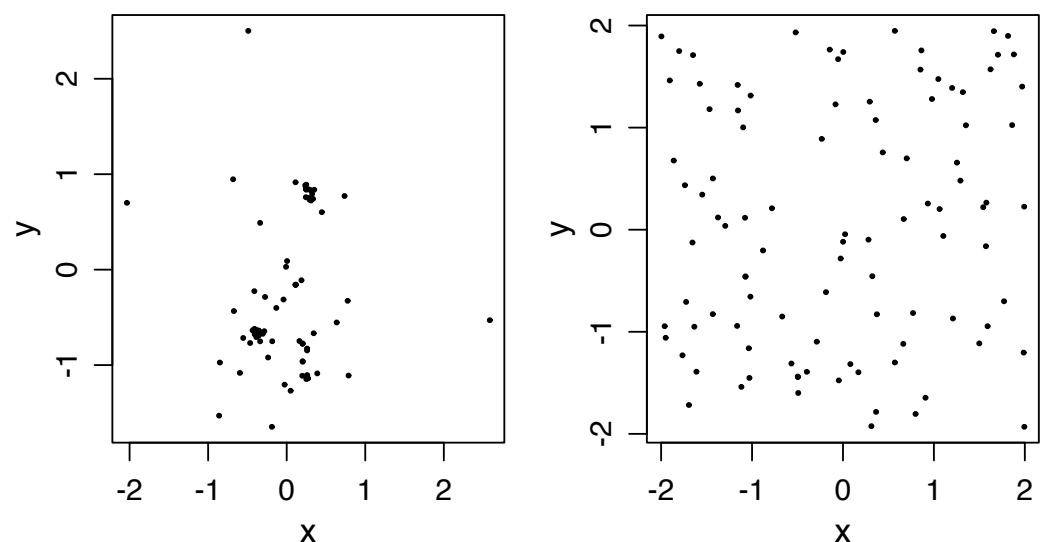
#### 3.1. Simulation Setting

In this section, we design a simulation study to evaluate the effectiveness of our proposed approach in capturing heterogeneity patterns and estimating group differences in data heterogeneity. To facilitate visualization, we generate two groups of point clouds

in a 2D domain, each following distinct heterogeneity structures. The first group consists of point clouds exhibiting two sub-clusters, while the second group contains point clouds that are homogeneous across the domain. Let  $N_i$  denote the number of observations (point clouds) in group  $i$  for  $i = 1, 2$ , and let  $n_{ij}$  denote the number of data points within the  $j$ th observation of group  $i$ , where  $j = 1, \dots, N_i$ .

To generate data with heterogeneous structures, we simulate  $N_1 = 30$  point clouds. Each point cloud is generated independently following a DDT process (without covariates), using the divergence function  $\alpha(t) = 0.25/(1 - t)$ . Each point cloud consists of  $n_{1j} = 100$  data points in  $\mathcal{R}^2$ , for  $j = 1, \dots, N_1$ . To generate data with homogeneous structures, we simulate  $N_2 = 30$  observations, each consisting of  $n_{2j} = 100$  data points in  $\mathcal{R}^2$ , for  $j = 1, 2, \dots, N_2$ . For each data point, the  $x$ - and  $y$ -coordinates are independently drawn from a common uniform distribution:  $x \sim \text{Unif}(-2, 2)$  and  $y \sim \text{Unif}(-2, 2)$ . This results in point clouds that lie in a similar spatial domain to those in the heterogeneous group. For observations in this group, we set the covariates to  $z_i = 0$  for  $i = 1, \dots, N_2$ .

Figure 4 displays two simulated point clouds, one from each group. The left panel shows a point cloud from the heterogeneous group, characterized by inhomogeneous spatial distribution with evident local clusters and sub-clusters. In contrast, the right panel presents a point cloud from the homogeneous group, where data points are uniformly distributed and exhibit no apparent substructure.



**Figure 4.** Plots of two simulated point clouds, one from each group. The **left panel** shows a point cloud in the heterogeneous group and the **right panel** shows a point cloud from the homogeneous group.

We pooled the simulated data from both groups and applied the proposed MCMC algorithm to obtain posterior samples. During implementation, we specified the divergence function as  $\alpha_i(t) = \exp(c_0 + c_1 z_i)/(1 - t)$ , for  $i = 1, 2, \dots, N_1 + N_2$ . The parameters  $(c_0, c_1)^T$  were initialized at  $(0, 0)^T$ , and the initial parent–child relationships were assigned randomly. Specifically, we assigned a unique index to each node and randomly initialized the parent–child relationships. Additionally, we set identical divergence times for internal nodes sharing the same index across all observations. This initialization ensures that, despite slight variations in tree topology due to randomized parent–child assignments, the divergence times remain identical across all latent trees. To improve mixing, we set the hyperparameter  $\sigma = 5$ , based on the posterior mean estimate obtained by running DDT on one randomly selected point cloud from the *heterogeneous* group.

We ran a total of 15,000 MCMC iterations, discarding the first 10,000 as burn-in. To reduce autocorrelation, we applied thinning by retaining one sample every five iterations, resulting in 1000 posterior samples per parameter. The posterior samples of the regression coefficient  $c_1$  are used to assess the differences in heterogeneity between the two groups. Specifically, if the 95% credible interval for  $c_1$  does not include zero, it indicates that the latent tree structures of the two groups have significantly different divergence behaviors, suggesting that the two groups are different in terms of the heterogeneity pattern. Conversely, if the 95% credible interval for  $c_1$  includes zero, it suggests no significant difference in data heterogeneity between the groups. Furthermore, the posterior estimate of  $\exp(c_0 + c_1 z_i)$  reflects the divergence probability of the latent tree. Smaller values indicate a lower likelihood of divergence, particularly in the early to middle stages of the latent tree process. In addition to the regression coefficients, the posterior samples of the latent trees offer richer insights into the underlying heterogeneity structure. Relevant summary statistics—such as divergence times or the “distance” between two randomly selected nodes—can be computed to characterize the heterogeneity within each point cloud. These summaries can then be compared across groups to assess differences in heterogeneity patterns. Based on our observations, divergence occurring at later stages of the latent tree process often indicates greater sub-structure in the data, with point clouds exhibiting more pronounced heterogeneity. Therefore, in this simulation, we use the divergence time as a summary statistic to quantify data heterogeneity. Specifically, we pooled the divergence times of all point clouds within each group and reported the 95% credible interval of the pooled divergence times.

### 3.2. Simulation Results

We summarize the simulation results using the statistics introduced in Section 3.1. Posterior estimates are reported in Table 1, which presents the 2.5%, 50%, and 97.5% quantiles of the regression coefficients  $c_0$  and  $c_1$  after the burn-in period. The 95% credible interval for  $c_1$  is estimated to be  $[-2.2270, -1.4199]$ , which does not include zero, indicating significant difference in data heterogeneity between the two groups. Specifically, the data in the homogeneous group yield an estimate of  $E(\exp(c_0)) = 2.376$ . In contrast, the heterogeneous data group yields an estimate of  $E(\exp(c_0 + c_1)) = 0.409$ , which corresponds to a more locally clustered, clumpy pattern compared to the homogeneous group. These results support the conclusion that, in our simulation study, the significance of the regression coefficient  $c_1$  effectively captures the differences in heterogeneity patterns between the two groups.

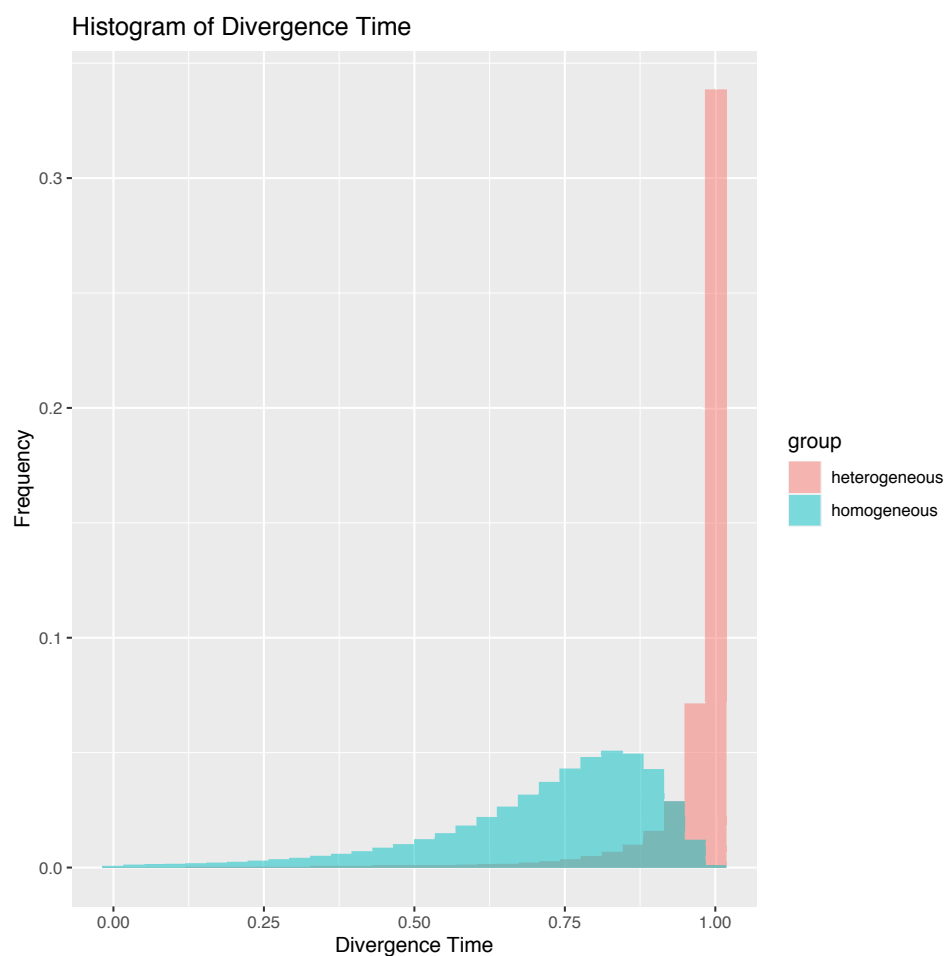
**Table 1.** Summary of posterior samples in the simulation study.  $\{t_1\}$  denotes the divergence times in the homogeneous group, and  $\{t_2\}$  denotes those in the heterogeneous group.

Parameter	Quantile		
	2.5%	50%	97.5%
$c_0$	0.6386	0.8041	1.2013
$c_1$	-2.2270	-1.6980	-1.4199
$\{t_1\}$	0.2236	0.7631	0.9492
$\{t_2\}$	0.6462	0.9963	0.9998

In addition to summary statistics for  $c_0$  and  $c_1$ , Table 1 also reports quantiles for the divergence times of posterior latent trees across all posterior samples in the two groups. Specifically, the 95% credible interval for divergence times in the heterogeneous group is  $[0.6462, 0.9998]$ , whereas that in the homogeneous group is  $[0.2236, 0.9492]$ . To better visualize the distribution of divergence times, Figure 5 presents a histogram of the diver-

gence times for the two groups. It is evident that the distribution of the heterogeneous group illustrates greater skewness, with the mode concentrated closer to one. This suggests that latent trees in the heterogeneous group tend to diverge later—closer to the leaf nodes—whereas in the homogeneous group, divergence tends to occur earlier—closer to the root. Intuitively, as illustrated in the left panel of Figure 4, the heterogeneous group exhibits more localized clusters, implying multiple local modes in the underlying intensity function, which corresponds to shorter segment lengths near the terminal nodes of the latent tree.

Beyond the summary statistics presented in Table 1, we also illustrated one posterior sample of a latent tree from the homogeneous group in Figure 6, and one from the heterogeneous group in Figure 7. In both figures, branch lengths are represented using a heat map: green indicates shorter branches, while red highlights longer ones. Comparing Figures 6 and 7, we observe that the latent tree from the homogeneous group generally has longer branches and sub-branches, with splits occurring earlier in the tree. In contrast, the latent tree from the heterogeneous group tends to exhibit later-stage divergence and shorter sub-branches, reflecting its more localized clustering structure. These results demonstrate that both the regression coefficients and the latent tree structures effectively capture data heterogeneity and reveal differences across groups of samples. Regarding computation, we implemented parallel computing using 30-cores to update the tree structures on a Linux server equipped with an Intel(R) Xeon(R) CPU E5-4627v2 @ 3.30 GHz and 252 GB of RAM. On average, completing 15,000 MCMC iterations required approximately 4.84 h.



**Figure 5.** Histograms of divergence times in posterior tree structures in the *homogeneous* group and the *heterogeneous* group.

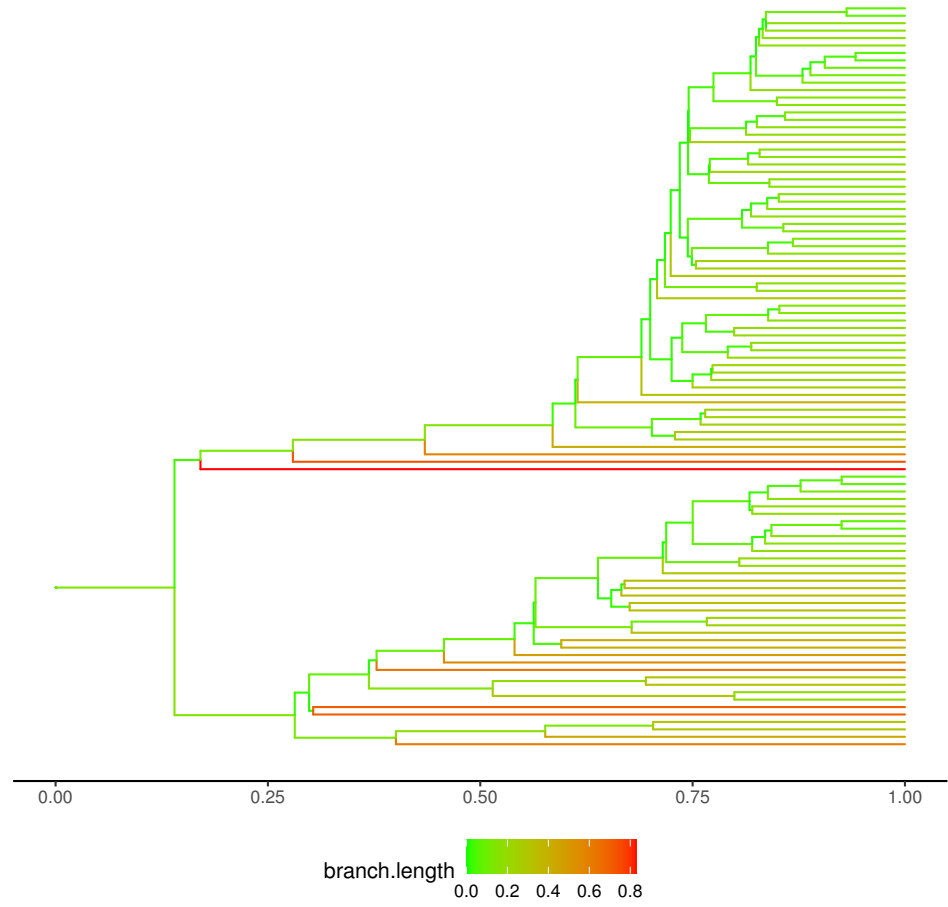


Figure 6. One posterior sample of a latent tree in the homogeneous group.

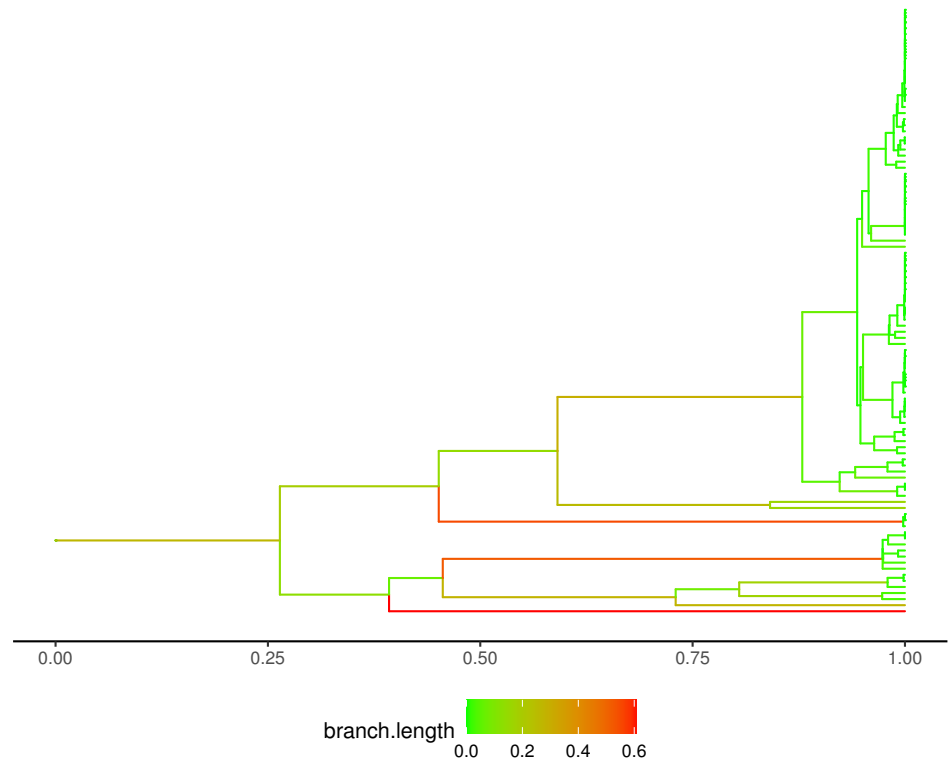
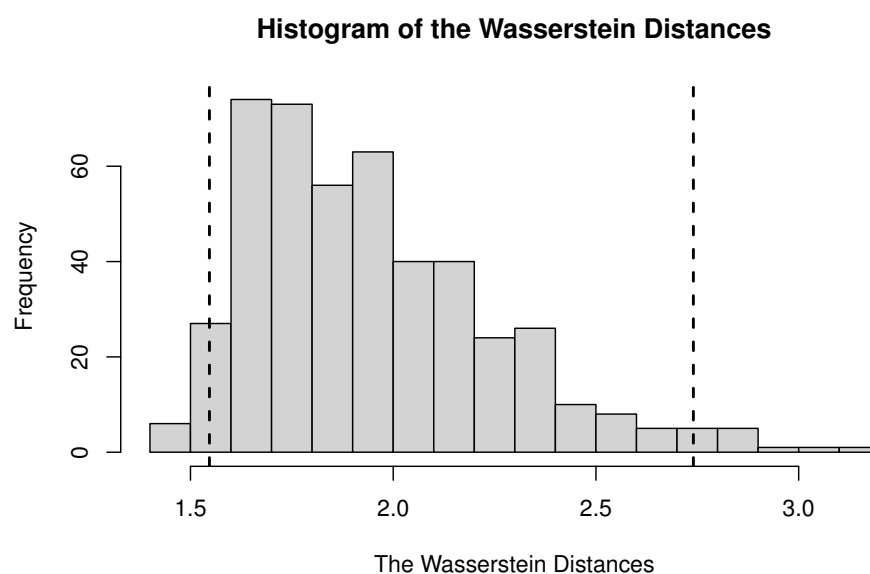


Figure 7. One posterior sample of a latent tree in the heterogeneous group.

It is worth noting that directly comparing the proposed method with an alternative is challenging due to the lack of a universal statistic for all methods. Nevertheless, for comparison purposes, we applied a simple alternative metric to further assess the differences in heterogeneity patterns between the two groups. Specifically, we treated each 2D point cloud as a random sample from an empirical distribution and computed the Wasserstein distances of order 2 between each pair of samples—one from the heterogeneous group and one from the homogeneous group. The Wasserstein distance of order 2 is defined as the square root of the total cost incurred when transporting measure  $\mu$  to measure  $\nu$  in an optimal way, where the cost of transporting a unit of mass from  $x$  to  $y$  is given as  $\|x - y\|^2$  [24]. A histogram of the resulting Wasserstein distances is shown in Figure 8, with the 0.025 and 0.975 quantiles indicated by vertical dashed lines. These quantiles define a 95% confidence interval for the Wasserstein distances: [1.547, 2.740]. Since this interval does not include zero, we conclude that with 95% probability, the Wasserstein distance between the empirical distributions of the two groups is greater than zero.



**Figure 8.** Histogram of pairwise Wasserstein distances between samples from the heterogeneous and the homogeneous groups. Dashed lines indicate the 0.025 and 0.975 sample quantiles.

Finally, to evaluate the sensitivity of the posterior inference to prior specifications, we also conducted a sensitivity analysis by rerunning the simulation study using two alternative values for the DDT parameter  $\sigma$ :  $\sigma = 1$  and  $\sigma = 10$ . Using the posterior samples, we performed a Kolmogorov–Smirnov (K-S) test to assess differences in the divergence times between the heterogeneous and homogeneous groups. The resulting K-S test statistics were 0.457 for  $\sigma = 1$  ( $p$ -value = 0) and 0.299 for  $\sigma = 10$  ( $p$ -value = 0). In contrast, the original simulation with  $\sigma = 5$  yielded a K-S test statistic of 0.803 ( $p$ -value = 0). These results suggest that all three choices of  $\sigma$  lead to consistent findings—the divergence times of the two groups differ significantly in terms of their empirical cumulative distribution functions.

#### 4. Real Data Application

We apply the proposed method to the brain tumor dataset described in Section 1. The imaging data were obtained from The Cancer Genome Atlas (TCGA) database. The website for the GBM study is at The Cancer Imaging Archive (<https://www.cancerimagingarchive.net/>, accessed on 7 August 2025). In addition to brain images, we acquired the patients’ clinical characteristics, including survival time, gender, age, and several genetic variables. The clinical data were retrieved from cBioPortal ([http:](http://)

[//www.cbiportal.org/](http://www.cbiportal.org/), accessed on 7 August 2025). The complete dataset—comprising T2-weighted/FLAIR images and clinic variables—includes 63 samples, of which 21 are female, and 42 are male. Several preprocessing steps were applied to the imaging data. The raw MRI scans were first processed using the Medical Image Processing Analysis and Visualization software; steps include spatial registration and intensity bias correction. Tumor regions were then segmented using the Medical Image Interaction Toolkit (MITK, <https://www.mitk.org/>, accessed on 7 August 2025). The resulting segmented tumor regions preserve high signal intensity and clearly differentiate between edema and tumor tissue. Images and their 3D tumor masks were subsequently resliced, and the axial slice with the largest tumor area was selected for analysis. Further details of the preprocessing steps are described in Bharath et al. [25], Bharath et al. [5], and the references therein.

Using the proposed model, we aim to characterize latent heterogeneity patterns in tumor pixel intensities and to identify differences in data heterogeneity between the long-survival (>12 months) and short-survival (≤12 months) patient groups. To this end, we randomly sampled approximately 15% of the pixel intensities from each segmented tumor region and treated the sampled values as a point cloud. This yielded observations with point counts ranging from 66 to 1043. The final dataset consists of 63 observations, with 37 from patients in the long-survival group and 26 from those in the short-survival group.

We implemented the proposed MCMC algorithm with the DDT parameter fixed at  $\sigma = 5$ , which helps improve the mixing of the Markov chain. We set  $z_i = 1$  if the sample belongs to a patient with long-survival time, and  $z_i = 0$  otherwise. Under this setup, the regression coefficient  $c_1$  captures the difference between the long-survival and short-survival groups. To improve computational efficiency, we implemented parallelization in the step of updating latent trees during MCMC iterations. The algorithm was run for 15,000 MCMC iterations, with the first 12,000 iterations discarded as burn-in. We further applied thinning by retaining one sample for every five iterations, resulting in a total of 600 posterior samples for each parameter. Posterior estimation results were summarized in the same manner as in the simulation study presented in Section 3.2. Table 2 reports the 2.5%, 50%, and 97.5% quantiles of the model parameters  $c_0$ ,  $c_1$ , and the divergence times of the latent trees after the burn-in period.

**Table 2.** Summary of posterior samples in real data application. Here,  $\{t_1\}$  denote the divergence times of latent trees for observations in the long-survival group, and  $\{t_2\}$  denote those in the short-survival group.

Parameter	Quantile		
	2.5%	50%	97.5%
$c_0$	0.3222	0.5282	0.8427
$c_1$	0.2143	0.4721	0.9579
$\{t_1\}$	0.4658	0.8439	0.9790
$\{t_2\}$	0.5138	0.9019	0.9927

As shown in Table 2, the 95% credible interval for  $c_1$  is [0.2143, 0.9579], indicating a modest positive effect associated with  $z_i = 1$ . This suggests that observations from the long-survival group tend to higher divergence parameters compared to those from the short-survival group. In addition to the quantile statistics for  $c_0$  and  $c_1$ , summary statistics of the latent trees offer further insight into the results. We report summary statistics of the divergence times for latent trees by pooling all samples in each group. Specifically, the 95% credible interval for the divergence times is [0.4648, 0.9790] for samples in the long-survival group, and [0.5138, 0.9927] for those in the short-survival group. To further illustrate the

distribution of divergence times, we present histograms of the pooled divergence times for the two groups in Figure 9. The distribution for the short-survival patients shows slightly greater skewness than that of the long-survival group. Additionally, the mode of the distribution for the long-survival group is lower than that of the short-survival group. This indicates that latent trees for patients with shorter survival times tend to have more branching near the leaves, suggesting that pixel intensities in the short-survival group may exhibit greater heterogeneity compared to those in the long-survival group. To further quantify the differences between the two groups shown in Figure 9, we performed a K-S test to assess whether the divergence times are significantly different. Our result yielded a K-S test statistic of 0.208 ( $p$ -value = 0), indicating that the two groups have significantly different empirical cumulative distribution functions. For better visualization, we also present one posterior sample of a latent tree for a short-survival patient in Figure 10, and one for a long-survival patient in Figure 11. The latent tree in Figure 10 exhibits frequent divergences near the leaves with small branches, suggesting higher local heterogeneity. These local heterogeneity structures provide valuable insights into clinical assessment and decision-making. For example, higher levels of heterogeneity often indicate higher variations in tumor structure, cellularity, angiogenesis, necrosis, or genetic diversity. They may also suggest that the tumor cells are biologically more complex, potentially more aggressive, and may have a worse prognosis. In terms of computation, the full MCMC run of 15,000 iterations required approximately 14.2 h to complete.

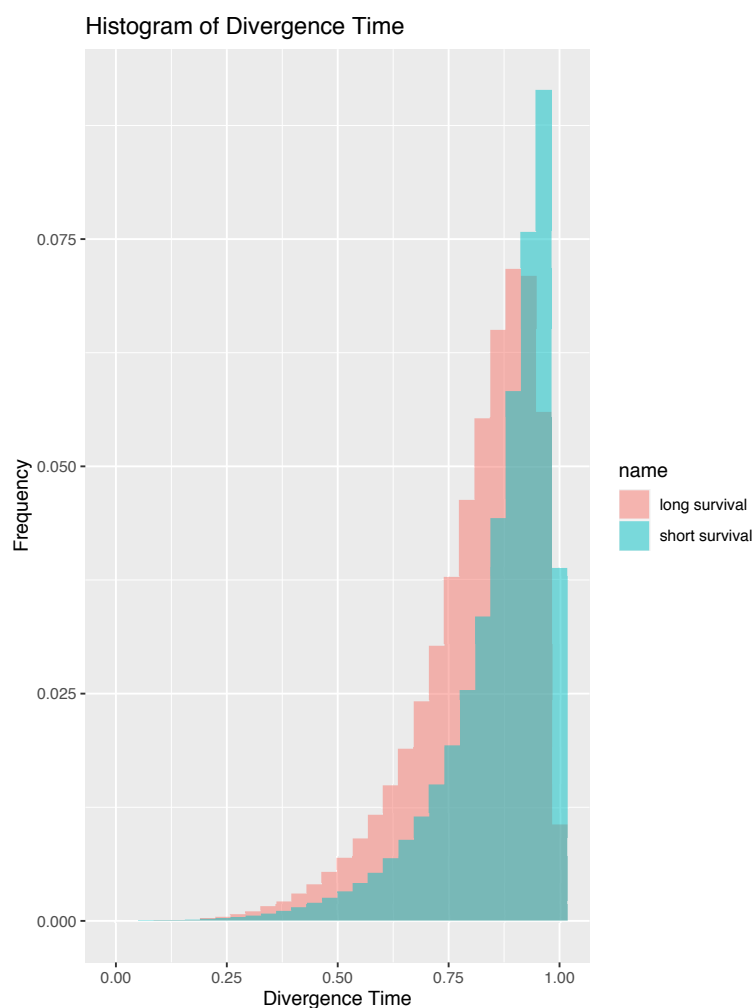


Figure 9. Histograms of divergence times in posterior tree structures.

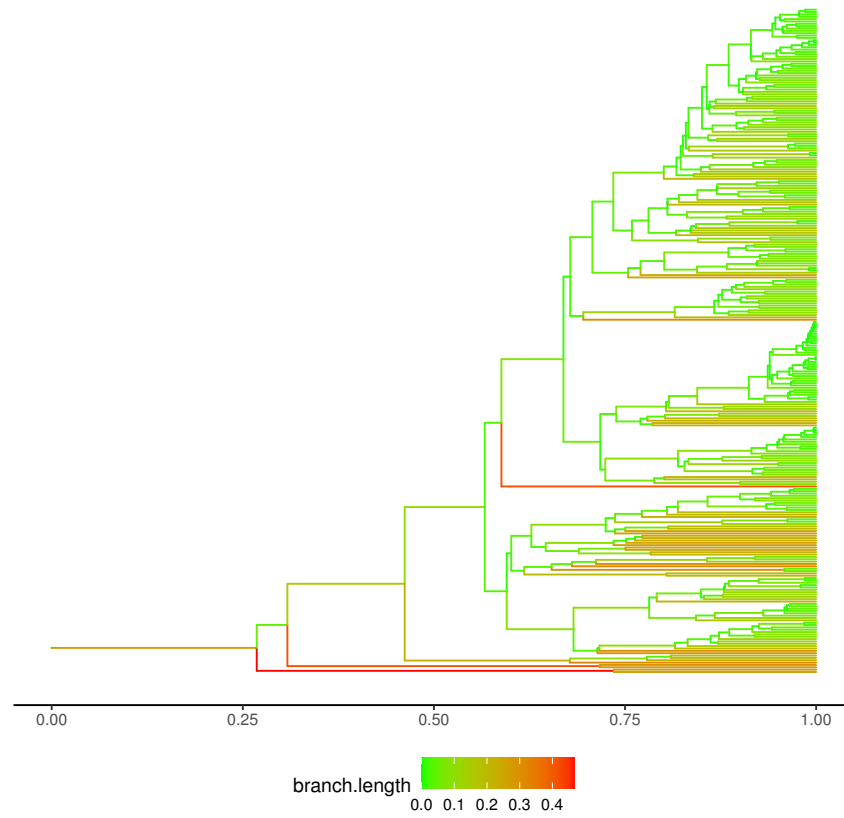


Figure 10. One posterior latent tree sample of one randomly selected short-survival patient.

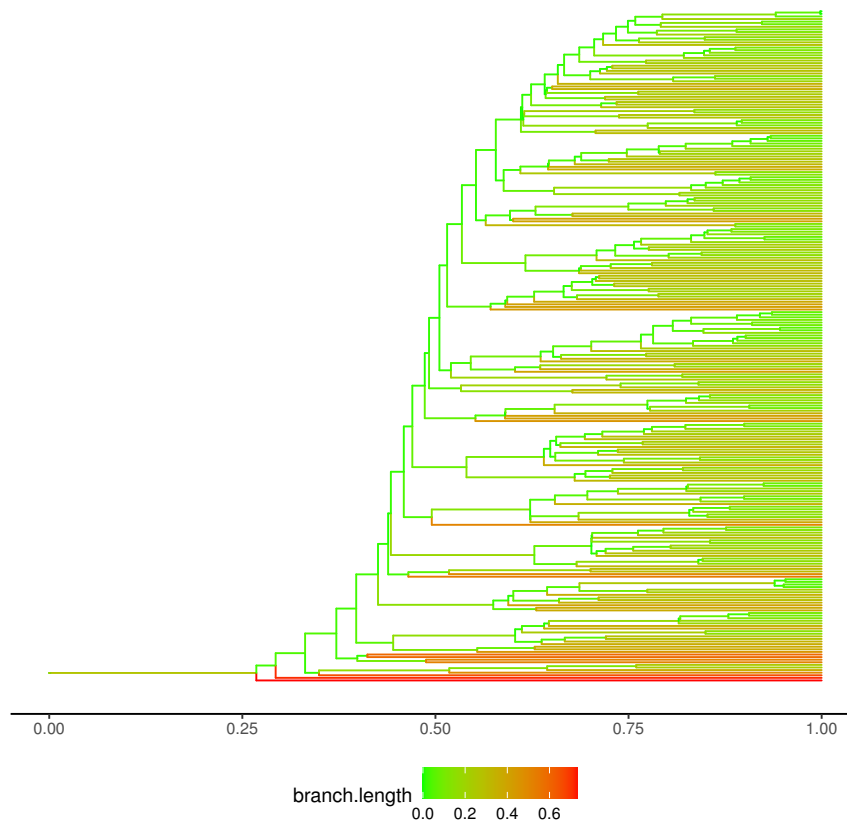


Figure 11. One posterior latent tree sample of one randomly selected long-survival patient.

## 5. Conclusions

We have proposed a Bayesian latent tree model to characterize complex heterogeneous structures in data. Such heterogeneous structures include, but are not limited to, local clusters, hierarchical organizations, and nested sub-clusters. We employed the Bayesian nonparametric DDT prior to capture the latent heterogeneity structure of each observation, and introduced a regression component that links covariates to the hyperparameters of the latent trees. We applied the proposed method to the segmented GBM tumor images to investigate differences in pixel intensity heterogeneity between long-survival and short-survival patient groups. The significance of the regression coefficient indicates differences in heterogeneity between the two groups, while the posterior tree structures provide deeper insights into latent heterogeneity patterns.

The novelty of the proposed work lies in two key aspects. First, to our knowledge, this is the first framework to adopt the DDT prior within a regression setup, enabling the characterization of systematic heterogeneity patterns by borrowing strength across groups of samples. Second, our application to brain tumor images is novel: the pixel intensities in each segmented tumor image are not measured on a common domain due to variations in tumor location within the brain, and the number of pixels varies across images. As a result, many traditional analyses such as principal component analysis and functional data analysis do not directly apply. Our proposed method characterizes data heterogeneity using latent trees and captures group differences through the regression coefficient. It therefore facilitates a deeper understanding of the underlying mechanisms of cell evolution and disease progression based on similar types of medical imaging data.

## 6. Discussion

In the proposed framework, we considered a discrete covariate; however, the model can be readily applied to continuous covariates for assessing associations with continuous predictors.

The proposed method employs MCMC to sample from the posterior distributions of both DDT parameters and the latent tree structures. However, the mixing of MCMC can be poor when dealing with large trees, posing computational challenges. To mitigate this, in the GBM data analysis, we uniformly subsampled the 2D tumor slices and worked with a subset of pixel intensities from each tumor image. Our current algorithm is able to handle real datasets with the number of leaf nodes ranging from 66 to 1043 and a sample size of 63. More efficient algorithms for sampling tree structures, such as annealing MCMC [26] or Bayesian approximation inference [27], may be further explored to improve computational scalability. Additionally, we recommend incorporating empirical Bayesian strategies to pre-specify the diffusion parameter  $\sigma$ . In our analysis, the value of  $\sigma$  was selected by fitting a DDT model without covariates to a representative tumor sample.

For the divergence function  $a(t)$ , we have assumed  $a_{z_i}(t) = \exp(c_0 + c_1 z_i) / (1 - t)$ , which extends the simple form  $a(t) = c / (1 - t)$  used in the one-sample setup. As suggested by Neal [22] and Neal [17], other formats of  $a(t)$  are possible, such as  $a(t) = b + d / (1 - t)^2$ . The properties of these alternative forms for estimating covariate effects remain unstudied. We leave this as a direction for future research.

In the real data application, we considered a binary covariate created by splitting the survival times at a 12-month cutoff. The cutoff was a practical choice based on the characteristics of the tumor data. A previous unsupervised analysis by Bharath et al. [5] on the same dataset identified two clusters based on tumor shapes. The clustering pattern appeared to be associated with survival times, with one cluster exhibiting longer survival times than the other. Motivated by this observation, Bharath et al. [25] and our current study adopted a cutoff of 12-months to divide patients into two groups: one corresponding

to shorter-survival times and the other to longer-survival times. We acknowledge that the results could differ if alternative cutoff values are used or if more than two groups are considered.

It is worth noting that the current study considers a regression setup with a binary covariate, where the group membership is assumed to be known, typically determined by the experimental design or clinical variables. However, in many real-world applications, the group assignments of the samples may be unknown. In such cases, one could apply unsupervised learning methods, such as k-means clustering, variational autoencoders, or Dirichlet process mixture models, to uncover the underlying group structure.

Finally, the DDT prior is based on binary trees; that is, when divergence occurs, a branch splits into two subbranches. This structure makes computation for the latent branching process tractable while still allowing the model to capture heterogeneous structures in the observed data. Extensions to polytomies—trees where a single node gives rise to more than two children—go beyond the assumptions of the DDT and require different nonparametric priors, such as the Pitman–Yor Diffusion Tree [18].

**Author Contributions:** Conceptualization, H.Z.; methodology, S.H. and H.Z.; software, S.H.; writing—original draft preparation, H.Z.; writing—review and editing, H.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by National Science Foundation of the United States grant number 1762577.

**Data Availability Statement:** The GBM brain tumor images used in this paper were obtained from The Cancer Genome Atlas (TCGA) database (<https://www.cancerimagingarchive.net/>, accessed on 7 August 2025). Clinical and genomic variables were obtained from cBioPortal (<http://www.cbioportal.org/>, accessed on 7 August 2025) and linked to the images.

**Acknowledgments:** The authors thank Veerabhadran Baladandayuthapani and Karthik Bharath for helpful discussions related to the brain tumor dataset.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Zhang, J.; Stevens, M.F.G.; Bradshaw, T.D. Temozolomide: Mechanisms of action, repair and resistance. *Curr. Mol. Pharmacol.* **2012**, *5*, 102–114. [[CrossRef](#)] [[PubMed](#)]
2. Just, N. Improving tumour heterogeneity MRI assessment with histograms. *Br. J. Cancer* **2014**, *111*, 2205–2213. [[CrossRef](#)]
3. Sachdeva, J.; Kumar, V.; Gupta, I.; Khandelwal, N.; Ahuja, C.K. A novel content-based active contour model for brain tumor segmentation. *Magn. Reson. Imaging* **2012**, *30*, 694–715. [[CrossRef](#)] [[PubMed](#)]
4. Zhou, M.; Hall, L.O.; Goldgof, D.B.; Gillies, R.J.; Gatenby, R.A. Survival time prediction of patients with glioblastoma multiforme tumors using spatial distance measurement. In Proceedings of the Medical Imaging 2013: Computer-Aided Diagnosis. International Society for Optics and Photonics, Lake Buena Vista, FL, USA, 12–14 February 2013; Volume 8670, p. 86702O.
5. Bharath, K.; Kurtek, S.; Rao, A.; Baladandayuthapani, V. Radiologic image-based statistical shape analysis of brain tumours. *J. R. Stat. Soc. Ser. C Appl. Stat.* **2018**, *67*, 1357–1378. [[CrossRef](#)] [[PubMed](#)]
6. Yang, H.; Baladandayuthapani, V.; Rao, A.U.K.; Morris, J.S. Quantile function on scalar regression analysis for distributional data. *J. Am. Stat. Assoc.* **2020**, *115*, 90–106. [[CrossRef](#)] [[PubMed](#)]
7. Poursaeed, R.; Mohammadzadeh, M.; Safaei, A.A. Survival prediction of glioblastoma patients using machine learning and deep learning: A systematic review. *BMC Cancer* **2024**, *24*, 1581. [[CrossRef](#)] [[PubMed](#)]
8. Felsenstein, J. Statistical inference of phylogenies. *J. R. Stat. Soc. Ser. A (General)* **1983**, *146*, 246–262. [[CrossRef](#)]
9. Teh, Y.W.; Daume, H., III; Roy, D.M. Bayesian agglomerative clustering with coalescents. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 3–6 December 2007; pp. 1473–1480.
10. Hu, Y.; Ying, J.L.; Daume, H., III; Ying, Z.I. Binary to bushy: Bayesian hierarchical clustering with the Beta coalescent. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 1079–1087.
11. Yang, Z.; Rannala, B. Bayesian phylogenetic inference using DNA sequences: A Markov Chain Monte Carlo method. *Mol. Biol. Evol.* **1997**, *14*, 717–724. [[CrossRef](#)] [[PubMed](#)]

12. Mau, B.; Newton, M.A.; Larget, B. Bayesian phylogenetic inference via Markov chain Monte Carlo methods. *Biometrics* **1999**, *55*, 1–12. [[CrossRef](#)] [[PubMed](#)]
13. Huelsenbeck, J.P.; Ronquist, F. MRBAYES: Bayesian inference of phylogenetic trees. *Bioinformatics* **2001**, *17*, 754–755. [[CrossRef](#)] [[PubMed](#)]
14. Chipman, H.A.; George, E.I.; McCulloch, R.E. Bayesian CART model search. *J. Am. Stat. Assoc.* **1998**, *93*, 935–948. [[CrossRef](#)]
15. Gramacy, R.B.; Lee, H.K.H. Bayesian treed Gaussian process models with an application to computer modeling. *J. Am. Stat. Assoc.* **2008**, *103*, 1119–1130. [[CrossRef](#)]
16. Aldous, D. Probability distributions on cladograms. In *Random Discrete Structures*; Springer: Berlin/Heidelberg, Germany, 1996; pp. 1–18.
17. Neal, R.M. Density modeling and clustering using Dirichlet diffusion trees. *Bayesian Stat.* **2003**, *9*, 619–629.
18. Knowles, D.A.; Ghahramani, Z. Pitman-Yor diffusion trees. *arXiv* **2011**, arXiv:1106.2494. [[CrossRef](#)]
19. Ghahramani, Z.; Jordan, M.I.; Adams, R.P. Tree-structured stick breaking for hierarchical data. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 6–11 December 2010; pp. 19–27.
20. Vikram, S.; Dasgupta, S. Interactive bayesian hierarchical clustering. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 2081–2090.
21. Kingman, J.F.C. The coalescent. *Stoch. Process. Their Appl.* **1982**, *13*, 235–248. [[CrossRef](#)]
22. Neal, R.M. *Defining Priors for Distributions Using Dirichlet Diffusion Trees*; Technical Report; University of Toronto: Toronto, ON, Canada, 2001.
23. Knowles, D.A.; Van Gael, J.; Ghahramani, Z. Message Passing Algorithms for the Dirichlet Diffusion Tree. In Proceedings of the International Conference on Machine Learning, Bellevue, WA, USA, 28 June–2 July 2011; pp. 721–728.
24. Schuhmacher, C.; Meier, J.S.; von der Heide, B.E. *Transport: Computation of Optimal Transport Plans and Wasserstein Distances*, R package version 0.13-2; R Foundation for Statistical Computing: Vienna, Austria, 2022.
25. Bharath, K.; Kambadur, P.; Dey, D.K.; Rao, A.; Baladandayuthapani, V. Statistical tests for large tree-structured data. *J. Am. Stat. Assoc.* **2017**, *112*, 1733–1743. [[CrossRef](#)] [[PubMed](#)]
26. Geyer, C.J.; Thompson, E.A. Annealing Markov chain Monte Carlo with applications to ancestral inference. *J. Am. Stat. Assoc.* **1995**, *90*, 909–920. [[CrossRef](#)]
27. Zhang, C.; Matsen, F.A., IV. A Variational Approach to Bayesian Phylogenetic Inference. *J. Mach. Learn. Res.* **2024**, *25*, 1–56.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.