Revista Colombiana de Estadística - Applied Statistics
July 2025, volume 48, issue 2, pp. 139–161
http://doi.org/10.15446/rce.v48n2.117153

MCMC Methods for Sample Generation from a New Bivariate Distribution

Métodos MCMC para generar muestras de una nueva distribución bivariada

Llerzy E. Torres Ome[1,a], Jose R. Tovar Cuevas[2,b], Paula A. Bran Cardona[3,c]

[1]Department of Mathematics, Faculty of Natural and Exact Sciences., Universidad del Valle, Cali, Colombia

[2]School of Statistics, Faculty of Engineering, Universidad del Valle, Cali, Colombia

[3]Department of Mathematics, Faculty of Natural and Exact Sciences, Universidad del Valle, Cali, Colombia

Abstract

This article introduces a novel bivariate probability distribution derived through a transformation-based approach, along with the closed-form expression of its l-th order joint moment. Although the distribution may be employed as a prior for the shape parameters of the Beta distribution, the main focus of this work lies in evaluating the convergence behavior of Markov Chain Monte Carlo (MCMC) algorithms designed to generate samples from this new distribution. A simulation strategy is analyzed, consisting of a Gibbs sampling scheme in which an adaptive random walk Metropolis–Hastings (ARWMH) algorithm is used to sample from one of the full conditional distributions, employing a four-parameter Beta distribution as the proposal. Convergence is assessed using diagnostics such as the effective sample size (ESS) and the potential scale reduction factor (R-hat). The results show that when the elements of the parameter vector ϕ differ, the empirical moments obtained from the chains approximate the theoretical values accurately. However, when all components of ϕ are equal, the estimates of variance and covariance deviate considerably, revealing a sensitivity to symmetry in the geometry of the new distribution. A brief application in a Bayesian context is also presented, in which the new distribution is used as a prior and the Beta distribution as the likelihood. This application confirms that the proposed sampling methods yield empirical moments that

[a]Ph.D(c). E-mail: llerzy.torres@correounivalle.edu.co
[b]Ph.D. E-mail: jose.r.tovar@correounivalle.edu.co
[c]Ph.D. E-mail: paula.bran@correounivalle.edu.co

are consistent with the theoretical ones, thus supporting the validity of the strategy. The contributions of this study are relevant to the design, evaluation, and implementation of MCMC techniques for sampling from complex distributions in Bayesian inference.

***Key words*:** Adaptive random walk; Bivariate probability distribution; Convergence diagnostics; Gibbs sampling; MCMC; Metropolis-Hastings.

Resumen

Este artículo presenta una nueva distribución de probabilidad bivariada construida mediante una estrategia basada en transformaciones, junto con la expresión cerrada de su momento conjunto de orden l. Aunque esta distribución puede utilizarse como distribución a priori para los parámetros de forma de la distribución Beta, el enfoque principal del trabajo se centra en evaluar el comportamiento de convergencia de algoritmos Monte Carlo Cadenas de Markov (MCMC) diseñados para generar muestras aleatorias de dicha distribución. Se analiza una estrategia de simulación que consiste en un esquema de muestreo de Gibbs, en el cual se emplea un algoritmo Metropolis–Hastings con caminata aleatoria adaptativa (MHCAA) para generar muestras de una de las distribuciones condicionales completas, utilizando como propuesta una distribución Beta de cuatro parámetros. La convergencia se evalúa mediante diagnósticos como el tamaño efectivo de muestra (ESS) y el factor de reducción de escala potencial (R-hat). Los resultados muestran que, cuando los componentes del vector de parámetros ϕ son diferentes, los momentos empíricos obtenidos aproximan adecuadamente los valores teóricos. En contraste, cuando todos los componentes de ϕ son iguales, se evidencian desviaciones significativas en las estimaciones de la varianza y la covarianza, lo que sugiere sensibilidad ante configuraciones simétricas de la nueva distribución. Se presenta además una aplicación breve en el contexto bayesiano, usando la nueva distribución como a priori y la distribución Beta como verosimilitud, confirmando que las muestras generadas reproducen momentos a priori empíricos coherentes con los valores teóricos. Las contribuciones del estudio son relevantes para el diseño, evaluación e implementación de técnicas MCMC en contextos de inferencia bayesiana con distribuciones complejas.

***Palabras clave*:** Caminata aleatoria adaptativa; Diagnósticos de convergencia; Distribución de probabilidad bivariada; Gibbs sampling; MCMC; Metropolis-Hastings.

1. Introduction

The development of new probability distributions remains a fundamental area of research in statistics due to their ability to model complex random phenomena and adapt to specific contexts. In recent years, the literature has shown increasing interest in extending or modifying existing distributions to enhance their flexibility and applicability. For example, Mandouh & Muhammed (2024) introduced a unit bivariate Marshall-Olkin distribution to model dependencies between variables with predefined marginal distributions. Martínez-Flórez et al. (2023) explored a

multivariate version of the Unit-Sinh-Normal distribution, while Althubyani et al. (2022) proposed a three-parameter univariate Beta distribution derived from an exponential transformation of the modified Sarhan-Zaindin Weibull-Beta distribution. Similarly, Bran-Cardona et al. (2011) presented a bivariate generalization of the Kummer-Beta distribution, highlighting the use of special functions such as the confluent hypergeometric function. In a related development, Moschen & Carvalho (2023) investigated parameter estimation for various bivariate Beta distributions, offering an updated review of their features and limitations.

Parallel to the advancement of new distributions, the construction of efficient algorithms for random sample generation remains an active area of research, particularly in scenarios where the joint density function lacks a closed form or is analytically intractable. Modeling the shape parameters of widely used distributions, such as the Beta or Weibull, often requires bivariate distributions capable of capturing the relationship between these parameters while allowing for the incorporation of external information—an essential aspect of subjective Bayesian inference. However, existing approaches, such as those by Kaminskiy & Krivtsov (2005), Ma & Leijon (2010) and Ma & Leijon (2011), have primarily addressed these issues through numerical approximations or non-subjective frameworks. In the context of posterior simulation for models with Beta likelihood and Gamma product priors, Bouguila et al. (2006) employed a Metropolis-Hastings scheme with a random walk, using a product of log-normal distributions as the proposal.

This study introduces a novel bivariate distribution for the random vector (Y_1, Y_2), which is associated with the shape parameters of the Beta distribution—a model widely used for bounded phenomena such as the proportion of defective items, vegetation cover percentage, disease severity indices, and educational test scores, among others. The proposed distribution is constructed through a bijective and continuously differentiable transformation that maps (Y_1, Y_2) to an auxiliary vector (X_1, X_2). The joint density of (X_1, X_2) is defined as the product of a standard Beta marginal density, $f_{X_1}(x_1|\phi_1 = (a, b))$, and a four-parameter conditional Beta density, $f_{X_2|X_1}(x_2|x_1, \phi_2 = (0, U(x_1), c, d))$, where the upper bound of X_2, $U(x_1)$, depends on the value of X_1. This structure provides a flexible framework for modeling the relationship between the shape parameters of the Beta distribution, following the construction methodology proposed by Arroyo et al. (2022), and facilitates the specification of informative prior distributions within a Bayesian context.

Since the resulting joint density lacks a closed-form expression for direct sampling, this work proposes a simulation scheme based on Markov Chain Monte Carlo (MCMC) methods, which are widely recognized for their ability to approximate complex distributions through iterative techniques (Robert & Casella, 1999, 2010). The procedure consists of two stages. In the first stage, two MCMC algorithms operate in conjunction within each iteration: Gibbs sampling is used to construct a Markov chain for the vector (X_1, X_2) based on its full conditional distributions, and within this scheme, an adaptive random walk Metropolis-Hastings (ARWMH) algorithm is employed to sample from the conditional distribution $f_{X_1|X_2}(x_1|x_2, \phi)$, for which direct sampling is not feasible. In the second stage, the inverse transformation is applied to the generated chain to obtain samples from the transformed vector (Y_1, Y_2).

The convergence of the generated chains is assessed using different diagnostic criteria, including comparisons between theoretical densities and empirical histograms, trace plots, cumulative averages (with both random and fixed seeds), autocorrelation analysis, heatmaps to assess the appropriate precision parameter in the ARWMH algorithm, the R-hat statistic, effective sample size, contour plots of the generated chains, and the comparison of numerical moments with their theoretical counterparts.

This paper is structured as follows: Section 2 introduces the novel bivariate distribution and provides the closed-form expression for its joint moments. Section 3 details the MCMC algorithms developed to generate random samples from the vector (Y_1, Y_2). Section 4 focuses on the convergence diagnostics of the proposed simulation methods. Section 5 illustrates a brief application within a Bayesian framework. Finally, Section 6 presents the main conclusions and outlines directions for future research.

2. Novel Distribution

Let (X_1, X_2) be a random vector with $X_1 \sim Beta(a, b)$ and $X_2|X_1 = x_1 \sim Beta(0, U(x_1), c, d)$, where $U(x_1) = x_1(1 - x_1)$. The joint probability density function (PDF) of the vector (X_1, X_2) is characterized by four shape parameters $\phi = (a, b, c, d) \in \mathbb{R}^4_+$ and the density function

$$f_{X_1,X_2}(x_1, x_2|\phi) = \frac{1}{\mathrm{B}(a, b) \cdot \mathrm{B}(c, d)} \frac{x_2^{c-1}[x_1(1 - x_1) - x_2]^{d-1}}{x_1^{c+d-a}(1 - x_1)^{c+d-b}}, \tag{1}$$

for $x_1 \in (0, 1)$ and $x_2 \in (0, U(x_1))$. By applying the Random Vector Transformation Theorem with the PDF (1) and the transformation

$$(Y_1, Y_2) := \left(X_1 \left(\frac{X_1(1 - X_1)}{X_2} - 1 \right), (1 - X_1) \left(\frac{X_1(1 - X_1)}{X_2} - 1 \right) \right), \tag{2}$$

we obtain the joint PDF for (Y_1, Y_2),

$$f_{Y_1,Y_2}(y_1, y_2|\phi) = k^{-1}(\phi)\ y_1^{a-1} y_2^{b-1} (y_1 + y_2)^{d-(a+b)} (y_1 + y_2 + 1)^{-c-d},\ y_1, y_2 \in \mathbb{R}_+, \tag{3}$$

where $k^{-1}(\phi)$ is the normalization constant with $k(\phi) = \mathrm{B}(a, b) \cdot \mathrm{B}(c, d)$. The joint moment of order l of the PDF (3) is presented in the following theorem.

Theorem 1. *Let (Y_1, Y_2) be a random vector with joint PDF given by (3). The joint moment of order $l = l_1 + l_2$ is*

$$E[Y_1^{l_1} Y_2^{l_2}|\phi] = k^{-1}(\phi) \cdot \mathrm{B}(c - l, l + d) \cdot \mathrm{B}(l_1 + a, l_2 + b),$$

where $k^{-1}(\phi)$ is a constant independent of l_1 and l_2, $\mathrm{B}(\cdot, \cdot)$ denotes the Beta function, and the result holds provided that $c > l$ to ensure absolute convergence.

Proof. Consider the joint moment of (Y_1, Y_2):

$$E[Y_1^{l_1} Y_2^{l_2}|\phi] = \int_0^\infty \int_0^\infty y_1^{l_1} y_2^{l_2} f_{Y_1,Y_2}(y_1, y_2|\phi)\, dy_1\, dy_2.$$

Substituting the joint density (3) yields:

$$E[Y_1^{l_1}Y_2^{l_2}|\phi] \propto \int_0^\infty \int_0^\infty y_1^{l_1+a-1} y_2^{l_2+b-1}(y_1+y_2)^{d-(a+b)}(y_1+y_2+1)^{-c-d}\,dy_1\,dy_2.$$

Apply the change of variables $u = y_1 + y_2$ and $v = y_1/y_2$, which leads to:

$$E[Y_1^{l_1}Y_2^{l_2}|\phi] \propto \int_0^\infty \int_0^\infty u^{l+d-1}(u+1)^{-c-d} \cdot \frac{v^{l_1+a-1}}{(1+v)^{l+a+b}}\,du\,dv.$$

Separating the integrals:

$$E[Y_1^{l_1}Y_2^{l_2}|\phi] \propto \left(\int_0^\infty u^{l+d-1}(u+1)^{-c-d}\,du\right)\left(\int_0^\infty v^{l_1+a-1}(1+v)^{-l-a-b}\,dv\right).$$

Both integrals are Mellin transforms of the form:

$$\mathcal{M}[(1+w)^{-\delta}; s] = \int_0^\infty w^{s-1}(1+w)^{-\delta}\,dw = \frac{\Gamma(\delta-s)\Gamma(s)}{\Gamma(\delta)},$$

valid for $\delta > s > 0$. Applying this to each integral:

$$E[Y_1^{l_1}Y_2^{l_2}|\phi] \propto \frac{\Gamma(c-l)\Gamma(l+d)}{\Gamma(c+d)} \cdot \frac{\Gamma(l_2+b)\Gamma(l_1+a)}{\Gamma(l+a+b)},$$

which can be expressed in terms of Beta functions as:

$$E[Y_1^{l_1}Y_2^{l_2}|\phi] \propto \mathrm{B}(c-l, l+d) \cdot \mathrm{B}(l_1+a, l_2+b),$$

where the omitted constant of proportionality, $k^{-1}(\phi)$, is independent of l_1 and l_2. The condition $c > l$ ensures absolute convergence of the Gamma functions.

□

3. Proposed MCMC Algorithm

Let (X_1, X_2) be a random vector with joint PDF given by (1), and (Y_1, Y_2) another random vector with joint PDF (3). To generate random samples from (Y_1, Y_2), we propose a two-stage procedure. In the first stage, Gibbs sampling (GS) is employed to draw a sample $(x_1, x_2)_{(n)}$ from the distribution of (X_1, X_2). In the second stage, the transformation defined in (2) is applied to each element of this sample, yielding realizations from the target vector (Y_1, Y_2).

The implementation of the GS algorithm for $f_{X_1,X_2}(x_1, x_2|\phi)$ requires generating random samples from the full conditional distributions (FCDs) $f_{X_1|X_2}(x_1|x_2, \phi)$ and $f_{X_2|X_1}(x_2|x_1, \phi)$. Owing to the structure of the joint density (1), the FCD $f_{X_2|X_1}(x_2|x_1, \phi)$ corresponds to a four-parameter Beta distribution. This distribution is readily available in several statistical software packages, such as the '`rBeta.4P`' function from the '`betafunctions`' package in R.

In contrast, the FCD $f_{X_1|X_2}(x_1|x_2,\phi)$, as expressed non-normalized in (4), does not correspond to any standard probability distribution and poses analytical challenges for direct sampling. Consequently, we employ the ARWMH algorithm to simulate draws from this distribution within the GS framework.

$$f_{X_1|X_2}(x_1|x_2,\phi) \propto x_1^{a-c-d}(x_1(1-x_1)-x_2)^{d-1}(1-x_1)^{b-c-d}. \tag{4}$$

The ARWMH algorithm implemented in this study employs a four-parameter Beta distribution as the proposal distribution. The density for the candidate state at iteration $t+1$ is denoted by $q(x_1|x_1^{(t)}) = \text{dBeta.4P}(x_1, x_l, x_u, a', b')$, where x_l and x_u represent the lower and upper bounds (location parameters), and a' and b' are the shape parameters. These parameters are dynamically adjusted at each iteration based on the current state $x_1^{(t)}$ of the chain, the conditioning value of the variable $X_2 = x_2$, and a precision parameter φ. These quantities will be defined below.

The bounds x_l and x_u are determined as functions of the conditioning value $x_2 = v$. Specifically, recalling that the upper limit function $U(x_1) = x_1(1-x_1)$ must satisfy $U(x_1) > x_2$, the feasible range for x_1 is obtained by solving the inequality $x_1(1-x_1) > v$ with respect to x_1, yielding:

$$x_l := 0.5 - 0.5\sqrt{1-4v} < x_1 < 0.5 + 0.5\sqrt{1-4v} =: x_u. \tag{5}$$

This condition holds as long as $v \in [0, 0.25]$, since x_1 must remain within the interval $(0,1)$ and the constraint $x_2 < U(x_1)$ must be satisfied. To determine the shape parameters a' and b' of the proposal distribution, we assume that the sequence $x_1^{(1)}, x_1^{(2)}, \ldots, x_1^{(t-1)}, x_1^{(t)}$, generated for the conditional distribution $X_1|X_2$, forms a random walk. The proposal distribution q at iteration $t+1$ is designed to approximate this walk, with its expected value centered at the current state $x_1^{(t)}$ and a variance σ_1^2, which is a user-defined parameter controlling the algorithm's precision. By equating the theoretical mean and variance of the Beta proposal distribution q with these target values, the following system of equations is derived:

$$\begin{cases} E[X_1] = \dfrac{a'x_u + b'x_l}{a'+b'} & = x_1^{(t)} \\ V[X_1] = \dfrac{a'b'(x_u-x_l)^2}{(a'+b')^2(a'+b'+1)} & = \sigma_1^2 \end{cases},$$

whose solution is

$$\begin{cases} a' & = \dfrac{x_1^{(t)}-x_l}{x_u-x_l}\left(\dfrac{(x_1^{(t)}-x_l)(x_u-x_1^{(t)})}{\sigma_1^2}-1\right) \\ b' & = \dfrac{x_u-x_1^{(t)}}{x_u-x_l}\left(\dfrac{(x_1^{(t)}-x_l)(x_u-x_1^{(t)})}{\sigma_1^2}-1\right) \end{cases}. \tag{6}$$

By introducing the quantity $\varphi := ((x_1^{(t)}-x_l)(x_u-x_1^{(t)})/\sigma_1^2 - 1)$, referred to as the precision parameter, the shape parameters a' and b' of the proposal distribution simplify to $a' = \varphi \cdot \frac{x_1^{(t)}-x_l}{x_u-x_l}$ and $b' = \varphi \cdot \frac{x_u-x_1^{(t)}}{x_u-x_l}$, respectively. Higher values of

φ correspond to smaller variances σ_1^2, leading to tighter proposals, while lower values of φ result in broader proposals. An excessively large φ may restrict the chain's exploration of the state space, increasing the acceptance rate but limiting variability, whereas a small φ may result in frequent rejection of proposed states.

Considering these dynamics, Algorithm 1 outlines the ARWMH procedure for generating random samples from the FCD of X_1 given X_2, with non-normalized density as specified in (4). In the algorithm, the quantities $\mathbf{a_{current}}$, $\mathbf{b_{current}}$, $\mathbf{a_{proposal}}$, and $\mathbf{b_{proposal}}$ represent the shape parameters a' and b'. The subscript `current` denotes parameters evaluated at the current state $x^{(t)}$ of the chain using the expressions in (6), while `proposal` refers to parameters evaluated at the candidate value w for state $t+1$, substituting w into (6). The function $h(x, x_l, x_u, a', b')$ denotes the PDF of the proposal distribution q, evaluated at x, with shape parameters a' and b', and location parameters x_l and x_u. The execution of Algorithm 1 requires the following inputs:

1. N: Total number of samples to be generated prior to applying thinning and burn-in procedures.

2. *precision*: The initial value of the precision parameter φ, which controls the variability of the proposal distribution in the ARWMH algorithm. During the burn-in period, φ is adaptively updated every 100 iterations to guide the acceptance rate toward a specified target value. Specifically, the observed acceptance rate over the last 100 iterations is computed, and the discrepancy from the target acceptance rate is calculated as (`target_acceptance` − `current_acceptance`). This discrepancy is scaled by the factor $1/$`iteration`, and the resulting adjustment is added to the current value of φ. This adaptive mechanism follows a heuristic approach inspired by the Robbins–Monro stochastic approximation method, aiming to improve the efficiency and stability of the sampling process during the initial stages.

3. $\phi = (a, b, c, d)$: Parameter vector defining the target full conditional distribution $f_{X_1|X_2}(x_1|x_2, \phi)$. The parameters a and b are shape parameters associated with the marginal Beta distribution of X_1, while c and d are shape parameters governing the conditional distribution of X_2 given X_1. These parameters remain constant throughout the sampling process and characterize the structural properties of the target distribution.

4. v: Current value of the conditioning variable X_2, utilized in evaluating the full conditional distribution $f_{X_1|X_2}(x_1|X_2 = v, \phi)$ within both the ARWMH and GS steps.

5. x_{10}_given: Initial value of the Markov chain. If not provided by the user, the algorithm generates this value randomly using the '`rBeta.4P`' function from the `betafunctions` package in `R`. The parameters for this four-parameter Beta distribution are defined as follows: the location parameters x_l and x_u correspond to the domain specified in (5), and the shape parameters are set to match those of the marginal distribution of X_1, namely a and b.

Algorithm 1 Adaptive random walk Metropolis-Hastings for sampling from the full conditional distribution $X_1|X_2, \phi$

Input: N, *precision*, a, b, c, d, v, *thinning*, *burn-in*, x_{10}_given, *target_ acceptance*, *option*, *dig_ tol*

Output: Depending on *option*, returns either the last chain value and acceptance rate or the full chain, adjusted precision, and acceptance statistics.

Initialization: Compute the domain of X_1:

$$x_l = 0.5 - 0.5\sqrt{1-4v}, \quad x_u = 0.5 + 0.5\sqrt{1-4v}$$

if x_{10}_given is not provided **then**
    Generate x_{10} from `rBeta.4P(1,` x_l`,` x_u`,` a`,` b`)`.
end if

MCMC iterations:
for $t \leftarrow 1$ to N **do**
    Compute $\mathbf{a_{current}}$ and $\mathbf{b_{current}}$ from $x_1^{(t)}$ using φ, x_l, and x_u.
    Propose candidate $w \sim$ `rBeta.4P(1,` x_l`,` x_u`,` $\mathbf{a_{current}}$`,` $\mathbf{b_{current}}$`)`.
    Compute $\mathbf{a_{proposal}}$ and $\mathbf{b_{proposal}}$ based on w.
    if candidate w is valid (based on *dig_ tol*) **then**
        Compute acceptance probability:

$$\rho = \frac{f_{X_1|X_2}(w|v,\phi) \cdot h(x_1^{(t)}|x_l, x_u, \mathbf{a_{proposal}}, \mathbf{b_{proposal}})}{f_{X_1|X_2}(x_1^{(t)}|v,\phi) \cdot h(w|x_l, x_u, \mathbf{a_{current}}, \mathbf{b_{current}})}$$

        if `Uniform(0,1)` $< \rho$ **then**
            Accept proposal: $x_1^{(t+1)} = w$
        else
            Reject proposal: $x_1^{(t+1)} = x_1^{(t)}$
        end if
    end if
    if $t \leq$ *burn-in* and $t \mod 100 = 0$ and *target_ acceptance* $\neq 0$ **then**
        Adjust φ based on the acceptance rate over the last 100 iterations.
    end if
end for

Post-processing: Apply *burn-in* and *thinning*.
if *option* = `'end'` **then**
    return Last chain state and overall acceptance rate.
else
    return Full chain, adjusted φ, global acceptance rate, and post-burn-in acceptance rate.
end if

6. *thinning*: Sampling interval, specifying the number of iterations to skip between stored samples. This procedure reduces autocorrelation within the Markov chain and improves the quality of the collected samples.

7. *burn-in*: Number of initial iterations to discard from the generated chain to minimize the influence of the initial state x_{10}_given. This value must be greater than zero and less than the total number of iterations N.

8. *target_acceptance*: Desired acceptance rate for the proposal distribution, used to guide the adaptive adjustment of the *precision* parameter during the burn-in period, as described in the definition of φ.

9. *option*: Specifies the type of output returned by the algorithm. The value '`end`' returns only the final state of the chain, while '`all`' returns the complete chain, the final adjusted precision, the overall acceptance rate, and the post-burn-in acceptance rate.

10. *dig_tol*: Number of decimal places used to control numerical precision and prevent indeterminacy in the evaluation of terms involved in the acceptance probability ρ. Specifically, it ensures that the expressions $-x_{10}^2+x_{10}-v$ and $-w^2+w-v$, which appear in the non-normalized conditional density (4), are appropriately distinguished from zero.

Continuing with the first stage, Algorithm 2 implements the GS procedure to generate N_1 realizations of the random vector (X_1, X_2), whose joint PDF is given by (1). At each iteration $t+1$, the value of X_1 is updated using the last state from a chain of size N_2, generated via Algorithm 1, with the conditioning variable set to $v = x_2^{(t)}$. All remaining inputs in Algorithm 2 are consistent with those used in Algorithm 1.

Finally, in the second stage, the transformation defined in (2) is applied to the sample $(x_1, x_2)_{(n)}$, generated by Algorithms 1-2, to obtain a random sample $(y_1, y_2)_{(n)}$ corresponding to the random vector (Y_1, Y_2) with joint PDF (3). The sample size n corresponds to the number of retained iterations, computed as $n = (N_1 - burn - in)/thinning$.

4. Convergence Monitoring

This section evaluates the convergence of the Markov chains generated by Algorithms 1-2. The results presented here were obtained using code developed in the R software environment[1]. Throughout this section, the parameter values specified in (7) for the joint PDF (3) are used, although the methodology is fully replicable with different parameter configurations. It is important to note that, according to Theorem 1, the existence of joint moments of order l is guaranteed provided that $c > l$.

$$a = 2.2, \quad b = 2.2, \quad c = 2.2, \quad d = 2.2. \tag{7}$$

[1] It is available for consultation and download at the following GitHub repository: code link

Algorithm 2 Gibbs sampling for the random vector (X_1, X_2)

1: **Input:** N_1, N_2, *precision*, a, b, c, d, *thinning*, *burn-in*, x_{10}_given, *target_ acceptance*, *option*, *dig_ tol*
2: **Output:** Generated chain for the random vector (X_1, X_2)

3: **Initialization:**
4: Generate initial values:

$$x_1^{(1)} = \texttt{rbeta}(1, a, b), \quad x_2^{(1)} = \texttt{rBeta.4P}(1, 0, x_1^{(1)}(1 - x_1^{(1)}), c, d)$$

5: **Gibbs sampling iterations:**
6: **for** $t \leftarrow 1$ to N_1 **do**
7: Update $x_1^{(t+1)}$ using Algorithm 1 (ARWMH), setting $N = N_2$, $v = x_2^{(t)}$, *option* = '`end`', and other parameters as defined.
8: Update $x_2^{(t+1)}$ by drawing:

$$x_2^{(t+1)} = \texttt{rBeta.4P}(1, 0, x_1^{(t+1)}(1 - x_1^{(t+1)}), c, d)$$

9: **end for**

10: **Output:** Return the generated chain for the random vector (X_1, X_2).

4.1. Convergence of Algorithm 1

The initial focus is placed on Algorithm 1, which is designed to generate random samples from the non-normalized FCD $f_{X_1|X_2}(x_1|x_2 = v, \phi)$. Figure 1 illustrates the shape of this conditional density for three different values of the conditioning variable $X_2 = v$. For values of v close to zero, the density exhibits a bimodal structure with an antimode near $x_1 = 0.5$. In contrast, as v approaches 0.25, the domain of $X_1|X_2 = v$ becomes narrower due to the boundary conditions established in (5). This behavior reflects the dependence of the conditional distribution's support and shape on the value of the conditioning variable X_2.

A Markov chain was generated using Algorithm 1 with the following simulation scheme:

$$N = 10^5, \quad \varphi = 3, \quad v = 0.05, \quad thinning = 25, \quad burn - in = 5 * 10^3, \quad target\_acceptance = 0.4, \quad (8)$$

which produced a post burn-in sample of size $n = 3800$. The kernel density estimate of this sample is displayed in Figure 2, generated using the '`density`' and '`plot`' functions in R. This figure illustrates the distributional shape of the full conditional density, reconstructed from the output of the Markov chain.

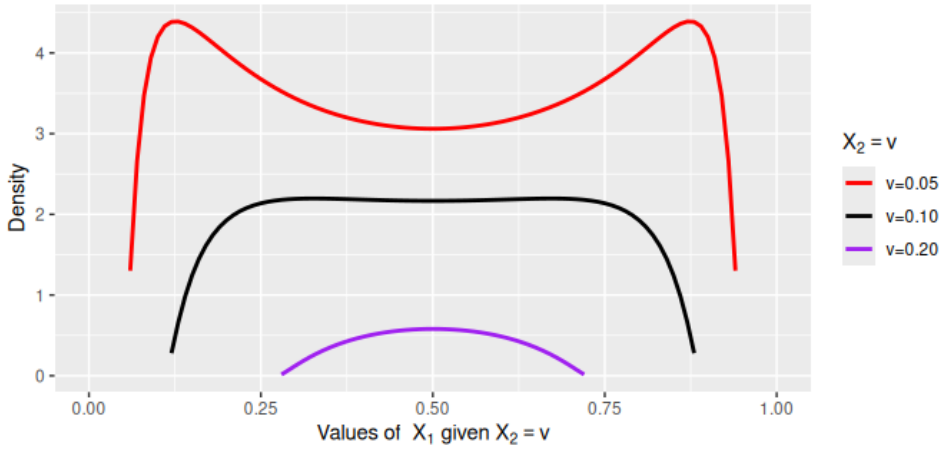

FIGURE 1: Non-normalized full conditional density $f_{X_1|X_2}(x_1|x_2 = v, \phi)$ evaluated for three different values of the conditioning variable $v \in \{0.05, 0.10, 0.20\}$, with fixed parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$. The horizontal axis represents the support of X_1 given $X_2 = v$, and the vertical axis corresponds to the density values.

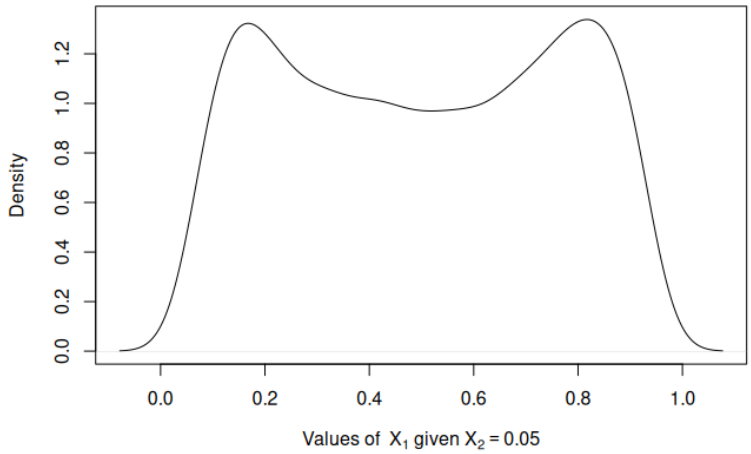

FIGURE 2: Kernel density estimate of a sample of $n = 3800$ observations drawn from the non-normalized full conditional distribution $f_{X_1|X_2}(x_1|x_2 = v, \phi)$, generated using Algorithm 1 with $v = 0.05$ and parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$. The horizontal axis represents the support of X_1 given $X_2 = v$, and the vertical axis corresponds to the estimated density values.

Algorithm 1 was executed for 200 different parameter combinations, generated from 10 values of the conditioning variable $X_2 = v$ (ranging from 0.01 to 0.24) and 20 values of the precision parameter φ (ranging from 1 to 20). Based on the results of these 200 runs, Figure 3 presents three heatmaps that illustrate the relationship

between the conditioning variable $X_2 = v$, the precision parameter φ, and key convergence diagnostics: effective sample size (ESS), acceptance rate, and R-hat.

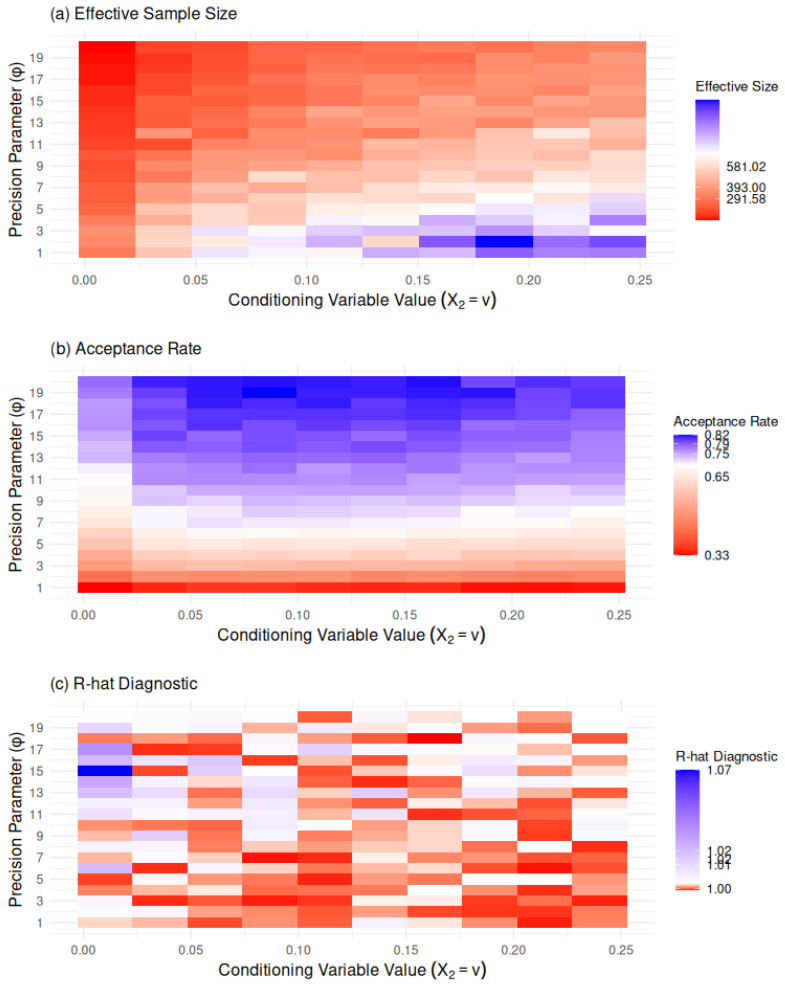

FIGURE 3: Diagnostic measures for the sampling procedure applied to the non-normalized full conditional distribution $f_{X_1|X_2}(x_1|x_2 = v, \phi)$, evaluated across combinations of 10 values of the conditioning variable $v \in [0.01, 0.25]$ and 20 values of the precision parameter φ. The panels show: (a) the effective sample size; (b) displays the acceptance rate; (c) presents the R-hat convergence diagnostic. Each measure was computed from Markov chains generated using Algorithm 1 with $N = 10^4$ iterations, thinning of 2, burn-in of 1000, and a target acceptance rate of 0.4. All plots include labeled axes: the horizontal axis represents the conditioning variable $X_2 = v$, and the vertical axis corresponds to the precision parameter φ. Color gradients reflect the magnitude of each diagnostic metric.

The ESS (panel a), computed using the '`mcmc`' and '`effectiveSize`' functions from the '`coda`' package in R, ranges from 291.58 to 581.02. The ESS reaches its maximum values (depicted in blue) for higher values of v (between 0.10 and 0.25) and moderate precision values (approximately 1 to 6), indicating more representative samples in these regions. However, for precision values greater than 5, the ESS decreases (shifting toward red), suggesting reduced sampling efficiency and lower sample quality.

Conversely, the acceptance rate (panel b) increases as the precision parameter φ rises, largely independent of v. Higher acceptance rates are observed for large φ values, while lower precision levels result in reduced acceptance rates. Although an acceptance rate between 30% and 70% is generally considered favorable, it must be balanced with an adequate ESS. An excessively high acceptance rate combined with a low ESS may indicate that the sampler is exploring a restricted region of the parameter space, potentially compromising the representativeness of the samples (see Robert & Casella, 1999, Robert & Casella, 2010).

The R-hat diagnostic (panel c) further assesses chain convergence by comparing between-chain variance to within-chain variance. The heatmap in Figure 3 displays R-hat values computed from three independent chains of length 4500, generated over the same grid of v and φ values. The maximum R-hat value observed was 1.07, with the 98th, 96th, and 94th percentiles corresponding to 1.02, 1.02, and 1.01, respectively. These results indicate satisfactory convergence across the chains, as R-hat values remain close to one throughout the grid.

4.1.1. Selection of the Precision Parameter φ

The initial selection of the precision parameter φ is based on the diagnostic heatmaps presented in Figure 3. Specifically:

- Panel (a) shows that ESS are notably higher when φ falls within the range $[1, 6)$, as indicated by the blue gradient.
- Panel (b) confirms that acceptance rates remain within the recommended range of $0.30 - 0.70$ for values of φ in the same interval.
- Panel (c) reveals R-Hat values close to 1.00 in the same interval of φ, suggesting convergence, except for $\varphi = 6$, where a blue-shaded cell indicates a value above 1.02, signaling lack of convergence.

Based on this evidence, an initial value $\varphi_0 \in [1, 6)$ is selected and adaptively updated throughout the burn-in phase to enhance convergence and sampling efficiency. The adaptation follows a Robbins–Monro-type stochastic scheme, in which φ is updated every $m = 100$ iterations using the recursion:

$$\varphi_{k+1} = \varphi_k + \frac{1}{t}(\text{target\_acceptance} - \text{acceptance\_rate\_last\_m}),$$

where t denotes the current iteration, k the number of updates (restricted to $k \leq$ burn-in/m), acceptance_rate_last_m is the observed acceptance rate over the last m iterations, and acceptance_target is the predefined target acceptance rate.

4.1.2. Convergence Results for a Generated Chain

Based on the results presented in Figure 3, the precision parameter was set to $\varphi = 3$ for $v = 0.05$. Algorithm 1 was then executed using the simulation scheme in (8), the parameter values defined in (7), and a random initial value generated from the distribution Beta.4P(x_l, x_u, a, b). The convergence diagnostics for the resulting Markov chain are summarized in Figure 4, which includes: the histogram alongside the kernel density estimate, the trace plot of the sampled chain, the cumulative average plot with convergence bounds, and the autocorrelation function (ACF) plot.

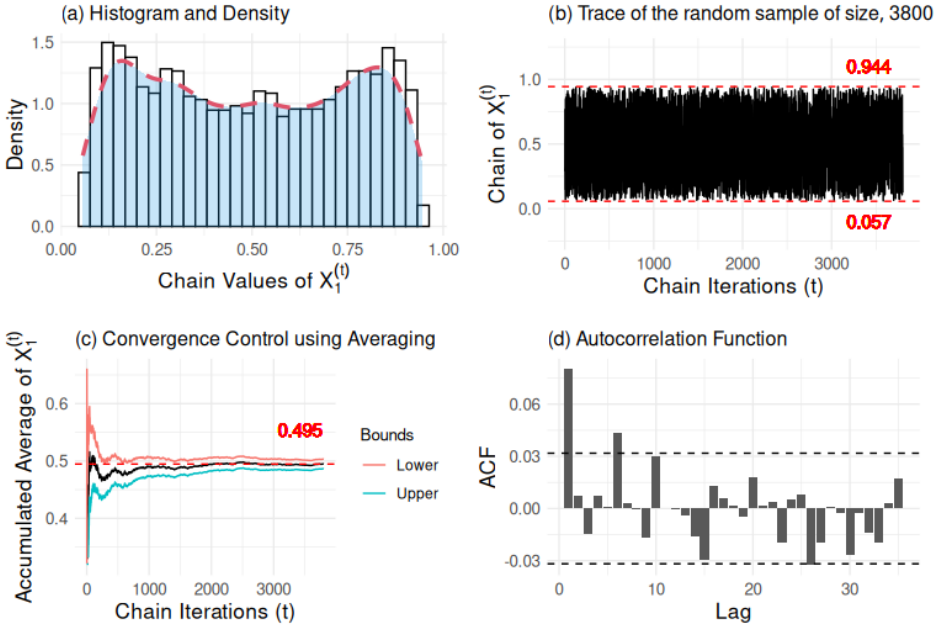

FIGURE 4: Convergence diagnostics for the Markov chain of X_1 generated by Algorithm 1 with $n = 3800$ post burn-in samples, parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$, and precision parameter $\varphi = 3$. The panels illustrate: (a) the histogram and kernel density estimate of the sample, with the horizontal axis representing $X_1^{(t)}$ and the vertical axis showing the density; (b) the trace plot of the chain across iterations t, highlighting the minimum and maximum sampled values; (c) the running mean plot, displaying the average of the chain over iterations t and its convergence bounds, with the horizontal axis for iterations and the vertical axis for the accumulated averaged value of $X_1^{(t)}$; and (d) the autocorrelation function (ACF) of the chain, where the horizontal axis indicates the lag and the vertical axis shows the ACF values.

The cumulative average of the Markov chain provides an effective means of assessing its convergence behavior. In contrast to the results shown in Figure 4, which illustrate the trajectory of a single chain initialized with a random seed,

Figure 5 focuses solely on the cumulative averages computed from nine independent chains, each initialized with a distinct fixed seed. Across all cases, the cumulative averages demonstrate convergence toward a common value, consistent across chains when rounded to one decimal place, regardless of the initial seed. The final average values attained by each chain are summarized in Table 1.

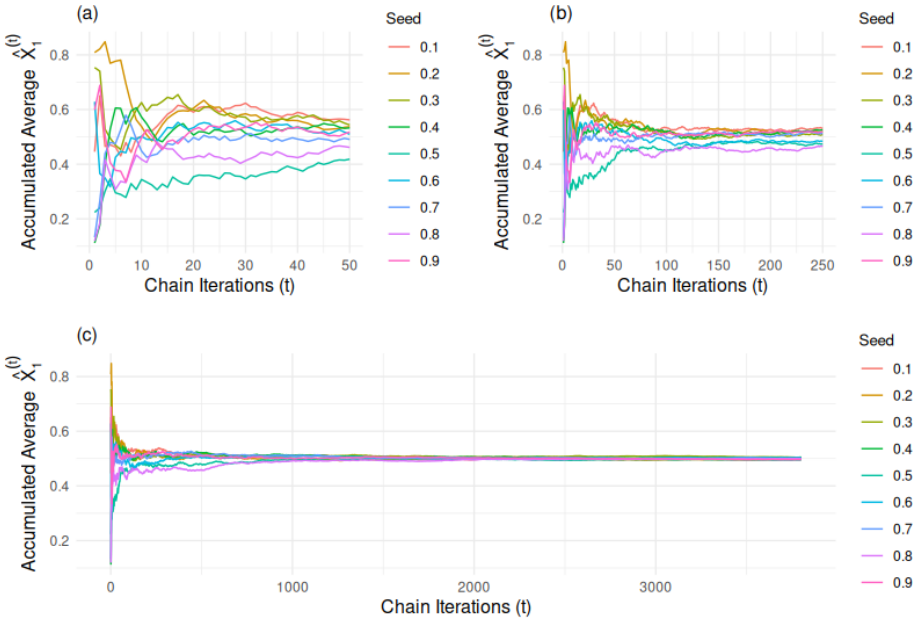

FIGURE 5: Convergence monitoring for Algorithm 1 using the parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$, simulation scheme $N = 10^5$, $\varphi = 3$, *thinning* = 25, *burn-in* = 5000, and a target acceptance rate of 0.4, applied across nine fixed seeds. The panels display the cumulative average of $X_1^{(t)}$ as a function of chain iterations t, illustrating convergence behavior at different scales: (a) from iteration 1 to 50; (b) from iteration 1 to 250; (c) from iteration 1 to 3800 (post burn-in sample size). The horizontal axis in each panel represents chain iterations t, while the vertical axis shows the accumulated average of $X_1^{(t)}$. Different colors correspond to the nine distinct seeds used for initialization.

TABLE 1: Averages obtained for nine chains generated for the conditional density function of X_1 given X_2, considering different fixed seeds in Algorithm 1.

Seed	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Average	0.500	0.502	0.498	0.500	0.503	0.494	0.499	0.501	0.502

4.2. Convergence of Algorithm 2

The convergence of the chain generated by Algorithm 2, which employs the GS method, is now assessed. Using the parameter values specified in (7) and the following simulation scheme:

$$N_1 = 10^5, \quad N_2 = 2, \quad \varphi = 3, \quad thinning = 2, \quad burn - in = 0, \quad \text{and} \quad target\_acceptance = 0.4 \tag{9}$$

a chain is generated and subsequently refined by applying a thinning factor of 25 and a burn-in of 5000 iterations, resulting in a final sample of $n = 3800$ observations from the random vector (X_1, X_2). Figures 6-11 present diagnostic assessments based on this sample. Specifically, Figure 6 shows the contour plot of the joint density function (1) superimposed with a scatter plot of the generated (X_1, X_2) sample, demonstrating strong agreement between the theoretical distribution and the empirical observations. While the marginal convergence diagnostics for each component of the random vector are provided in Figures 7 and 8. These figures display the kernel density estimates, trace plots, cumulative averages, and autocorrelation functions for X_1 and X_2, respectively.

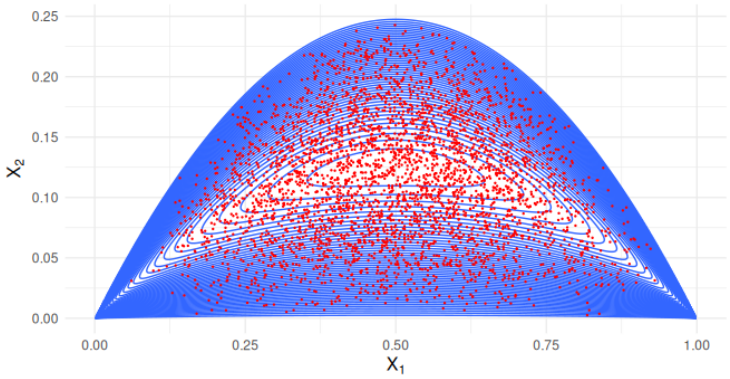

FIGURE 6: Contour plot of the joint density function $f_{X_1,X_2}(x_1, x_2)$ with parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$, overlaid with a scatter plot of a random sample of size $n = 3800$ generated using Algorithm 2. The contour lines indicate density levels, while the scatter points visualize the sampled values, illustrating the alignment between the theoretical distribution and the generated sample.

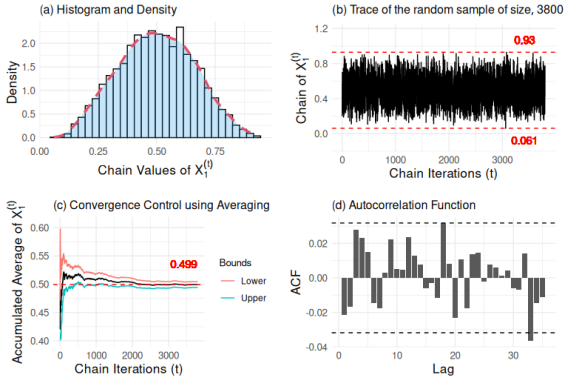

FIGURE 7: Convergence diagnostics for the Markov chain of X_1 generated using Algorithm 2 with parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$. The panels include: (a) the histogram and kernel density estimate of the sampled values of X_1; (b) the trace plot of the chain across iterations t, highlighting the minimum and maximum sampled values; (c) the running mean plot, showing the accumulated average of $X_1^{(t)}$ over iterations t, with convergence bounds; and (d) the autocorrelation function (ACF) of the chain. The horizontal axes represent iterations (panels a–c) or lag (panel d), and the vertical axes represent density, chain values, accumulated averages, or ACF values, respectively.

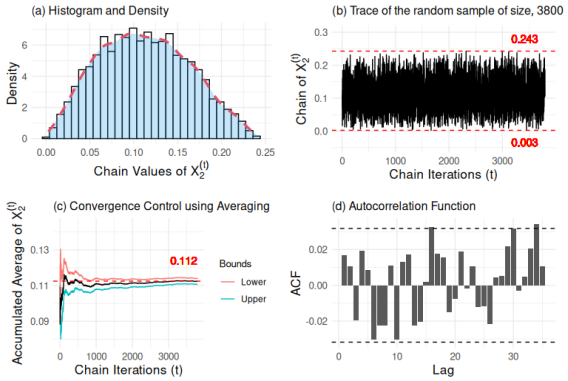

FIGURE 8: Convergence diagnostics for the Markov chain of X_2 generated using Algorithm 2 with parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$. The panels display: (a) the histogram and kernel density estimate of the sampled values of X_2; (b) the trace plot of the chain across iterations t; (c) the running mean plot of $X_2^{(t)}$, with convergence bounds; and (d) the autocorrelation function (ACF). The horizontal axes represent iterations (panels a–c) or lag (panel d), and the vertical axes represent density, chain values, accumulated averages, or ACF values, respectively.

Applying the transformation (2) to the previously generated chain for the vector (X_1, X_2), a new chain for the random vector (Y_1, Y_2) is obtained. Figure 9 displays the scatter plot of this transformed sample overlaid with the corresponding contour lines of the joint density.

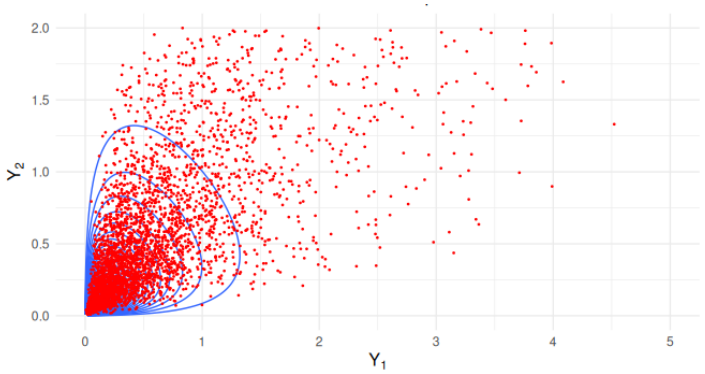

FIGURE 9: Contour plot of the joint density function $f_{Y_1,Y_2}(y_1, y_2)$ with parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$, overlaid with a scatter plot of a random sample of size $n = 3800$ generated by applying the transformation in (2) to the samples from Algorithms 1 and 2. The plot illustrates the alignment between the theoretical bivariate distribution and the generated sample.

The marginal convergence diagnostics for the chains of Y_1 and Y_2 are shown in Figures 10 and 11, respectively. These figures confirm that both chains exhibit convergence and stability. Specifically, the histograms and kernel density estimates align well, the trace plots display stable behavior without significant fluctuations, and the cumulative averages stabilize at approximately 0.886 for Y_1 and 0.898 for Y_2. Furthermore, the autocorrelation functions show decay, indicating low correlation between successive samples. These diagnostic results suggest that the chains adequately explore the target distribution and that the generated samples are representative of the theoretical model.

4.3. Comparison Between Theoretical Moments and MCMC-Based Estimates

Using the generated chains for Y_1 and Y_2, we estimated the mean, variance, and covariance, along with their corresponding standard errors (SE) and 95% credible intervals (CI). The SE for the variance and covariance was calculated using the batch means method with 100 partitions, while the credible intervals were obtained via bootstrap resampling with 1000 replications. For the means, the SE was computed as the posterior standard deviation divided by the square root of the ESS, and the CIs were based on posterior percentiles under the assumption of convergence. These empirical estimates, along with the corresponding theoretical values derived from Theorem 1, are summarized in Table 2.

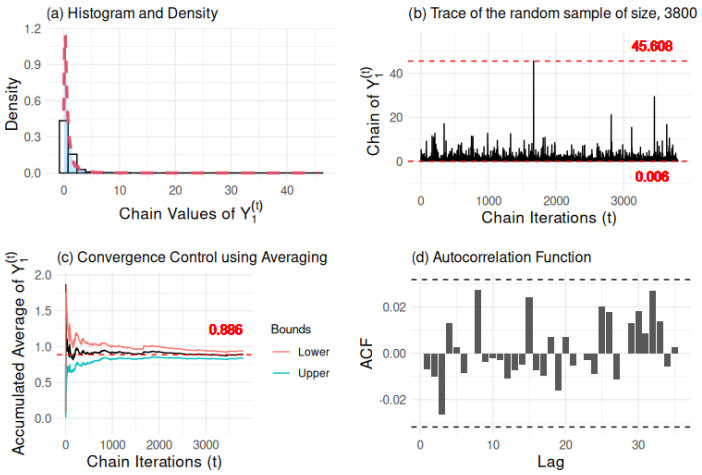

FIGURE 10: Convergence diagnostics for the Markov chain of Y_1 with parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$, generated through the transformation of samples from Algorithms 1 and 2. The panels show: (a) the histogram and kernel density estimate of Y_1; (b) the trace plot of the chain across iterations t; (c) the running mean plot with convergence bounds; and (d) the autocorrelation function (ACF). The horizontal axes represent iterations (panels a–c) or lag (panel d), and the vertical axes represent density, chain values, accumulated averages, or ACF values, respectively.

TABLE 2: Empirical and theoretical moments for the joint PDF (3) using parameter values $\phi = (2.2, 2.2, 2.2, 2.2)$. Empirical estimates include standard errors (SE) and 95% credible intervals (CI). The effective sample sizes (ESS) for Y_1 and Y_2 were 3800 and 3567, respectively, with Gelman–Rubin diagnostics (R-hat) of 0.999 and 1.000.

Moment	Empirical Mean	SE	Credible Interval (95%)	Theoretical Value	Difference
$E[Y_1]$	0.935	0.033	(0.052, 4.735)	0.917	-0.018
$V[Y_1]$	4.077	1.471	(1.784, 7.465)	7.851	3.774
$E[Y_2]$	0.919	0.035	(0.052, 4.672)	0.917	-0.003
$V[Y_2]$	4.649	2.413	(1.823, 9.894)	7.851	3.202
$Cov(Y_1, Y_2)$	3.243	1.510	(1.420, 6.540)	5.135	1.892

The posterior means approximate the theoretical expectations with reasonable accuracy. However, the variance and covariance estimates deviate more substantially. This discrepancy persists even after exploring a wide range of configurations for the MCMC sampling scheme, including variations in the precision parameter φ (selected as described in Subsection 4.1.1), the number of inner samples N_2, the burn-in period, the thinning interval, and the adaptation strategy for φ. These inconsistencies suggest that the parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$ may lead to geometries of the new distribution that are more difficult to explore effectively, even under well-tuned sampling conditions.

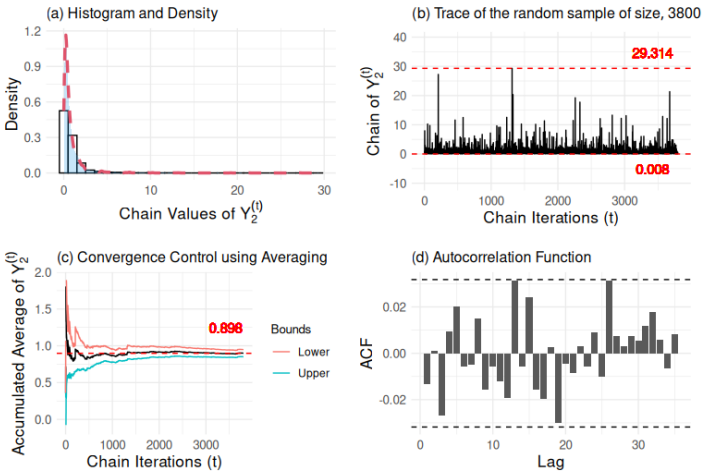

FIGURE 11: Convergence diagnostics for the Markov chain of Y_2 with parameter vector $\phi = (2.2, 2.2, 2.2, 2.2)$, generated through the transformation of samples from Algorithms 1 and 2. The panels include: (a) the histogram and kernel density estimate of Y_2; (b) the trace plot of the chain across iterations t; (c) the running mean plot with convergence bounds; and (d) the autocorrelation function (ACF). The horizontal axes represent iterations (panels a–c) or lag (panel d), and the vertical axes represent density, chain values, accumulated averages, or ACF values, respectively.

To investigate the effect of the simulation scheme on moment approximation, we conducted an additional experiment with $\phi = (3, 6, 3, 6)$. Algorithm 2 was executed under the same scheme as in (9), with a burn-in of 5 000 and a thinning factor of 5. This resulted in a final sample size of $n = 19\,000$. The effective sample sizes for both Y_1 and Y_2 were 19 000, and the Gelman–Rubin diagnostics (R-Hat) were 0.999 for each variable, indicating convergence. The results are summarized in Table 3. In this second scenario, the empirical estimates more closely match the theoretical values, confirming that the choice of parameter ϕ plays a key role in the accuracy of posterior moment approximations.

TABLE 3: Empirical and theoretical moments for the joint PDF (3) using parameter values $\phi = (3, 6, 3, 6)$. Empirical estimates include standard errors (SE) and 95% credible intervals (CI). Both Y_1 and Y_2 achieved effective sample sizes (ESS) of 19 000, with R-hat values equal to 0.999 for each variable.

Moment	Empirical Mean	SE	Credible Interval (95%)	Theoretical Value	Difference
$E[Y_1]$	1.083	0.010	(0.146, 4.111)	1.000	-0.083
$V[Y_1]$	1.957	0.246	(1.519, 2.495)	1.800	-0.157
$E[Y_2]$	1.923	0.018	(0.310, 7.004)	2.000	0.077
$V[Y_2]$	5.948	1.425	(4.068, 9.305)	5.800	-0.148
$Cov(Y_1, Y_2)$	2.574	0.480	(1.843, 3.675)	2.200	-0.374

5. Application

A dataset concerning the proportion of defective items is presented by Defiatri & Damayanti (2023). Assuming the data follow a standard Beta distribution, and that the bivariate distribution proposed in this study can serve as a prior for the shape parameters of the Beta, the values of the hyperparameter vector ϕ in the joint density (3) were determined by solving a system of equations based on bootstrap-derived quantile intervals for the sample mean and variance. Specifically, the method of moments was used to determine the values of (a, b), while the approach proposed by Tovar (2012) was applied to obtain the values of (c, d), thereby fully specifying the joint distribution in (3). Subsequently, Algorithms 1 and 2 were applied to simulate chains for Y_1 and Y_2, and the prior moments—both their empirical estimates and theoretical values—are summarized in Table 4.

TABLE 4: Empirical and theoretical moments for the joint PDF (3) with hyperparameter values $a = 141.6457$, $b = 2263.205$, $c = 39.48316$, and $d = 3092.636$. The simulation scheme was set to 10^5 iterations, with a burn-in of 5000 and a thinning interval of 5. Empirical estimates include standard errors (SE) and 95% credible intervals (CI). Both Y_1 and Y_2 chains achieved effective sample sizes (ESS) of 19000, with R-hat values equal to 1.000 and 0.999, respectively.

Moment	Empirical Mean	SE	Credible Interval (95%)	Theoretical Value	Difference
$E[Y_1]$	4.775	0.006	(3.323, 6.757)	4.733	-0.042
$V[Y_1]$	0.767	0.011	(0.749, 0.787)	0.758	-0.009
$E[Y_2]$	75.553	0.090	(54.737, 103.445)	75.630	0.077
$V[Y_2]$	154.296	2.015	(150.659, 158.065)	154.651	0.355
$Cov(Y_1, Y_2)$	9.580	0.137	(9.330, 9.838)	9.517	-0.063

6. Conclusions

This article introduced a new bivariate distribution along with the derivation and proof of its l-th order joint moment. In addition to the theoretical characterization, the study focused on the implementation and evaluation of two MCMC algorithms for generating random samples from this distribution: a Gibbs sampling scheme that incorporates an adaptive random walk Metropolis–Hastings algorithm for one of the full conditional distributions, using a four-parameter Beta proposal. A central contribution of this work lies in the convergence diagnostics applied to the generated chains, highlighting the stability and effectiveness of the simulation framework under various parameter configurations.

Through empirical monitoring, it was shown that the first-order moments (means) estimated from the simulated chains accurately approximate their theoretical values. However, the estimation of second-order moments, such as variances and covariances, proved to be more sensitive—particularly when symmetric configurations are used (i.e., all components of ϕ are equal). This sensitivity highlights a key limitation of the approach and the need for careful parameter specification in symmetric cases.

While the distribution was originally motivated as a prior for the shape parameters of the Beta distribution, the main focus of this study was not on a full Bayesian inference framework, but rather on presenting the distribution as a standalone bivariate model with a tractable moment structure and a reliable simulation strategy. A brief application was included, where the distribution served as a prior within a Beta likelihood setting. In this context, the empirical moments from the simulated chains once again aligned closely with the theoretical values, reinforcing the internal coherence and validity of the distribution as a prior.

In summary, the principal contributions of this work are: (i) the proposal and theoretical development of a new bivariate distribution, (ii) the formulation of a simulation methodology for generating samples from it, and (iii) the assessment of convergence behavior under various algorithmic and parametric configurations. These results provide a foundation for future studies that may incorporate the distribution more fully into Bayesian inference, explore its use as a prior for other families, or extend its structure to higher-dimensional or more complex settings.

Acknowledgements

We would like to express our gratitude to Universidad del Valle for the teaching assistantships granted to the doctoral student Llerzy Esneider Torres Ome, which enabled him to pursue his doctoral studies.

[Received: October 2024 — Accepted: May 2025]

References

Althubyani, F. A., El-Bar, A. M. T. A., Fawzy, M. A. & Gemeay, A. M. (2022), 'A new 3-parameter bounded beta distribution: Properties, estimation, and applications', *Axioms* **11**(10), 504.

Arroyo, L. G., Lasso, F. A. & Tovar, J. R. (2022), 'Propuesta para obtener distribuciones previas para los parámetros de la distribución beta', *Investigación Operacional* **43**(1), 51–63.

Bouguila, N., Ziou, D. & Monga, E. (2006), 'Practical bayesian estimation of a finite beta mixture through gibbs sampling and its applications', *Statistics and Computing* **16**(2), 215–225.

Bran-Cardona, P. A., Orozco-Castañeda, J. M. & Nagar, D. K. (2011), 'Generalización bivariada de la distribución kummer-beta', *Revista Colombiana de Estadística* **34**(3), 497–512.

Defiatri, D. & Damayanti, R. W. (2023), Acceptance sampling mil-std 105e for quality control: A case study, *in* 'E3S Web of Conferences', Vol. 465, EDP Sciences, p. 02043.

Kaminskiy, M. P. & Krivtsov, V. V. (2005), 'A simple procedure for bayesian estimation of the weibull distribution', *IEEE Transactions on Reliability* **54**(4), 612–616.

Ma, Z. & Leijon, A. (2010), Expectation propagation for estimating the parameters of the beta distribution, *in* '2010 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)', IEEE, pp. 2082–2085.

Ma, Z. & Leijon, A. (2011), 'Bayesian estimation of beta mixture models with variational inference', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(11), 2160–2173.

Mandouh, R. M. & Muhammed, H. Z. (2024), 'On a bivariate bounded distribution: Properties and estimation', *Computational Journal of Mathematical and Statistical Sciences* **3**(1), 125–144.

Martínez-Flórez, G., Vergara-Cardozo, S., Tovar-Falón, R. & Rodriguez-Quevedo, L. (2023), 'The multivariate skewed log-birnbaum–saunders distribution and its associated regression model', *Mathematics* **11**(5), 1095.

Moschen, L. M. & Carvalho, L. M. (2023), 'Bivariate beta distribution: Parameter inference and diagnostics', *arXiv preprint* **arXiv:2303.01271**. `https://arxiv.org/abs/2303.01271`.

Robert, C. P. & Casella, G. (1999), *Monte Carlo Statistical Methods*, 1st edn, Springer.

Robert, C. P. & Casella, G. (2010), *Introducing Monte Carlo Methods with R*, Use R!, Springer.

Tovar, J. R. (2012), 'Eliciting beta prior distributions for binomial sampling', *Revista Brasileira de Biometria* **30**(1), 159–172.

Repository

All code and resources necessary to replicate the results of this study are available in the following GitHub repository: Estimation-Parameter-Beta. This repository contains the implemented simulation algorithms, supporting scripts, and detailed documentation to facilitate reproducibility. Instructions for running the simulations and regenerating the figures presented throughout the article are provided.

In particular, the R Markdown file used for random sample generation is available at `randomsamples.md`. The custom functions called within this file are defined in the accompanying script `requiredfunctions.R`.