IRAQI STATISTICIANS JOURNAL

https://isj.edu.iq/index.php/isj

ISSN: 3007-1658 (Online)

Bayesian Methods in Statistics : A Review

Najlaa Saad Ibrahim AL-Sharabi [1], Sura Mohamed Jamal-Alden Hussain[1]

[1]Department of Statistics and Informatics, College of Computer Science and Mathematics, University of Mosul, Mosul, Iraq

najlaa.s.a@uomosul.edu.iq
sura.alalwlia@uomosul.edu.iq

ARTICLE INFO

Article history:
Received 17 April 2026,
Revised 17 April 2026,
Accepted 1 July 2026
Available online 10 July 2026

Keywords:
Bayesian inference, Posterior distribution, Markov Chain Monte Carlo (MCMC), Gibbs Sampling, Hamiltonian Monte Carlo, No-U-Turn Sampler.

ABSTRACT

Bayesian statistics provides a comprehensive intellectual framework, redefining probability as a statement of individual certainty. This method redefines probability as a statement of individual certainty. The philosophical underpinnings of Bayesian statistics are first examined in this methodical research, with a focus on the cognitive updating process. In order to create updated views expressed through posterior distributions, this approach relies on the methodical integration of previous beliefs (expressed through prior distributions) with fresh data obtained from real-world experiments (using a probability function). In sophisticated computations, Bayesian statistics presents difficulties despite its many benefits. However, these difficulties have been largely overcome because to developments in simulation methods like the Markov Chain Monte Carlo (MCMC). The Metropolis-Hastings algorithm, Gibbs sampling, Hamiltonian Monte Carlo (HMC), and the No-U-Turn Sampler (NUTS) technique are the four main algorithms that are examined in this paper within the MCMC framework. It explains how they developed historically and examines their theoretical and practical traits. The study compares the benefits of the Bayesian approach, such as model design flexibility and uncertainty management, with its drawbacks, such as prior distribution selection and computational complexity. In view of the increased usage of artificial intelligence technologies and big data, the study also examines the prospects for the Bayesian approach in the future. In order to help researchers and professionals in the field stay up to date with the latest advancements in this quickly growing field, the review finishes with practical advice. These recommendations include identifying interesting research directions.

1. Introduction

Statistics has been under some major methodological and philosophical change over the past few decades. The Bayesian model combines a different philosophical and methodological perspective than the old iterative framework. It offers a new definition of probability. This development reflects a need for more flexible tools capable of handling the complexity of data management, and making confident predictions. (Gellman et al., 2013). And the only difference between the Bayesian approach is that probability is

Corresponding author E-mail address:
https://doi.org/10.62933/kqy88t52

not just the frequency of occurrence of the event but also as a value that demonstrates the degree of confidence and conviction. This kind of response may be viewed as a more reasonable reaction to the challenges of the real-life experience where the way of responding to uncertainties and probabilities should be more pragmatic. This is a paradigm shift in statistics. However, technological issues have complicated this strategy in terms of prevalence because of its complexity and need of computer resources since the 1970s.(Bernardo and Smith, 1994). This has become controllable because the creation of stochastic algorithms and simulation techniques such as Markov Chain Monte Carlo has been developed quickly (Howson and Urbach, 2006; Oqbah and Al-Rassam, 2025). This has increased the application range of this approach and simplified the task of developing multilevel models that could respond to issues of truly complex problems. The impact of Bayesian approach has expanded rapidly beyond computation to a wide range of applications in real-life many fields. It has aided in reducing health hazards pertaining to illnesses and raising levels of diagnosis in medicine and health care. (Van de Schoot et al., 2021; Kruschke, 2014). It has played a critical role in deciphering complex human behavioral patterns both in the social sciences and the economic sciences. Machine learning and artificial intelligence have also highly benefited with the assistance of Bayesian techniques to solve hard problems with large amounts of uncertainty. These examples show the number of ways in which Bayesian methods can be used to solve many complicated problems (Brooks et al., 2011). Over the past twenty years, Bayesian methods have been studied from the theoretical perspective to the applied. In 2013, Gelman et al. provided an overview of the Bayesian methodology from the theoretical viewpoint and reviewed the computational aspects of the method. The authors presented Metropolis-Hastings and Gibbs Sampling algorithms, but did not discuss the details of the later algorithms HMC or NUTS. Robert and Casella (2010) presented the mathematical foundations of Monte Carlo techniques, but their review was written for a statistical audience. In the field of advanced algorithms, Hoffman and Gelman (2014) examined how the NUTS algorithm is constructed and evaluated, and Betancourt (2017) provided a theoretical overview of the HMC algorithm. They focused on one particular algorithm and did not analyze any alternatives.

This chapter will address the basic philosophical and methodological issues surrounding the Bayesian statistics, its history and use in solving current problems such as big data and artificial intelligence.It further alludes to the challenges, which continue to impede the broader application of this option in remarking its unique capabilities in the processing and forecasting of the data in future. The end of the study proposes a vision of this interesting topic in the future as a robust and efficient means that will not be ineffective to satisfy the demands of the ever-changing cyber world.

2. Bayesian reasoning formula

In its methodology, analysis, and conclusions, Bayesian theory depends on data from observations as well as data from individual beliefs, or "prior information." In contrast to the conventional school, the Bayesian school views parameters in probability distributions as random variables with a probability distribution (McGrayne, 2011). The parameter φ is an unknown number that is the focus of Bayes' estimate approach. Although the parameter has a real value, it is random as the parameters are unaware of it. It focuses on the distribution $P(\varphi)$, which expresses the parameter's lack of initial knowledge. To learn more about the parameter, historical data, knowledge, and qualifying beliefs are measured and analyzed

to generate the previous distribution. Assuming that the parameter φ has a certain value, the probability of observing data x is $P(x|\varphi)$, which quantifies the likelihood of each potential value of the parameter in light of the actual data investigated. However, the well-known Bayes' Theorem, which may be expressed mathematically as the following formula (Efron, 2013) (Howson & Urbach, 2006), is used to determine the posterior distribution, which asserts that the initial unknown of φ is updated in light of the new information:

$$P(\varphi|x) = \frac{P(x|\varphi)P(\varphi)}{P(x)} \qquad (1)$$

The normalizing constant, or $P(x)$, is the factor that guarantees the total integral of the ensuing distribution equals one (the sum of the probabilities of the values of φ equals one). Additionally, it makes it possible to compare several models.

3. The Development of Computational Bayesian Algorithms

For numerous decades, a major barrier to the broad application of Bayesian models was the computational complexity of constructing subsequent distributions. But since the end of the 20th century, computational developments have drastically altered this subject, making sophisticated Bayesian models broadly usable. Stochastic simulation techniques, especially Markov Chain Monte Carlo (MCMC) methods, have been used to manipulate multiparameter and nonlinear statistical models, overcoming the limitations of traditional analytical models and overcoming long-standing computational challenges associated with Bayesian inference.

The following are some basic techniques utilized in Bayesian Monte Carlo estimators for Markov Chain Monte Carlo (MCMC):

3.1 The Algorithm of Metropolis-Hastings

Metropolis created this algorithm in 1953, and Hastings improved it in 1970. It is among the most popular techniques for putting MCMC, often known as MH, into practice. The algorithm operates by taking a Markov chain of values φ. Only its prior state, φ_{t-1}, determines each value φ_t. The simulation is carried out on the assumption of a probability density $\pi(\varphi)$, referred to as the target density. A new proposed sample, z, is created using the function $q(\mathrm{z}|\varphi)$, which depends on the current value of φ and indicates the proposed distribution (normal or uniform). A proposed sample is accepted with a probability of $\alpha(\mathrm{z}, \varphi)$, and rejected to ensure the process is reversible. Since the proposed distribution is usually a random distribution, it cannot be expected to benefit the following target density equation $\pi(\varphi)$ (Chib & Greenberg, 1995):

$$q(\mathrm{z}|\varphi)\pi(\varphi) \neq q(\varphi|\mathrm{z})\pi(\mathrm{z}) \qquad (2)$$

In order to balance the inequality, we define the factor $\alpha(\mathrm{z}, \varphi) < 1$, assuming that $q(\mathrm{z}|\varphi)\pi(\varphi) > q(\varphi|\mathrm{z})\pi(\mathrm{z})$.

$$q(\mathrm{z}|\varphi)\pi(\varphi)\alpha(\mathrm{z}, \varphi) = q(\varphi|\mathrm{z})\pi(\mathrm{z}) \qquad (3)$$

As a result, we get the value of $\alpha(\mathrm{z},\varphi)$, also known as the likelihood of acceptance, which has the following form:

$$\alpha(\mathrm{z},\varphi)=\min\left(1,\frac{q(\varphi|z)\pi(z)}{q(z|\varphi)\pi(\varphi)}\right) \qquad (4)$$

The Metropolis-Hastings method can be described as follows and only requires the definition of $\pi(\varphi)$.

1. When t = 0, take any initial value φ_0 such that $\pi(\varphi_0)>0$.
2. Assume φ_t is the sequence's current value for the point sample z_t from the suggested distribution $q(z|\varphi)$.
3. Using the uniform distribution, create a random variable U.
4. Update and produce new values in the manner described below:
   i. When $\varphi_{t+1}=\mathrm{z}_t$, if $U\le\alpha(\varphi_t,\mathrm{z}_t)$.
   ii. Set $\varphi_{t+1}=\varphi_t$ if the preceding step is not met.
5. Raise the t values.
6. Go to the second step if t < N; if not, return $\{\varphi_1,\varphi_2,\dots,\varphi_N\}$.

The target density expression $\pi(\varphi)$ is the distribution that follows the formation of a Markov chain via the Metropolis-Hastings algorithm. Assuming that $p_{\varphi z}$ is the probability of moving from φ to z, this can be demonstrated as follows:

$$p_{\varphi z}=q(\mathrm{z}|\varphi)\alpha(\mathrm{z},\varphi)=q(\mathrm{z}|\varphi)\ \min\left(1,\frac{q(\varphi|z)\pi(z)}{q(z|\varphi)\pi(\varphi)}\right) \qquad (5)$$

According to the equation above, if the algorithm reaches detailed equilibrium, then:

$$p_{\varphi z}\pi(\theta)=p_{z\varphi}\pi(z)$$
$$q(\mathrm{z}|\varphi)\alpha(\varphi,z)\pi(\varphi)=q(\varphi|z)\alpha(\mathrm{z},\varphi)\pi(z) \qquad \forall\,\varphi,z \quad (6)$$

Under the following circumstances, the samples of the target distribution equal the constant distribution of the transition probability:

i. The full equilibrium equation will be constant if $\alpha(\varphi,z)=\alpha(\mathrm{z},\varphi)=1$ and $q(\mathrm{z}|\varphi)\pi(\varphi)=q(\varphi|z)\pi(z)$.

$$p_{\varphi z}\pi(\varphi)=q(\mathrm{z}|\varphi)\pi(\varphi)\ ,p_{z\varphi}\pi(z)=q(\varphi|z)\pi(z) \qquad (7)$$

This results in:

$$p_{\varphi z}\pi(\varphi)=p_{z\varphi}\pi(z) \qquad (8)$$

ii. The full equilibrium equation will stay constant if $\alpha(\varphi,z)=\frac{q(\varphi|z)\pi(z)}{q(z|\varphi)\pi(\varphi)}$, $\alpha(\mathrm{z},\varphi)=1$, and $q(\mathrm{z}|\varphi)\pi(\varphi)>q(\varphi|z)\pi(z)$.then

$$p_{\varphi z}\pi(\varphi)=q(\mathrm{z}|\varphi)\alpha(\varphi,z)\pi(\varphi)=q(\mathrm{z}|\varphi)\frac{q(\varphi|z)\pi(z)}{q(z|\varphi)\pi(\varphi)}\pi(\varphi) \qquad (9)$$

$$p_{z\varphi}\pi(z)=q(\varphi|z)\alpha(\mathrm{z},\varphi)\pi(z)=q(\varphi|z)\pi(z) \qquad (10)$$

iii. The full equilibrium equation will stay constant if $\alpha(\varphi, z) = 1$, $\alpha(\mathrm{z}, \varphi) = \frac{q(z|\varphi\theta)\pi(\varphi)}{q(\varphi|z)\pi(z)}$, and $q(\mathrm{z}|\varphi)\pi(\varphi) < q(\varphi|z)\pi(z)$.

$$p_{\varphi z}\pi(\varphi) = q(\mathrm{z}|\varphi)\alpha(\varphi, z)\pi(\varphi) = q(\mathrm{z}|\varphi)\pi(\varphi) \quad (11)$$

$$p_{z\varphi}\pi(z) = q(\varphi|z)\alpha(z, \varphi)\pi(z) = q(\varphi|z)\frac{q(z|\varphi)\pi(\varphi)}{q(\varphi|z)\pi(z)}\pi(z) \quad (12)$$

Since the equilibrium equation is constant in the aforementioned scenarios, we may deduce that the algorithm is a Markov chain (Ahmad & Saieed, 2024).

3.2 The Gibbs Sampling (GS) Algorithm

This algorithm is a case of the Metropolis-Hastings algorithm, which is shortened to (GS) and named for the physicist Josiah Willard Gibbs. While the conditional distribution is known, it is used to produce a sequence of samples from the unknown joint probability distribution of two or more random variables. As long as the present values of the other random variables exist, values are produced from the distribution of each random variable. As a result, the series of samples forms a Markov chain with a constant distribution, the common distribution. The intended outcome is the boundary probability density function, $g_{\varphi}(\varphi)$, and the algorithm's mechanism assumes a joint probability density function, $g_{\varphi,z_1,\dots,z_N}(\varphi, z_1, \dots, z_N)$. Integration is used to estimate $g_{\varphi}(\varphi)$ (Casella & George, 1992).

$$g_{\varphi}(\varphi) = \int \dots \int g_{\varphi,z_1,\dots,z_N}(\varphi, z_1, \dots, z_N) dz_1 \dots dz_N \quad (13)$$

The Gibbs sampling process is an alternative technique for generating the boundary probability density function because integrals might be challenging to calculate in some situations.

3.3 The Monte Carlo Hamiltonian Algorithm

The Hamiltonian Monte Carlo algorithm reduces between sample correlation while keeping a high acceptance rate for the Metropolis-Hastingsmethod. It does so by using a Hamiltonian dynamical system. where the target density is the proposition for the Markov Chain Monte Carlo (MCMC) simulation of the motion of a particle in a parameter space. An extension of this procedure is to add momentum variables q, that are assumed to have a Gaussian distribution, to the target distribution. These variables are equal in size but are not affected by the variables (Betancourt, 2017). As the formula indicates, the formula for the common target distribution is:

$$p(\varphi, q) \propto e^{-U(\varphi)-\frac{q'q}{2}} \equiv e^{-H(\varphi,q)} \quad (14)$$

where the potential energy is represented by $U(\varphi) = \log(\pi(\varphi))$. The symbol for the kinetic energy is $\frac{q'q}{2}$. The MCMC algorithm on the space (φ, q), where φ is interpreted as

the position coordinates of a physical particle with total energy $H(\varphi, q)$, given by the sum of its potential energy $U(\varphi)$ and its kinetic energy $\frac{q'q}{2}$, was proposed using Hamiltonian dynamics as a result of the expansion of the target distribution. The Markov chain is built by taking into account the Markov chain's state (φ_n, q_n) at time n. Resampling the momentum $q_n \sim n(0,1)$yields the new state for time n+1, which is then proposed using the subsequent procedures (Neal, 2012):

i. Using the differential equation to simulate a deterministic Hamiltonian flow with regard to time t and an initial condition (φ_n, q_n):

$$\frac{d}{dt}\begin{bmatrix}\varphi_t \\ q_t\end{bmatrix} = \begin{bmatrix}0 & I \\ -I & 0\end{bmatrix} = \begin{bmatrix}\nabla_\varphi H\big(q(t), \varphi(t)\big) \\ \nabla_q H(q(t), \varphi(t))\end{bmatrix} \equiv \begin{bmatrix}q(t) \\ -\nabla_\theta U\big(\varphi(t)\big)\end{bmatrix} \qquad (15)$$

To get:

$$\big(\varphi'_n, q'_n\big) = \delta_{t,H}(\varphi_n, q_n) \qquad (16)$$

In this case, adopting a certain pair of position and momentum as initial conditions for the Hamiltonian flow associated with H for time t is denoted by $\delta_{t,H}$.

ii. Use the following function to reverse the momentum component that results:

$$(\tilde{\varphi}_{n+1}, \tilde{q}_{n+1}) = \delta_q\big(\varphi'_n, q'_n\big) = \tilde{\delta}_{t,H}(\varphi_n, q_n) \qquad (17)$$

iii. To impose detailed equilibrium for the target distribution, use the Metropolis-Hastings algorithm's acceptance or rejection phases.

iv. Use δ_q to reverse the momentum once more so that it points in the original direction.

The function $\tilde{\delta}_{t,H}$ is self-reversible because it is time-reversible (i.e., if the path q(t), φ(t) is a solution to Equation (15), then the reversed momentum path q(-t), φ(-t) is likewise a solution). Consequently, it has been demonstrated that the acceptance ratio in step (3), which upholds detailed equilibrium, has the following structure (Green, 1995):

$$\min\left(1, \frac{e^{-H(\tilde{\varphi}_{n+1}, \tilde{q}_{n+1})}}{e^{-H(\varphi_n, q_n)}} \left|\det \nabla_{\varphi,q} \tilde{\delta}_{t,H}(\varphi_n, q_n)\right|\right) \qquad (18)$$

Observe that by considering the determinant of the aforementioned equation, the momentum inversion operator $\tilde{\delta}_{t,H}$ and the Hamiltonian flux conserve magnitude. As a result, the acceptance ratio (3) has a Jacobi limit of 1. As a result, the following formula represents the acceptance probability:

$$\min\left(1, e^{H(\varphi_n, q_n) - H(\tilde{\varphi}_{n+1}, \tilde{q}_{n+1})}\right) \qquad (19)$$

Additionally, the acceptance ratio is equal to 1 since the Hamiltonian flow defined in (15) conserves energy (i.e., $H(\varphi_n, q_n) \equiv H(\tilde{\varphi}_{n+1}, \tilde{q}_{n+1})$). Furthermore, the common

distribution is maintained by the momentum inversion in step (4) and the momentum resampling in step (1).

3.4 Algorithm for the No-U-Turn Sampler (NUTS)

The Hamiltonian Monte Carlo (HMC) approach is expanded upon by the (NUTS) algorithm. The procedure is iterative and creates a set of possible candidate points that span a large portion of the target distribution. At the beginning of the turn, it automatically stops and starts over. This algorithm suggests

$$\frac{d}{dt}\frac{(\tilde{\varphi}-\varphi).(\tilde{\varphi}-\varphi)}{2} = (\tilde{\varphi}-\varphi).\tilde{q} \qquad (20)$$

This amount is directly proportional to the distance we go from the beginning point φ when the simulation is run for an infinitesimally small additional time period. As a result, the HMC algorithm needs several steps to make the amount in equation (20) less than zero. In addition, the algorithm's lack of time reversibility keeps it from getting close to the proper distribution. The NUTS technique was developed to solve this issue by doing the HMC simulation in both forward and backward time while preserving reversibility. In order for the NUTS method to function, a slice variable w with a conditional distribution $p(w|\varphi,q) = uniform(w, e^{-U(\varphi)-\frac{q'q}{2}})$ is introduced. This causes the conditional distribution $p(\varphi,q|w) = uniform(\varphi,q;\{\varphi',q' \left| e^{-U(\varphi)-\frac{q'q}{2}}\right\} \geq w)$.

The NUTS algorithm's derivation and implementation are made simpler by this step. The NUTS method uses a jump integer to plot a road ahead and backward in a hypothetical time after sampling $(w|\varphi,q)$. The path first moves forward or backward one step, then forward or backward two steps, then forward or backward four steps, and so on. Position and momentum states are represented by the terminal nodes of a balanced binary tree that is implicitly created a criterion that determines the adequate simulation duration, i.e., when executing the simulation for a higher number of steps does not result in an increase in the gap between the suggested value $\tilde{\varphi}$ and the original value of φ. As seen in the following image, this criterion is based on the dot product of the current momentum $\tilde{q}$ and the vector from the beginning location to the present position $(\tilde{\varphi}-\varphi)$ computed with respect to time (in the Hamiltonian).

by this multiplication procedure. When any balanced subtree of the overall binary tree starts to spin on itself from left to right (i.e., the virtual particle starts to rotate "U"), the multiplication stops. At this stage, the simulation is stopped by the NUTS algorithm, which carefully maintains detailed balance by sampling from the collection of locations determined during the simulation (Matthew and Andrew, 2014).

4. Evaluation of the Four MCMC Methods

The following standards will be used to compare the four MCMC algorithms:

1) Computational Efficiency**:** The ability of an algorithm to solve a problem with the least amount of time, memory, and money is known as computational efficiency. The computational complexity of an algorithm, which is reflected in its execution speed and memory usage, is commonly used to gauge its efficiency.
2) Algorithm Flexibility and Applicability**:** This is as far as the solutions of the algorithms are concerned to achieve the desired results. It also denotes the ability of the algorithm to process different

types of data and their requirements such as prior knowledge of the target distribution, and its ability to give the best model, which makes it competitive with the rest of the algorithms. A flexible algorithm can be applied to all problems without modifying its basic structure (Schneider et al., 2013).

3) Ease of Implementation**:** This describes how much work the algorithm puts into coming up with straightforward, understandable solutions as well as how sensitive it is to different testing.
4) Maintainability: This is the algorithm's ability to modify, improve, and correct errors, by measuring the autocorrelation between successive samples (where low values indicate sample independence), the mixing rate (which indicates the speed at which the Markov chain moves through the parameter space), and parameter space coverage (which indicates the algorithm's ability to detect the target distribution space).
5) Theoretical Properties of the method: This pertains to the precision of the data generated by the method, including parameter stability (the results do not improve with a change in the data) and convergence to actual solutions(after a certain number of iterations the series reaches the target distribution).

The four algorithms are compared using the aforementioned criteria in the following table.

Table (1): Comparison between the four algorithms.

Algorithms	Compared criteria				
	Computational Efficiency	**Flexibility and Applicability**	**Ease of Implementation**	**Maintainability**	**Theoretical Properties**
MH	**2**	**5**	**5**	**2**	**4**
GS	**3**	**2**	**3**	3	**3**
HMC	**4**	**4**	**2**	4	**4**
NUTS	**5**	**4**	**2**	**5**	**5**
The values on the scale indicate: 1 = Very poor, 2 = Poor, 3 = Average, 4 = Good, 5 = Excellent					

The following is indicated by the results in the above table:

1. Due to the method used to calculate the path length, the NUTS algorithm was the best according to the computational efficiency criterion. (Hoffman & Gelman, 2014). The MH algorithm was ranked the last, the HMC being the second and GS being the third.
2. The MH algorithm was the most flexible and applicable, followed by the GS algorithm (Chib & Greenberg, (1995). HMC and NUTS, which are equally versatile, were third because they deal with differentiable models (Betancourt, 2017).
3. The MH method was the ease to perform when the algorithms were compared. The GS algorithm came next, while the HMC and NUTS algorithms, because they require a large amount of computational work,

came in third place (Carpenter et al., 2017).

4. The NUTS method came in first, followed by HMC, the GS algorithm, and the MH algorithm when comparing the autocorrelation measurement between consecutive samples and the mixing rate (Geyer, 2011; Robert & Casella, 2010).
5. According to the theoretical characteristics of four algorithms, NUTS occupies the first place, and MH and HMC are in the second and third places, respectively.

The following is advised in light of the four algorithms' comparative results above:

1. Use the MH algorithm to find basic models since it can identify several effective modeling techniques due to its high execution speed and flexibility.

2. NuTS algorithm is the most widely used in industrial applications due to its general stability, accuracy, and ability to approximate real-world solutions.

5. Conclusions

The following conclusions are drawn from a review of some basic MCMC algorithms and Bayesian approaches in statistics:

1. The NUTS algorithm demonstrated significant superiority compared to the other three algorithms in terms of computational efficiency, optimization potential, and theoretical properties.
2. The MH algorithm showed the highest flexibility, applicability, and ease of implementation.
3. The lowest computational efficiency resulted from the MH algorithm, suggesting an orderly relationship between ease of implementation and computational efficiency.
4. This review is important as it provides a comparative evaluation of the best Bayesian algorithms and illustrates the development of these algorithms in terms of their ideas and history.
5. The study identified research gaps and suggested research directions that practitioners and researchers could pursue.

6. Recommendations

1. The MH algorithm is recommended as a starting point to find basic models. Because it is fast and versatile, it can pick up many effective modelling techniques.
2. Using real-world data, create algorithms based on the data.

References

[1] Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2013). *"*Bayesian Data Analysis (3rd ed.)*"*. *Chapman and Hall/CRC*. https://doi.org/10.1201/b16018

[2] Bernardo, J. M., & Smith, A. F. M. (1994). "Bayesian Theory*"*. *JOHN WILEY & SONS, LTD*. http://www.wiley.co.uk or http://www.wilcy.com

[3] Oqbah , M.A. & Al_Rassam , R.S. (2025)."Bayesian Estimation of Stress-Strength Reliability Function P(T<X<Z) for EIR Distribution". *Rimah Journal for Research and Studies* ,Vol.3, No.122. DOI:10.59799/SQZH5356

[4] Howson, C., & Urbach, P. (2006). *"*Scientific Reasoning: The Bayesian Approach (3rd ed.)". *Open Court Publishing*. http://fitelson.org/probability/howson_and_urbach_3rd_ed.pdf

[5] Kruschke, J. K. (2014). "Doing Bayesian Data Analysis*"*. *A Tutorial with R, JAGS, and Stan (2nd ed.)*. http://dx.doi.org/10.1016/B978-0-12-405888-0.00001-5

[6] van de Schoot, R., Depaoli, S., King, R. & Kramer, B., (2021). "Bayesian statistics and modelling". *Nature Reviews Methods Primers*, Vol. 1, No. 1, PP.1-26. DOI:10.1038/s43586-020-00001-2

[7] Brooks, S., Gelman, A., Jones, G., & Meng, X. L. (2011). "Handbook of Markov Chain Monte

Carlo". *Chapman and Hall/CRC*. https://doi.org/10.1201/b10905

[8] McGrayne, S. B. (2013). "The Theory That Would Not Die: How Bayes' Rule Cracked the Enigma Code, Hunted Down Russian Submarines, and Emerged Triumphant from Two Centuries of Controversy". *The European Legacy*, Vol.18. No.7, pp. 965–966. DOI:10.1080/10848770.2013.817786

[9] Efron, B. (2013). "Bayes' Theorem in the 21st Century". *Science*, Vol.340, No.6137, PP.1177-1178. DOI:10.1126/science.1236536

[10] Chib, S. & Greenberg, E., (1995)." Understanding the Metropolis-Hastings Algorithm", *The American Statistician*, Vol. 49, No.4, pp. 327-335.http://links.jstor.org/sici?sici=0003-1305%28199511%2949%3A4%3C327%3AUTMA%3E2.0.CO%3B2-2.

[11] Ahmad, H. A. & Saieed, H. A.,(2024)," Bayesian Estimation of TLR distribution under Sq and E linex loss". *Iraqi Journal of Statistical Sciences*, Vol. 21, No. 2, PP.64-71 http://creativecommons.org/licenses/by/4.0/

[12] Casella,G. & George, E.,(1992) "Explaining the Gibbs Sampler*", The American Statistician*, vol. 46, No. 3. http://www.jstor.org/

[13] Betancourt, M., (2017)."A Conceptual Introduction to Hamiltonian Monte Carlo". *arXiv*:1701.02434. https://doi.org/10.48550/arXiv.1701.02434

[14] Neal, R., (2012)." MCMC using Hamiltonian Dynamics*". Handbook of Markov Chain Monte Carlo*. DOI:10.1201/b10905-6

[15] Green, P. J.,(1995)."Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination". *Biometrika*, Vol. 82, No .4, PP.711–732. https://doi.org/10.1093/biomet/82.4.711

[16] Matthew, D. H. and Andrew, G., (2014)." The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo". *Journal of Machine Learning Research*, No.15, PP.1593-1623. https://www.jmlr.org/papers/volume15/hoffman14a/hoffman14a.pdf

[17] Schneider, M., Grahl, J., Francas, D. & Vigo, D., (2013)." A problem-adjusted genetic algorithm for flexibility design". *International Journal of Production Economics*, Vol. 141, No.1. https://doi.org/10.1016/j.ijpe.2012.05.017

[18] Hoffman, M. D., & Gelman, A. (2014). "The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo". *Journal of Machine Learning Research,* Vol.15, No.47, PP.1593-1623. https://www.jmlr.org/papers/volume15/hoffman14a/hoffman14a.pdf

[19] Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M. A., Guo, J., Li, P., & Riddell, A. (2017). "Stan: A Probabilistic Programming Language". *Journal of Statistical Software*, Vol. 76, No.1, PP.1-32. *doi: 10.18637/jss.v076.i01*

[20] Geyer, C. J. (2011). "Introduction to Markov Chain Monte Carlo". *In S. Brooks, A. Gelman, G. Jones, & X. L. Meng (Eds.), Handbook of Markov Chain Monte Carlo* (pp. 3-48). Boca Raton: CRC Press. https://scispace.com/pdf/introduction-to-markov-chain-monte-carlo-4i4ybfwvbi.pdf

[21] Robert, C. P., & Casella, G. (2010). "Monte Carlo Statistical Methods (2nd ed.) ". *New York: Springer.* https://mcube.lab.nycu.edu.tw/~cfung/docs/books/robert2004monte_carlo_statistical_methods.pdf