

Article

BSTZINB: A Bayesian Framework for Negative-Binomial Modeling of Spatio-Temporal Zero-Inflated Count Data in Epidemiology

Suman Majumder ¹ , Yoonbae Jun ² , Sounak Chakraborty ^{3,*} , Chae Young Lim ⁴ and Tanujit Dey ^{5,*}

¹ Interdisciplinary Statistical Research Unit, Indian Statistical Institute, Kolkata 700108, India; smajumder@isical.ac.in

² School of Public Health, University of Nevada, Reno, NV 89557, USA; yoonbaej@unr.edu

³ Department of Statistics and Data Science, University of Missouri, Columbia, MO 65211, USA

⁴ Department of Statistics, Seoul National University, Seoul 03080, Republic of Korea; twinwood@snu.ac.kr

⁵ Center for Surgery and Public Health, Department of Surgery, Brigham and Women's Hospital, Harvard Medical School, Boston, MA 02145, USA

* Correspondence: chakrabortys@missouri.edu (S.C.); tdey@bwh.harvard.edu (T.D.)

Abstract

Modern Bayesian hierarchical methodologies allow us to leverage spatio-temporal dependencies between observations, enhancing both health effect estimation and map visualization in efficient and flexible ways. However, the necessary levels of statistical software are often unavailable or difficult to access. We have recently examined Bayesian spatio-temporal models to estimate the association between COVID-19 death counts and various social and environmental risk factors, including ambient air pollution exposure. Typically, it is very common that in an infection disease mapping problem with count data, we have excessive zeros, and it is usually for over-dispersed count outcome variables. Furthermore, the theory suggests that the excess zeros are generated by a separate process from the count values and that the excess zeros need to be modeled independently. Our proposed models are specially designed to handle the zero-inflation and over-dispersion in count data through Zero-Inflated Negative Binomial regression with random effects that vary across time and space within a Markov Chain Monte Carlo framework. Drawing on our knowledge and experience, we aim to provide a simple, unified, and publicly available software that can be applied in various disease mapping studies under the contemporary Bayesian framework.

Keywords: epidemiology; spatio-temporal; incidence-count data; zero-inflation; Markov Chain Monte Carlo; R package; Bayesian inference



Academic Editors: Bogdan Oancea, Adrian Pană and Cătălina Liliana Andrei

Received: 6 May 2026

Revised: 30 June 2026

Accepted: 14 July 2026

Published: 20 July 2026

Copyright: © 2026 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

This paper introduces a new R package, BSTZINB (available through CRAN <https://cran.r-project.org/web/packages/BSTZINB/index.html>, accessed on 13 July 2026), for statistical modeling and inference with various Bayesian spatio-temporal models specially designed for zero-inflated count data [1]. We are strongly motivated by recent catastrophic world events like the COVID-19 pandemic. In any infectious disease, the number of infected cases and related deaths vary spatially and temporally as the disease spreads over time and adjacent regions. In addition, the spread of the disease and the related degree of severity of the health outcomes can be heavily affected by several biological, socioeconomic, and environmental factors [2–4]. For example, in a recently published study

on modeling the COVID-19 pandemic, we found evidence that long-term exposure to air pollution might have increased the severity of COVID-19 weekly death counts in the early period of the pandemic in the United States [1].

The problem of zero-inflation arises when the large numbers of zeros in count data cannot be accommodated properly by a standard discrete distribution like the Poisson or Negative Binomial distribution [5–7]. When count data contain an unusually large proportion of zero observations, often referred to as “excess zeros,” standard Poisson and even conventional Negative Binomial models may inadequately capture the observed frequency of zeros, resulting in poor model fit and biased inference [5,8]. In addition, the non-zero observations frequently exhibit over-dispersion relative to the Poisson distribution, which can lead to biased parameter estimates, underestimated standard errors, and inflated Type I error rates when ignored [9,10]. In count-data applications, over-dispersion occurs when the conditional variance exceeds the conditional mean, thereby violating the equidispersion assumption of the Poisson model [9,10].

Bayesian modeling has emerged as a very popular tool for analyzing spatio-temporal data due to its flexibility in incorporating prior information, modeling complex dependency structures, and handling zero-inflation and over-dispersion cases, which are common in epidemiological studies [11–14]. There is an enormous need, especially among epidemiologists, for simple, unified, and publicly available software for such modern epidemiological research [11,14,15]. In that respect, [16] provides a related Bayesian zero-inflated Negative Binomial framework for spatio-temporal count data, which is methodologically relevant to the present manuscript. However, their [16] software is not distributed as a standard R package through CRAN and therefore does not provide the same level of direct accessibility, installation, documentation, and integration into routine R workflows.

Bayesian hierarchical models have become increasingly popular in epidemiological research because they can simultaneously accommodate over-dispersion, excess zeros, spatial correlation, temporal dependence, and uncertainty quantification within a unified inferential framework [1,17]. Such models have been successfully used to study disease incidence, mortality, and environmental health outcomes observed across geographic regions and over time. Our BSTZINB package translates these methodological advances into an accessible software tool, enabling researchers to fit a broad class of Bayesian count-data models without requiring extensive expertise in Bayesian computation. Specifically, the package implements Bayesian Negative Binomial (BNB), Bayesian Zero-Inflated Negative Binomial (BZINB), Bayesian Spatio-Temporal Negative Binomial (BSTNB), and Bayesian Spatio-Temporal Zero-Inflated Negative Binomial (BSTZINB) models using efficient Markov Chain Monte Carlo (MCMC) algorithms, while also providing tools for model assessment, visualization, disease mapping, and statistical reporting.

The BSTZINB package provides a suite of tools for data description, Bayesian modeling, and result summaries for spatio-temporal health outcomes. It is developed with the understanding that the end-user may not be an expert in Bayesian statistics or spatio-temporal modeling. Therefore, special care is provided for the package to communicate the model-based “knowledge discovery” to the end-user. In summary, this package offers a broad range of models, including (a) BNB—a Bayesian Negative Binomial model; (b) BZINB—a Bayesian Zero-Inflated Negative Binomial model; (c) BSTNB—a Bayesian Spatiotemporal Negative Binomial model; and (d) BSTZINB—a Bayesian Spatiotemporal Zero-Inflated Negative Binomial model. To our knowledge, no other package offers similar user-friendly Bayesian modeling for zero-inflated spatio-temporal epidemiology data like our BSTZINB package, complete with both inferential and visualization tools. In our BSTZINB package, we have carefully optimized the MCMC procedure to enhance speed and ensure efficient, tractable inference for all models within a reasonable computation time. This optimization

supports practical implementation in real data analysis settings, especially for general practitioners who may not be experts in Bayesian MCMC-based model fitting methods.

The rest of the paper is organized as follows: in Section 2, we provide a brief background knowledge on different types of count data, zero-inflated, and over-dispersed models. In Section 3, we describe the general application of our software and how an end-user can effortlessly use our package. Section 4 provides a walk-through example, including each step from data input to inference. It covers the entire process: starting with data input, then setting up (prior distribution and MCMC parameter choices), and running the model fitting. It continues with post-processing of the model outputs, diagnostics, and inference of the model or its parameters. Section 5 concludes the paper with a general helpful discussion of our findings and suggestions for possible future extensions. In the Supplementary Materials document we have provided additional simulation studies, benchmarking with other competing methods/packages, and MCMC convergence diagnostics and studies.

2. Methods

Depending on a spatio-temporal dependency and a zero-inflated nature of data, we can construct a Bayesian spatio-temporal model that assumes either standard or spatio-temporal (ST), with or without a Zero-Inflated (ZI) model component. We now provide a generic model structure that covers nested sub-models in the following illustration.

Suppose that we observe $k = 1, \dots, n$ samples, and each of them is assigned to the i th region (e.g., county) at time j . Let us consider the following data model equation [17]:

$$y_{ij} \sim (1 - q_{ij}) \mathbb{I}_{(w_{ij}=0 \wedge y_{ij}=0)} + q_{ij} \text{NB}(p_{ij}, r) \mathbb{I}_{(w_{ij}=1)}, \quad (1)$$

where y represents a disease count, and q represents the probability at risk of exposure that follows a Negative Binomial risk distribution, denoted as NB. The probability of contracting the disease among the individuals in the exposed population is given by p and the mean-variance dispersion parameter is given by r . A random variable w is a binary label with the disease at-risk probability q , and $\wedge$ is a symbol of logical conjunction. $\mathbb{I}_{(\cdot)}$ is a zero-one indicator made from the condition in the subscript condition. Now, we introduce the following model structures according to the way of modeling p and q . Note that in Equation (1), k indexes the observation-level record, while (i, j) indexes the corresponding spatial unit and time point.

2.1. Bayesian Zero-Inflated Negative Binomial Model (BZINB)

If an outcome is independent of both space and time, then we do not need to specify the space-time structure of the equation, so that the generic model (1) can be simplified into $y_k \sim (1 - q_k) \mathbb{I}_{(w_k=0 \wedge y_k=0)} + q_k \text{NB}(p_k, r) \mathbb{I}_{(w_k=1)}$. Consequently, we model q_k and p_k as,

$$\text{logit}(q_k) = \beta_{10} + \sum_{d=1}^{D_1} \beta_{1d} X_{kd}, \quad (2)$$

$$\text{logit}(p_k) = \beta_{20} + \sum_{d=1}^{D_2} \beta_{2d} X_{kd}, \quad (3)$$

where β_{1d} represents the coefficient of the fixed effect covariates $(\beta_{11}, \dots, \beta_{1D_1})$ as well as an intercept (β_{10}) in the q -model component for probability-at-risk, while β_{2d} represents the same but in the p -model component for at-risk population. Here, D_1 denotes the number of covariates included in the at-risk or zero-inflation component, while D_2 denotes the number of covariates included in the Negative Binomial count component. Note that the

two sets of covariates may be identical, partially overlapping, or different depending on the scientific question. If the same covariates are used in both components, the model allows the same predictors to have potentially different effects on the probability of being at risk and on the count intensity among at-risk observations. If different covariates are used, the user can model separate mechanisms for structural zeros and positive counts.

2.2. Bayesian Spatio-Temporal Negative Binomial Model (BSTNB)

If we do not consider excessive zero counts, then we only need to specify a model for p_{ij} , since $q_{ij} \equiv 1$ in Equation (1). Thus, we assume $y_{ij} \sim \text{NB}(p_{ij}, r)$ and consider

$$\text{logit}(p_{ij}) = \beta_0 + \sum_{d=1}^D \beta_d X_d + \sum_{m=1}^M \beta_{(D+m)} T_m + b_{i1} + b_{i2} t_{ij}, \quad (4)$$

where this model further requires the definition of a basis function T and a random coefficient b . Given the number of basis functions, m , $T_1, \dots, T_m$ are determined by principles of cubic B-spline basis functions [18]. This is used to capture the non-linear fixed time effect. b_{i1} refers to a random intercept within the i th region, while b_{i2} refers to a random slope along the time points t_{ij} within the i th region. These are used to capture random effects of space and time, which might interact with each other. The cubic B-spline basis provides sufficient flexibility to capture nonlinear temporal patterns, such as peaks, plateaus, and changing rates of increase or decrease over time. Consequently, the spline representation allows the model to describe complex temporal dynamics commonly observed in epidemiological count data while maintaining smoothness and computational stability. The random effects b_{i1} and b_{i2} further allow each spatial region to have its own intercept and temporal trajectory around this overall population-level trend.

2.3. Bayesian Spatio-Temporal Zero-Inflated Negative Binomial Model (BSTZINB)

Following Equation (1), we model q and p as,

$$\text{logit}(q_{ij}) = \beta_{10} + \sum_{d=1}^{D_1} \beta_{1d} X_d + \sum_{m=1}^M \beta_{1(D_1+m)} T_m + b_{1i1} + b_{1i2} t_{ij}, \quad (5)$$

$$\text{logit}(p_{ij}) = \beta_{20} + \sum_{d=1}^{D_2} \beta_{2d} X_d + \sum_{m=1}^M \beta_{2(D_2+m)} T_m + b_{2i1} + b_{2i2} t_{ij}, \quad (6)$$

$$(7)$$

where $\mathbf{b}_{1i} = (b_{1i1}, b_{1i2})^T$ and $\mathbf{b}_{2i} = (b_{2i1}, b_{2i2})^T$ represent the random effects corresponding to q_{ij} and p_{ij} , respectively. We impose a multivariate ICAR prior structure [1] on $\boldsymbol{\phi}_i = (\mathbf{b}_{1i}^T, \mathbf{b}_{2i}^T)^T$ as,

$$\boldsymbol{\phi}_i \mid \boldsymbol{\phi}_{(-i)}, \boldsymbol{\Gamma} \sim N_4 \left(\frac{1}{m_i} \sum_{l \in \partial_i} \boldsymbol{\phi}_l, \frac{1}{m_i} \boldsymbol{\Gamma} \right), \quad (8)$$

where $\boldsymbol{\Gamma}/m_i$ is a 4×4 unknown conditional covariance matrix and is estimated through the MCMC algorithm. This multivariate ICAR allows spatial dependence and region-specific random time trends within count components and excess-zero components as well as dependence between them. This model will be referred to as BSTZINB-NLT in the rest of the paper, where NLT refers to the non-linear fixed time trend for q .

We hypothesize that the assumption of a non-linear fixed time trend for q might not be necessary in some cases due to model parsimony. Thus, we further assume a linear

time-fixed trend for q , i.e., $\beta_{1(D_1+1)} = \beta_{1(D_1+2)} = \dots = \beta_{1(D_1+M)} = 0$. In other words, we consider

$$\text{logit}(q_{ij}) = \beta_{10} + \sum_{d=1}^{D_1} \beta_{1d} Z_d + b_{1i1} + b_{1i2} t_{ij}. \quad (9)$$

This model will be referred to as BSTZINB-LT in the rest of the paper, where LT refers to a linear fixed time trend for q .

Conditional posterior sampling algorithms are mostly adopted from [1], and it is straightforward to implement our algorithms to all the illustrated approaches. For instance, we draw an at-risk indicator w_{ij} from a Bernoulli distribution with $\eta_{ij} = \frac{q_{ij}^r(1-p_{ij})}{q_{ij}^r(1-p_{ij})+p_{ij}}$, where q_{ij} and p_{ij} were defined in Section 2.3 for zero-inflated models, while we always draw $w = 1$ for every iteration in practice for the non Zero-Inflated models. To update β_1 , we draw a latent variable ξ_{1ij} from a Pólya-Gamma distribution $\text{PG}(1, \eta_{1ij})$, as shown in [19]. Given w and ξ_1 , the full conditional distribution of β_1 is

$$\Pr(\beta_1 | w, \xi_1) \propto \pi(\beta_1) \exp \left[-\frac{1}{2} (z_1 - X\beta_1)^T \Omega_1 (z_1 - X\beta_1) \right], \quad (10)$$

where X is a $n \times p$ design matrix, $\pi(\beta_1)$ is the prior distribution $N_p(\beta_0, \Sigma_0)$, $z_1 = (w - 1/2)/\xi_1$, and Ω_1 is a $n \times n$ precision matrix, $\text{diag}(\xi_1)$. Therefore, we update β_1 from $N_p(\mu, \Sigma)$ where $\Sigma = (\Sigma_0^{-1} + X^T \Omega_1 X)^{-1}$ and $\mu = \Sigma (\Sigma_0^{-1} \beta_0 + X^T \Omega_1 z_1)$ with given z_1 . We update β_2 by a similar process using the corresponding Pólya-Gamma distribution $\text{PG}(y_{ij} + r, \eta_{2ij})$ as discussed in [1,17,20].

3. Software Description

To use our BSTZINB R-package (≥ 3.5) we need datasets that include a response variable as a count variable of interest (e.g., disease counts), covariates (e.g., risk factors), spatial coordinates, and time points. Also, spatial map information, such as the shape of the areal units and the adjacency information among the areal units, is necessary.

For the descriptive analysis (Part 1) Figure 1, the BSTZINB R-package reshapes original data for model fitting and constructs the adjacency matrix to account for spatial dependency. After that, it can provide conventional summary statistics, including histograms and informative statistics, such as Moran's I [21,22]. Moran's I is used as an exploratory diagnostic to assess whether neighboring spatial units tend to have similar outcome values. A significant positive Moran's I provides empirical motivation for including spatially structured random effects in the Bayesian model. We can investigate whether the response variable has either significant spatio-temporal dependency or a high proportion of zero rates so that our spatio-temporal zero-inflated hierarchical modeling may improve the model fitting compared to the conventional ones.

For the primary modeling (Part 2) Figure 1, the BSTZINB R-package provides Bayesian Negative Binomial regression model solutions: BNB, BZINB, BSTNB, and BSTZINB. In Table 1 we provide a comprehensive list of all available functions in our BSTZINB R-package and their short description.

In our BSTZINB package, the four Bayesian Negative-Binomial regression models (BNB, BZINB, BSTNB, and BSTZINB) require input variables such as a spatio-temporally indexed response variable (y), covariates (X), and a spatial-unit adjacency matrix (A). Our package allows for the choice of a linear/nonlinear temporal structure for the at-risk probability modeling (LinearT), and the various Bayesian MCMC parameter settings, such as the number of MCMC chains (nchain), the number of iterations (niter), the number of burn-in samples (nburn), and the number of thinning samples (nthin) [23].

Table 1. Summary of functions in the BSTZINB package.

Function	Descriptions
BNB	Fit a Bayesian Negative Binomial model
BZINB	Fit a Bayesian Zero-Inflated Negative Binomial model
BSTNB	Fit a Bayesian Spatio-Temporal Negative Binomial model
BSTZINB	Fit a Bayesian Spatio-Temporal Zero Inflated Negative Binomial model
USDmapCount	Produce a disease map over the counties in the US
get_adj_mat	Provide the adjacency matrix for the study counties
qRankPar	Produce a descending order of bar plot for time-averaged log-q estimates over quantile-representative counties
qRankParTop	Produce a descending order of bar plot for time-averaged log-q estimates over the top ranking counties
TimetrendCurve	Produce a time-trend curve over the study time domain for a county
ResultTableSummary	Produce a concise summary table for a model fit
ResultTableSummary2	Produce a full result summary table for all the model-fit objects
compute_ZINB_DIC	Compute a DIC for a zero-inflated Negative Binomial model-fit object
compute_NB_DIC	Compute a DIC for a standard Negative Binomial model-fit object
conv.test	Provide test results of MCMC convergence for specific model-fit object

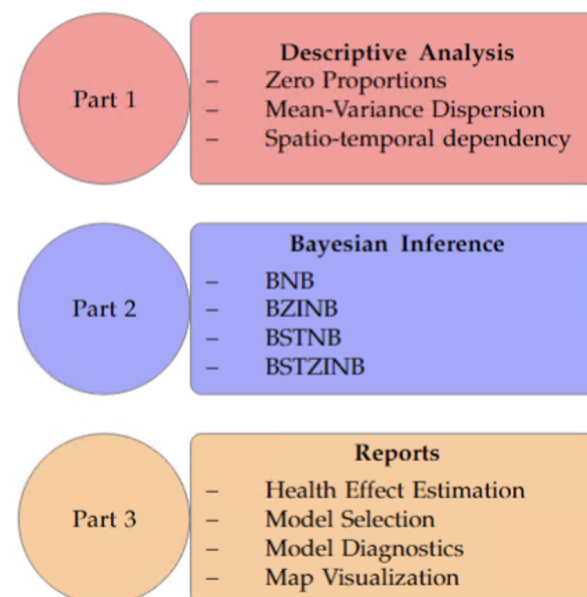


Figure 1. A flow chart of the BSTZINB R-package: Part 1. Descriptive analysis—we explore the data to determine the most effective statistical methods; Part 2. Bayesian inference—we give a solution to apply several Bayesian models such as Bayesian Zero-Inflated Negative Binomial (BZINB), Bayesian Spatio-Temporal Negative Binomial (BSTNB), and Bayesian Spatio-Temporal Negative Zero-Inflated Negative Binomial (BSTZINB); Part 3. Reports—we provide various solutions to health effect estimation, model selection, model diagnostics, and map visualizations for an epidemiological study.

The BSTZINB model outcome provides MCMC samples of coefficients for the fixed effect covariates in the structural zero model component (α) and in the negative-binomial model component (β). It also includes MCMC samples of the dispersion parameter. These MCMC samples are obtained by a random-walk Metropolis–Hastings update [17] (R) and by two-stage Gibbs update [24] (R2). The MCMC samples of the conditional covariance coefficients (Σ_{ϕ}) and country-specific random effects ($\phi_1, \phi_2, \phi_3, \phi_4$), the MCMC samples for logit of the estimated probability at risk (η_1), logit of the Negative Binomial probability (η_2) and at-risk indicators (I) are also provided as model outputs for downstream statistical inference and estimation. BSTNB and BNB are similar functions but do not consider zero-inflated modeling. It requires the same input variables as BSTZINB,

and there are no outcomes for the structural zero model component, such as α , PHI1 , PHI2 , Eta2 , by definition. Appendix A includes brief backgrounds of Bayesian Inference and methodological illustrations for interested readers.

For the model outputs (Part 3), our package provides various model-fitting results and plots focusing on spatial and temporal effect estimation within the Bayesian MCMC regression framework. Appendix B illustrates specific arguments and output of this package's main (Appendix B.2) and auxiliary functions (Appendix B.3), as well as imports and suggests of libraries (Appendix B.4). In the next section, we present one vignette with a simulated data set.

4. An Illustrative Example

In this section, we provide a step-by-step guide on how to use our BSTZINB R package with the help of a synthetic data set.

4.1. Installing BSTZINB Package

The BSTZINB package is available on CRAN and is ready to be installed and used by the user. The easiest way to install the package is running the following command in the R Console:

```
> install.packages('BSTZINB')
```

Once the BSTZINB package is installed, it can be loaded like any other R packages:

```
> library(BSTZINB)
```

After the library BSTZINB is loaded, all warning messages as listed above can be ignored as they have no bearing in running our models.

4.2. A Toy Data Set for Step-by-Step Demonstration

We created a synthetic data set that resembles a typical zero-inflated count data over space and time that we often see in many epidemiological studies.

```
> data(county.adjacency)
> data(USAcities)

> IAcities <- USAcities %>% filter(state_id=="IA")
> countyname <- unique(IAcities$county_name)
> A <- get_adj_mat(county.adjacency, countyname, c("IA"))
> m <- apply(A, 1, sum)
```

The BSTZINB package utilizes user-supplied geospatial map information and creates toy adjacency information of the areal units in this simulated study domain. In this example, we use the counties from the state of Iowa in the United States to create a spatial adjacency matrix A . We import the county name and the adjacency information from the US census data made in 2010 (available in the package as `county.adjacency` and `USAcities`; See Appendix B.1), and one of our functions, `get_adj_mat`, transform that information into an adjacency matrix A for the counties of our interest. Thereafter, we consider 16 evenly spaced time points across 99 spatial units, which correspond to all the counties in the state of Iowa. Consequently, the complete dataset has 1584 observations with four variables, including county index(`sid`), time index(`tid`), outcome disease counts (`y`), and a covariate (`x`).

```

## Spatial effects
> library(spam)
> library(mvtnorm)
#Spatial dependence parameter ~ 1 for intrinsic CAR
> kappa<-.99

# Conditional Cov of phi1 and phi2 given phi_{-i}
> cov <- matrix(c(.5,.10,.10,-.10,
                  .10,.15,.10,.10,
                  .10,.10,.5,.10,
                  -.10,.10,.10,.15),4,4)
> Q <- as.spam(diag(m)-kappa*A)
> covphi <- solve(Q)%x%cov # 4n x 4n covariance ofphis
> phi <- rmvnorm(1,sigma=covphi) # 4n vector of spatial effects
> phitmp <- matrix(phi,ncol=4,byrow=T) # n x 4 matrix of spatial effects

> true.phi1 <- phi1 <- phitmp[,1]-mean(phitmp[,1]) # n x 1 phi1 vector --
  ↪ Centered
> true.phi2 <- phi2 <- phitmp[,2]-mean(phitmp[,2]) # n x 1 phi2 vector,
  ↪ etc.
> true.phi3 <- phi3 <- phitmp[,3]-mean(phitmp[,3])
> true.phi4 <- phi4 <- phitmp[,4]-mean(phitmp[,4])
> Phi1 <- rep(phi1,nis)
> Phi2 <- rep(phi2,nis)
> Phi3 <- rep(phi3,nis)
> Phi4 <- rep(phi4,nis)
> true.phi <- cbind(true.phi1,true.phi2,true.phi3,true.phi4)

```

For the further data-generating process of our simulation, we set an intrinsic conditional autoregressive (ICAR) structure with a spatial dependence parameter $\sigma_b^2 = 1$ and a pre-specified conditional covariance matrix for $\Phi_1, \Phi_2, \Phi_3, \Phi_4$, defined in Section 2.3. This is set to check whether our model can properly adjust for the spatial dependence of the outcome.

```

## Fixed Effects and Covariates
> library(boot)
> tpop <- USACities %>% filter(state_id=="IA") %>% group_by(county_fips)
  ↪ %>% summarise(tpop=population) %>% .[,2] %>% unlist %>% as.numeric
> x <- rep(log(tpop),nis)
> x <- (x-mean(x))/sd(x)
> # set.seed(2023); x <- rnorm(N)
> t <- tid / max(tid)
> X <- cbind(1,x) # Design matrix, can add additional covariates (e.g.,
  ↪ race, age)
> colnames(X) <- c("Intercept","x")
> Xtilde <- cbind(1,x,sin(t))
> p <- ncol(Xtilde)

# Binomial part

```

```

> true.alpha <- rep(0.1,p)
> true.alpha[1:2] <- c(-0.25,0.25)
> # true.alpha <- c(-0.25,0.25,-0.5,0.25)
> eta1 <- Xtilde%*%true.alpha+Phi1+Phi2*t
> pii <- inv.logit(eta1) # 1-pr("structural zero")
> u <- rbinom(N,1,pii) # at-risk indicator
> N1 <- sum(u)
> pstruct0 <- 1-mean(u) # Proportion of structural zeros

> # Count Part
> true.beta <- rep(0.1,p)
> true.beta[1:2] <- c(.5,-.25)
> tmp <- u
> tmp[u==0] <- 0 # If in structural group then set tmp to 0
> nis1 <- tapply(tmp,sid,sum) # Number of at risk observations in each
  ↪ county

> eta2 <- Xtilde[u==1,]%*%true.beta + Phi3[u==1] + Phi4[u==1]*t[u==1] #
  ↪ Linear predictor for count part
> true.r <- 1.25 # NB dispersion
> psi <- exp(eta2)/(1+exp(eta2)) # Prob of success
> mu <- true.r*psi/(1-psi) # NB mean

## Data Generation
> y <- rep(0,N) # Response
> y[u==1] <- rnbinom(N1,size=true.r,mu=mu) # If at risk, draw from NB
> pzero <- length(y[y==0])/N # Proportion of zeros

```

Following the data model Equation (1) and process model Equation (5), we specify the underlying at-risk-probability model as

$$\text{logit}(q_{ij}) = -0.25 + 0.25x_{ij} + 0.10\sin(t_{ij}) + \Phi_{1i} + \Phi_{2i}t,$$

for $i = 1, \dots, 99$ and $j = 1, \dots, 24$, and success-probability model as

$$\text{logit}(p_{ij}) = 0.5 - 0.25x_{ij} + 0.10\sin(t_{ij}) + \Phi_{3i} + \Phi_{4i}t,$$

for $i = 1, \dots, 99$ and $j = 1, \dots, 24$, where $\Phi_{1i}, \dots, \Phi_{4i}$ are mean-centered, random-generated quantities from the conditional covariance matrices $\Phi_1, \Phi_2, \Phi_3, \Phi_4$ based on Section 2.3. Finally, y_{ij} , i.e., an outcome count at the i th spatial unit and the j th time point, is assigned to either zero if $w_{ij} = 0$, or a realized value from the Negative Binomial distribution. We use both a mean parameter of an inverse logit of p_{ij} and a pre-specified dispersion parameter $r = 1.25$, which consider both regression modeling in the risk probability and over-dispersion in outcome counts. We then combine all variables to create our data frame:

```

# Dataframe
> this_dat <- data.frame(sid,tid,y,X)
> head(this_dat)
  sid tid y Intercept x
1 1 1 1 1 -0.9924003
2 1 2 0 1 -0.9924003

```

```
3 1 3 0 1 -0.9924003
4 1 4 3 1 -0.9924003
5 1 5 0 1 -0.9924003
6 1 6 0 1 -0.9924003
```

4.3. Descriptive Analysis

Table 2 shows a descriptive summary of data using the function `tbl_summary` from the `gtsummary` package. A continuous value is summarized by median and inter-quantile range (IQR), while a discrete value is summarized by counts and proportion.

```
> gtsummary::tbl_summary(this_dat) %>% modify_header(label = "**
  ↳ Coefficients**")
```

Table 2. A descriptive summary table for the generated synthetic data.

Coefficients	N = 1584 ¹
sid	50 (25, 75)
tid	9 (5, 13)
y	0.00 (0.00, 1.00)
Intercept	1584 (100%)
x	−0.20 (−0.68, 0.38)

¹ Median (IQR); n (%)

```
# Histogram
> tmp <- table(y)/n*100 # convert to percents
barplot(tmp, ylab="Percent", xlab="Count", col="lightblue")
```

This script shows a histogram of all outcome counts (Figure 2a) for checking the proportions of zero counts. We observe a high proportion of zero counts (66.7%), supporting our zero-inflated distributional assumptions.

```
# Moran's I
> library(ape)
> library(reshape)
> this_dat2 <- cast(this_dat, sid~tid, sum, value="y")
```

This script investigates the spatial aggregation of the disease counts over the counties. Moran's I is from the R-package `ape`. The estimated Moran's I statistics for the time-averaged disease counts are statistically significant because its p-value = 4.46×10^{-16} is lower than the 5% nominal Type I error. Also, Moran's I statistics calculated for each time index are all less than 5% nominal Type I error. These results support our spatial random effect modeling of the target outcome since we observe a spatial clustering tendency of the outcome, albeit the covariates are independent of spatial locations.

```
> Moran.I(rowMeans(this_dat2), A)
$observed
[1] 0.2704696

$expected
[1] -0.01020408
```

```
$sd
[1] 0.05190279

$p.value
[1] 6.384654e-08
```

```
> # US Disease Mapping Cumulative Death Counts
> USDmapCount(state.sel = c("IA"),
               dat = this_dat,
               scol = 1,
               tcol = 2,
               cname = countyname,
               uplim = 150)
```

USDmapCount in our package visualizes a map of disease characteristics in the United States. For example, the above script recognizes the Iowa state as a study domain, *this_dat* as input data, including spatial index in the column *scol* and time index in the column *tcol*. We can manually set the name of counties using the *cname* argument and scale of the legend color using *uplim*. If we further specify *tset*, we can select counts in a specific time point. Otherwise, counts are averaged over time. Figure 2b–f implies the spatio-temporal relation of the disease counts. Compared to the time-average case from the above statement (Figure 2b), all the time-specific cases (Figure 2c–f) show different patterns, which may indicate spatio-temporal dependence of the disease counts. This descriptive analysis implies that naive Negative Binomial modeling without incorporating spatio-temporal dependence may produce biased and/or inefficient estimates.

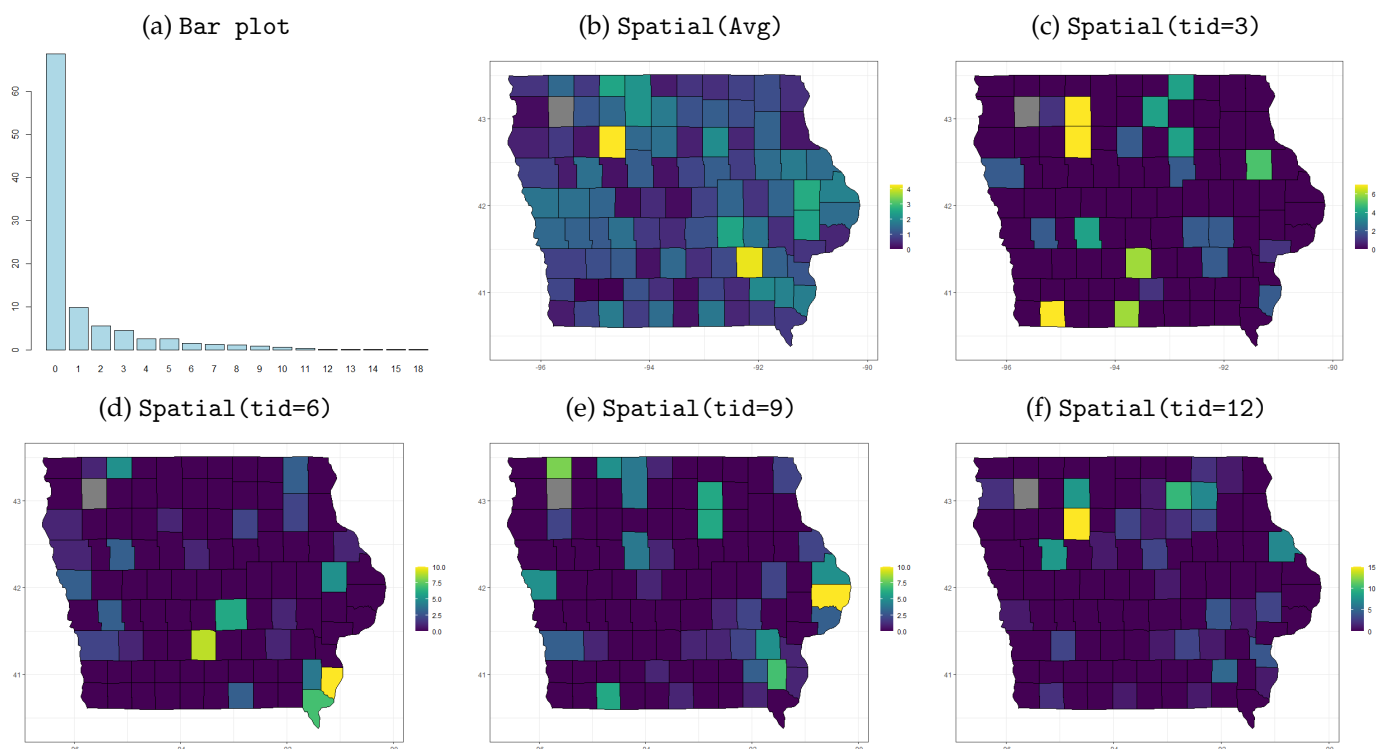


Figure 2. An example of descriptive summary of the outcome counts (*y*) for our simulation study. (a) Histogram of all outcome counts; (b) time-averaged spatial map visualization of outcome counts; (c–f) spatial map visualization of outcome counts at (c) *tid* = 3, (d) *tid* = 6, (e) *tid* = 9, (f) *tid* = 12.

4.4. Bayesian Spatio-Temporal Modeling

We implement the Bayesian regression modeling of the disease counts that allows not only over-dispersion but also a high proportion of zero counts and a spatio-temporal nature. Our package provides several Bayesian Negative Binomial models such as the BNB function for a standard model; the BZINB function for considering a zero-inflated model component (see Section 2.1); the BSTNB function for the considering the spatio-temporal random effect model component (see Section 2.2); and the BSTZINB function for the Bayesian spatio-temporal zero-inflated model (see Section 2.3). For instance, we apply the BSTZINB function to this example as follows :

```
> # Bayesian STZINB Model with Linear Time Trend
> res3 = BSTZINB(y, X, A, LinearT=TRUE, nchain=2, nsim=50, nburn=10,
  ↪ nthin=1)
> glimpse(res3)
List of 11
 $ Alpha : num [1:40, 1:3, 1:2] -0.622 -0.707 -0.607 -0.646 -0.574 ...
  ..- attr(*, "dimnames")=List of 3
  .. ..$ : NULL
  .. ..$ : chr [1:3] "" "x" "t"
  .. ..$ : NULL
 $ Beta : num [1:40, 1:3, 1:2] 1.006 0.79 0.982 0.988 1.14 ...
  ..- attr(*, "dimnames")=List of 3
  .. ..$ : NULL
  .. ..$ : chr [1:3] "" "x" "t"
  .. ..$ : NULL
 $ R : num [1:40, 1:2] 0.989 0.972 0.982 1 0.991 ...
 $ Sigphi: num [1:40, 1:16, 1:2] 0.516 0.523 0.517 0.63 0.507 ...
 $ PHI1 : num [1:40, 1:99, 1:2] -0.0277 0.217 -0.4054 -0.5354 -0.9193 ...
 $ PHI2 : num [1:40, 1:99, 1:2] -0.674 0.243 -0.134 0.243 0.208 ...
 $ PHI3 : num [1:40, 1:99, 1:2] -0.1815 -0.1597 -0.0146 -0.071 -0.0394 ...
  ↪
 $ PHI4 : num [1:40, 1:99, 1:2] 0.0314 -0.3003 0.0224 -0.2369 0.1086 ...
 $ Eta1 : num [1:40, 1:1584, 1:2] -0.69 -0.639 -1.2 -1.261 -1.631 ...
 $ Eta2 : num [1:40, 1:1584, 1:2] 1.38 1.05 0.89 1.21 1.21 ...
 $ I : num [1:40, 1:1584, 1:2] 1 1 1 1 1 1 1 1 1 1 ...
```

We have added an example for LinearT=FALSE in the online vignette that is available with the R package from CRAN: https://cran.r-project.org/web/packages/BSTZINB/vignettes/BSTZINB_vignette.html (accessed on 17/07/2026), look into the example `res4 <- BSTZINB(y, X, A, LinearT=FALSE, nchain=2, niter=50, nburn=10)`.

We illustrate the major interpretations and visualizations in the following subsections based on the above run model.

4.4.1. Main Results

ResultTableSummary transforms the BSTZINB object into a brief summary table, as shown in Figure 3. The first column of the table shows a list of regression coefficients, where combinations of an alphabet and a specified column name are used. The alphabet is either 'a' or 'b,' where 'a' refers to the structural zero model coefficients, and 'b' refers to the exposure model coefficients. If a column name is not specified, no letters will appear right after the period, but otherwise, the specified name will followed by the period.

For example, “b.x” means the coefficient of the covariate named “x” in the exposure model components. “a.” means the coefficient without any specified name in the structure-zero model components. On the other hand, the second column of the table shows relevant statistics. The first row shows the total number of observations, and the other rows provide point estimates and a 90% credible interval in parentheses.

BSTZINB Model Coefficients	
Coefficients	Estimates [†]
a.	-0.585 (-0.792, -0.346)
a.x	0.227 (0.155, 0.292)
a.t	0.718 (0.409, 1.019)
b.	0.896 (0.757, 1.037)
b.x	-0.273 (-0.346, -0.200)
b.t	-0.135 (-0.327, 0.032)
[†] Point estimates (90% credible intervals)	

Figure 3. ResultTableSummary produces a brief summary table of the BSTZINB object. Here ‘a’ refers to the structural zero model coefficients and ‘b’ refers to the exposure model coefficients. ‘a.’ refers to the intercept coefficient for the structural zero model, while ‘b.x’ refers to the coefficient related to covariate x in the exposure model.

```
> # Result Summary Table (simple)
> ResultTableSummary(res3)
> # Result Summary Table (Full)
> ResultTableSummary2(y, X, A, nchain=2, niter=2000, nburn=1000, nthin=1)
```

ResultTableSummary2 performs all Bayesian spatio-temporal model procedures available in the package, including BNB, BZINB, BSTNB, and BSTZINB, providing a full summary table, as shown in Figure A1. Note that the model coefficients of the time spline basis vectors are followed by those of the input covariates. For example, assuming the BSTZINB-NLT model (Section 2.3), $(a., a.x, a.t1, a.t2, a.t3, a.t4, a.t5, a.t6, a.t7)$ corresponds to $(\beta_{10}, \beta_{11}, \beta_{12}, \beta_{13}, \beta_{14}, \beta_{15}, \beta_{16}, \beta_{17})$ in Equation (5), where $D_1 = 1$, and $M = 6$. A red-colored text represents a failure to achieve MCMC convergence using Geweke statistic [25]. Also, a footnote provides the Deviance Information Criterion (DIC) as a model selection tool. The fourth model BSTZINB has the lowest DIC value with the given example data.

```
> # Bar Plot of vn-quantile Probability at Risk of disease
> qRankPar(state.set=c("IA"), cname=countyname, bstfit=res3, vn=12, cex.
  ↪ title=18, cex.lab=12, cex.legend=12)
```

qRankPar produces a bar plot of time-averaged estimated disease probabilities at risk, taking into account the quantiles and the descending order of the rankings among spatial units. For example, the argument $vn = 12$ decides the number of selected counties. That is, we choose the counties out of a given map considering $vn = 12$ quantiles of estimated disease probabilities from the BSTZINB object.

qRankParTop produces a bar plot for time-averaged estimated disease probabilities at risk for selected counties in descending order (Figure 4a). The options `cex.title`, `cex.lab` and `cex.legend` control the text size for the title, labels, and legend, respectively.

```
> # Line Plot of Nonlinear Time effect abundance (smoothed version)
> TimetrendCurve(res3, countyname, vn=5, smooth.mode=TRUE, cex.title=18,
  ↪ cex.lab=12, cex.legend=12)
```

TimetrendCurve produces the estimated nonlinear mixed time effects in the Negative Binomial model component (See Figure 3 in [1]).

The argument smooth.mode=T produces the cubic-spline interpolated nonlinear mixed time effects (Figure 4b). The options cex.title, cex.lab and cex.legend work as described previously.

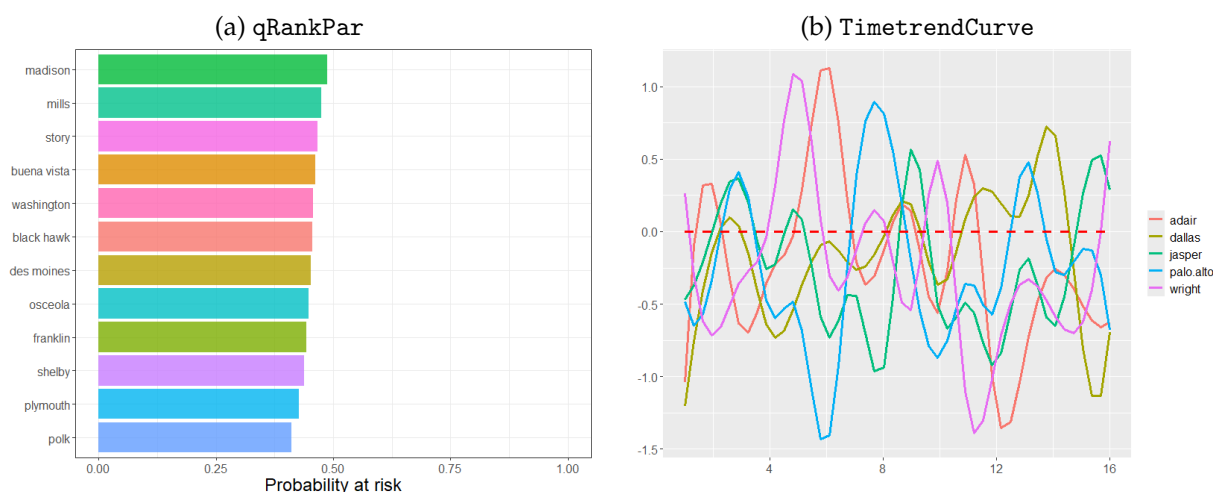


Figure 4. An illustrative example of eta-related results. (a) The left panel shows a result from the function **qRankPar**, producing a bar plot for time-averaged estimated disease probability at risk considering the rankings' quantiles and descending order. (b) The right panel shows a result from the function **TimetrendCurve**, producing the estimated nonlinear mixed time effects in the Negative Binomial model component.

4.4.2. Map Visualization

We describe several map visualizations based on the results above. The function **USDmapCount** visualizes the time-averaged disease counts over the counties in the United States. The x-axis represents longitude, the y-axis represents latitude, and the color indicates the level of disease counts. Missing values are shown in gray.

```
> # Map Visualization : Time-averaged Eta
> this_result_dat <- this_dat
> this_result_dat$y <- apply(res3$Eta1,2,mean)
> this_result_dat %>% head
> USDmapCount(state.sel = c("IA"),
  dat = this_result_dat,
  scol = 1,
  cname = countyname)
```

If a user sets a dat argument as η_1 from the BSTZINB output, then the outcome will provide a spatial visualization of the logit of the estimated probability at risk on average over time. Likewise, we can provide a map of the following characteristics from the model fitting (see Figure 5).

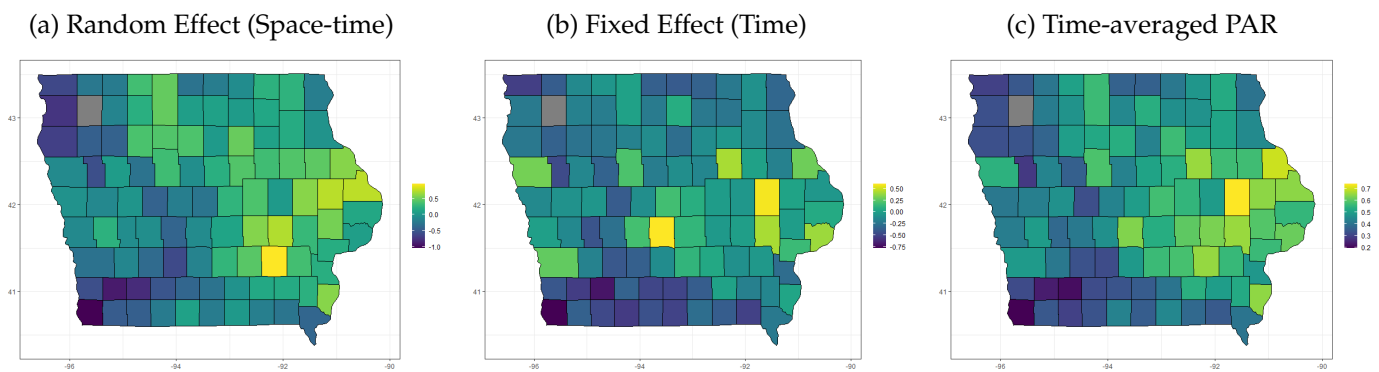


Figure 5. Map visualizations of various estimates regarding spatio-temporal aspects of health outcome. (a) The level of spatio-temporal random effect; (b) the level of temporal fixed effect; (c) time-averaged disease probability at risk (PAR).

4.4.3. MCMC Convergence and Diagnostics

The BNB, BZINB, BSTNB, and BSTZINB functions can provide console messages about work progress while running. This includes chains, iterations, and diagnostics in MCMC samplings. For example, we can see the following line on the R-console:

```
> res3 = BSTZINB(y, X, A, LinearT=TRUE, nchain=2, nsim=50, nburn=10,
  ↪ nthin=1)
```

This means that the first chain out of the two chains (specified from the argument `nchain`) is currently processing, 85% of the samples are completed to be generated, and those samples passed the Geweke convergence test. In particular, TRUE means pass, FALSE means fail, and NA means not applicable since the MCMC samplings are still in the burn-in period.

Our package also provides the Geweke convergence test after completing the Bayesian model fitting. For example, we can check MCMC convergence for each of the BSTZINB model parameters as follows:

```
> conv.test(res3$Alpha) ## Convergence test for Alpha parameters
[1] FALSE TRUE TRUE
> conv.test(res3$Beta) ## Convergence test for Beta parameters
[1] TRUE FALSE TRUE
> conv.test(res3$R) ## Convergence test for R parameters
[1] FALSE
> par(mfrow=c(1,2));plot(res3$R[,1],main="R");plot(res3$R[,2],main="R")
> par(mfrow=c(1,2));plot(res3$Alpha[,1,1],type='o',main="Alpha");plot(
  ↪ res3$Alpha[,1,2],type='o',main="Alpha")
```

Recall that `res3` is made from a Bayesian spatio-temporal zero-inflated Negative Binomial model with a linear time trend. This model has three alpha coefficients for the structural zero model component, three beta coefficients for the Negative Binomial model component, and one dispersion parameter. Our built-in function `conv.test` shows test results for each parameter sample. TRUE means statistical evidence on the convergence of the samples based on the Geweke test. In addition, you can generate trace plots to check stationarity and a degree of mixingness and visualize convergence behaviors of the MCMC samples [26].

```
> compute_ZINB_DIC(y,res3)
[1] 3972.34
```

`compute_NB_DIC` and `compute_ZINB_DIC` compute the Deviance Information Criterion specifically designed for the BSTNB and BSTZINB model outputs, respectively [27,28].

5. Discussions and Conclusions

The BSTZINB R-package tackles spatial and temporal analysis of health outcomes in epidemiology. It offers a three-phase framework: Phase 1 checks if your data suits the model by examining zero proportions and spatial/temporal dependencies. Phase 2 tackles data with zero inflation or non-standard dispersion using Bayesian inference. Finally, Phase 3 generates reports including health effect estimates, maps, and results from the model.

The BSTZINB R-package stands out for its focus on usability, making it accessible to researchers in epidemiology and beyond, even those without a deep dive into statistics. This practitioner-friendly design prioritizes ease of use, allowing researchers from any academic or industrial field to leverage powerful spatio-temporal methods without getting bogged down in complex mathematical details. The package comes equipped with clear, well-designed case studies that serve as practical guides for users to follow, ensuring they can confidently apply these methods to their own research questions.

The BSTZINB R-package goes beyond user-friendliness. It streamlines the workflow by unifying existing R tools under a coherent Bayesian framework, boosting efficiency and consistency in spatio-temporal modeling and inference. This powerhouse caters to the complexities of epidemiological data by incorporating zero-inflated count data and flexible random effects that can vary across space and time. Moreover, the freely available, open-source nature of the package makes it accessible to a broad range of researchers, fostering collaboration within the field. Looking ahead, addressing potential limitations like computational hurdles for massive datasets and the need for further development in advanced tools like R-Stan [29] will solidify BSTZINB's position as a premier tool for spatio-temporal analysis in epidemiology. It is important to note here that any Stan-based development would require careful reformulation, such as marginalizing over the discrete at-risk indicators where possible, using continuous relaxations, or embedding Stan within a larger MCMC scheme that alternates between updates of discrete latent indicators and continuous model parameters.

Despite its flexibility, BSTZINB inherits the computational limitations common to MCMC-based Bayesian spatio-temporal models. The current implementation is best suited for small- to moderate-scale areal datasets, where reliable posterior inference can be obtained within reasonable computation times. As the number of spatial units, temporal observations, and model parameters increases, computational cost grows accordingly. Future enhancements will focus on improving scalability, expanding prior flexibility, and incorporating more computationally efficient Bayesian inference strategies.

It is important to emphasize that the primary objective of the BSTZINB package is to provide Bayesian inference tools for spatio-temporal zero-inflated count outcomes commonly encountered in epidemiology, public health, and environmental studies. Consequently, survival-analysis frameworks based on lifetime distributions such as the exponential or Weibull distributions, as well as competing-risk models, fall outside the intended scope of the current package and correspond to fundamentally different classes of outcomes and inferential objectives. Rather than focusing on time-to-event data, BSTZINB is specifically designed for count data exhibiting excess zeros, over-dispersion, and spatio-temporal dependence. Nevertheless, the underlying Bayesian framework is sufficiently flexible that future extensions could incorporate broader classes of outcomes, including survival, recurrent-event, and competing-risk data, thereby expanding the applicability of the package to a wider range of biomedical and epidemiological research settings.

In closing, the BSTZINB R-package is a significant advancement in analyzing data that changes over space and time in epidemiology. Even researchers new to statistics can

easily use it. BSTZINB combines modern statistical methods with helpful visualization tools. This makes it a powerful and flexible option for analyzing such data in many fields, including epidemiology, biostatistics, and environmental science. While BSTZINB excels at making complex methods easier to use and understand, there's room for improvement. Future development can address handling very large datasets and include even more advanced models. These improvements will solidify BSTZINB's position as a leading tool for spatio-temporal analysis in epidemiology and beyond.

Supplementary Materials: The following supporting information (Supplementary Materials) can be downloaded at: <https://drive.google.com/file/d/19B0kc0AZSIHnYGLPUXPF6CjVhs28twrY/view> (accessed on 13 July 2026), additional simulation studies, benchmarking with other competing methods/packages, and MCMC convergence diagnostics and studies.

Author Contributions: Conceptualization, S.C. and T.D.; methodology, S.C., C.Y.L. and T.D.; software, S.M. and Y.J.; validation, S.M.; formal analysis, S.M. and Y.J.; data curation, S.M.; writing—original draft preparation, S.M., Y.J., S.C. and T.D.; writing—review and editing, S.M., Y.J., S.C., C.Y.L. and T.D.; visualization, S.M. and Y.J.; supervision, S.C., T.D. and C.Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declares no conflicts of interest.

Appendix A. Methodological Backgrounds

Appendix A.1. Bayesian Inference

We illustrate prior and hyper-parameter specifications and Markov Chain Monte Carlo (MCMC) settings under the STZINB model framework. For the latent at-risk indicators w_{ij} , probability was given as $\exp(\theta_{1ij}) / [1 + \exp(\theta_{1ij})]$, where θ_{1ij} is defined as either Equation (5) or (9). Prior distributions for β_1 and β_2 were assumed to be $N_p(\beta_0 = \mathbf{0}, \Sigma_0 = 100\mathcal{I}_p)$, respectively, where $\mathcal{I}_p$ is a $p \times p$ identity matrix. For r in Negative Binomial component, a uniform prior was considered. To construct time-basis functions $T_m, m = 1, \dots, M$, we standardized time points t_{ij} to be within the range of 0 to 1, i.e., $t_{ij} = \frac{j}{J}$, where $J = 23$ is the number of weeks during the study period, and $j = 1, \dots, J = 23$ is a week indicator. We set three internal knot points, 0.25, 0.50, and 0.75, considering 0 and 1 as boundaries, so that $M = 7$ throughout our analysis. DIC is calculated as described in [30] for model comparison.

For each model, we ran three MCMC chains with 11,000 iterations and 1000 burn-in. For each model, the convergence of each model was determined by conventional MCMC diagnostics such as trace plots and Geweke z-statistics.

Appendix A.2. Model Fitting Principles

We adopt a posterior sampling algorithm from Neelon et al. (2019) [17] and Chakraborty et al. (2022) [1], which can be extended straightforwardly to other simpler models. As outlined in Chakraborty et al. (2022) [1], we update several key parameters including at-risk indicators w , coefficients β_1 for the binary model component, coefficients β_2 for the count model component, a dispersion parameter r for the Negative Binomial distribution, and spatial random effects ϕ with Γ . The following steps describe the MCMC algorithm used to update these parameters.

- Update at-risk indicators w

We draw w_{ij} from a Bernoulli distribution with probability η_{ij} defined as:

$$\eta_{ij} = \frac{q_{ij}^r(1 - p_{ij})}{q_{ij}^r(1 - p_{ij}) + p_{ij}} \quad (\text{A1})$$

where q_{ij} and p_{ij} are functions illustrated in Section 2.3, given current parameter values. To avoid numerical issues, q_{ij} and p_{ij} are adjusted to be within (0.001, 0.999) in practice.

- Update $\beta = (\beta_1, \beta_2)$

We draw a latent variable ξ_{1ij} from a Pólya-Gamma distribution $\text{PG}(1, \theta_{1ij})$, and update β_1 using its full conditional distribution based on observed data and prior distributions. Updating β_2 follows a similar process using the Pólya-Gamma distribution $\text{PG}(y_{ij} + r, \theta_{2ij})$.

- Update r

We employ a random-walk Metropolis-Hastings step for updating r , as detailed in [17], with a uniform prior for computational efficiency.

- Update ϕ, Γ

We update ϕ , which consists of random effects vectors, from their normal full conditional distributions. The update involves applying sum-to-zero constraints where necessary. The spatial structure Γ is updated using an inverse-Wishart full conditional distribution (described in detail [1]).

Appendix A.3. Additional Tables and Figures

Coefficients	BNB [†]	BZINB [†]	BSTNB [†]	BSTZINB [†]
a.t				
a.		-0.182 (-0.213, -0.153)		-0.011 (-0.142, 0.127)
a.x		0.320 (0.283, 0.379)		0.302 (0.220, 0.371)
a.t1				-0.807 (-1.055, -0.475)
a.t2				-0.229 (-0.446, 0.029)
a.t3				-0.391 (-0.619, -0.180)
a.t4				0.006 (-0.079, 0.241)
a.t5				-0.563 (-0.799, -0.175)
a.t6				0.973 (0.743, 1.320)
a.t7				-0.107 (-0.339, 0.130)
b.t				
b.	0.253 (0.204, 0.324)	0.919 (0.866, 0.952)	0.080 (0.010, 0.123)	0.848 (0.770, 1.278)
b.x	-0.088 (-0.107, -0.077)	-0.285 (-0.312, -0.246)	-0.095 (-0.130, -0.076)	-0.231 (-0.271, -0.183)
b.t1				0.067 (-0.383, 0.315)
b.t2				0.066 (-0.081, 0.235)
b.t3				-0.452 (-0.749, -0.039)
b.t4				0.145 (0.023, 0.395)
b.t5				-0.552 (-1.032, -0.194)
b.t6				-0.027 (-0.185, 0.171)
b.t7				0.070 (-0.114, 0.232)

[†] Point estimates (90% credible intervals) DIC1 = 4468.5 ; DIC2 = 3973.5 ; DIC3 = 4284.5 ; DIC4 = 4162.1

Figure A1. ResultTableSummary2 produces a full summary table of the BSTZINB object. For the column “Coefficients”, “a.*” stands for structural zero model coefficients, and “b.*” stands for exposure model coefficients. “*.t” implies coefficients for time variables, and “*.x” implies coefficients

for covariates x 's, and "*" means an intercept. From ".t1" to ".t7" represents the corresponding spline basis functions applied in the BSTZINB models (when `LinearT = FALSE`). Each cell shows the point estimate and the 90% credible interval within the parenthesis. Red colors represent the poor convergence of the estimates. Other terminologies are defined as follows: BSTP—Bayesian Spatio-Temporal Poisson; BSTNB—Bayesian Spatio-Temporal Negative Binomial; BSTZINB(LT)—Bayesian Spatio-Temporal Zero Inflated Negative Binomial with linear temporal fixed effect on the structural zero model component; BSTZINB(NLT)—Bayesian Spatio-Temporal Zero Inflated Negative Binomial with a non-linear temporal fixed effect on the structural zero model component.

Appendix B. List of Function Arguments

Appendix B.1. Data

A data frame with 4 variables: **Countyname**—2010 state and county name; **fips.County**—2010 state and county FIPS codes; **neighborname**—2010 state and county name of each neighboring county or county equivalent; **fipsneighbor**—2010 state and county FIPS codes of each neighboring county or county equivalent (See this link from the census.gov: <https://www.census.gov/geographies/reference-files/2010/geo/county-adjacency.html#:~:text=The%202010%20county%20adjacency%20file%20is%20a%20national%2C%20tab%20delimited,the%20end%20of%20the%20list>, accessed on 13 July 2026).

```
> data(county.adjacency)
```

```
> data(USAcities)
```

A data frame with 17 variables: **city**—The name of the city/town; **city_ascii**—"city" as an ASCII string; **state_id**—the state or territory's USPS postal abbreviation; **state_name**—the name of the state or territory that contains the city/town; **county_fips**—the 5-digit FIPS code for the primary county. The first two digits correspond to the state's FIPS code; **county_name**—the name of the primary county (or equivalent) that contains the city/town; **lat**—the latitude of the city/town; **lng**—the longitude of the city/town; **population**—an estimate of the city's urban population; **density**—the estimated population per square kilometer; **source**—for some cities, our data is generated from a polygon representing the city, for others we simply have a point; **military**—TRUE if this place is a military establishment such as a fort or base; **incorporated**—TRUE if the place is a city/town. FALSE if the place is just a commonly known name for a populated area; **timezone**—the city's time zone in the tz database format (e.g., America/Los_Angeles); **ranking**—an integer from 1-5 that captures the importance of a city (1 is most important, 5 least important); **zips**—a string containing all five-digit zip codes in the city/town, delimited by a space. Learn more; **id**—a 10-digit unique id generated by SimpleMaps. It is consistent across releases and databases (e.g., World Cities Database) (see this link for US cities: <https://simplemaps.com/data/us-cities>, accessed on 13 July 2026).

Appendix B.2. Main Functions

BNB, BZINB, BSTNB, and BSTZINB are considered as main functions in our package. Here, we are only illustrating the BSTZINB function since the other functions are just a few variants.

```
> BSTZINB(y,X,A,LinearT = TRUE,
nchain=3, niter=1000, nburn=0, nthin=1)
```

- **y**: A response variable in a vector form of size n . Our model assumes y is arranged in ascending order of spatial unit and temporal unit. For example, if we assume s

spatial units and t temporal unit, then we consider y of size $n = st$ as arranged in the following order: $(1,1), (1,2), \dots, (1,t), (2,1), (2,2), \dots, (s,t)$

- **X**: A set of covariates in a matrix form. It has n rows indicating spatio-temporal unit, and p columns indicating each covariate.
- **A**: Adjacency matrix of which (i,i') element is either 1 if spatial unit i and i' are neighbors, or 0 otherwise.
- **LinearT**: A logical indicator. The default option is TRUE. TRUE means that we assume a linear time trend structure. FALSE means that we assume a non-linear time trend structure represented by B-spline functions (See the reference).
- **nchain**: The number of chains in MCMC settings.
- **niter**: The number of iterations in MCMC settings.
- **nburn**: The number of burn-in samples in MCMC settings.
- **nthin**: The number of thinning samples in MCMC settings.

The output of the BSTZINB function consists of the following:

- **Alpha**: MCMC sampled coefficients in the binary (structure zero) model component.
- **Beta**: MCMC sampled coefficients in the count (Negative Binomial) model component.
- **R**: MCMC sampled dispersion parameters using the random-walk Metropolis–Hastings approach.
- **R2**: MCMC sampled dispersion parameter using the two-stage Gibbs approach.
- **Sigphi**: MCMC Sampled values of the 4×4 conditional covariance matrix Γ .
- **PHI1**: MCMC Sampled values of random intercepts for the binary (structure zero) model component.
- **PHI2**: MCMC Sampled values of random slopes for the binary (structure zero) model component.
- **PHI3**: MCMC Sampled values of random intercepts for the count (Negative Binomial) model component.
- **PHI4**: MCMC Sampled values of random slopes for the count (Negative Binomial) model component.
- **ETA1**: MCMC Sampled values of $\text{logit}(q_{ij})$ for the binary (structure zero) model component.
- **ETA2**: MCMC Sampled values of $\text{logit}(p_{ij})$ for the count (Negative Binomial) model component.

Appendix B.3. Auxiliary Functions

The following are descriptions for a list of functions other than the main functions provided in Table 1.

```
> USDmapCount(state.sel,dat,scol,tcol=NULL,
tset=NULL,cname,uplim=NULL)
```

- **state.sel**: A name set of states in the United States
- **dat**: A dataset with spatial and/or temporal unit variables.
- **scol**: An index or indexes of spatial unit variables in dat.
- **tcol**: An index of temporal unit variables in dat.
- **tset**: A value of temporal units in dat. **tset=NULL** is a default option, which shows a time-averaged output. You may specify a specific time value to show a time-specific output.
- **cname**: A name or a set of names of US counties.
- **uplim**: An upper limit value of outcome variables to visualize. **uplim=NULL** is a default option, which considers a maximum value of outcome variables to visualize.

```
> get_adj_mat(county.adjacency, Countyvec, Statevec)
```

- `county.adjacency`: A dataset of US counties including county names and their neighbors. `county.adjacency` (See Appendix B.1) is an exemplary dataset provided in our package.
- `Countyvec`: A set of names of US counties on your data analysis. All the counties should belong to the US states specified in the `Statevec` argument.
- `Statevec`: A set of names of US states on your data analysis.

`qRankPar` and `qRankParTop` have similar arguments to use, so we will only illustrate `qRankParTop` as their representative one.

```
> qRankParTop(state.set, cname, bstfit, vn=12, cex.title=18, cex.lab=12, cex.
  ↪ legend=12)
```

- `state.set`: A name set of states in the United States.
- `cname`: A name or a set of names of US counties.
- `bstfit`: One of the `BSTNB`, or `BSTZINB` objects in our package.
- `vn`: The number of selected counties to visualize.
- `cex.title`: Font size for the title of the graph (in points).
- `cex.lab`: Font size for the labels on the axes (in points).
- `cex.legend`: Font size for the legend text (in points).

```
> TimetrendCurve(bstfit, cname, vn=5, smooth.mode=TRUE, cex.title=18, cex.lab
  ↪ =12, cex.legend=12)
```

- `bstfit`: One of the `BNB`, `BZINB`, `BSTNB`, or `BSTZINB` objects in our package.
- `cname`: A name or a set of names of US counties.
- `vn`: The number of selected counties to visualize.
- `smooth.mode`: `TRUE` if you apply smoothing spline approach to visualize η_1 (default); `FALSE` if you visualize a raw line of η_1
- `cex.title`: Font size for the title of the graph (in points).
- `cex.lab`: Font size for the labels on the axes (in points).
- `cex.legend`: Font size for the legend text (in points).

```
> ResultTableSummary(bstfit)
```

- `bstfit`: One of the `BNB`, `BZINB`, `BSTNB`, or `BSTZINB` objects in our package.

```
> ResultTableSummary2(y, X, A, LinearT=FALSE, nchain=3, niter=10, nburn=0,
  ↪ nthin=1){
```

- `y`: A response variable in a vector form of size n . Our model assumes `y` is arranged in ascending order of spatial unit and temporal unit. For example, if we assume s spatial units and t temporal unit, then we consider `y` of size $n = st$ as arranged in the following order: $(1, 1), (1, 2), \dots, (1, t), (2, 1), (2, 2), \dots, (s, t)$
- `X`: A set of covariates in a matrix form. It has n rows indicating spatio-temporal unit, and p columns indicating each covariate.
- `A`: Adjacency matrix of which (i, i') element is either 1 if spatial unit i and i' are neighbors, or 0 otherwise.

- **LinearT**: A logical indicator. The default option is TRUE. TRUE means that we assume a linear time trend structure. FALSE means that we assume a non-linear time trend structure represented by B-spline functions (see the reference).
- **nchain**: The number of chains in MCMC settings.
- **niter**: The number of iterations per chain in MCMC settings.
- **nburn**: The number of burn-in samples per chain in MCMC settings.
- **nthin**: The number of thinning samples in MCMC settings.

`compute_ZINB_DIC` and `compute_NB_DIC` have similar arguments to use, so we will only illustrate `compute_ZINB_DIC` as their representative one.

```
> compute_ZINB_DIC(y, bstfit, lastit, nchain)
```

- **y**: A response variable in a vector form of size n . Our model assumes y is arranged in ascending order of spatial unit and temporal unit. For example, if we assume s spatial units and t temporal units, then we consider y of size $n = st$ as arranged in the following order: $(1, 1), (1, 2), \dots, (1, t), (2, 1), (2, 2), \dots, (s, t)$
- **bstfit**: One of the BZINB, or BSTZINB objects in our package.
- **lastit**: The number of iterations minus the number of burn-in samples per chain in MCMC settings.
- **nchain**: The number of chains in MCMC settings.

```
> conv.test(params, nchain=3, thshold=1.96)
```

- **params**: An array of MCMC samples in parameters. This can be recovered from parameters in BNB, BZINB, BSTNB, or BSTZINB objects.
- **nchain**: The number of chains in MCMC settings.
- **thshold**: Decision criteria of Geweke statistics.

Appendix B.4. Imports and Suggests of R- Libraries

The following are the libraries required to run BSTZINB.

```
> library(BayesLogit)
> library(boot)
> library(coda)
> library(dplyr)
> library(ggplot2) ## For maps
> library(gt)
> library(gtsummary) ## For summary tables
> library(maps)
> library(matrixcalc)
> library(MCMCpack)
> library(msm)
> library(reshape)
> library(spam)
> library(viridis)
```

The following libraries are suggested for BSTZINB package:

```

> library(knitr)
> library(rmarkdown)
> library(CorrMixed)
> library(ape)
> library(testthat)

```

References

1. Chakraborty, S.; Dey, T.; Jun, Y.; Lim, C.Y.; Mukherjee, A.; Dominici, F. A spatiotemporal analytical outlook of the exposure to air pollution and COVID-19 mortality in the USA. *J. Agric. Biol. Environ. Stat.* **2022**, *27*, 419–439.
2. Yates, E.F.; Zhang, K.; Naus, A.; Forbes, C.; Wu, X.; Dey, T. A review on the biological, epidemiological, and statistical relevance of COVID-19 paired with air pollution. *Environ. Adv.* **2022**, *8*, 100250. <https://doi.org/https://doi.org/10.1016/j.envadv.2022.100250>.
3. Dey, T.; Dominici, F. COVID-19, Air Pollution, and Racial Inequity: Connecting the Dots. *Chem. Res. Toxicol.* **2021**, *34*, 669–671. <https://doi.org/10.1021/acs.chemrestox.0c00432>.
4. Nicola, M.; Alsafi, Z.; Sohrabi, C.; Kerwan, A.; Al-Jabir, A.; Iosifidis, C.; Agha, M.; Agha, R. The socio-economic implications of the coronavirus pandemic (COVID-19): A review. *Int. J. Surg.* **2020**, *78*, 185–193. <https://doi.org/https://doi.org/10.1016/j.ijsu.2020.04.018>.
5. Lambert, D. Zero-Inflated Poisson Regression, with an Application to Defects in Manufacturing. *Technometrics* **1992**, *34*, 1–14.
6. Min, Y.; Agresti, A. Random effect models for repeated measures of zero-inflated count data. *Stat. Model.* **2005**, *5*, 1–19. <https://doi.org/10.1191/1471082X05st084oa>.
7. Cameron, A.C.; Trivedi, P.K. *Regression Analysis of Count Data*, 2 ed.; Econometric Society Monographs; Cambridge University Press: Cambridge, UK, 2013. <https://doi.org/10.1017/CBO9781139013567>.
8. Hall, D.B. Zero-Inflated Poisson and Binomial Regression with Random Effects: A Case Study. *Biometrics* **2000**, *56*, 1030–1039. <https://doi.org/10.1111/j.0006-341X.2000.01030.x>.
9. Cameron, A.C.; Trivedi, P.K. *Regression Analysis of Count Data*; Cambridge University Press: Cambridge, UK, 1998. <https://doi.org/10.1017/CBO9780511814365>.
10. Hilbe, J.M. *Negative Binomial Regression*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2011. <https://doi.org/10.1017/CBO9780511973420>.
11. Lawson, A.B. *Bayesian Disease Mapping: Hierarchical Modeling in Spatial Epidemiology*, 3rd ed.; CRC Press: Boca Raton, FL, 2018.
12. Banerjee, S.; Carlin, B.P.; Gelfand, A.E. *Hierarchical Modeling and Analysis for Spatial Data*, 2nd ed.; CRC Press: Boca Raton, FL, USA, 2015.
13. Wakefield, J. Disease Mapping and Spatial Regression with Count Data. *Biostatistics* **2007**, *8*, 158–183. <https://doi.org/10.1093/biostatistics/kxl008>.
14. Lee, D. CARBayes: An R Package for Bayesian Spatial Modeling with Conditional Autoregressive Priors. *J. Stat. Softw.* **2013**, *55*, 1–24. <https://doi.org/10.18637/jss.v055.i13>.
15. Rue, H.; Martino, S.; Chopin, N. Approximate Bayesian Inference for Latent Gaussian Models by Using Integrated Nested Laplace Approximations. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2009**, *71*, 319–392. <https://doi.org/10.1111/j.1467-9868.2008.00700.x>.
16. He, Q.; Huang, H.H. A Framework of Zero-Inflated Bayesian Negative Binomial Regression Models for Spatiotemporal Data. *J. Stat. Plan. Inference* **2024**, *229*, 106098. <https://doi.org/10.1016/j.jspi.2023.106098>.
17. Neelon, B. Bayesian Zero-Inflated Negative Binomial Regression Based on Pólya-Gamma Mixtures. *Bayesian Anal.* **2019**, *14*, 829–855. <https://doi.org/10.1214/18-BA1132>.
18. Gordon, W.J.; Riesenfeld, R.F. B-spline curves and surfaces. In *Computer Aided Geometric Design*; Elsevier: Amsterdam, The Netherlands, 1974; pp. 95–126.
19. Polson, N.G.; Scott, J.G.; Windle, J. Bayesian Inference for Logistic Models Using Pólya-Gamma Latent Variables. *J. Am. Stat. Assoc.* **2013**, *108*, 1339–1349. <https://doi.org/10.1080/01621459.2013.829001>.
20. Pillow, J.; Scott, J. Fully Bayesian inference for neural models with negative-binomial spiking. In *Advances in Neural Information Processing Systems*; Bartlett, P., Pereira, F., Burges, C., Bottou, L., Weinberger, K., Eds.; MIT Press: Cambridge, MA, USA, 2012; Volume 25, pp. 1907–1915.
21. Moran, P.A. Notes on continuous stochastic phenomena. *Biometrika* **1950**, *37*, 17–23.
22. Li, H.; Calder, C.A.; Cressie, N. Beyond Moran’s I: Testing for spatial dependence based on the spatial autoregressive model. *Geogr. Anal.* **2007**, *39*, 357–375.
23. Johnson, L.T.; Geyer, C.J. Variable transformation to obtain geometric ergodicity in the random-walk Metropolis algorithm. *Ann. Stat.* **2012**, *40*, 3050–3076.

24. Zhou, M.; Carin, L. Negative binomial process count and mixture modeling. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *37*, 307–320.
25. Geweke, J. *Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments*; Technical Report; Federal Reserve Bank of Minneapolis: Minneapolis, MN, USA, 1991.
26. Geyer, C.J. Introduction to markov chain monte carlo. In *Handbook of Markov Chain Monte Carlo*; Routledge: London, UK, 2011; pp. 1–46.
27. Gelman, A.; Carlin, J.B.; Stern, H.S.; Rubin, D.B. *Bayesian Data Analysis*; Chapman and Hall/CRC: Boca Raton, FL, USA, 1995.
28. Spiegelhalter, D.J.; Best, N.G.; Carlin, B.P.; Van Der Linde, A. Bayesian measures of model complexity and fit. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2002**, *64*, 583–639.
29. Stan Development Team. *RStan: The R Interface to Stan*; R package version 2.32.6.; RStan: New York, NY, USA, 2024.
30. Gelman, A.; Carlin, J.B.; Stern, H.S.; Dunson, D.B.; Vehtari, A.; Rubin, D.B. *Bayesian Data Analysis*; CRC Press: Boca Raton, FL, USA, 2013.

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.