Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2022.022098

check for updates

ARTICLE

Bayesian Computation for the Parameters of a Zero-Inflated Cosine Geometric Distribution with Application to COVID-19 Pandemic Data

Sunisa Junnumtuam, Sa-Aat Niwitpong* and Suparat Niwitpong

Department of Applied Statistics, Faculty of Applied Science, King Mongkut's University of Technology North Bangkok, Bangkok, 10800, Thailand

*Corresponding Author: Sa-Aat Niwitpong. Email: sa-aat.n@sci.kmutnb.ac.th

Received: 21 February 2022 Accepted: 08 June 2022

ABSTRACT

A new three-parameter discrete distribution called the zero-inflated cosine geometric (ZICG) distribution is proposed for the first time herein. It can be used to analyze over-dispersed count data with excess zeros. The basic statistical properties of the new distribution, such as the moment generating function, mean, and variance are presented. Furthermore, confidence intervals are constructed by using the Wald, Bayesian, and highest posterior density (HPD) methods to estimate the true confidence intervals for the parameters of the ZICG distribution. Their efficacies were investigated by using both simulation and real-world data comprising the number of daily COVID-19 positive cases at the Olympic Games in Tokyo 2020. The results show that the HPD interval performed better than the other methods in terms of coverage probability and average length in most cases studied.

KEYWORDS

Bayesian analysis; confidence interval; gibbs sampling; random-walk metropolis; zero-inflated count data

1 Introduction

Over-dispersed count data with excess zeros occur in various situations, such as the number of torrential rainfall incidences at the Daegu and the Busan rain gauge stations in South Korea [1], the DMFT (decayed, missing, and filled teeth) index in dentistry [2], and the number of falls in a study on Parkinson's disease [3]. Classical models such as Poisson, geometric, and negative binomial (NB) distributions may not be suitable for analyzing these data, so two classes of modified count models (zero-inflated (ZI) and hurdle) are used instead. Both can be viewed as finite mixture models comprising two components: for the zero part, a degenerate probability mass function is used in both, while for the non-zero part, a zero-truncated probability mass function is used in hurdle models and an untruncated probability mass function is used in ZI models. Poisson and geometric hurdle models were proposed and used by [4] to analyze data on the daily consumption of various beverages; in the intercept-only case (no regressors appear in either part of the model), the ZI model is equivalent to the hurdle model, with the estimation yielding the same log-likelihood and fitted probabilities. Furthermore, several comparisons with classical models have been reported in the literature. The efficacies of ZI and hurdle models have been explored by comparing least-squares regression with transformed outcomes (LST), Poisson regression, NB regression, ZI Poisson (ZIP), ZINB, zero-altered Poisson (ZAP) (or Poisson hurdle), and zero-altered NB (ZANB) (or NB hurdle) models [5]; the results from using both simulated and real data on health surveys show that the ZANB and ZINB models performed better than the others when the data had excess zeros and were over-dispersed. Recently, Feng [6] reviewed ZI and hurdle models and highlighted their differences in terms of their data-generating process; they conducted simulation studies to evaluate the performances of both types of models, which were found to be dependent on the percentage of zero-deflated data points in the data and discrepancies between structural and sampling zeros in the data-generating process.

The main idea of a ZI model is to add a proportion of zeros to the baseline distribution [7,8], for which various classical count models (e.g., ZIP, ZINB, and ZI geometric (ZIG)), are available. These have been studied in several fields and many statistical tools have been used to analyze them. The ZIP distribution originally proposed by [9] has been studied by various researchers. For instance, Ridout et al. [10] considered the number of roots produced by 270 shoots of *Trajan* apple cultivars (the number of shoots entries provided excess zeros in the data), and analyzed the data by using Poisson, NB, ZIP, and ZINB models; the fits of these models were compared by using the Akaike information criterion (AIC) and the Bayesian information criterion (BIC), the results of which show that ZINB performed well. Yusuf et al. [11] applied the ZIP and ZINB regression models to data on the number of falls by elderly individuals; the results show that the ZINB model attained the best fit and was the best model for predicting the number of falls due to the presence of excess zeros and over-dispersion in the data. Iwunor [12] studied the number of male rural migrants from households by using an inflated geometric distribution and estimated the parameters of the latter; the results show that the maximum likelihood estimates were not too different from the method of moments values. Kusuma et al. [13] showed that a ZIP regression model is more suitable than an ordinary Poisson regression model for modeling the frequency of health insurance claims.

The cosine geometric (CG) distribution, a newly reported two-parameter discrete distribution belonging to the family of weighted geometric distributions [14], is useful for analyzing over-dispersed data and has outperformed some well-known models such as Poisson, geometric, NB, and weighted NB. In the present study, the CG distribution was applied as the baseline and then a proportion of zeros was added to it, resulting in a novel three-parameter discrete distribution called the ZICG distribution.

Statistical tools such as confidence intervals provide more information than point estimation and p-values for statistical inference [15]. Hence, they have often been applied to analyze ZI count data. For example, Wald confidence intervals for the parameters in the Bernoulli component of ZIP and ZAP models were constructed by [16], while Waguespack et al. [17] provided a Wald-based confidence interval for the ZIP mean. Moreover, Srisuradetchai et al. [18] proposed the profile-likelihood-based confidence interval for the geometric parameter of a ZIG distribution. Junnumtuam et al. [19] constructed Wald confidence intervals for the parameters of a ZIP model; in an analysis of the number of daily COVID-19 deaths in Thailand using six models: Poisson, NB, geometric, Gaussian, ZIP, and ZINB, they found that the Wald confidence intervals for the ZIP model were the most suitable. Furthermore, Srisuradetchai et al. [20] proposed three confidence intervals: a Wald confidence interval and score confidence intervals using the profile and the expected or observed Fisher information for the Poisson parameter in a ZIP distribution; the latter two outperformed the Wald confidence interval in terms of coverage probability, average length, and the coverage per unit length.

Besides the principal method involving maximum likelihood estimation widely used to estimate parameters in ZI count models, Bayesian analysis is also popular. For example, Cancho et al. [21] provided a Bayesian analysis for the ZI hyper-Poisson model by using the Markov chain Monte Carlo

(MCMC) method; they used some noninformative priors in the Bayesian procedure and compared the Bayesian estimators with maximum likelihood estimates obtained by using the Newton-Raphson method and found that all of the estimates were close to the real values of the parameters as the sample size was increased, which means that their biases and mean-squared errors (MSEs) approached zero under this circumstance. Recently, Workie et al. [22] applied the Bayesian analytic approach by using MCMC simulation and Gibbs' sampling algorithm for modeling the Bayesian ZI regression model determinants to analyze under-five child mortality.

Motivated by these previous studies, we herein propose Wald confidence intervals based on maximum likelihood estimation, Bayesian credible intervals, and highest posterior density (HPD) intervals for the three parameters of a ZICG distribution. Both simulated data and real-world data were used to compare the efficacies of the proposed methods for constructing confidence intervals via their coverage probabilities and average lengths.

2 Methodology

2.1 The ZICG Distribution

The CG distribution is a two-parameter discrete distribution belonging to the family of weighted geometric distributions [14]. The probability mass function (PMF) for a CG distribution is given by

$$P(Y = y) = C_{p,\theta} p^y [\cos(y\theta)]^2, y \in \mathbb{N}, \tag{1}$$

where $\theta \in \left[0, \frac{\pi}{2}\right]$ and

$$C_{p,\theta} = \frac{2(1-p)(1-2p\cos(2\theta)+p^2)}{2+p((p-3)\cos(2\theta)+p-1)}.$$

If $\theta = 0$, then we can obtain $C_{p,\theta} = 1 - p$ and Y is a standard geometric distribution. Let X be a random variable following a ZICG distribution with parameters $\omega \in (0, 1)$, $p \in (0, 1)$, and $\theta \in \left[0, \frac{\pi}{2}\right]$. Subsequently, we can construct a new three-parameter discrete distribution with CG as the baseline distribution, and so the pmf of X is given by

$$P(X = x) = \begin{cases} \omega + (1-\omega) C_{p,\theta} & x = 0 \\ (1-\omega) C_{p,\theta} p^x [\cos(x\theta)]^2 & x = 1, 2, 3, \ldots \end{cases}, \tag{2}$$

where ω is the probability of zeros and $0 \le \omega \le 1$. Moreover, one can easily prove $\sum_{x=0}^{\infty} P(X = x) = 1$. Fig. 1 provides pmf plots for the ZICG distribution for different parameter combinations. It can be seen that even though the proportion of zeros (ω) is small (i.e., $\omega = 0.1$), the probability of zeros is still high; e.g., $\omega = 0.1, p = 0.5$, and $\theta = 1$ provides $P(X = 0) > 0.5$. Moreover, when p is large, the dispersion is high. Overall, the ZICG distribution, which is suitable for data that are over-dispersed with excess zeros, can be used to analyze ZI count data.

2.2 Statistical Properties

This section provides the cumulative distribution function (CDF), moment generating function (MGF), mean, and variance of a ZICG distribution, which are derived from the CG distribution [14].

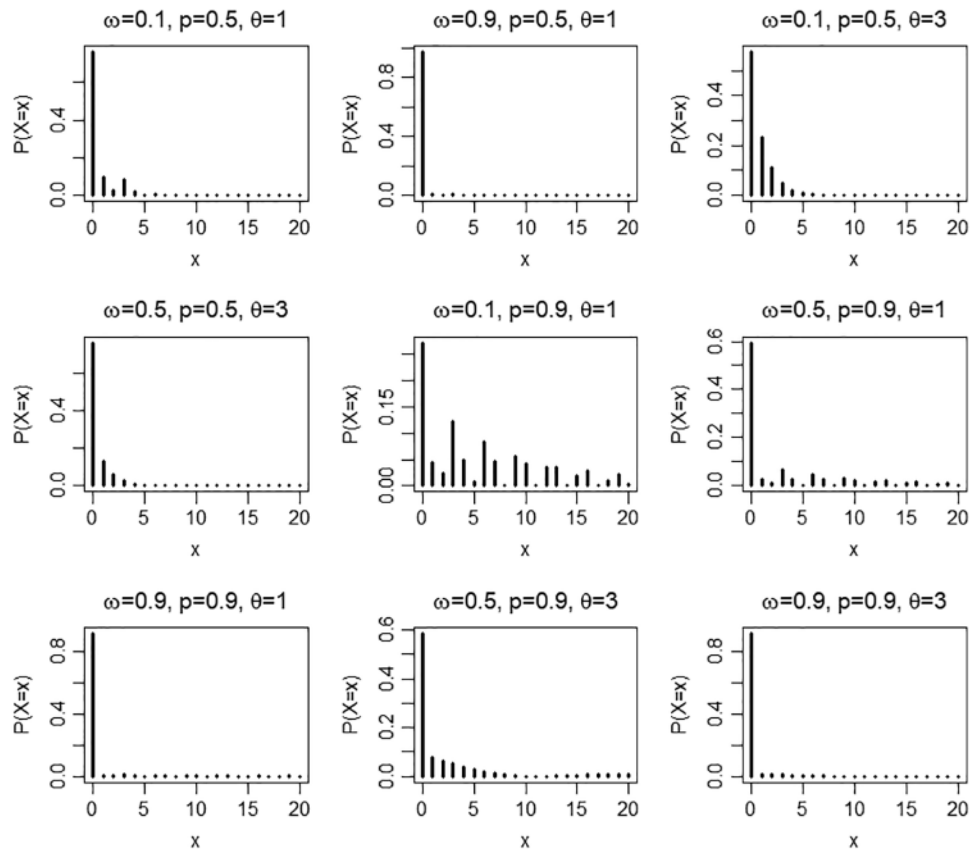

Figure 1: Pmf plots of the ZICG distribution for different values of the parameters ω, p, and θ

Proposition 2.1. The cdf of a ZICG distribution with parameters ω, p, and θ is given by

$$F(m) = P(X \le m) = \sum_{k=0}^{m} P(X = k) = \omega + \sum_{k=0}^{m} (1-\omega)\, C_{p,\theta} p^k [\cos(k\theta)]^2$$

$$= \omega + (1-\omega)\, C_{p,\theta} \sum_{k=0}^{m} p^k [\cos(k\theta)]^2$$

$$= \omega + (1-\omega)\, C_{p,\theta} \sum_{k=0}^{m} p^k \left(\frac{1+\cos(2k\theta)}{2}\right)$$

$$= \omega + (1-\omega)\, \frac{C_{p,\theta}}{2} \left[\sum_{k=0}^{m} p^k + Re\left(\sum_{k=0}^{m} \left(pe^{i2\theta}\right)^k\right)\right]$$

$$= \omega + (1-\omega)\, \frac{C_{p,\theta}}{2} \left[\frac{1-p^{m+1}}{1-p} + Re\left[\frac{1-p^{m+1}e^{i2(m+1)\theta}}{1-pe^{i2\theta}}\right]\right]$$

$$= \omega + (1-\omega)\, \frac{C_{p,\theta}}{2} \left[\frac{1-p^{m+1}}{1-p} + \frac{1-p^{m+1}\cos(2(m+1)\theta) - p\cos(2\theta) + p^{m+2}\cos(2m\theta)}{1-2p\cos(2\theta)+p^2}\right]. \quad (3)$$

Proposition 2.2. The mgf of the ZICG distribution with parameters ω, p, and θ is given by

$$M_X(t) = E\left(e^{tX}\right) = \sum_{k=0}^{\infty} e^{tk} P(X = k)$$

$$= e^0\left(\omega + (1-\omega) C_{p,\theta}\right) + \sum_{k=1}^{\infty} e^{tk} (1-\omega) C_{p,\theta} p^k [\cos(k\theta)]^2$$

$$= \omega + (1-\omega) C_{p,\theta} + (1-\omega) C_{p,\theta} \sum_{k=1}^{\infty} \left(pe^t\right)^k [\cos(k\theta)]^2$$

$$= \omega + (1-\omega) C_{p,\theta} \sum_{k=0}^{\infty} \left(pe^t\right)^k [\cos(k\theta)]^2$$

$$= \omega + (1-\omega) C_{p,\theta} \sum_{k=0}^{\infty} \left(pe^t\right)^k \left(\frac{1+\cos(2k\theta)}{2}\right)$$

$$= \omega + (1-\omega) \frac{C_{p,\theta}}{2} \left[\sum_{k=0}^{\infty} \left(pe^t\right)^k + Re\left(\sum_{k=0}^{\infty} \left(pe^t e^{i2\theta}\right)^k\right)\right]$$

$$= \omega + (1-\omega) \frac{C_{p,\theta}}{2} \left[\frac{1}{1-pe^t} + \frac{1-pe^t\cos(2\theta)}{1-2pe^t\cos(2\theta)+(pe^t)^2}\right]. \tag{4}$$

Since the explicit expression for the moment using equality is $\mu_r' = E(X^r) = M^{(r)}(t)\mid_{t=0}$, then the first two moments respectively become

$$M_X'(t) = (1-\omega)\, pe^t \frac{C_{p,\theta}}{2} \left[\frac{1}{(1-pe^t)^2} + \frac{\cos(2\theta)(1+p^2e^{2t}) - 2pe^t}{(1-2pe^t\cos(2\theta)+(pe^t)^2)^2}\right], \tag{5}$$

$$M_X''(t) = (1-\omega)\, pe^t \frac{C_{p,\theta}}{2} \left[\left(\frac{1+pe^t}{(1-pe^t)^3}\right) - \frac{\cos(2\theta)((pe^t)^4-1) + (pe^t)^3(\cos(4\theta-3))}{(1-2pe^t\cos(2\theta)+(pe^t)^2)^3} + \frac{pe^t(3-\cos(4\theta))}{(1-2pe^t\cos(2\theta)+(pe^t)^2)^3}\right]. \tag{6}$$

Since $M_X'(t=0) = E(X)$, then

$$E(X) = (1-\omega)\, p \frac{C_{p,\theta}}{2} \left[\frac{1}{(1-p)^2} + \frac{\cos(2\theta)(1+p^2) - 2p}{(1-2p\cos(2\theta)+p^2)^2}\right], \tag{7}$$

and $M_X''(t=0) = E(X^2)$, then

$$E\left(X^2\right) = (1-\omega)\, p \frac{C_{p,\theta}}{2} \left[\frac{1+p}{(1-p)^3} - \frac{\cos(2\theta)(p^4-1) + p^3(\cos(4\theta-3))}{(1-2p\cos(2\theta)+p^2)^3} + \frac{p(3-\cos(4\theta))}{(1-2p\cos(2\theta)+p^2)^3}\right]. \tag{8}$$

Since $V(X) = E(X^2) - (E(X))^2$, then

$$V(X) = (1-\omega)\, p \frac{C_{p,\theta}}{2} \left[\frac{1+p}{(1-p)^3} - \frac{\cos(2\theta)(p^4-1) + p^3(\cos(4\theta-3)) + p(3-\cos(4\theta))}{(1-2p\cos(2\theta)+p^2)^3}\right] - \left[(1-\omega)\, p \frac{C_{p,\theta}}{2} \left[\frac{1}{(1-p)^2} + \frac{\cos(2\theta)(1+p^2) - 2p}{(1-2p\cos(2\theta)+p^2)^2}\right]\right]^2. \tag{9}$$

The index of dispersion (D), a measure of dispersion, is defined as the ratio of the variance to the mean

$$D = \frac{V(X)}{E(X)}. \tag{10}$$

The values of D for selected values of the parameters ω, p, and θ are provided in Table 1. When the value of parameter p increases, the index of dispersion also increases, and so the value of p affects D much more than parameters ω and θ.

Table 1: Index of dispersion of the ZICG distribution for different values of parameters

ω	$p=0.1$			$p=0.5$			$p=0.9$		
	$\theta=1$	$\theta=3$	$\theta=5$	$\theta=1$	$\theta=3$	$\theta=5$	$\theta=1$	$\theta=3$	$\theta=5$
0.1	1.2456	1.1078	1.7272	2.7955	1.7121	2.7514	11.4168	13.8715	11.3870
0.2	1.2491	1.1185	1.7296	2.8589	1.7963	2.8172	12.2676	14.5714	12.2392
0.3	1.2525	1.1293	1.7319	2.9222	1.8804	2.8831	13.1184	15.2713	13.0915
0.4	1.2560	1.1401	1.7343	2.9855	1.9645	2.9490	13.9691	15.9712	13.9437
0.5	1.2595	1.1508	1.7367	3.0489	2.0487	3.0149	14.8199	16.6711	14.7960
0.6	1.2629	1.1616	1.7390	3.1122	2.1328	3.0808	15.6707	17.3710	15.6482
0.7	1.2664	1.1724	1.7414	3.1756	2.2169	3.1466	16.5215	18.0708	16.5004
0.8	1.2699	1.1831	1.7437	3.2389	2.3010	3.2125	17.3723	18.7707	17.3527
0.9	1.2733	1.1939	1.7461	3.3023	2.3852	3.2784	18.2231	19.4706	18.2049

2.3 Maximum Likelihood Estimation for the ZICG Model with No Covariates

The likelihood function of the ZICG distribution is

$$L(\omega, p, \theta) = \prod_{i=1}^{n} \left[\omega_i + (1-\omega_i) C_{p_i,\theta}\right]^{I(x_i=0)} \prod_{i=1}^{n} \left[(1-\omega_i) C_{p_i,\theta} p^{x_i} \left[\cos(x_i\theta)\right]^2\right]^{I(x_i>0)}, \quad (11)$$

while the log-likelihood function of the ZICG distribution can be expressed as

$$l = \sum_{i=1}^{n} \log\left[\omega_i + (1-\omega_i) C_{p_i,\theta}\right]^{I(x_i=0)} + \sum_{i=1}^{n} \log\left[(1-\omega_i) C_{p_i,\theta} p^{x_i} \left(\cos(x_i\theta)\right)^2\right]^{I(x_i>0)}. \quad (12)$$

In the case of a single homogeneous sample (p, θ, and ω are constant or have no covariates), the log-likelihood function can be written as

$$l = n_0 \log\left(\omega + (1-\omega) C_{p,\theta}\right) + \sum_{j=1}^{J} n_j \log\left[(1-\omega) C_{p,\theta} p^j \left(\cos(j\theta)\right)^2\right], \quad (13)$$

where J is the largest observed count value; n_j is the frequency of each possible count value; $j = x = 0, 1, 2, \ldots, J$; n_0 is the number of observed zeros; and $\sum_{j=0}^{J} n_j = n$ is the total number of observations or the sample size. Based on log-likelihood function (13), maximum likelihood estimates $\hat{\omega}$, $\hat{p}$, and $\hat{\Theta}$ are the roots of equations $\frac{\partial l}{\partial \omega} = 0$, $\frac{\partial l}{\partial p} = 0$, and $\frac{\partial l}{\partial \theta} = 0$, respectively.

Here, we have

$$\frac{\partial l}{\partial \omega} = \frac{n_0\left(1 - C_{p,\theta}\right)}{\omega + (1-\omega) C_{p,\theta}} + \sum_{j=1}^{J} \frac{n_j\left(-C_{p,\theta} p^j \left(\cos(j\theta)\right)^2\right)}{(1-\omega) C_{p,\theta} p^j \left(\cos(j\theta)\right)^2}$$

$$= \frac{n_0\left(1 - C_{p,\theta}\right)}{\omega + (1-\omega) C_{p,\theta}} - \sum_{j=1}^{J} \frac{n_j}{(1-\omega)}, \quad (14)$$

$$\frac{\partial l}{\partial p} = \frac{n_0 (1-\omega) C_{p,\theta}^{(p)}}{\omega + (1-\omega) C_{p,\theta}} + \sum_{j=1}^{J} \frac{n_j (1-\omega) (cos (j\theta))^2 \left(C_{p,\theta} j p^{j-1} + p^j C_{p,\theta}^{(p)} \right)}{(1-\omega) C_{p,\theta} p^j (cos (j\theta))^2}$$

$$= \frac{n_0 (1-\omega) C_{p,\theta}^{(p)}}{\omega + (1-\omega) C_{p,\theta}} + \sum_{j=1}^{J} \frac{n_j \times j}{p} + \sum_{j=1}^{J} \frac{n_j C_{p,\theta}^{(p)}}{C_{p,\theta}}, \quad (15)$$

$$\frac{\partial l}{\partial \theta} = \frac{n_0 (1-\omega) C_{p,\theta}^{(\theta)}}{\omega + (1-\omega) C_{p,\theta}} + \sum_{j=1}^{J} n_j \left[\frac{C_{p,\theta}{}^{(\theta)}}{C_{p,\theta}} - 2j \frac{\sin (j\theta)}{\cos (j\theta)} \right]$$

$$= \frac{n_0 (1-\omega) C_{p,\theta}^{(\theta)}}{\omega + (1-\omega) C_{p,\theta}} + \sum_{j=1}^{J} n_j \left[\frac{C_{p,\theta}{}^{(\theta)}}{C_{p,\theta}} - 2j \tan (j\theta) \right]. \quad (16)$$

Algorithm 1: Obtaining the maximum likelihood estimates of ω, p, and θ.

1. Fit a geometric model to obtain initial value $p^{(0)}$ for p of the CG model.
2. Fit a CG model to obtain initial values $p^{(1)}$ for p and $\theta^{(1)}$ for θ of the ZICG model.
3. Iterate the schemes for $\hat{p}$ and $\hat{\Theta}$ until convergence by using stopping rule $| \hat{p}^{(m+1)} - \hat{p}^{(m)} | < \varepsilon$, where $\hat{p}^{(m)}$ and $\hat{p}^{(m+1)}$ are estimates of p at the $(m)^{th}$ and $(m+1)^{th}$ iterations, respectively.
4. Obtain $\hat{\omega}$ by substituting $\hat{p}$ and $\hat{\Theta}$ for p and θ.

This provides the closed-form expression for $\hat{\omega}$, and so iteration is not required. However, since there are no closed-form expressions for $\hat{p}$ and $\hat{\Theta}$, they are solved by using an educated version of trial-and-error. The general idea is to start with an initial educated guess of the parameter value, calculate the log-likelihood for that value, and then iteratively find parameter values with larger and larger log-likelihoods until no further improvement can be achieved. There are a variety of fast and reliable computational algorithms for carrying out these procedures, one of the most widely implemented being the Newton-Raphson algorithm [23]. In this study, the maximum likelihood estimates of $\hat{p}, \hat{\omega}$, and $\hat{\Theta}$ can be obtained by solving the resulting equations simultaneously by using the *nlm* function in [24].

2.4 The Wald Confidence Intervals for the ZICG Parameters

In this study, we assume that there is more than one unknown parameter. Meanwhile, the assumed parameter vector is $\tilde{\beta} = (\beta_1, \ldots, \beta_k)^T$ and the maximum likelihood estimator for it is $\widehat{\beta_i}; i = 1, 2, \ldots, k$, where k is the number of parameters. Thus,

$$\frac{\widehat{\beta_i} - \beta_i}{se\left(\widehat{\beta_i}\right)} \sim N(0, 1), \quad (17)$$

where standard error $se\left(\widehat{\beta_i}\right) = \sqrt{\left[I^{-1}\left(\widehat{\beta_i}\right)\right]_{ii}}$ and $\left[I^{-1}\left(\widehat{\beta_i}\right)\right]_{ii}$ is the inverse of the expected Fisher information matrix for parameter order i. When n is large enough, the observed information converges in probability to the expected information according to the law of large numbers, at which point the expected Fisher information matrix is replaced by the observed Fisher information matrix. Since the ZICG distribution has three parameters (i.e., $\beta_1 = \omega, \beta_2 = p$, and $\beta_3 = \theta$), then the elements of observed Fisher information matrix $J_{(\beta_1, \beta_2, \beta_3)} = J_{(\omega,p,\theta)}$ are given by

$$J_{(\omega,p,\theta)} = -\begin{bmatrix} J_{\omega\omega} & J_{\omega p} & J_{\omega\theta} \\ J_{p\omega} & J_{pp} & J_{p\theta} \\ J_{\theta\omega} & J_{\theta p} & J_{\theta\theta} \end{bmatrix}, \tag{18}$$

where

$$J_{\omega\omega} = \frac{\partial^2 l}{\partial \omega^2}, J_{\omega p} = J_{p\omega} = \frac{\partial^2 l}{\partial \omega \partial p}, J_{\omega\theta} = J_{\theta\omega} = \frac{\partial^2 l}{\partial \omega \partial \theta}$$

$$J_{pp} = \frac{\partial^2 l}{\partial p^2}, J_{p\theta} = J_{\theta p} = \frac{\partial^2 l}{\partial p \partial \theta}, J_{\theta\theta} = \frac{\partial^2 l}{\partial \theta^2},$$

$$\frac{\partial^2 l}{\partial \omega^2} = -\frac{n_0(1 - C_{p,\theta})^2}{(\omega + (1-\omega)C_{p,\theta})^2} - \frac{\sum_{j=1}^{J} n_j}{(1-\omega)^2}, \tag{19}$$

$$\frac{\partial^2 l}{\partial p^2} = \frac{(\omega + (1-\omega)C_{p,\theta}) n_0 (1-\omega)(C_{p,\theta}^{(p)})^{(p)}}{(\omega + (1-\omega)C_{p,\theta})^2} - \frac{(n_0(1-\omega)C_{p,\theta}^{(p)})((1-\omega)C_{p,\theta}^{(p)})}{(\omega + (1-\omega)C_{p,\theta})^2}$$
$$- \frac{\sum_{j=1}^{J} n_j \times j}{p^2} + \sum_{j=1}^{J} n_j \frac{C_{p,\theta}(C_{p,\theta}^{(p)})^{(p)} - (C_{p,\theta}^{(p)})^2}{(C_{p,\theta})^2}, \tag{20}$$

where $C_{p,\theta}^{(p)} = \frac{\partial C_{p,\theta}}{\partial p}$, and $\left(C_{p,\theta}^{(p)}\right)^{(p)} = \frac{\partial C_{p,\theta}^{(p)}}{\partial p}$.

$$\frac{\partial^2 l}{\partial \theta^2} = \frac{(\omega + (1-\omega)C_{p,\theta}) n_0 (1-\omega)(C_{p,\theta}^{(\theta)})^{(\theta)}}{(\omega + (1-\omega)C_{p,\theta})^2} - \frac{(n_0(1-\omega)C_{p,\theta}^{(\theta)})((1-\omega)C_{p,\theta}^{(\theta)})}{(\omega + (1-\omega)C_{p,\theta})^2}$$
$$+ \sum_{j=1}^{J} n_j \left[\frac{C_{p,\theta}(C_{p,\theta}^{(\theta)})^{(\theta)} - (C_{p,\theta}^{(\theta)})^2}{(C_{p,\theta})^2} - 2j^2 \sec^2(j\theta) \right], \tag{21}$$

where $C_{p,\theta}^{(\theta)} = \frac{\partial C_{p,\theta}}{\partial \theta}$, and $\left(C_{p,\theta}^{(\theta)}\right)^{(\theta)} = \frac{\partial C_{p,\theta}^{(\theta)}}{\partial \theta}$.

$$\frac{\partial^2 l}{\partial \omega \partial p} = \frac{(\omega + (1-\omega)C_{p,\theta})(-n_0 C_{p,\theta}^{(p)}) - (n_0(1 - C_{p,\theta}))((1-\omega)C_{p,\theta}^{(p)})}{(\omega + (1-\omega)C_{p,\theta})^2}, \tag{22}$$

$$\frac{\partial^2 l}{\partial \omega \partial \theta} = \frac{(\omega + (1-\omega)C_{p,\theta})(-n_0 C_{p,\theta}^{(\theta)}) - (n_0(1 - C_{p,\theta}))((1-\omega)C_{p,\theta}^{(\theta)})}{(\omega + (1-\omega)C_{p,\theta})^2}, \tag{23}$$

$$\frac{\partial^2 l}{\partial p \partial \theta} = \frac{(\omega + (1-\omega)C_{p,\theta}) n_0 (1-\omega)(C_{p,\theta}^{(p)})^{(\theta)} - (n_0(1-\omega)C_{p,\theta}^{(p)})((1-\omega)C_{p,\theta}^{(\theta)})}{(\omega + (1-\omega)C_{p,\theta})^2}$$
$$+ \sum_{j=1}^{J} n_j \frac{C_{p,\theta}(C_{p,\theta}^{(p)})^{(\theta)} - C_{p,\theta}^{(p)} C_{p,\theta}^{(\theta)}}{(C_{p,\theta})^2}. \tag{24}$$

Hence, the $(1-\alpha)$ 100% Wald confidence interval can be constructed as

$$\widehat{\beta}_i \pm z_{1-\frac{\alpha}{2}} \sqrt{\left[J^{-1}\left(\widehat{\beta}_i\right)\right]_{ii}}, \tag{25}$$

where $z_{1-\frac{\alpha}{2}}$ is the $\left(1-\frac{\alpha}{2}\right) 100^{th}$ percentile of a standard normal distribution. The maximum likelihood estimators and their standard errors for ZICG can be obtained by using the *nlm* function, which minimizes them by using a Newton-type algorithm [24]. Thus, the simplified algorithm for the Wald confidence intervals for the ZICG parameters becomes

Algorithm 2: Establishing the Wald confidence intervals for the ZICG parameters.

1. Fit a geometric model to obtain initial value $p^{(0)}$ for p of the CG model.
2. Fit a CG model by using the *nlm* function to obtain initial values $p^{(1)}$ for p and $\theta^{(1)}$ for θ, and then plug in $p^{(1)}$ and $\theta^{(1)}$ to obtain $\omega^{(1)}$.
3. Fit the ZICG model by using the initial values from Step 2 to obtain $\hat{\omega}$, $\hat{p}$, and $\hat{\Theta}$ and their standard errors.
4. Calculate the Wald confidence intervals for the parameters of the ZICG distribution by substituting in the estimates from Step 3.

2.5 Bayesian Analysis for the Confidence Intervals for the ZICG Parameters

Suppose $X = x_1, x_2, \ldots, x_n$ is a sample from ZICG(ω, p, θ), then the likelihood function for the observed data is given by

$$L\left(data \mid \omega, p, \theta\right) = \prod\nolimits_{i=1}^{n} \left[\omega_i + (1-\omega_i)\, C_{p_i,\theta_i}\right]^{I(x_i=0)} \prod\nolimits_{i=1}^{n} \left[(1-\omega_i)\, C_{p_i,\theta_i} p^{x_i} \left[cos\left(x_i\theta_i\right)\right]^2\right]^{I(x_i>0)}. \quad (26)$$

Let $A = x_i : x_i = 0, i = 1, \ldots, n$ and m be the numbers in set A, then the likelihood function for ZICG can be written as

$$L\left[\omega, p, \theta\right] = \left[\omega + (1-\omega)\, C_{p,\theta}\right]^m (1-\omega)^{n-m} \prod_{x_i \notin A} C_{p,\theta} p^{x_i} \left[cos\left(x_i\theta\right)\right]^2. \quad (27)$$

Since the elements in set A can be generated from two different parts: (1) the real zeros part and (2) the CG distribution, after which the an unobserved latent allocation variable can be defined as

$$I_i = \begin{cases} 1; & p(\omega, p, \theta) \\ 0; & 1 - p(\omega, p, \theta) \end{cases}, \quad (28)$$

where $i = 1, \ldots, m$ and

$$p\left(\omega, p, \theta\right) = \frac{\omega}{\omega + (1-\omega)\, p\left(0 \mid p, \theta\right)}. \quad (29)$$

Thus, the likelihood function based on augmented data $D = \{X, I\}$, where $I = (I_1, \ldots, I_m)$ [25] becomes

$$\begin{aligned} L\left[\omega, p, \theta \mid D\right] &= L\left[\omega, p, \theta\right] \prod\nolimits_{i=1}^{m} p\left(\omega, p, \theta\right)^{I_i} \left(1 - p\left(\omega, p, \theta\right)\right)^{1-I_i} \\ &= \omega^S (1-\omega)^{n-S} p\left(0 \mid p, \theta\right)^{m-S} \prod_{x_i \notin A} p\left(x_i \mid p, \theta\right), \end{aligned} \quad (30)$$

where $S = \sum_{i=1}^{m} I_i \sim Bin\left[m, p\left(\omega, p, \theta\right)\right]$. Thus, the likelihood function for ZICG based on the augmented data becomes

$$L\left[\omega, p, \theta \mid D\right] \propto \omega^S (1-\omega)^{n-S} \prod_{x_i \notin A} C_{p,\theta} p^{x_i} \left[cos\left(x_i\theta\right)\right]^2, \quad (31)$$

and

$$p\left(\omega, p, \theta\right) = \frac{\omega}{\omega + (1-\omega)\, C_{p,\theta}}. \quad (32)$$

Since there is no prior information from historic data or previous experiments, we use the noninformative prior for all of the parameters. The prior distributions for ω and p are assumed to be beta distributions while that of θ is assumed to be a gamma distribution. Thus, the joint prior distribution for ZICG is

$$\pi\left(\omega,p,\theta\right)=\frac{\omega^{a-1}(1-\omega)^{b-1}}{B(a,b)}\times\frac{p^{c-1}(1-p)^{d-1}}{B(c,d)}\times\frac{1}{\beta^{\alpha}\Gamma(\alpha)}\theta^{\alpha-1}e^{\frac{-\theta}{\beta}},\tag{33}$$

where $B(a,b)=\frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$ and $B(c,d)=\frac{\Gamma(c)\Gamma(d)}{\Gamma(c+d)}$. In this study all of the parameters are assumed to have prior specifications, which are $\omega\sim Beta(1.5,1.5), p\sim Beta(2,5)$, and $\theta\sim Gamma(2,1/3)$. Since the posterior distributions for the parameters can be formed as [7]

$$posterior\propto likelihood\times prior,\tag{34}$$

the joint posterior distribution for parameters ω, p, and θ can be written as

$$P\left(\omega,p,\theta\right)\propto\omega^{S}\left(1-\omega\right)^{n-S}\prod_{x_i\notin A}C_{p,\theta}p^{x_i}\left[cos\left(x_i\theta\right)\right]^2\times\frac{\omega^{a-1}\left(1-\omega\right)^{b-1}}{B\left(a,b\right)}\times\frac{p^{c-1}(1-p)^{d-1}}{B(c,d)}\times\frac{1}{\beta^{\alpha}\Gamma(\alpha)}\theta^{\alpha-1}e^{\frac{-\theta}{\beta}}.\tag{35}$$

Since the joint posterior distribution in (35) is analytically intractable for calculating the Bayes estimates similarly to using the posterior distribution method, MCMC simulation can be applied to generate the parameters [26,27]. The Metropolis-Hastings algorithm is an MCMC method for obtaining a sequence of random samples from a probability distribution from which direct sampling is difficult. Subsequently, the obtained sequence can be used to approximate the desired distribution. Moreover, the Gibbs' sampler, which is an alternative to the Metropolis-Hastings algorithm for sampling from the posterior distribution of the model parameters, can be used. Hence, the Gibbs' sampler can be applied to generate samples from the joint posterior distribution in (35). Clearly, the marginal posterior distribution of ω given p and θ is

$$P\left(\omega\mid p,\theta\right)\propto\omega^{S+a-1}\left(1-\omega\right)^{n-S+b-1}.\tag{36}$$

Thus, the marginal posterior distribution of ω is $Beta(S+a,n-S+b)$, and the marginal posterior distribution of p given ω and θ is

$$P\left(p\mid\omega,\theta\right)\propto\prod_{x_i\notin A}C_{p,\theta}p^{x_i}\left[cos\left(x_i\theta\right)\right]^2\frac{p^{c-1}(1-p)^{d-1}}{B(c,d)},\tag{37}$$

and the marginal posterior distribution of θ given ω and p is

$$P\left(\theta\mid\omega,p\right)\propto\prod_{x_i\notin A}C_{p,\theta}p^{x_i}\left[cos\left(x_i\theta\right)\right]^2\frac{1}{\beta^{\alpha}\Gamma(\alpha)}\theta^{\alpha-1}e^{\frac{-\theta}{\beta}}.\tag{38}$$

Here, we applied the random-walk Metropolis (RWM) algorithm to generate p and θ. RWM is defined by using transition probability $p(x\rightarrow y)$ for one value x to y so that the distribution of points converges to $\pi(x)$. Since RWM is a special case of the Metropolis-Hastings algorithm with $p(x,y)=p(y,x)$ (symmetric) [28], then the acceptance probability can be calculated as

$$\alpha\left(X_{j-1},Y_j\right)=min\left(\frac{\pi\left(y\right)}{\pi\left(x\right)},1\right).\tag{39}$$

The process proceeds as follows:

1. Choose trial position $Y_j=X_{j-1}+\epsilon_j$, where ϵ_j is a random perturbation with distribution g that is symmetric (e.g., a normal distribution).

2. Calculate $r = \frac{\pi(y)}{\pi(x)}$.
3. Generate U_j from *Uniform*[0, 1].
4. If $U_j \leq \alpha(X_{j-1}, Y_j)$, then accept the change and let $X_j = Y_j$, else let $X_j = X_{j-1}$.

The Gibbs' sampling steps are as follows:

Algorithm 3: Establishing the Bayesian credible intervals.

1. Take the initial values of ω, p, and $\theta(\omega_0, p_0$ and θ_0, respectively).
2. Take the values ω_i, p_i and θ_i for ω, p and θ at the ith step, then
   (a) Calculate $p(\omega_0, p_0, \theta_0) = \frac{\omega_0}{\omega_0 + (1-\omega_0)C(p_0,\theta_0)}$.
   (b) Generate S_i from $Bin(m, p(\omega_{i-1}, p_{i-1}, \theta_{i-1}))$.
   (c) Generate ω_i from $Beta(S_i + a, n - S_t + b)$ and obtain ω_{i+1}.
   (d) Generate p_{i+1} by using the RWM algorithm.
   (e) Generate θ_{i+1} by using the RWM algorithm.
3. Repeat Step 2, N times.
4. Posterior analysis:
   (a) Calculate the Bayesian estimators of $g(\omega, p, \theta)$ by using $\frac{1}{N-M}\sum_{i=M+1}^{N} g(\omega_i, p_i, \theta_i)$, where M is the number of burn-in samples.
   (b) Calculate the 100(1−α)% confidence interval as $(g_{(\alpha/2)}, g_{(1-\alpha/2)})$, where $g_{(\alpha/2)}$ is the $\frac{\alpha}{2}$-th quantile of $g(\omega_i, p_i, \theta_i)$, and $g_{(1-\alpha/2)}$ is the $1-\frac{\alpha}{2}$-th quantile of $g(\omega_i, p_i, \theta_i), i = M+1, \ldots, N$.

2.6 The Bayesian-Based HPD Interval

The HPD interval is the shortest Bayesian credible interval containing $100(1-\alpha)\%$ of the posterior probability such that the density within the interval has a higher probability than outside of it. The two main properties of the HPD interval are as follows [29]:

1. The density for each point inside the interval is greater than that for each point outside of it.
2. For a given probability (say $1-\alpha$), the HPD interval has the shortest length.

Bayesian credible intervals can be obtained by using the MCMC method [30]. Hence, we used it to construct HPD intervals for the parameters of a ZICG distribution. This approach only requires MCMC samples generated from the marginal posterior distributions of the three parameters: ω, p, and θ. In the simulation and computation, the HPD intervals were computed by using the *HDInterval* package version 0.2.2 [31] from the R statistics program.

Algorithm 4: Establishing the HPD intervals by using the MCMC algorithm.

1. Take the initial values of ω, p, and $\theta(\omega_0, p_0$, and θ_0, respectively).
2. Take the values ω_i, p_i, and θ_i for ω, p, and θ at the ith step, then
   (a) Calculate $p\,(\omega_0, p_0, \theta_0) = \frac{\omega_0}{\omega_0 + (1-\omega_0)C(p_0,\theta_0)}$.
   (b) Generate S_i from $Bin\,(m, p\,(\omega_{i-1}, p_{i-1}, \theta_{i-1}))$.
   (c) Generate ω_i from $Beta(S_i + a, n - S_t + b)$ and obtain ω_{i+1}.
   (d) Generate p_{i+1} by using the RWM algorithm.
   (e) Generate θ_{i+1} by using the RWM algorithm.

(Continued)

Algorithm 4: (Continued)

3. Repeat Step 2, N times.
4. Posterior analysis:
   (a) Calculate the Bayesian estimators of $g(\omega, p, \theta)$ by using $\frac{1}{N-M}\sum_{i=M+1}^{N} g(\omega_i, p_i, \theta_i)$, where M is the number of burn-in samples.
   (b) Calculate the 100(1 − α% HPD intervals for the parameters by using the *HDInterval* package in R program.

2.7 The Efficacy Comparison Criteria

Coverage probabilities and average lengths were used to compare the efficacies of the confidence intervals. Suppose the nominal confidence level is $1-\alpha$, then confidence intervals that provide coverage probabilities of $1-\alpha$ or better are selected. In addition, the shortest average length identifies the best confidence interval under the provided conditions. Let $C_{(s)} = 1$ if the parameter values fall within the confidence interval range, else $C_{(s)} = 0$. The coverage probability is computed by

$$CP = \frac{1}{M}\sum_{s=1}^{M} C_{(s)}, \tag{40}$$

and the average length is computed by

$$AL = \frac{U_{(s)} - L_{(s)}}{M}, \tag{41}$$

where $U_{(s)}$ and $L_{(s)}$ are the upper and lower bounds of the confidence interval for loop s, respectively.

3 Results and Discussion

3.1 Simulation Study

Sample size $n = 50$ or 100; proportion of zeros $\omega = 0.1$, 0.5, or 0.9; $p = 0.5$ or 0.9; and $\theta = 1$ or 3 were the parameter values used in the simulation study. The simulation data were generated by using the inverse transform method [32], with the number of replications set as 1,000 and the nominal confidence level as 0.95. A flowchart of the simulation study is presented in Fig. 2, while the coverage probabilities and average lengths of the methods are reported in Table 2.

For sample size $n = 50$ or 100, the Bayesian credible intervals and the HPD intervals performed better than the Wald confidence intervals because they provided coverage probabilities close to the nominal confidence level (0.95) and obtained the shorter average lengths for almost all of the cases. However, when the proportion of zeros was high (i.e., $\omega = 0.9$), none of the methods performed well. In addition, the average lengths of all of the methods decreased for $n = 100$ compared to $n = 50$. Overall, the Wald confidence intervals did not perform well, which might have been caused by poor optimization that can sometimes occur. Thus, the optimization was not a good choice in this case. Similarly, Srisuradetchai et al. [20] found that when the Poisson parameter has a low value and the sample size is small, the Wald confidence interval for the Poisson parameter of a ZIP distribution was inferior to the other two intervals tested. Moreover, for a small sample size, the coverage probabilities of the Wald confidence intervals tended to decrease as the proportion of zeros was increased. Likewise, Daidoji et al. [33] showed that the Wald-type confidence interval for the Poisson parameter of a zero-truncated Poisson distribution performed unsatisfactorily because its coverage probability was below the nominal value when the Poisson mean and/or sample size was small. Hence, in the present study, estimations by using the Bayesian credible intervals and the HPD intervals were more accurate than the Wald confidence interval for all of the test settings.

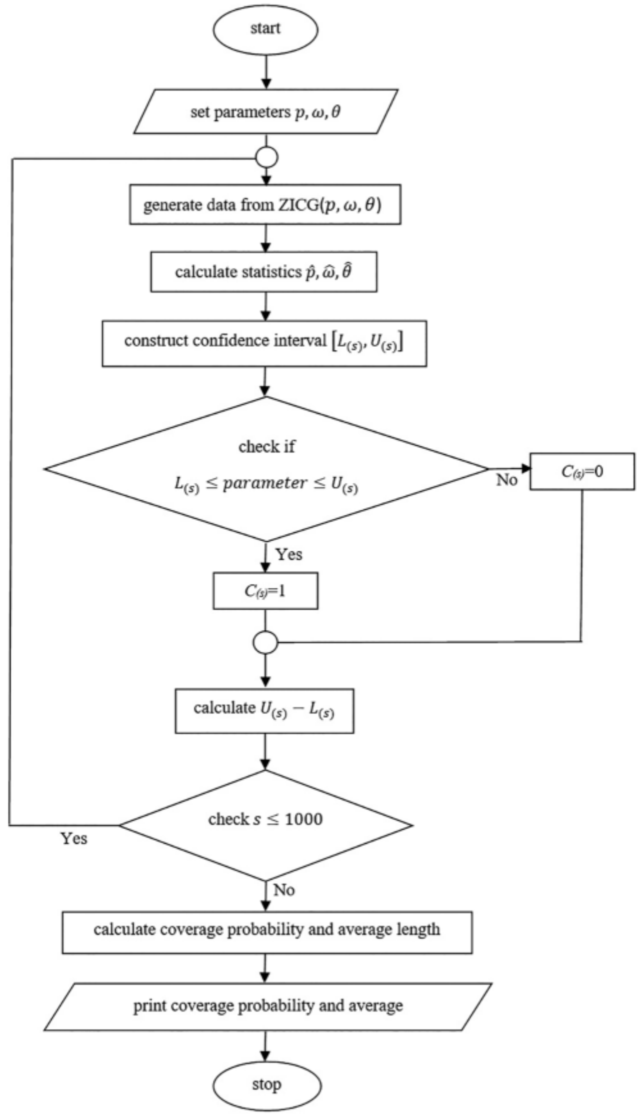

Figure 2: A flowchart of the simulation study

Table 2: The coverage probabilities and average lengths of the 95% confidence interval for parameters of the ZICG distribution

n	p	θ	ω	Method	CPs			ALs		
					ω	p	θ	ω	p	θ
50	0.5	1	0.1	Wald	0.527	0.809	0.026	114.231	47.931	2.743
				Bayesian	**0.989**	0.732	0.874	0.562	0.177	4.043
				HPD	**0.985**	0.748	0.946	0.508	0.168	3.928

(Continued)

Table 2 (continued)

n	p	θ	ω	Method	CPs			ALs		
					ω	p	θ	ω	p	θ
			0.5	Wald	0.5812	0.882	0.096	143.937	85.863	26.715
				Bayesian	**0.983**	0.875	0.866	0.752	0.283	4.020
				HPD	0.941	0.819	0.946	0.706	0.267	3.908
			0.9	Wald	0.754	0.925	0.637	82.840	59.022	9.714
				Bayesian	0.921	0.844	0.864	0.886	0.512	3.985
				HPD	0.855	0.65	0.897	0.843	0.469	3.836
50	0.9	1	0.1	Wald	0.544	0.556	0.031	15865.933	512.228	0.011
				Bayesian	**0.986**	**0.999**	0.878	0.2507	0.111	3.878
				HPD	**0.965**	**0.999**	**0.976**	0.226	0.111	3.722
			0.5	Wald	0.402	0.400	0.007	5550.402	483.188	0.019
				Bayesian	**0.973**	**0.997**	0.906	0.375	0.149	3.962
				HPD	**0.970**	**0.993**	**0.981**	0.371	0.148	3.863
			0.9	Wald	0.722	0.722	0.027	790.778	544.465	0.136
				Bayesian	**0.960**	**0.971**	0.886	0.291	0.289	3.951
				HPD	**0.976**	0.948	**0.956**	0.267	0.280	3.823
50	0.5	3	0.1	Wald	0.872	0.819	0.004	174.914	65.643	0.871
				Bayesian	**0.999**	0.673	**0.999**	0.369	0.163	4.646
				HPD	**0.999**	0.652	**0.999**	0.323	0.160	4.460
			0.5	Wald	**0.956**	0.911	0.024	59.560	20.094	1.538
				Bayesian	0.804	0.896	**0.999**	0.592	0.204	4.447
				HPD	0.645	0.850	**0.999**	0.537	0.195	4.300
			0.9	Wald	**0.952**	**0.954**	0.631	97.290	28.399	69.825
				Bayesian	0.634	0.642	**0.999**	0.889	0.385	4.075
				HPD	0.528	0.452	**0.999**	0.851	0.351	3.947
50	0.9	3	0.1	Wald	0.487	0.393	0.000	13654.005	502.619	0.014
				Bayesian	**0.985**	**0.997**	**0.999**	0.240	0.116	4.720
				HPD	**0.955**	**0.998**	**0.999**	0.215	0.115	4.525
			0.5	Wald	0.438	0.364	0.002	4698.163	463.454	0.027
				Bayesian	0.937	**0.963**	**0.999**	0.390	0.158	4.694
				HPD	0.921	**0.961**	**0.999**	0.383	0.157	4.518
			0.9	Wald	0.775	0.712	0.013	628.775	428.654	0.242
				Bayesian	0.889	0.815	**0.999**	0.411	0.318	4.060
				HPD	0.892	0.783	**0.999**	0.383	0.308	3.939
100	0.5	1	0.1	Wald	0.480	0.694	0.005	116.671	55.914	0.211
				Bayesian	**0.989**	0.765	0.807	0.465	0.134	3.935
				HPD	**0.984**	0.773	0.903	0.418	0.128	3.774
			0.5	Wald	0.485	0.778	0.016	90.840	67.565	1.003
				Bayesian	0.936	0.846	0.810	0.666	0.230	3.966

(Continued)

Table 2 (continued)

n	p	θ	ω	Method	CPs			ALs		
					ω	p	θ	ω	p	θ
				HPD	0.873	0.801	0.898	0.625	0.221	3.841
			0.9	Wald	0.707	0.767	0.397	351.284	6417.277	15.518
				Bayesian	0.926	0.821	0.841	0.756	0.484	3.991
				HPD	0.877	0.688	0.909	0.700	0.454	3.856
100	0.9	1	0.1	Wald	0.480	0.483	0.011	13083.192	408.086	0.006
				Bayesian	**0.965**	**0.998**	0.899	0.202	0.082	3.855
				HPD	0.927	**0.998**	**0.992**	0.187	0.082	3.679
			0.5	Wald	0.380	0.387	0.007	4091.424	341.966	0.011
				Bayesian	**0.952**	**0.999**	0.867	0.263	0.111	3.876
				HPD	**0.951**	**0.999**	**0.979**	0.260	0.111	3.716
			0.9	Wald	0.640	0.665	0.019	700.136	561.476	0.059
				Bayesian	0.947	**0.986**	0.908	0.159	0.216	3.985
				HPD	**0.960**	**0.985**	**0.955**	0.152	0.214	3.888
100	0.5	3	0.1	Wald	0.725	0.720	0.006	88.315	31.167	0.325
				Bayesian	**0.999**	0.885	**0.999**	0.286	0.152	4.706
				HPD	**0.996**	0.866	**0.999**	0.250	0.149	4.510
			0.5	Wald	0.895	0.830	0.008	52.324	19.127	1.030
				Bayesian	0.786	0.908	**0.999**	0.570	0.206	4.579
				HPD	0.643	0.865	**0.999**	0.526	0.199	4.404
			0.9	Wald	0.927	0.889	0.301	116.852	50.701	14.860
				Bayesian	0.568	0.525	**0.999**	0.848	0.349	4.130
				HPD	0.456	0.404	**0.999**	0.811	0.327	4.020
100	0.9	3	0.1	Wald	0.401	0.276	0.000	10599.191	376.330	0.008
				Bayesian	**0.959**	**0.999**	**0.999**	0.190	0.088	4.717
				HPD	0.908	**0.999**	**0.999**	0.175	0.087	4.513
			0.5	Wald	0.380	0.298	0.001	3484.650	334.218	0.012
				Bayesian	**0.956**	**0.999**	**0.999**	0.252	0.116	4.718
				HPD	**0.955**	**0.999**	**0.999**	0.250	0.115	4.519
			0.9	Wald	0.755	0.700	0.005	489.154	330.362	0.089
				Bayesian	0.927	0.901	**0.999**	0.207	0.237	4.346
				HPD	0.938	0.894	**0.999**	0.196	0.235	4.235

3.2 Applicability of the Methods When Using Real COVID-19 Data

Data for new daily COVID-19 cases during the Tokyo 2020 Olympic Games from 01 July 2021 to 12 August 2021 were used for this demonstration. The data are reported by the Tokyo Organizing Committee on the Government website (https://olympics.com/en/olympic-games/tokyo-2020) and they are shown in Table 3, with a histogram of the data provided in Fig. 3.

Table 3: The number of daily COVID-19-positive cases during the olympic games in Tokyo 2020

The number of COVID-19 positive case	0	1	2	3	4	6	7	8	9	10	12	15	16	17	18	19	21	22	24	26	27	28	29	31
Frequency	9	2	4	2	1	2	1	1	1	2	1	2	2	1	2	1	1	1	1	1	1	1	2	1

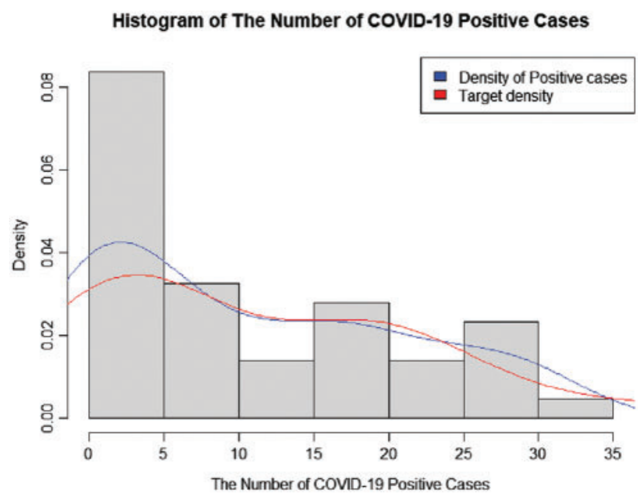

Figure 3: A histogram of the number of COVID-19-positive case during the olympic games in Tokyo 2020

3.2.1 Analysis of the COVID-19 Data

The information in Table 4 shows that the data are over-dispersed with an index of dispersion of 9.7149. The suitability of fitting the data to ZICG, ZIG, ZIP, ZINB, CG, geometric, Poisson, NB, and Gaussian distributions was assessed by using the AIC computed as $\text{AIC} = 2k - 2l\left(\hat{\Theta}\right)$ and the corrected AIC (AICc) computed as AICc $=$AIC $+2k(k+1)/(n-k-1)$ based on the log-likelihood function $\left(l\left(\hat{\Theta}\right)\right)$, where k is the number of parameters to fit. As can be seen in Table 5, the AIC and AICc values for ZICG were very similar (290.1166 and 290.7320) and the lowest recorded, thereby inferring that it provided the best fit for the data.

Table 4: Descriptive statistics

N	Mean	Variance	SD	Skewness	Kurtosis	ID
43	10.6744	103.7010	10.1834	0.5478	1.9347	9.7149

Table 5: Log-likelihood (l), AIC, and AICc values

Distribution	ZICG	ZIG	ZIP	ZINB	CG	Geometric	Poisson	NB	Gaussian
$-l$	142.0583	143.2602	217.0000	142.8000	146.3532	146.7716	302.7065	145.1130	160.3010
AIC	290.1166	290.5204	437.9509	291.6952	296.7065	295.5432	607.4131	294.2259	324.6019
AICc	290.7320	290.8204	438.2509	292.3106	297.0065	295.6408	607.5107	294.5259	324.9019

The 95% confidence intervals for the parameters of a ZICG distribution constructed by using the three estimation methods are provided in Table 6. The maximum likelihood estimates for parameters ω, p, and θ were 0.0743, 0.9238, and 0.4967, respectively. From the simulation results in Table 2 for $n = 50, \omega = 0.1, p = 0.9$, and $\theta = 1$, the HPD intervals provided a coverage probability greater than 0.95 for all of the parameters. Hence the HPD intervals are recommended for constructing the 95% confidence intervals for the parameters in this scenario.

Table 6: Estimation of the number of daily COVID-19-positive cases during the olympic games in Tokyo 2020

Method	ω estimation		p estimation		θ estimation	
	95% CI	Length of CI	95% CI	Length of CI	95% CI	Length of CI
Wald CI	(−0.0760, 0.3007)	0.2247	(0.8994, 0.9483)	0.0489	(0.4938, 0.4996)	0.0058
Bayesian CI	(0.0004, 0.2413)	0.2409	(0.8683, 0.9689)	0.1006	(0.0217, 4.8781)	4.8564
HPD interval	(0, 0.2111)	0.2111	(0.8667, 0.9667)	0.1000	(0.0153, 4.7192)	4.7038

4 Conclusions

We proposed a new mixture distribution called ZICG and presented its properties, namely the mgf, mean, variance, and Fisher information. According to the empirical study results, the ZICG distribution is suitable for over-dispersed count data containing excess zeros, such as occurred in the number of daily COVID-19-positive cases at the Tokyo 2020 Olympic Games. Confidence intervals for the three parameters of the ZICG distribution were constructed by using the Wald confidence interval, the Bayesian credible interval, and the HPD interval. Since the maximum likelihood estimates of the ZICG model parameters have no closed form, the Newton-Raphson method was applied to estimate the parameters and construct the Wald confidence intervals. Furthermore, Gibbs' sampling with the RWM algorithm was utilized in the Bayesian computation to approximate the parameters and construct the Bayesian credible intervals and the HPD intervals. Their performances were compared in terms of coverage probabilities and average lengths. According to the simulation results, the index of dispersion plays an important role: when it was small (e.g., $p = 0.5$), the Bayesian credible intervals and HPD intervals provided coverage probabilities greater than the nominal confidence level (0.95) in some cases whereas the Wald confidence interval did not perform at all well except for one case. Therefore, the Wald confidence interval approach is not recommended for constructing the confidence intervals for the ZICG parameters. Overall, the HPD interval approach is recommended for constructing the 95% confidence intervals for the parameters of a ZICG distribution since it provided coverage probabilities close to the nominal confidence level and the smallest average lengths in most cases. However, there are some cases where none of the methods performed well and so, in future research, other methods for estimating the parameters of a ZICG distribution will be investigated. For example, the prior part of the Bayesian computation should be further investigated to improve the efficiency of the Bayesian analysis. Furthermore, count data with more than one inflated value such as zeros-and-ones can occur, and so the zero-and-one inflated CG distribution could be interesting in this case.

Acknowledgement: The first author acknowledges the generous financial support from the Science Achievement Scholarship of Thailand (SAST).

Funding Statement: This research has received funding support from the National Science, Research and Innovation Fund (NSRF), and King Mongkut's University of Technology North Bangkok (Grant No. KMUTNB-FF-65-22).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Lee, C., Kim, S. (2017). Applicability of zero-inflated models to fit the torrential rainfall count data with extra zeros in South Korea. *Water, 9(2),* 123. DOI 10.3390/w9020123.
2. Böhning, D., Dietz, E., Schlattmann, P., Mendonca, L., Kirchner, U. (1999). The zero-inflated poisson model and the decayed, missing and filled teeth index in dental epidemiology. *Journal of the Royal Statistical Society: Series A (Statistics in Society), 162(2),* 195–209. DOI 10.1111/1467-985X.00130.
3. Ashburn, A., Fazakarley, L., Ballinger, C., Pickering, R., McLellan, L. D. et al. (2006). A randomised controlled trial of a home based exercise programme to reduce the risk of falling among people with Parkinson's disease. *Journal of Neurology, Neurosurgery & Psychiatry, 78(7),* 678–684. DOI 10.1136/jnnp.2006.099333.
4. Mullahy, J. (1986). Specification and testing of some modified count data models. *Journal of Econometrics, 33(3),* 341–365. DOI 10.1016/0304-4076(86)90002-3.
5. Yang, S., Harlow, L. L., Puggioni, G., Redding, C. A. (2017). A comparison of different methods of zero-inflated data analysis and an application in health surveys. *Journal of Modern Applied Statistical Methods, 16(1),* 518–543. DOI 10.22237/jmasm/1493598600.
6. Feng, C. X. (2021). A comparison of zero-inflated and hurdle models for modeling zero-inflated count data. *Journal of Statistical Distributions and Applications, 8(8),* 1–19.
7. Cameron, A. C., Trivedi, P. K. (2013). *Regression analysis of count data*. New York, USA: Cambridge University Press.
8. Böhning, D., Ogden, H. (2021). General flation models for count data. *Metrika, 84,* 245–261. DOI 10.1007/s00184-020-00786-y.
9. Lambert, D. (1992). Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics, 34(1),* 1–14. DOI 10.2307/1269547.
10. Ridout, M. S., Demetrio, C. G. B., Hinde, J. P. (1998). Models for counts data with many zeros. *The 19th International Biometric Conference*, Cape Town, South Africa.
11. Yusuf, O. B., Bello, T., Gureje, O. (2017). Zero inflated poisson and zero inflated negative binomial models with application to number of falls in the elderly. *Biostatistics and Biometrics Open Access Journal, 1(4),* 69–75.
12. Iwunor, C. C. (1995). Estimation of parameters of the inflated geometric distribution for rural out-migration. *Genus, 51,* 253–260.
13. Kusuma, R. D., Purwono, Y. (2019). Zero-inflated poisson regression analysis on frequency of health insurance claim pt. xyz. *12th International Conference on Business and Management Research*, Bali, Indonesia.
14. Chesneau, C., Bakouch, H., Hussain, T., Para, B. (2021). The cosine geometric distribution with count data modeling. *Journal of Applied Statistics, 48(1),* 124–137. DOI 10.1080/02664763.2019.1711364.
15. Das, S. K. (2019). Confidence interval is more informative than p-value in research. *International Journal of Engineering Applied Sciences and Technology, 4(6),* 278–282. DOI 10.33564/IJEAST.2019.v04i06.045.
16. Srisuradetchai, P., Junnumtuam, S. (2020). Wald confidence intervals for the parameter in a Bernoulli component of zero-inflated poisson and zero-altered poisson models with different link functions. *Science & Technology Asia, 25(2),* 1–14.

17. Waguespack, D., Krishnamoorthy, K., Lee, M. (2020). Tests and confidence intervals for the mean of a zero-inflated poisson distribution. *American Journal of Mathematical and Management Sciences, 39(4),* 383–390. DOI 10.1080/01966324.2020.1777914.
18. Srisuradetchai, P., Dangsupa, K. (2021). Profile-likelihood-based confidence intervals for the geometric parameter of the zero-inflated geometric distribution. *The Journal of KMUTNB, 31(3),* 527–538. DOI 10.14416/j.kmutnb.
19. Junnumtuam, S., Niwitpong, S. A., Niwitpong, S. (2021). The Bayesian confidence interval for coefficient of variation of zero-inflated poisson distribution with application to daily COVID-19 deaths in Thailand. *Emerging Science Journal, 5,* 62–76. DOI 10.28991/esj-2021-SPER-05.
20. Srisuradetchai, P., Tonprasongrat, K. (2022). On interval estimation of the poisson parameter in a zero-inflated poisson distribution. *Thailand Statistician, 20(2),* 357–371.
21. Cancho, V. G., Yiqi, B., Fiorucci, J. A., Barriga, G. D. C., Dey, D. K. (2017). Estimation and influence diagnostics for zero-inflated hyper-poisson regression model: Full Bayesian analysis. *Communications in Statistics–Theory and Methods, 47(11),* 2741–2759. DOI 10.1080/03610926.2017.1342839.
22. Workie, M. S., Azene, A. G. (2021). Bayesian zero-inflated regression model with application to under-five child mortality. *Journal of Big Data, 8(4),* 1–23.
23. Bilder, C. R., Loughin, T. M. (2014). *Analysis of categorical data with R*. Boca Raton, FL, USA: CRC Press.
24. Team, R. C. (2021). R: A language and environment for statistical computing. *R Foundation for Statistical Computing*, Vienna, Austria. https://www.R-project.org/.
25. Tanner, M. A., Wong, W. H. (1987). The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association, 82(398),* 528–540. DOI 10.1080/01621459.1987.10478458.
26. Hastings, W. K. (1970). Monte carlo sampling methods using markov chains and their applications. *Biometrika, 57(1),* 97–109. DOI 10.1093/biomet/57.1.97.
27. Metropolis, N., Ulam, S. (1949). The monte carlo method. *Journal of the American Statistical Association, 44(247),* 335–341. DOI 10.1080/01621459.1949.10483310.
28. Maima, D. H. M. (2015). *On the random walk metropolis algorithm (Master's Thesis)*. Islamic University of Gaza, Gaza.
29. Box, G. E. P., Tiao, G. C. (1992). *Bayesian inference in statistical analysis*. New York, USA: Wiley.
30. Chen, M. H., Shao, Q. M. (1999). Monte carlo estimation of Bayesian credible and HPD intervals. *Journal of Computational and Graphical Statistics, 8(1),* 69–92.
31. Meredith, M., Kruschke, J. (2020). HDInterval: Highest (Posterior) density intervals. https://CRAN.R-project.org/package=HDInterval. R package version 0.2.2.
32. Casella, G., Berger, R. L. (2002). *Statistical inference*. USA: Thomson Learning, Duxbury.
33. Daidoji, K., Iwasaki, M. (2012). On interval estimation of the poisson parameter in a zero-truncated poisson distribution. *Journal of the Japanese Society of Computational Statistics, 25(1),* 1–12. DOI 10.5183/jjscs.1103002_193.

Appendix A. R code for simulation study

```
rCGD<−function(n, p, theta){
X = rep(0, n)
for (j in 1:n) {
i = 0
#step 1: Generated Uniform(0, 1)
U = runif(1, 0, 1)
```

```
#step 2: Computed F(k∗ − 1) and F(k∗ )
cdf<−(2 ∗ (1 − p) ∗ (1 − 2 ∗ p ∗ cos(2 ∗ theta) + p^2))
/(2 + p ∗ ((p − 3) ∗ cos(2 ∗ theta) + p − 1))
while(U >= cdf)
{i = i + 1;
cdf = cdf + f(i, p, theta) }
X[j] = i;
}
return(X)
}
C<−function(p, theta) {
(2 ∗ (1 − p) ∗ (1 − 2 ∗ p ∗ cos(2 ∗ theta) + p^2))/(2 + p ∗ ((p − 3) ∗ cos(2 ∗ theta) + p − 1))}
#Calculating log likelihood of CG
loglikeCG<−function(x, y) {
p<−x[1]
theta<−x[2]
loglike<− n ∗ log(2) + n ∗ log(1 − p)+
n ∗ log(1 − 2 ∗ p ∗ cos(2 ∗ theta) + p^2)−
n ∗ log(2 + p ∗ ((p − 3) ∗ cos(2 ∗ theta) + p − 1))+
log(p) ∗ sum(y) + 2 ∗ sum(log(cos(y ∗ theta)))
# note use of sum
loglike<− −loglike
}
#Calculating the log-likelihood for ZICG
loglikeZICG<−function(x, y) {
w<−x[1]
p<−x[2]
theta<−x[3]
n0<−length(y[which(y == 0)])
ypos<−y[which(y != 0)] #y > 0
loglike<− n0 ∗ log(w + (1 − w) ∗ C(p, theta))
+ sum(log((1 − w) ∗ C(p, theta) ∗
(p^ypos) ∗ cos(ypos ∗ theta)^2))
loglike<− −loglike
}
# Bayesian Confidence Interval
```

```
gibbs<−function(y = y, sample.size, n, p, theta, w){
nonzero_values = y[which(y != 0)]
m<−n−length(nonzero_values)
ypos = y[which(y != 0)]
prob.temp = numeric()
S.temp = numeric()
theta.temp = numeric(sample.size)
w.temp = numeric(sample.size)
p.temp = numeric(sample.size)
p.temp[1]<−0.5 #initial p value
theta.temp[1]<−0.5 #initial theta
w.temp[1]<−0.5 #initial w value
prob.temp[1]<−w.temp[1]/(w.temp[1] + (1−w.temp[1]) ∗ (2 ∗ (1 − p.temp[1])
∗ 1 − 2 ∗ p.temp[1] ∗ cos(2 ∗ theta.temp[1]) + p.temp[1]^2))/(2 + p.temp[1] ∗ ((p.temp[1] − 3)
∗ cos(2 ∗ theta.temp[1]) + p.temp[1] − 1)))
S.temp[1]<−rbinom(1, m, prob.temp[1])
w.temp[1]<−rbeta(1,S.temp[1] + 0.5, n−S.temp[1] + 0.5)
p.samp<−GenerateMCMC.p(y = y, N = 1000, n = n, m = m, w = w.temp[1], theta = theta.temp[1],
sigma = 1)
p.temp[1]<−mean(p.samp[501:1000])
theta.samp<−GenerateMCMCsample(ypos = ypos,N = 1000, n = n, m = m, w = w.temp[1], p = p.temp[1],
sigma = 1)
theta.temp[1]<−mean(theta.samp[501:1000])
for(i in 2:(sample.size)){
prob.temp[i]<−w.temp[i − 1]/(w.temp[i − 1] + (1−w.temp[i − 1]) ∗ C(p.temp[i − 1],
theta.temp[i − 1]))
S.temp[i]<−rbinom(1, m, prob.temp[i])
w.temp[i]<−rbeta(1,S.temp[i] + 0.5
, n−S.temp[i] + 0.5)
p.samp<−GenerateMCMC.p(y = y, N = 1000, n = n, m = m, w = w.temp[i],
theta = theta.temp[i − 1], sigma = 1)
p.temp[i]<−mean(p.samp[501:1000])
theta.samp<−GenerateMCMCsample(ypos = ypos,N = 1000, n = n, m = m, w = w.temp[i],
p = p.temp[i], sigma = 1)
theta.temp[i]<−mean(theta.samp[501:1000])}
```

```
return(cbind(w.temp, p.temp, theta.temp))}
#random walk Metropolis for sample theta
GenerateMCMCsample<−function(ypos,N, n, m, w, p, sigma){
prior <− function(x) dgamma(x, shape = 2, rate = 1/3)
theta.samp <− numeric(length = N)
theta.samp[1] <− runif(n = 1, min = 0, max = 5)
for (j in 2:N) {
Yj<−theta.samp[j − 1] + rnorm(n = 1, mean = 0, sigma)
if(0< = Yj&&Yj< = pi/2){
alpha.cri <−((w + (1 − w) ∗ C(p,Yj))^m ∗ ((1 − w) ∗ (2 ∗ (1 − p) ∗ (1 − 2 ∗ p ∗ cos(2 ∗ Yj) + p^2))/
(2 + p ∗ ((p − 3) ∗ cos(2 ∗ Yj) + p − 1)))^(n − m) ∗ prod((cos(ypos ∗ Yj))^2) ∗ prior(Yj))/
((w + (1 − w) ∗ C(p, theta.samp[j − 1]))^m ∗ ((1 − w) ∗ (2 ∗ (1 − p) ∗
(1 − 2 ∗ p ∗ cos(2 ∗ theta.samp[j − 1]) + p^2))/
(2 + p ∗ ((p − 3) ∗ cos(2 ∗ theta.samp[j − 1]) + p − 1)))^(n − m) ∗
prod((cos(ypos ∗ theta.samp[j − 1]))^2) ∗ prior(theta.samp[j − 1]))
}else{alpha.cri<− 0}
U <− runif(1)
if(is.na(alpha.cri)){theta.samp[j]<− theta.samp[j − 1]} else {
if (U < min(alpha.cri, 1))
{theta.samp[j]<− Yj
}else theta.samp[j]<− theta.samp[j − 1]}
} return(theta.samp)}
##Random Walk Metropolis sampler for p
GenerateMCMC.p <− function(y,N, n, m, w, theta, sigma){
prior<−function(x)dbeta(x, shape1 = 2,
shape2 = 5)
p.samp <− numeric(length = N)
p.samp[1] <− runif(n = 1, min = 0, max = 1)
for (j in 2:N) {
Yj<−p.samp[j − 1] + rnorm(n = 1, mean = 0, sigma)
if(0< = Yj&&Yj< = 1){
alpha.cri <−((w + (1 − w) ∗ C(Yj, theta))^m ∗ ((1 − w) ∗ C(Yj, theta))^(n − m) ∗ Yj^(sum(y)))
∗ prior(Yj)/((w + (1 − w) ∗ C(p.samp[j − 1], theta))^m ∗
((1 − w) ∗ C(p.samp[j − 1], theta))^(n − m) ∗ p.samp[j − 1]^(sum(y))) ∗ prior(p.samp[j − 1])
}else{alpha.cri<− 0 }
U <− runif(1)
```

```
if(is.na(alpha.cri)){
p.samp[j]<− p.samp[j − 1]}
else {if (U < min(alpha.cri, 1)) {
p.samp[j]<− Yj
}else p.samp[j]<− p.samp[j − 1]} }
return(p.samp)
}
i = 0
while(i<M){
suscept<−rbinom(n, size = 1, prob = 1 − w)
count<−rCGD(n , p, theta)
y = suscept ∗ count
if(max(y) == 0){
next
}i = i + 1
p_hat<−egeom(y, method = "mle")
p0 = p_hat$parameters;theta0 = 0.5;
# store starting values
intvalues1 = c(p0, theta0)
resultCG<−nlm(loglikeCG, intvalues1 , y, hessian = TRUE, print.level = 1)
mleCG<−resultCG$estimate
mleCG.p <− c(mleCG[1])
mleCG.theta<−c(mleCG[2])
#formula for estimating w (omega)
n0<−length(y[which(y == 0)])
p1 = mleCG.p ; theta1 = mleCG.theta;
c_hat<−C(p1, theta1)
w1<−(n0 − n ∗ c_hat)/(n ∗ (1 − c_hat))
intvalues2 = c(w1, p1, theta1)
resultZICG<−nlm(loglikeZICG, intvalues2, y, hessian=TRUE, print.level = 1)
mleZICG<−resultZICG$estimate
hess<−resultZICG$hessian
cov<−solve(hess, tol = NULL)
stderr<−sqrt(diag(cov))
mle.w <− c(mleZICG[1])
mle.p <− c(mleZICG[2])
```

```
mle.theta<−c(mleZICG[3])
sd.w <− stderr[1]
sd.p <− stderr[2]
sd.theta <−stderr[3]
#Wald confidence interval for w
#lower bound of Wald CI for w
Wald.w.L[i]<−mle.w − qnorm(1 − alpha/2) ∗ sd.w
#Upper bound of Wald CI for w
Wald.w.U[i]<−mle.w + qnorm(1 − alpha/2) ∗ sd.w
Wald.CI.w = rbind(c(Wald.w.L[i],Wald.w.U[i]))
Wald.CP.w[i] = ifelse(Wald.w.L[i]<w&&w<Wald.w.U[i], 1, 0)
Wald.Length.w[i] = Wald.w.U[i] − Wald.w.L[i]
#Wald confidence interval for p
#lower bound of Wald CI for p
Wald.p.L[i]<−mle.p − qnorm(1 − alpha/2) ∗ sd.p
#Upper bound of Wald CI for p
Wald.p.U[i]<−mle.p + qnorm(1 − alpha/2) ∗ sd.p
Wald.CI.p = rbind(c(Wald.p.L[i],Wald.p.U[i]))
Wald.CP.p[i] = ifelse(Wald.p.L[i]<p&&<Wald.p.U[i], 1, 0)
Wald.Length.p[i] = Wald.p.U[i] − Wald.p.L[i]
#Wald confidence interval for theta
#lower bound of Wald CI for theta
Wald.theta.L[i]<−mle.theta − qnorm(1 − alpha/2) ∗ sd.theta
#Upper bound of Wald CI for theta
Wald.theta.U[i]<−mle.theta + qnorm(1 − alpha/2) ∗ sd.theta
Wald.CI.theta = rbind(c( Wald.theta.L[i],Wald.theta.U[i]))
Wald.CP.theta[i] = ifelse(Wald.theta.L[i]<theta&&theta<Wald.theta.U[i], 1, 0)
Wald.Length.theta[i] = Wald.theta.U[i] − Wald.theta.L[i]
#########End Wald CI#############
test<−gibbs(y = y, sample.size = sample.size,
n = n, p = p, theta = theta, w = w)
#burn-in w estimator
w.mcmc<−test[, 1][1001:3000]
#estimator of w
w.bayes<−mean(w.mcmc)
#burn-in p estimator
```

```
p.mcmc<-test[, 2][1001:3000]
#estimator of p
p.bayes<-mean(p.mcmc)
#burn-in theta estimator
theta.mcmc<-test[, 3][1001:3000]
#estimator of theta
theta.bayes<-mean(theta.mcmc)
#########Construct Bayesian confidence interval########
L.w[i] = quantile(w.mcmc, alpha/2, na.rm = TRUE)
U.w[i] = quantile(w.mcmc,(1 - alpha/2), na.rm = TRUE)
CIr1 = rbind(c(L.w[i],U.w[i]))
Bayes.CP.w[i] = ifelse(L.w[i]<w&&w<U.w[i], 1, 0)
Bayes.Length.w[i] = U.w[i] - L.w[i]
L.p[i] = quantile(p.mcmc, alpha/2, na.rm = TRUE)
U.p[i] = quantile(p.mcmc,(1 - alpha/2), na.rm = TRUE)
CIr2 = rbind(c(L.p[i],U.p[i]))
Bayes.CP.p[i] = ifelse(L.p[i]<p&&<U.p[i], 1, 0)
Bayes.Length.p[i] = U.p[i] - L.p[i]
L.the[i] = quantile(theta.mcmc, alpha/2, na.rm = TRUE)
U.the[i] = quantile(theta.mcmc,(1 - alpha/2), na.rm = TRUE)
CIr3 = rbind(c(L.the[i],U.the[i]))
Bayes.CP.the[i] = ifelse(L.the[i]<theta&&theta<U.the[i], 1, 0)
Bayes.Length.the[i] = U.the[i] - L.the[i]
#########Construct HPD interval########
w.hpd = hdi(w.mcmc, 0.95)
L.w.hpd[i] = w.hpd[1]
U.w.hpd[i] = w.hpd[2]
CIr4 = rbind(c(L.w.hpd[i],U.w.hpd[i]))
CP.w.hpd[i] = ifelse(L.w.hpd[i]<w&&w<U.w.hpd[i], 1, 0)
Length.w.hpd[i] = U.w.hpd[i] - L.w.hpd[i]
p.hpd = hdi(p.mcmc, 0.95)
L.p.hpd[i] = p.hpd[1]
U.p.hpd[i] = p.hpd[2]
CIr5 = rbind(c(L.p.hpd[i],U.p.hpd[i]))
CP.p.hpd[i] = ifelse(L.p.hpd[i]<p&&<U.p.hpd[i], 1, 0)
Length.p.hpd[i] = U.p.hpd[i] - L.p.hpd[i]
```

```
theta.hpd = hdi(theta.mcmc, 0.95)
L.the.hpd[i] = theta.hpd[1]
U.the.hpd[i] = theta.hpd[2]
CIr6 = rbind(c(L.the.hpd[i],U.the.hpd[i]))
CP.the.hpd[i] = ifelse(L.the.hpd[i]<theta&&theta<U.the.hpd[i], 1, 0)
Length.the.hpd[i] = U.the.hpd[i] – L.the.hpd[i]
}
```