Bayesian Inference and MCMC Methods in Astrophysics

AGASTYA GAUR

University of Illinois at Urbana-Champaign

ABSTRACT

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Keywords: Astrophysics, Astrostatistics, Bayesian Statistics, Markov Chain Monte Carlo, Big Data

1. INTRODUCTION

1.1. *The History of Astrostatistics*

In the 4th century BC, Hipparchus, attempting to estimate the length of a year, found the middle of the range of a scattered set of Babylonian solstice measurements (Feigelson & Babu 2004). An achievement for the time, Hipparchus's measurement marked the beginning of what would become a long-standing marriage between astronomy and statistics. In the centuries to come, a number of breakthroughs in astrostatistics followed. Notably, Tycho Brahe in the late 1500s made repeated positional measurements of stars using naked-eye observations. The data were so precise that their modeling led to Kepler's new laws of planetary motion (Leavesley & Tárnok 2018). Furthermore, in the 1770s, Laplace rediscovered Bayesian statistics, and over the next decade he expanded upon the theory, using it to modernize Newton's theory of gravity (Stigler 1975).

The biggest advance in astrostatistics before the era of computing was in 1805 when Legendre published the method of least squares regression to model the orbit of comets (Feigelson & Babu 2004). He theorized that the model best fit to a set of data was one that minimized the sum of the squares of the errors. Though Legendre did not provide a formal proof of the method, regarding it only as a convenient trick, later works by Robert Adrain developed formal mathematical proofs of the method (Merriman 1877). In 1809, Gauss published his own work on least squares, using it to calculate the orbit of the dwarf planet Ceres. Controversially, he also insisted that he had discovered

the method years before Legendre (Stigler 1981). Through its impact, least squares regression has cemented itself in history as one of the most important leaps in astrostatistics.

The recurring theme is clear: progress in astronomy often hinges on solving problems of statistical estimation. By the end of the century, astronomy had firmly established itself as a quantitative science, driven by the refinement of statistical methods to identify regularities in scattered measurements, fitting orbital models, and quantifying uncertainty in the presence of noise.

1.2. *The Impact of Astrophysics*

Later years brought two developments that reshaped the relationship between astronomy and statistics: the rise of physics as the foundation of astronomy and the advent of computing, which enabled unprecedented scales of data analysis. As astronomy grew increasingly intertwined with the theories of physics, the field transformed into what we now call astrophysics. Though a niche field called statistical astronomy persisted, the majority of astronomers made insufficient use of statistics in their work (Feigelson & Babu 2004). The focus shifted to deriving physical models from first principles, and statistical methods were often seen as secondary or even unnecessary. Hubble (1930) determined the fit for the light curve of elliptical galaxies by trial-and-error instead of regression. Zwicky (1937) first observed dark matter using a curve fitted only by eye.

The disdain for statistics stemmed from most astrophysicists' stubborn adherence to Newtonian determinism. Physics was regarded as the fundamental law of nature and an elegant basis for astronomy, while statistics was seen as rough, approximate, and imperfect. Statistics also flourished in the social sciences, further alienating it from astrophysics.

However, statistics would not be kept away from astronomy for long. As computing machines developed, astronomers increasingly adopted new tools for both calculation and simulation. In the 1920s, one of the earliest applications appeared in the production of lunar tables. Previously, astronomers calculated the position of the Moon using complex, error-prone methods that required extensive manual computation (Duncombe 1988). By the 1930s, however, Comrie (1932) demonstrated how punch card computing machines could automate the process, making lunar ephemerides faster and more reliable. From then on to the 1970s, Comrie's work was continued by Wallace Eckert, who improved the punch-card calculations using IBM computers (Olley 2018). Like lunar calculations, galactic simulations were also performed with computational devices as early as the 1940s. Holmberg (1940) modeled gravitational interactions using lightbulbs to represent galaxies, demonstrating how spiral structures could emerge. These analog demonstrations laid the groundwork for digital computer N-body simulations in the 1970s, such as Toomre & Toomre (1972), who explained tidal tails and bridges in interacting galaxies.

With these advances in computing came a natural resurgence of statistical methods in astronomy. Given the ability to automate calculations, handle larger datasets, and simulate complex systems, statistical analysis became indispensable. In the mid-20th century, the growth of galaxy surveys encouraged quantitative modeling of structure and dynamics. Early work such as Lynden-Bell (1967) applied statistical mechanics to stellar systems, laying the foundation for the study of galaxy formation and equilibrium. By the 1970s, statistics was increasingly recognized as a distinct methodological pillar of astronomy. Peebles (1973) systematically catlogued and analyzed extragalactic objects using power spectra and correlation functions, pioneering the statistical study of large-scale structure.

While the early history of the field was dominated by statistical reasoning and theory, the growth of physics and digital computation broadened this into what we now call quantitative analysis (QA). Quantitative analysis thus represents the merging of three traditions that once stood apart: the deductive rigor of physics, the inferential power of statistics, and the scalability of computation. In modern astronomy, progress often relies not on one of these strands in isolation but on their integration. QA therefore serves as both a methodological framework and a philosophy of practice, emphasizing reproducibility, uncertainty quantification, and the ability to extract physical meaning from complex data.

1.3. *The Data Deluge*

Today, astrophysics sits in a universe of complex statistical problems that demand new quantitative approaches and more computing power with each passing day. These are compounded by an unprecedented era of astronomical data generation in the 21st century. Sky surveys like Gaia DR3 alone provide astrometry and photometry for nearly two billion stars, plus more than ten million variable sources (Gaia Collaboration et al. 2023). The nineteenth data release of the Sloan Digital Sky Survey collected spectroscopic data from over 6 million objects (Collaboration et al. 2025). Advances in CCD detectors will see data from sky surveys increase in the next decade from gigabytes to terabytes today, and possibly to petabytes in the near future. The same trend can be seen in data from NASA's Solar Dynamics Observatory, which now generates over a terabyte of data per day, and the Rubin LSST, generating close to 30 terabytes per day (Borne 2009). Compared to the Henry Draper Catalogue (Cannon & Pickering 1918)—a century-old counterpart that cataloged roughly 200,000 stars—the explosion in data is striking.

The leap from hundreds of thousands of stars in the Henry Draper Catalogue to billions in Gaia represents more than a change in scale: it is a qualitative transformation in what science becomes possible. With the Draper Catalogue, astronomers could classify stellar spectra and trace broad patterns in stellar populations. With Gaia, it is now possible to reconstruct the full three-dimensional structure and kinematics of the Milky Way, identify hypervelocity stars, and test theories of Galactic evolution with unprecedented detail. Where older surveys allowed the identification of a few rare stellar types, modern surveys allow systematic searches for extreme outliers across billions of objects, transforming the statistical character of astronomy.

This data deluge makes QA indispensable. Large-scale surveys now span the entire electromagnetic spectrum, from radio (e.g., LOFAR, ALMA) to X-ray and gamma-ray observatories such as Chandra and Fermi. Multi-messenger astronomy adds yet another layer, with gravitational waves detected by LIGO/Virgo and high-energy neutrinos from IceCube (Abbasi et al. 2023). Time-domain surveys such as ZTF and the upcoming LSST produce streams of transient and variable sources, producing large data volumes and data rates. Each dataset has distinct noise properties, resolutions, and systematic biases, making multi-wavelength integration a formidable statistical task. The ability to extract meaningful insights from these massive datasets is crucial for advancing our understanding of the universe.

1.4. *Statistical Challenges in Modern Astrophysics*

Two recurring types of statistical challenges emerge across astrophysics. The first challenge is that noisy, incomplete, and often degenerate data have uncertain theoretical statistical distributions. In

addition, the number, complexity, and degeneracy of physical parameters poses major challenges (Schafer 2015). For example, in exoplanet studies, radial velocity measurements can only determine a planet's minimum mass because they do not contain orbital inclination (Lovis & Fischer 2010). In cosmology, measurements of the cosmic microwave background couple the effects of the Hubble constant, the amount of ordinary matter, and dark energy, so isolating any one factor requires prior assumptions about the others (Christensen & Meyer 2022). Even within galaxies, rotation curve studies must weight the balance between visible stars and invisible dark matter. Each of these cases requires careful statistical treatment to avoid misleading conclusions.

The second challenge is that the large volume of data and model complexity creates equally daunting problems of computing time and power. As noted above, the Rubin Observatory LSST will generate tens of terabytes of imaging data per night (Borne 2009), while Gaia has already released petabyte-scale catalogs (Gaia Collaboration et al. 2023). Brute-force exploration of parameter spaces is simply impossible at these scales. Efficient algorithms and scalable statistical methods are required to render analysis computationally tractable (Huijse et al. 2014). Together, these issues create a need for QA frameworks that can handle degeneracy in complex parameter spaces, model complexities, and scale efficiently with massive datasets.

1.5. *Bayesian Inference and MCMC in Context*

In this domain, Bayesian inference via Markov Chain Monte Carlo (MCMC) methods naturally emerges as a potential solution. Bayesian inference offers a principled framework for parameter estimation in complex systems. MCMC methods provide an effective way to explore parameter spaces by sampling from posterior distributions. This has become one of the most widely used and versatile approaches (Von Toussaint 2011). Computational models are becoming far more complex, and the data being analyzed is often noisy and incomplete. Bayesian methods, with their ability to incorporate prior parameter constraints and handle uncertainty, are particularly well-suited to these challenges. MCMC methods, in particular, provide a practical way to sample from complex posterior distributions that arise in Bayesian analysis. This makes them invaluable for parameter estimation, model comparison, and uncertainty quantification.

Another reason for the appeal of Bayesian methods is their contrast with frequentist approaches. Frequentist methods, which were dominant through much of the 20th century, emphasize point parameter estimates and confidence intervals derived from repeated sampling arguments (Trotta 2008). Bayesian inference, in contrast, provides full posterior probability distributions for parameters, naturally incorporating prior information from physics or earlier observations. This framework is particularly powerful where data are sparse, noisy, and incomplete.

The wider adoption of Bayesian statistics in astronomy gained momentum in the late 20th century. Cosmologists have applied Bayesian methods to the cosmic microwave background to extract cosmological parameters from noisy sky maps (Tegmark et al. 1997). In exoplanet science, Bayesian inference became common in the 1990s and 2000s for modeling radial velocity curves and transit signals in the era of larger and more precise datasets (Gregory 2005). Pulsar timing, supernova cosmology, and gravitational lens modeling similarly saw wider use of Bayesian methods. The trend reflects a broader recognition that many of astronomy's hardest problems demand not just point estimates, but principled uncertainty quantification.

We review here Bayesian inference and MCMC within an astrophysical perspective because of their flexibility, principled uncertainty quantification, and growing ubiquity across astrophysics. In Section II, the methodology section develops a working foundation: here we review Bayesian statistics, priors, and likelihood construction in realistic astronomical settings. We cover toy examples before escalating to domain-relevant formulations. Then, we introduce Monte Carlo methods and build to Markov Chain Monte Carlo, deriving the Metropolis family and related samplers, providing step-by-step Python implementations on simple problems.

The next three sections present focused case studies that illustrate how Bayesian–MCMC pipelines advance frontiers in different subfields. Section III outlines the observational context and challenges facing the direct detection of exoplanets, such as disentangling planetary signals from stellar activity, classifying threshold for non-detections, and classifying imaged exoplanets. It then outlines the methodology and advantages of Bayesian solutions to these problems. Section IV covers CMB parameter estimation, considering geometric degeneracies in the power spectrum. It explores how parallelizable frameworks for MCMC and Bayesian treatments of neural networks enable efficient posterior calculations, including tradeoffs between sampler sophistication and wall-clock efficiency. Finally, Section V covers a non-exhaustive set of other astrophysical applications, including gravitational wave detection, dark matter searches, and black hole modeling.

Finally, in Section VI and VII we compare cases where Bayesian–MCMC excels with those where complementary methods are more appropriate, identify methodological gaps revealed by the case studies, and outline opportunities for future work in astrophysics and related fields that face similar statistical and computational challenges.

2. METHODOLOGY

2.1. *Bayesian Preliminaries*

The aim of Bayesian statistics is to determine $P(H|D)$, or the probability of a hypotheses H being true given data D. A hypothesis is any statement that can be true or false, and data is any information that can be used to evaluate the hypothesis. For example, for the roll of a single die, an hypothesis H would that the die roll is a 3. The data D is the result of the die roll.

Hypotheses live in the *hypothesis space*, which is the set of all possible hypotheses of a system (Brewer 2018). For the die example, the hypothesis space is $\{1, 2, 3, 4, 5, 6\}$. The hypothesis space will also have a probability distribution, or a *prior*, written as $P(H)$ (Brewer 2018). The prior is the probability of each hypothesis being true absent any data. For a fair die, the prior is uniform: $\frac{1}{6}$ for all H in the hypothesis space. In other words, $P(H)$ is the probability of the hypothesis being true.

The data D also has a probability distribution called the *evidence*, $P(D)$ (Brewer 2018). The evidence is the probability of the data absent any hypothesis. In the die example, $P(D)$ is $\frac{1}{6}$ for rolling a 3 on a fair die. The evidence lives in the *data space*, which is the set of all possible data outcomes. In the die example, the data space is also $\{1, 2, 3, 4, 5, 6\}$.

A *likelihood*, $P(D|H)$, is the probability of the data assuming that the hypothesis is true (Brewer 2018). In the die example, if H is that the die roll is a 3, then $P(D|H)$ is unity if the die roll is a 3 and zero otherwise. We use the prior, evidence, and likelihood to calculate $P(H|D)$, which is the probability of a hypothesis being true given the data. This is called the *posterior* (Brewer 2018). In

the die example, if you roll a 3, then $P(H|D)$ is unity if the hypothesis H is that the die roll is a 3. It is zero otherwise.

The posterior $P(H|D)$ is defined by Bayes' Theorem. It can be derived using two rules, as outlined in Cox (1946). Firstly, the probability that a hypothesis is true $P(H)$ and the probability that it is not true $P(\tilde{H})$ add up to unity:

$$P(H) + P(\tilde{H}) = 1. \tag{2.1}$$

The second rule is the product rule for conditional probabilities, which states:

$$P(H)P(D|H) = P(D)P(H|D). \tag{2.2}$$

This can be rearranged to give Bayes' Theorem:

$$P(H|D) = \frac{P(H)P(D|H)}{P(D)}. \tag{2.3}$$

2.1.1. *Example: The Double-Headed Coin*

For the previous example of a die roll, the data, the result of the die roll, completely determined the hypothesis. Bayesian statistics becomes more useful when the data are incomplete, as is often the case in astrophysics.

Consider five coins, four of which are fair, and one of which is double-headed. A random coin lands heads. We calculate the probability that the double headed coin was randomly selected.

The first step is to determine H and D from their respective spaces. The hypothesis space is {Picked Fair, Picked Double-Headed}, which can be written concisely as {fair, double}. We hypothesize that the double-headed coin was picked (H = 'double'). The data space is {heads, tails}, and D is 'heads'.

For four fair coins and one double-headed coin the priors are: $P(\text{fair}) = \frac{4}{5} = 0.8$ and $P(\text{double}) = \frac{1}{5} = 0.2$. The evidence is more complex to find in this case. Since the chance of flipping heads or tails includes the case that the double headed coin was picked, the evidence cannot be 50-50. We must calculate the probability of getting heads and tails across all the coins:

$$P(\text{heads}) = \frac{4(0.5) + 1(1)}{5} = 0.6$$
$$P(\text{tails}) = \frac{4(0.5) + 1(0)}{5} = 0.4$$

Note the implicit rule for any probability space:

$$\sum_n P(x_n) = 1, \tag{2.4}$$

where x_n is a value in the space.

The likelihood $P(\text{heads}|\text{double})$ is unity, as it is only possible to get heads from the double-headed coin. Using Bayes' Theorem (Equation 2.3):

$$P(\text{double}|\text{heads}) = \frac{P(\text{double})P(\text{heads}|\text{double})}{P(\text{heads})} = \frac{0.2}{0.6} = \boxed{0.\bar{3}}$$

Before incorporating the data, the probability of picking the double headed coin was 0.2, but by using Bayesian statistics, including information from the data increases the probability to $0.\bar{3}$. This example demonstrates the power of Bayesian statistics for incomplete data.

2.2. Bayesian Statistics in Astrophysics

In an astrophysical context, Bayes' Theorem is used to calculate the probability of *parameters*, θ, of a model rather than a hypothesis (Brewer 2018).

$$P(\theta|D) = \frac{P(\theta)P(D|\theta)}{P(D)}. \tag{2.5}$$

Consider measurements of the radial velocity of a star over time. When a start hosts an orbiting planet, both bodies orbit a common center of mass. This induces a wobble in the star's motion which is detected as periodic Doppler shifts in the star's spectrum, also known as the *radial velocity*. The radial velocity $v_r(t)$ of a star with a single planet in a strictly circular orbit can be modeled by the following equation (Lovis & Fischer 2010):

$$v_r(t) = K \sin\left(\frac{2\pi t}{T} + \phi\right), \tag{2.6}$$

where K is the velocity semi-amplitude, T is the orbital period, and ϕ is an orbital phase offset. Thus, the parameters of this model are $\theta = \{K, P, \phi\}$. This is simplified to omit the parameters for eccentricity, system velocity, and argument of periapsis.

Consider the simulated data for radial velocity of a star shown in Figure 1. Bayes' Theorem (Equation 2.5) becomes:

$$P(K, T, \phi|D) = \frac{P(K, T, \phi)P(D|K, T, \phi)}{P(D)}. \tag{2.7}$$

Assuming for simplicity that the parameters are independent, the prior can be written as:

$$P(K, T, \phi) = P(K)P(T)P(\phi). \tag{2.8}$$

Absent any information, we assume the priors are uniformly distributed in a certain range. From inspection, the velocity semi-amplitude K is between 5 and 15 m/s, the period T is between 20 and 30 days, and the phase offset ϕ is between 0 and 2π. Thus, the adopted uniform priors are:

$$P(K) = \begin{cases} \frac{1}{10} & 5 < K < 15 \\ 0 & \text{otherwise} \end{cases} \qquad P(T) = \begin{cases} \frac{1}{10} & 20 < T < 30 \\ 0 & \text{otherwise} \end{cases} \qquad P(\phi) = \begin{cases} \frac{1}{2\pi} & 0 < \phi < 2\pi \\ 0 & \text{otherwise} \end{cases}$$
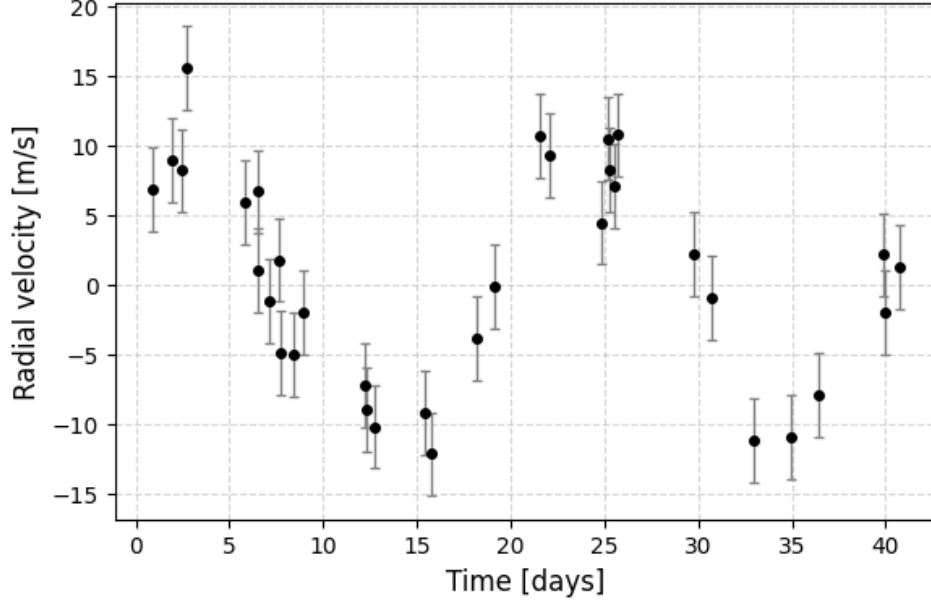
and the combined prior is then:

Figure 1. Simulated radial velocity data of a star with a single orbiting planet (Equation 2.6). Each point has an error of three standard deviations. The data are noisy and incomplete, making it difficult to determine the parameters of the model.

$$P(K, T, \phi) = \begin{cases} \frac{1}{200\pi} & 5 < K < 15, 20 < T < 30, 0 < \phi < 2\pi \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

The likelihood, $P(D|K, T, \phi)$, is the probability of the data given the parameters. For a given set of parameters, the model prediction is the mean of a normal distribution with standard deviation equal to the error of each data point. This provides the probability P_i of obtaining a data point d_i at time t_i. The likelihood for the full dataset is:

$$P(D|K, T, \phi) = \prod_{i=1}^{N} P_i \tag{2.10}$$

This takes the form:

$$P(D|K, T, \phi) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(D_i - v_r(t_i; K, T, \phi))^2}{2\sigma_i^2}\right) \tag{2.11}$$

As a probability, the posterior integrated over the entire parameter space must equal unity. Thus, the evidence $P(D)$ can be found as follows:

$$1 = \iiint P(K,T,\phi|D)dKdTd\phi$$

$$1 = \iiint \frac{P(K,T,\phi)P(D|K,T,\phi)}{P(D)}dKdTd\phi$$

$$\therefore P(D) = \iiint P(K,T,\phi)P(D|K,T,\phi)dKdTd\phi \tag{2.12}$$

This integral is analytically intractable, so it must be calculated numerically.

Using the data in Figure 1, the posterior distribution can be calculated using the code in Appendix A. The code uses the log-prior and log-likelihood to avoid any underflow errors. The posterior can be calculated by adding the log-prior and log-likelihood, and then exponentiating the result, then normalizing with the evidence.

$$P(K,T,\phi|D) = \frac{e^{\log(P(K,T,\phi))+\log(P(D|K,T,\phi))}}{P(D)} \tag{2.13}$$

The program evaluates the posterior on a grid of 100 points for each parameter, giving a total of $100^3 = 1,000,000$ points. The posterior is also cumulatively summed at every evaluated point, giving a numerical estimate of the evidence $P(D)$. Finally, the marginal posteriors of each parameter are calculated from the total posterior by integrating over the other two parameters:

$$P(K|D) = \iint P(K,T,\phi|D)dTd\phi \tag{2.14}$$

$$P(T|D) = \iint P(K,T,\phi|D)dKd\phi \tag{2.15}$$

$$P(\phi|D) = \iint P(K,T,\phi|D)dKdT \tag{2.16}$$

The resulting posterior distributions are shown in Figure 2.

The exact parameters of a model are always unknown, but posterior distributions field a close estimate. The true parameters, $K = 10$ m/s, $T = 21.4$ days, and $\phi = \frac{\pi}{4}$, fall within one standard deviation of the mean of each posterior distribution (Figure 2), indicating that the Bayesian method was successful in estimating the parameters.

Figure 3 shows the true model and the mean parameter prediction from the posterior distributions. Bayes' Theorem is thus useful to estimate the parameters for astrophysical models. However, this method is computationally expensive. Computing the posterior numerically requires integrating over the full parameter space. For n parameters and m possible values for each parameter, this algorithm would have a time complexity of $O(m^n)$, which is infeasible for even small n and m. To solve this problem, we can use *Monte Carlo methods* (Brewer 2018).

2.3. *Monte Carlo Sampling*

As noted above, the primary computational challenge in brute force Bayesian inference arises from evaluating the posterior probability at every point in the parameter space. One way to reduce this
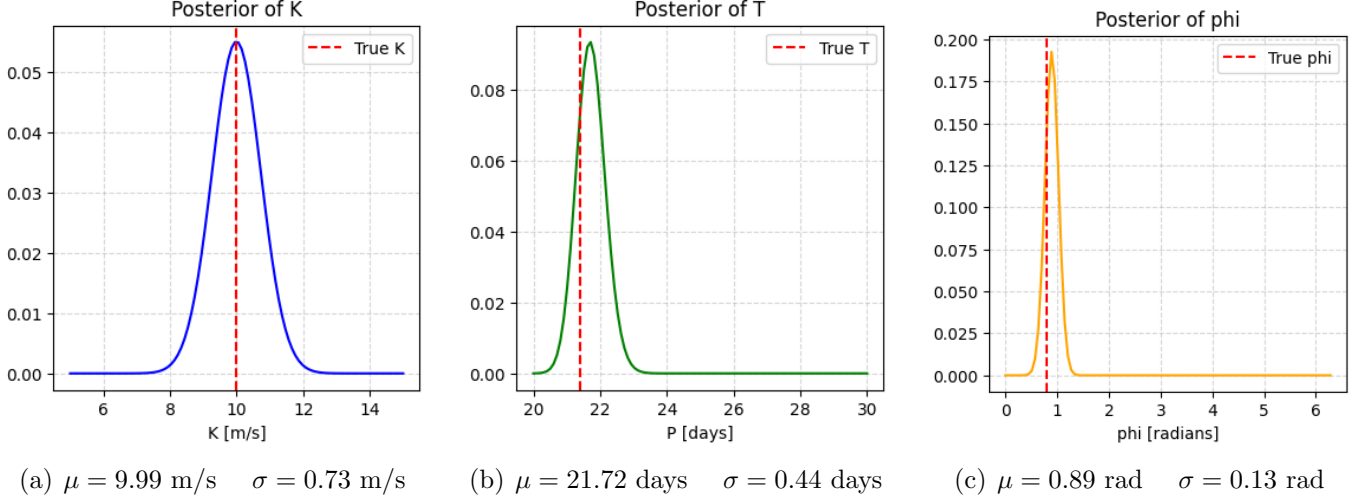
(a) $\mu = 9.99$ m/s $\sigma = 0.73$ m/s   (b) $\mu = 21.72$ days $\sigma = 0.44$ days   (c) $\mu = 0.89$ rad $\sigma = 0.13$ rad

Figure 2. The marginal posterior distributions of the parameters K, T, and ϕ given the data in Figure 1. These posteriors are normalized with a numerically calculated evidence.
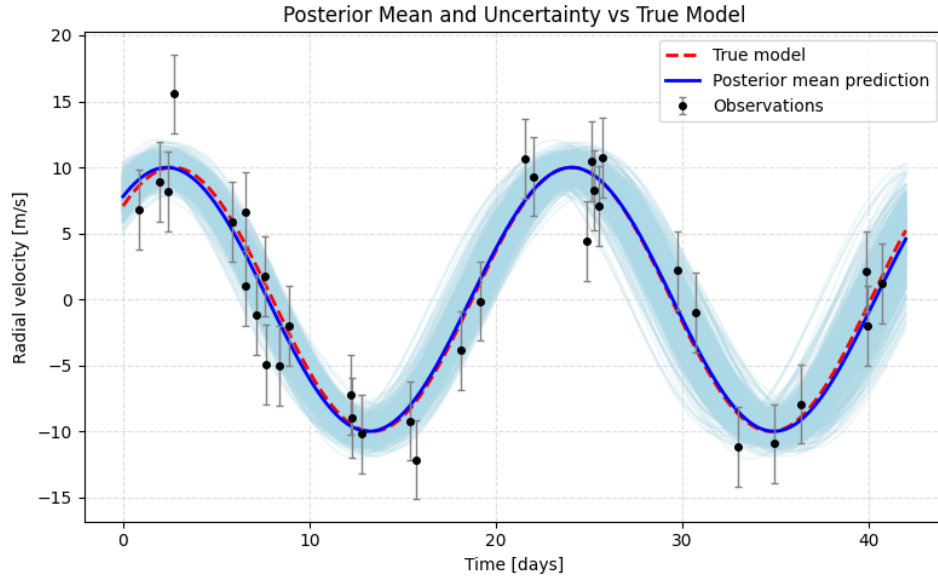

Figure 3. The radial velocity data from Figure 1 along with the mean prediction from the posterior distributions in Figure 2. The shaded region represents one standard deviation from the mean prediction.

cost is to sample only a subset of points, distributed according to the posterior itself (Von Toussaint 2011). These points are called *draws*. These draws make it possible to calculate expectation values from the explicit posterior using only the sampled points (Von Toussaint 2011). This defines *Monte Carlo sampling*, which can be used to approximate integrals and expectations that would otherwise require explicit evaluation over a continuous space.

For a known probability density, the expectation value of some variable x can be calculated as:

$$\langle x \rangle = \int x P(x|\theta) dx. \tag{2.17}$$

x	Description
θ_i	Expectation value of parameter θ_i
θ_i^2	Variance of parameter θ_i
$f(\theta_1, \theta_2, \ldots, \theta_n)$	Expectation value of any function of the parameters
$\mathbb{1}(\theta_i = a)$	Probability that $\theta_i = a$
$\mathbb{1}(a \leq \theta_i \leq b)$	Probability that θ_i is in the range $[a, b]$
$\mathbb{1}(\text{condition})$	Probability that the condition is true

Table 1. Forms of x to calculate expectation values of various quantities.

For N values of x drawn from P using Monte Carlo, this can be discretely approximated as:

$$\langle x \rangle \approx \frac{1}{N} \sum_{i=1}^{N} x_i, \tag{2.18}$$

which generalizes naturally to higher-dimensional parameter spaces, though convergence can become slower as dimensionality increases (Von Toussaint 2011).

x is a function of the parameters, and can be defined arbitrarily to calculate the expectation value of any value of interest. Given a probability density $P(\theta_1, \theta_2, \ldots, \theta_n)$, some forms of x are listed in Table 1.

The usefulness of Monte Carlo sampling becomes apparent when considering the computation of marginal posteriors. In the previous example, we used the posterior to calculate the marginal posteriors of each parameter. In the brute force method, this was found by integrating the posterior over all other parameters. Monte Carlo makes this simpler. If we have a set of N draws from the posterior of a parameter space $\{\theta_1, \theta_2, \ldots, \theta_n\}$, we can obtain draws for a certain parameter θ_i by ignoring all other parameters (Brewer 2018). A histogram of the draws of θ_i then gives the marginal posterior of θ_i. This is much simpler than integrating over all other parameters, especially in high-dimensional parameter spaces.

2.4. *Markov Chain Monte Carlo*

Markov Chain Monte Carlo (MCMC) is one of the many methods used to obtain posterior draws, and in modern astrophysics, has become a standard tool for Bayesian inference. Using MCMC, the posterior of even a high-dimensional parameter space can computed efficiently (Trotta 2008). MCMC does this by creating a sequence of draws in the form of a *Markov Chain* (Neal 1993).

2.4.1. *Markov Chains*

A Markov Chain is a sequence of random variables, $x_0, x_1, \ldots, x_n$, where the probability of each variable only depends on the previous variable (Neal 1993). This is known as the *Markov property*, and can be written mathematically as:

$$P(x_{n+1} | x_n, x_{n-1}, \ldots, x_0) = P(x_{n+1} | x_n). \tag{2.19}$$

To create a Markov chain, we must first decide an initial value x_0 using an initial probability distribution $P_0(x_0)$. The probability distribution of the next value is evolved iteratively from the previous probability distribution using a *transition probability* (Neal 1993) $T_n(x_n, x_{n+1})$:

$$P(x_{n+1}) = P(x_n)T_n(x_n, x_{n+1}) \tag{2.20}$$

If the transition probability does not depend on the point in the chain, the chain is *homogenous*, or *stationary* (Neal 1993).

Markov chains can be shown to converge to a *stationary distribution* $P^*(x)$, which is independent of the initial distribution (Trotta 2008). A stationary distribution, once reached, does not change as the chain evolves further.

Markov chains can also be *ergodic* (Neal 1993), which means that a stationary distribution can be reached from any initial distribution. A ergodic chain is irreducible and aperiodic (Neal 1993). Irreducibility means that it is possible to reach any point in the space from any other point, and aperiodicity means that the chain does not get stuck in cycles (Von Toussaint 2011).

For Bayesian inference, the goal of MCMC is to construct a Markov chain whose stationary distribution is the unknown posterior distribution (Brewer 2018). To ensure that the chain converges to the desired posterior, the choice of transition probability $T(x_n, x_{n+1})$ is crucial. The transition rule must satisfy the condition of *detailed balance*, which guarantees that, at equilibrium, the probability flow between any two states is symmetric:

$$P(x_n)T(x_n, x_{n+1}) = P(x_{n+1})T(x_{n+1}, x_n). \tag{2.21}$$

This condition ensures that the chain does not drift away from the target distribution once it has been reached (Von Toussaint 2011).

2.4.2. *The Metropolis-Hastings Algorithm*

One of the simplest and most widely used algorithms that satisfies detailed balance is the *Metropolis algorithm* (Metropolis et al. 1953), more specifically, its generalization, the *Metropolis–Hastings algorithm* (Hastings 1970). The algorithm constructs a Markov chain by iteratively proposing new states in the parameter space and deciding whether to accept or reject them based on the target distribution. Brewer (2018) outlines the steps of the Metropolis-Hastings algorithm as follows:

1. Initialize the chain with a starting point x_0.

2. Generate a candidate point x_{n+1} from a proposal distribution $q(x_{n+1}|x_n)$ based on the current state x_n.

3. Accept or reject the proposal

4. Repeat steps 2 and 3 for a large number of iterations to generate a sequence of samples until convergence.

The algorithm depends on the proposal distribution $q(x_{n+1}|x_n)$ and the acceptance criterion. Commonly, a 'random walk' proposal is used to generate candidates by adding a small random perturbation to the current state. For an n-dimensional parameter space, the random walk proposal is a

symmetric (usually Gaussian) distribution, also of n-dimensions, with the current state as its mean (Von Toussaint 2011). If the proposal distribution is indeed symmetric, then the following is implied:

$$q(x_{n+1}|x_n) = q(x_n|x_{n+1}). \tag{2.22}$$

In this case, the Metropolis-Hastings algorithm simplifies to the original Metropolis algorithm (Brewer 2018). Asymmetric proposal distributions are more general, and can also be used. However, they are less common in practice due to their specificity to the problem.

The acceptance criterion is based on the ratio of the posterior at the proposed and current states, adjusted by the proposal distribution. This is the probability of the proposal being accepted. Mathematically, this is (Brewer 2018):

$$\alpha = \min\left(1, \frac{P(x_{n+1}|D)q(x_n|x_{n+1})}{P(x_n|D)q(x_{n+1}|x_n)}\right) = \min\left(1, \frac{P(D|x_{n+1})P(x_{n+1})q(x_n|x_{n+1})}{P(D|x_n)P(x_n)q(x_{n+1}|x_n)}\right). \tag{2.23}$$

If the proposal distribution is symmetric, we can further apply Equation 2.22 to obtain (Brewer 2018):

$$\alpha = \min\left(1, \frac{P(D|x_{n+1})P(x_{n+1})}{P(D|x_n)P(x_n)}\right). \tag{2.24}$$

Simplified with log-priors and log-likelihoods:

$$\log(\alpha) = \min\left(0, \log(P(D|x_{n+1})) + \log(P(x_{n+1})) - \log(P(D|x_n)) - \log(P(x_n))\right). \tag{2.25}$$

This criterion accepts any proposal that increases the posterior probability, while proposals that decrease it are accepted with a probability proportional to the decrease. If a proposal is rejected, the current state is counted again. This allows the chain to sample high-probability regions more frequently while still having the ability to escape local maxima and explore the full parameter space (Brewer 2018).

The choice of proposal distribution can also significantly affect the efficiency of the algorithm (Von Toussaint 2011). If the proposal steps are too small, the chain will explore the parameter space slowly, leading to high correlation between samples. Conversely, if the steps are too large, many proposals will be rejected, also resulting in inefficient sampling. Tuning the proposal distribution to balance exploration and acceptance rates is often necessary for optimal performance (Von Toussaint 2011). A common heuristic is to adjust the proposal distribution to achieve an acceptance rate between 0.2 and 0.5, depending on the dimensionality of the parameter space (Gelman et al. 1997).

The Metropolis-Hastings algorithm has a *burn-in* period. The initial samples of the chain may not be representative of the target distribution, especially if the starting point is far from high-probability regions. To mitigate this, it is common practice to discard a certain number of initial samples, from the burn-in period, before using the remaining samples for inference (van Ravenzwaaij et al. 2018).

The Metropolis-Hastings algorithm outputs a Markov chain of samples that approximate the target posterior distribution (Hastings 1970). Since the probability is only calculated for proposed points instead of the full parameter space, the algorithm is much more computationally efficient than brute force methods. The time complexity is reduced to $O(n*m)$, where n is the number of samples drawn and m is the number of walks, which is feasible for even high-dimensional parameter spaces.

2.5. *MCMC in Astrophysics*

Revisiting the radial velocity example, we can use MCMC to obtain draws from the posterior distribution of the parameters $\{K, T, \phi\}$ from Equation 2.6 given the data in Figure 1. Using the Metropolis-Hastings algorithm, 10 chains were run, each with 12,000 samples. The first 1,000 samples of each chain were discarded as burn-in. The code used to run the MCMC is given in Appendix B.
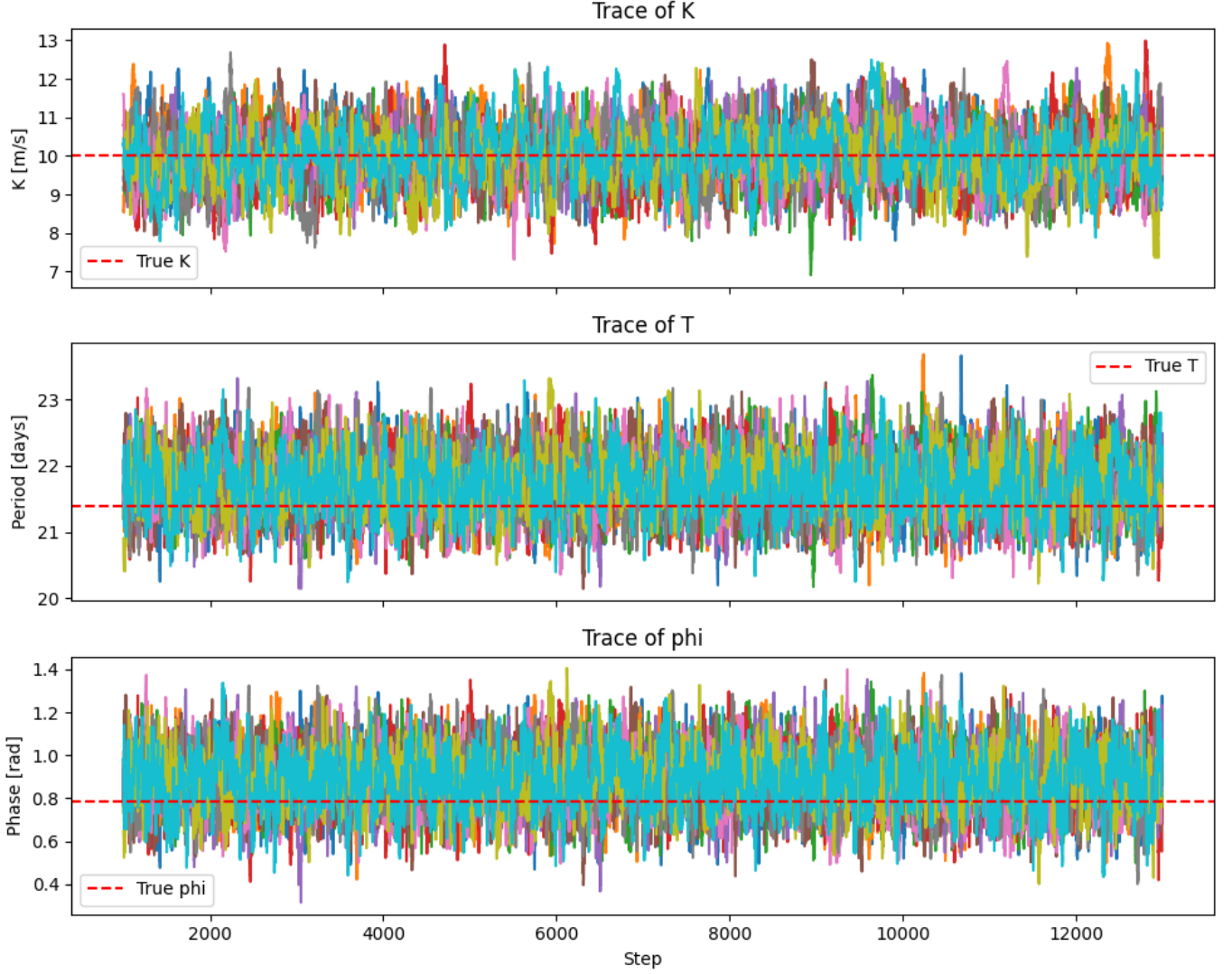

Figure 4. Trace plots of the Markov chains for each parameter. A total of 10 chains were run, each with 12,000 samples. The first 1,000 samples of each chain were discarded as burn-in.

The trace plots of the chains are shown in Figure 4. These are visualizations of the Markov chains for each parameter. The chains appear to be well-mixed and stationary, indicating that they have converged to the target distribution. The marginal posterior distributions of each parameter is shown in Figure 5.

Comparing these posteriors to the ones obtained using the brute-force method, we see that they are very similar. Table 2 summarizes the comparison between the true parameter values, the estimates
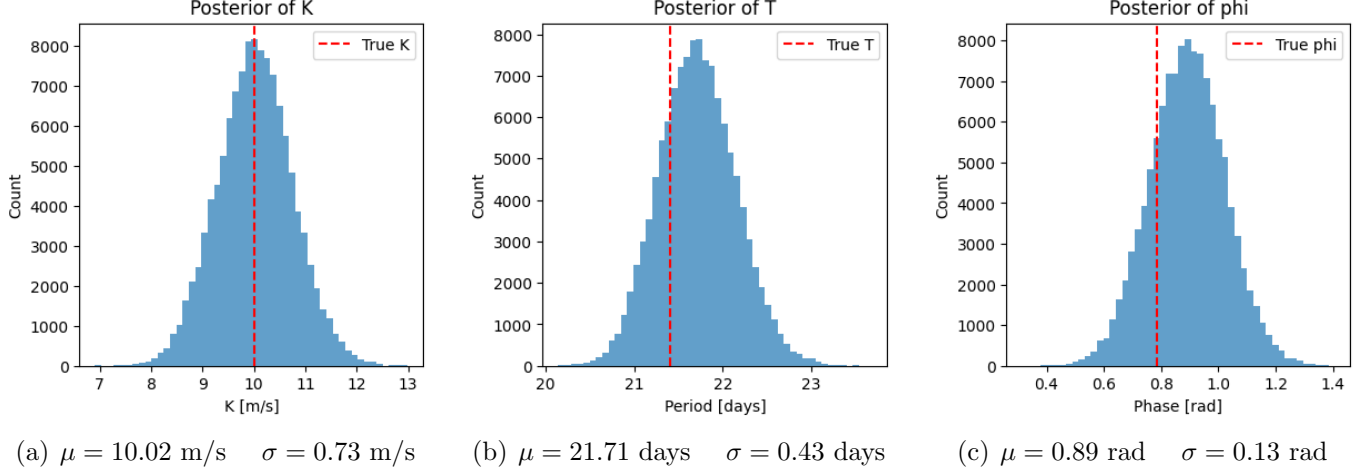
(a) $\mu = 10.02$ m/s $\sigma = 0.73$ m/s   (b) $\mu = 21.71$ days $\sigma = 0.43$ days   (c) $\mu = 0.89$ rad $\sigma = 0.13$ rad

Figure 5. The histograms of the draws from the posterior distributions of the parameters K, T, and ϕ given the data in Figure 1.

Parameter	True	Brute-Force	Brute-Force Z-score	MCMC	MCMC Z-score
K (m/s)	10.00	9.99 ± 0.73	-0.01	10.02 ± 0.73	0.03
T (days)	21.4	21.72 ± 0.44	0.68	21.71 ± 0.43	0.66
ϕ (rad)	0.25π	0.89 ± 0.13	0.77	0.89 ± 0.13	0.77

Table 2. Comparison of the true parameter values, the estimates from the brute-force method, and the estimates from the MCMC method.

from the brute-force method (Figure 2), and the estimates from the MCMC method (Figure 5). The two algorithms are compared using the mean and standard deviation of each posterior distribution, as well as the Z-score of the true value from the mean of each posterior distribution. The Z-score is a measure of how many standard deviations a value is from the mean, and is calculated as:

$$Z = \frac{X - \mu}{\sigma}, \tag{2.26}$$

where X is the value being compared, μ is the mean of the distribution, and σ is the standard deviation of the distribution. A Z-score of zero indicates that the value is equal to the mean, while a Z-score of 1 indicates that the value is one standard deviation above the mean. A Z-score of -1 indicates that the value is one standard deviation below the mean. A Z-score within the range of -2 to 2 is generally considered to be acceptable, as it indicates that the value is within two standard deviations of the mean.

Both methods give very similar results (Table 2). The Z-scores of the true values are all within 1, indicating that both methods were successful in estimating the parameters. The MCMC method, however, was far more computationally efficient. As described above, the brute-force method had to evaluate the posterior at $100^3 = 1,000,000$ points in the parameter space. The MCMC method, as implemented in Appendix B, only had to evaluate the posterior at $10 \times 12,000 = 120,000$ points, a reduction of nearly an order of magnitude, to provide virtually the same results.

Generalizing this to higher-dimensional parameter spaces, the computational savings of MCMC become even more apparent. In a hypothetical 10-dimensional parameter space with 100 possible values for each parameter, the brute-force method would have to evaluate the posterior at $100^{10} = 10^{20}$ points, which is infeasible. The MCMC method, on the other hand, would still provide reasonably accurate results with no significant increase in computing cost. The number of required samples and walks increases with the number of dimensions based on the discretion of the experimenter. Usually, this increase is sub-exponential, even linear with respect to an n-dimensional parameter space.

2.6. *Applications in Research*

Using Bayesian inference and MCMC in astrophysics is far more complex than the radial velocity example given above. Real-world astrophysical models often have many more parameters, complex likelihood functions, and intricate prior distributions. In practice, astrophysical analyses must contend with challenges such as correlated noise in observations, hierarchical modeling of populations, and significant computational bottlenecks when scaling to high-dimensional or large datasets. The adaptability of the Bayesian–MCMC framework makes it useful in these conditions to tackle problems of parameter estimation, model selection, and uncertainty quantification across various astrophysical subfields.

In the following sections, we will explore some case studies where Bayesian inference and MCMC have been successfully applied, including exoplanet detection, cosmological parameter estimation, and gravitational wave data analysis.

3. DIRECT IMAGING OF EXOPLANETS

3.1. *Introduction*

Exoplanets are planets that orbit stars outside our solar system, with each being a unique laboratory for studying planetary formation and evolution (Kaushik et al. 2025). Studying their atmospheres, compositions, and orbital dynamics can provide insights into the conditions necessary for life and the diversity of planetary systems in the universe. These properties are key to understanding not only the planets themselves but also the broader processes that govern planetary system formation and evolution (Madhusudhan 2019).

Exoplanets are detected using various methods, including transit photometry, radial velocity measurements, direct imaging, and other less common techniques (Wei 2018). Each method probes a different region of exoplanet parameter space, with transit and radial velocity methods being most sensitive to close-in planets, while direct imaging is sensitive to young, massive planets at wide separations (Fischer et al. 2014). Of these, direct imaging is the most straightforward in concept, as it does not rely on indirect signatures. The brightness of an exoplanet, whether reflected starlight or thermal emission, is directly imaged by a telescope (Kaushik et al. 2025). This approach enables direct spectroscopic analysis, offering unique access to the planet's atmospheric composition, temperature, and cloud structure. Although the yield of direct imaging is lower than that of indirect methods, directly imaged exoplanets constitute a crucial subset of targets for atmospheric and dynamical characterization (Currie et al. 2023).

Direct imaging has achieved several notable successes since its first confirmed detections in the early 2000s. The HR 8799 system, imaged by Marois et al. (2008), revealed four massive exoplanets orbiting a young A-type star, marking a milestone in multi-planet imaging. Other examples include

β Pictoris b (Lagrange et al. 2010) and 51 Eridani b (Macintosh et al. 2015), both of which have become benchmark systems for understanding the atmospheres of young gas giants. More recently, the James Webb Space Telescope (JWST) [**ADD FOOTNOTE**] and ground-based instruments such as SPHERE [**ADD FOOTNOTE**] on the Very Large Telescope (VLT) [**ADD FOOTNOTE**] and GPI [**ADD FOOTNOTE**] on Gemini South have achieved higher contrasts and enabled spectroscopy of fainter, cooler companions such as HIP 65426 b (Carter et al. 2023).

However, direct imaging faces substantial observational and technical challenges. The primary limitation arises from the extreme contrast between the stellar and planetary brightness: even a bright exoplanet is typically 10^9 times fainter than its host star (Fischer et al. 2014). A dominant obstacle is also the *point spread function* (PSF), which is the diffraction pattern created by the telescope optics (Fischer et al. 2014). The PSF spreads starlight over the detector and can easily obscure faint companions (Fischer et al. 2014). PSFs come in various shapes, ranging from circular Airy rings to more complex forms depending on the specifics of the telescope aperture (Fischer et al. 2014). Various image processing techniques are employed to enhance the visibility of the exoplanet against the stellar background. These techniques can be broadly categorized into *preprocessing* and *postprocessing* methods.

3.1.1. *Preprocessing methods*

Preprocessing methods aim to reduce the intensity of the central PSF and improve the contrast between the star and the exoplanet. This was first performed using a *coronagraph* (Lyot 1939). This optical device is placed in the light path of the telescope to block out the light from the central star, allowing the much dimmer light from surrounding objects, such as exoplanets, to become more prominent. This reduces the intensity of the central star by up to 2 orders of magnitude (Chauvin 2023).

Changing the shape of the telescope aperture can also advantageously change the shape of the PSF. Zanoni & Hill (1965) argue for a square aperture, thereby constaining the PSF to the x and y-axes and suppressing the diagonals, which reduce the intensity of diffracted light from the central star by four orders of magnitude.

Modern preprocessing methods include *apodization*, which is a technique to reduce the brightness of the PSF without blocking out the host star. This is accomplished by altering the shape and transmission of a telescope aperture (Nisenson & Papaliolios 2001). A number of apodization techniques are used, ranging from pupil-plane masks (Reddy & Hashemi 2018), to phase-induced approaches (Guyon et al. 2005).

3.1.2. *Postprocessing methods*

Once an image has been acquired, postprocessing methods play a crucial role in distinguishing faint planetary signals from residual starlight. These techniques operate on the recorded data to remove the effect of the PSF, revealing features masked by the intense starlight (Lafreniere et al. 2007). A simple approach is *reference differential imaging* (RDI), which involves taking images of a reference star with similar properties to the target star. The reference image is then subtracted from the target image to remove the stellar contribution (Follette 2023). However, in the presence of temporal variations in the atmosphere and instrument, RDI may not accurately capture the PSF variations, leading to residual artifacts.

To account for this, more advanced techniques such as *angular differential imaging* (ADI) (Marois et al. 2006) and *spectral differential imaging* (SDI) (Follette 2023) have been developed. ADI leverages the rotation of the field of view during observations to distinguish between static PSF features and rotating planetary signals (Marois et al. 2006). SDI, on the other hand, exploits the spectral differences between the star and planet by observing at multiple wavelengths, allowing for differential subtraction based on color contrasts (Follette 2023).

In addition to these methods, statistical PSF modeling techniques have become widely adopted. Locally Optimized Combination of Images (LOCI) (Lafreniere et al. 2007) constructs a linear combination of reference frames that minimizes residual noise within localized regions of the image. Principal Component Analysis (PCA) or Karhunen-Loève Image Projection (KLIP) extended this concept, representing the PSF as a combination of orthogonal basis images derived from large reference libraries (Follette 2023).

3.2. *Challenges*

Even after filtering out the stellar emission, several challenges remain in the direct imaging of exoplanets. One major challenge is the presence of *speckle noise*, which arises from residual wavefront errors (microscopic imperfections in the telescope aperture) and residual refractive atmospheric errors (Fischer et al. 2014). These errors create discrete speckles that can mimic the appearance of point sources, making it difficult to distinguish between true planetary signals and noise artifacts (Fischer et al. 2014). Furthermore, overfiltering during PSF subtraction can lead to *self-subtraction*, where part of the planet's signal is inadvertently removed along with the stellar PSF, leading to *non-detections* (Kaushik et al. 2024). This effect is particularly pronounced for planets close to their host stars.

Furthermore, spectroscopy of directly imaged exoplanets requires significant observational time to classify planetary features (Ruane et al. 2017; Bixel & Apai 2019). This is further exacerbated by the need to observe multiple spectral bands to capture the full range of atmospheric features, which can be time-consuming and resource-intensive.

Bayesian inference and MCMC methods provide a powerful framework to address these challenges in direct imaging. Rather than relying solely on algorithmic subtraction, modern postprocessing analyses construct generative residual models of the image, allowing the use of Bayesian hypothesis testing and MCMC sampling to infer contrasts, spectra, and uncertainties (Ruffio et al. 2018). Using these methods high-priority exoplanets are determined for follow-up spectroscopic observations based on their inferred atmospheric properties (Bixel & Apai 2019).

3.3. *Bayesian Classification of Non-Detections*

Direct imaging surveys currently produce non-detections roughly 98% of the time (Ruffio et al. 2018). These non-detections are determined by a defined upper limit to the signal-to-noise ratio (SNR), typically 5σ (Meshkat et al. 2013; Ruffio et al. 2018). SNR is calculated by dividing the measured flux density at a given location by the standard deviation of the noise in an annulus around that location (Meshkat et al. 2013):

$$\text{SNR} = \frac{F_{\text{planet}}}{\sigma(r)\sqrt{\pi r_{\text{ap}}^2}} \tag{3.1}$$

where F_{planet} is the measured flux at the location of interest, $\sigma(r)$ is the standard deviation of the noise at radius r from the star, and r_{ap} is the radius of the photometric aperture used to measure the flux.

Image features below the 5σ limit are classified as non-detections. However, this method does not account for the possibility of false negatives, where a planet is present but undetected due to noise or other factors. To address this, Ruffio et al. (2018) developed a Bayesian framework to classify non-detections, described below.

3.3.1. *Non-Detections at Known Locations*

In some cases, an exoplanet may be detected in a spectral band, but not be detected in another band, possibly due to overfiltering of the PSF or excessive noise in the image (Ruffio et al. 2018). Bayesian inference can be used to determine a more probable upper limit on the planet brightness in the non-detection band.

Assuming the planet has a true flux density F, which is unknown. At its known location, we measure a flux $\tilde{F}$ with some uncertainty σ. The posterior $P(F|\tilde{F})$ is the probability distribution of the true flux F given the measured flux $\tilde{F}$. Using Bayes' Theorem, Ruffio et al. (2018) writes this as:

$$P(F|\tilde{F}) = \frac{P(\tilde{F}|F)P(F)}{P(\tilde{F})} \tag{3.2}$$

The prior $P(F)$ is the assumption that the flux is positive:

$$P(F) = \begin{cases} \alpha & F > 0 \\ 0 & F \leq 0 \end{cases} \tag{3.3}$$

where α is a constant depending on the range of F values being considered.

Assuming Gaussian noise, the likelihood $P(\tilde{F}|F)$ is given by:

$$P(\tilde{F}|F) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\tilde{F}-F)^2}{2\sigma^2}\right) \tag{3.4}$$

which is the probability of measuring flux density $\tilde{F}$ given a true flux density F and uncertainty σ.

The evidence $P(\tilde{F})$ is a normalization constant, given by:

$$P(\tilde{F}) = \int_0^\infty P(\tilde{F}|F)P(F)dF \tag{3.5}$$

Using the posterior, the Bayesian upper limit F_{lim} can be calculated depending on a desired confidence level, or *cutoff probability* η, which is the probability that the true flux of the planet is below the upper limit (Ruffio et al. 2018). This is given by:

$$\eta = \int_0^{F_{\text{lim}}} P(F|\tilde{F})dF \tag{3.6}$$

Solving for F_{lim} gives the Bayesian upper limit on the planet's flux at the desired confidence level. Conversely, using a predefined upper limit F_{lim} will yield the corresponding cutoff probability η. This allows for (a) a more accurate upper limit on the planet's brightness in the non-detection band, and (b) a quantification of the probability of a false non-detection at 5σ.

3.3.2. *Non-Detections on Known Orbits*

The orbit of a theoretical exoplanet may be known from the existence of a gap in a protoplanetay disk (Ruane et al. 2017) or an other astrometric indicator. However, direct imaging may yield a non-detection at the expected location of the planet. Bayesian inference can be used to determine the probability that a planet is present on the known orbit.

In this case, the data D is the entire image with a non-detection at the expected location of the planet. Ruane et al. (2017) models this as a random vector:

$$D = Fm(S) + N \tag{3.7}$$

where F is the unknown flux density of the planet, $m(S)$ is the PSF model at the expected location S of the planet, and N is the noise vector in the full image. Using Bayes' Theorem, the posterior $P(F|D)$ is given by:

$$P(F|D) = \int \frac{P(D|F,S)P(F,S)}{P(D)} dS \tag{3.8}$$

Note that the prior $P(F,S)$ includes both the flux density and location of the planet. It is reasonable to assume that the flux desntiy and location are independent, so that the posterior can be written as:

$$P(F|D) = \int \frac{P(D|F,S)P(F)P(S)}{P(D)} dS \tag{3.9}$$

The flux density prior $P(F)$ matches Equation 3.3. The location prior $P(S)$, however, is based on the known orbit of the planet. The orbit defines the projected velocity of the planet $v_{\mathrm{proj}}(S)$. Using this and the orbital period T, Ruane et al. (2017) writes the location prior as:

$$P(S) = \frac{1}{T v_{\mathrm{proj}}(S)} \tag{3.10}$$

This prior gives higher probability to locations where the planet moves slowly, and lower probability to locations where the planet moves quickly.

Since the data are a random vector, Ruane et al. (2017) expresses the likelihood $P(D|F,S)$ using a multivariate Gaussian distribution:

$$P(D|F,S) = \frac{1}{\sqrt{2\pi|\Sigma|}} \exp\left(-\frac{1}{2}(D - Fm(S))^T \Sigma^{-1}(D - Fm(S))\right) \tag{3.11}$$

This equation is the matrix form of Equation 3.4, where Σ is the covariance matrix of the noise in the image, and T denotes the transpose. Assuming the noise is independent of the data, the covariance matrix is diagonal.

The evidence $P(D)$ is a normalization constant, calculated using the same integration method as Equation 3.5.

By integrating over all values of S, the Bayesian upper limit on the planet's flux F_{lim} can be calculated at a desired cutoff probability η, as in Equation 3.6.

Case	Observational Data	Information Gained
1	Direct imaging detection in a single band	
2	+ Radial velocity observations	Period and semi-amplitude
3	+ Observations of a debris disk	Orbital inclination
4	+ Multiple direct imaging epochs	Phase, semi-major axis, eccentricity
5	+ Astromentry data	Mass
6	+ Color information from multiple bands	Color-based discrimination
7	All of the above	

Table 3. The observables used to classify exo-Earth candidates. Each case includes additional information, with case 7 being the most comprehensive.

3.4. *Bayesian Classification of Exo-Earths*

Direct imaging allows spectroscopic characterization of exoplanet atmospheres (Currie et al. 2023). This is particularly important for the detection of biosignatures, which are spectral features that indicate the presence of life. Common biosignatures include oxygen, ozone, methane, and water vapor (Seager et al. 2016). The detection of these features in the atmosphere of an exoplanet could indicate the presence of life, or as evidence of habitability.

Two telescopes currently in development, the Habitable Exoplanet Observatory (HabEx) [**ADD FOOTNOTE**] and the Large UV/Optical/IR Surveyor (LUVOIR) [**ADD FOOTNOTE**], aim to directly image Earth-like exoplanets, with an expected yield of dozens of exo-Earths. However, exo-Earth detection suffers from a false-positive rate of up to 77% when using detection data from a single spectral band, and 47% with prior constrains on the planet's orbit (Guimond & Cowan 2018). Furthermore, reducing the false-positive rate by collecting data from more bands is a time-consuming process of up to days or weeks per target (Bixel & Apai 2019).

In order to minimize the telescope time required to confirm exo-Earth candidates (EECs), Bixel & Apai (2019) developed a Monte Carlo framework to optimize the observing strategy. This framework uses Bayesian inference to classify EECs based on their observed spectra to prioritize targets for follow-up observations.

The framework compares two competing hypotheses: that the planet is an EEC, or that it is not. The comparison uses both the intrinsic properties θ (that determine if a planet is an EEC) and the observable quantities x shown in Table 3. They give this as a posterior probability $P(\theta|x)$ using Bayes' Theorem:

$$P(\theta|x) = \frac{P(x|\theta)P(\theta)}{P(x)} \tag{3.12}$$

To determine the prior $P(EEC)$, Bixel & Apai (2019) constructed a Monte Carlo population of simulated planetary systems. They assigned each predicted planet a radius, mass, orbital period, eccentricity, and albedo drawn from the Kepler mission occurrence rates. From these properties, the corresponding observable quantities x_{sim} were calculated, producing a prior sample of what future imaging surveys are likely to detect.

Bixel & Apai (2019) calculated the likelihood $P(x|\theta)$ by comparing real observed quantities x_{obs} to the simulated quantities x_{sim} using a Gaussian distribution:

$$P(x|\theta) = \prod_i \exp\left(\frac{(x_{obs,i} - x_{sim,i})^2}{2\sigma_{obs,i}^2}\right) \tag{3.13}$$

where $\sigma_{obs,i}$ is the uncertainty in the observed quantity $x_{obs,i}$.

The evidence $P(x)$ is a normalization constant, calculated by integrating over all values of θ:

$$P(x) = \int P(x|\theta)P(\theta)d\theta \tag{3.14}$$

The posterior probability $P(\theta|x)$ can then be used to classify the planet as an EEC or not. For each detected planet, the subset of simulated planets consistent with its observed brightness and separation (within measurement uncertainties) is identified, and the distribution of their physical properties defines the posterior. A threshold probability can be set, above which the planet is classified as an EEC, and below which it is classified as a non-EEC. This allows for the prioritization of targets for follow-up observations, reducing the telescope time required to confirm EECs.

Overall, Bixel & Apai (2019) found that with single-band detections alone, even genuine exo-Earths cannot be identified with high confidence—typical posterior probabilities were below 50%. However, adding incorporating more information about the exoplanet provides an additional powerful discriminant between Earth-like and non-habitable planets. In Case 7 (the most data-rich scenario), the average confidence in correctly identifying exo-Earths exceeded 80–85%. This is equivalent to reducing the false-positive rate to below 20%, a significant improvement over direct imaging alone.

4. CMB PARAMETER ESTIMATION

4.1. *Introduction*

The Cosmic Microwave Background (CMB) is the residual thermal radiation from the Big Bang, permeating the entire universe. It provides a snapshot of the universe when it was just 380,000 years old, offering invaluable insights into its early conditions, composition, and evolution (Staggs et al. 2018). The CMB is remarkably uniform, with temperature fluctuations at the level of one part in 100,000. These tiny anisotropies, discovered by Smoot et al. (1992), encode information about the density fluctuations that eventually led to the formation of galaxies and large-scale structures in the universe (Planck Collaboration et al. 2020). Such structures have been attributed to the gravitational collapse of primordial overdensities, which are believed to have originated from quantum fluctuations during the inflationary epoch (Guth 1981). These overdensities are described by a power spectrum $P(k)$, which quantifies the amplitude of fluctuations as a function of spatial scale (wavenumber k) (Staggs et al. 2018).

The analysis of CMB data has been instrumental in establishing the standard model of cosmology, known as the Lambda Cold Dark Matter (ΛCDM) model. This model posits that the universe is composed of approximately 5% ordinary matter, 27% dark matter, and 68% dark energy (Planck Collaboration et al. 2016). ΛCDM is based on six fundamental parameters: the physical densities of baryons ($\Omega_b h^2$) and cold dark matter ($\Omega_c h^2$), the angular size of the sound horizon at recombination (θ_s), the optical depth due to reionization (τ), the amplitude of the primordial power spectrum (A_s),

Parameter	Value	Description
$\Omega_b h^2$	0.0224 ± 0.0001	Physical baryon density
$\Omega_c h^2$	0.120 ± 0.001	Physical cold dark matter density
θ_s	0.01041 ± 0.00001	Angular size of sound horizon
τ	0.054 ± 0.007	Optical depth to reionization
$\ln(10^{10} A_s)$	3.044 ± 0.014	Amplitude of primordial power spectrum
n_s	0.9649 ± 0.0042	Scalar spectral index

Table 4. The six parameters of the ΛCDM model as measured by the Planck satellite (Planck Collaboration et al. 2020).

and the scalar spectral index (n_s) (Planck Collaboration et al. 2020). The final Planck analysis provides the most precise measurements of these parameters to date, with uncertainties at the sub-percent level for most parameters (Planck Collaboration et al. 2020). The values are shown in Table 4.

4.2. Challenges

Despite the remarkable success of CMB observations in constraining the six-parameter ΛCDM model, extracting cosmological parameters with greater precision from CMB data confronts several substantial challenges that complicate the analysis.

The most fundamental challenge arises from geometric and acoustic degeneracies among cosmological parameters. Since the CMB power spectrum encodes information about how sound waves propagate through the early universe, different combinations of parameters can produce nearly indistinguishable anisotropy patterns (Efstathiou & Bond 1999). The primary geometric degeneracy relates the angular size of the sound horizon at recombination r_s to the comoving anglular-diameter distance $D_A(z_*)$ through combinations of total matter density Ω_m, dark energy density Ω_Λ, and the Hubble parameter H_0 (Efstathiou & Bond 1999), where

$$\Omega_m = \frac{\Omega_b h^2 + \Omega_c h^2}{h^2} \qquad \text{and} \qquad \Omega_\Lambda = 1 - \Omega_m - \Omega_k$$

where Ω_k is the curvature density, which is consistent with zero based on CMB observations (Planck Collaboration et al. 2020). This means that increasing Ω_Λ while simultaneously reducing Ω_m in the right proportion keeps the position of acoustic peaks nearly unchanged, producing a strong correlation between these parameters.

Furthermore, extracting the six ΛCDM parameters from CMB data requires sampling from a high-dimensional posterior probability distribution in the presence of strong parameter correlations. MCMC methods are the standard approach, but they face significant computational challenges when the number of dimensions exceeds six or seven (Wandelt 2003). This is beacause the likelihood function for CMB data is extremely complex, with each likelihood evaluation requiring running a relativistic Boltzmann solver (such as CAMB or CLASS) to generate predicted CMB power spectra as a function of proposed parameters (Wandelt 2003; Arico' et al. 2022). For high-precision analyses,

millions of likelihood evaluations may be required to adequately sample the posterior, leading to total computation times of weeks or months on high-performance computing clusters.

CMB cosmology stands at the frontier of precision: the challenges of parameter degeneracies and computational complexity must be overcome to extract maximal science return from exquisite measurements. Future missions such as CMB-S4 [**ADD FOOTNOTE**], LiteBIRD[**ADD FOOT-NOTE**], and Simons Observatory[**ADD FOOTNOTE**] will push sensitivity and resolution further, demanding ever-more sophisticated data analysis strategies and highlighting the necessity of MCMC and Bayesian methods for robustly inferring fundamental cosmological parameters (Dunkley et al. 2005). In order to meet these challenges, improvements to MCMC algorithms have been proposed to reduce the computational cost of building likelihoods with Boltzmann solvers (Akeret et al. 2013). On the other hand, novel methods such as Bayesian Neural Networks (BNNs) respresent a divergence from traditional MCMC appriaches (Hortúa et al. 2020).

4.3. *Optimizations to MCMC*

Traditionally, CMB parameter estimation has relied on MCMC methods such as the Metropolis-Hastings algorithm. Lewis & Bridle (2002) developed CosmoMC, a widely used MCMC engine specifically designed for cosmological parameter estimation from CMB data, based on the Metropolis-Hastings algorithm and the Boltzmann solver CAMB (Lewis et al. 2000). While CosmoMC has been highly successful, it faces significant computational challenges due to the high dimensionality of the parameter space and the complexity of the likelihood function (Akeret et al. 2013). Metropolis-based MCMC methods reqire the evaluation of the likelihood function at each proposed step, which involves running a Boltzmann solver to generate predicted CMB power spectra. This can be computationally expensive, especially when millions of likelihood evaluations are needed to adequately sample the posterior distribution (Wandelt 2003).

To address these challenges, Akeret et al. (2013) developed CosmoHammer, an MCMC framework that leverages parallel computing to accelerate the sampling process. CosmoHammer uses the affine-invariant ensemble sampler implemented in the `emcee` package (Foreman-Mackey et al. 2013), which is well-suited for high-dimensional parameter spaces with strong correlations. The methodology of CosmoHammer is covered in this section.

Given a prior $q(\theta)$ and a likelihood function $\mathcal{L}(\theta)$, the posterior distribution is given by Bayes' Theorem:

$$P(\theta) \propto \mathcal{L}(\theta)q(\theta) \tag{4.1}$$

In this case, θ represents the set of cosmological parameters being estimated. The evidence has been omitted, as it is a normalization constant that does not affect the sampling process.

$\mathcal{L}(\theta)$ depends on the CMB power spectra, which can only be evaluated numerically via a Boltzmann solver such as CAMB, which has alread been established to be computationally expensive at large scales. To mitigate this, a parellelized approach is used based on the GW algorithm (Goodman & Weare 2010), where multiple chains (or *walkers*) are run simultaneously on different processors, and the `emcee` package (Foreman-Mackey et al. 2013), which further optimizes likelihood evaluations by dividing walkers into two groups that update their positions based on the other group's positions. This allows for efficient exploration of the parameter space while minimizing the number of Boltzmann solvers required.

CosmoHammer wraps the `emcee` sampler and CAMB Boltzmann solver into a modular framework that can be easily deployed on high-performance computing clusters. The framework allows for the parallel execution of multiple walkers, with each walker evaluating the likelihood function independently on a separate processor. This parallelization significantly reduces the overall computation time required to sample the posterior distribution. Furthermore, CosmoHammer includes features such as adaptive step sizes and convergence diagnostics to ensure efficient sampling and reliable parameter estimates.

CosmoHammer achieves comparable statistical precision to CosmoMC for while dramatically reducing wall-clock time by exploiting ensemble parallelism. With 350 walkers and 250 post-burn-in steps (87,500 samples), CosmoHammer attains mean errors around 3.4% of each parameter's marginal standard deviation. CosmoMC achieves similar precision with 35,000 post-burn-in samples, but it pays substantial tuning overhead if not pre-tuned, requiring on the order of 2000 adaptation steps before efficient sampling begins. In wall time, analysis that takes 30 hours on a dual-core laptop with a single long Metropolis-Hastings chain can be completed in 16 minutes using CosmoHammer on 2048 cores.

4.4. *Bayesian Neural Networks*

Alternatives to MCMC methods involving neural networks have also been proposed for CMB parameter estimation. A typical neural network is a graph of interconnected nodes or *neurons* organized into layers. Each neuron applies a nonlinear activation function to a weighted sum of its inputs, allowing the network to learn complex mappings from inputs to outputs (Gurney 2018). Neural networks have been used extensively in cosmology for tasks such as model discrimination (Schmelzle et al. 2017) and cosmic string detection (Ciuca et al. 2019). However, traditional neural networks provide point estimates of parameters without quantifying uncertainties, and they can be prone to overfitting, especially when trained on limited data (Hortúa et al. 2020). The most promising solution to these issues is the use of Bayesian Neural Networks (BNNs) for parameter estimation (Gal & Ghahramani 2016). In a BNN, the weights of the network are treated as random variables with prior distributions, allowing for uncertainty quantification in the predictions (Hortúa et al. 2020). The BNN approach to CMB parameter estimation developed by Hortúa et al. (2020) is covered in this section.

Given the prior distribution over weights $p(w)$ and the likelihood function $p(D|w)$, the posterior distribution over weights given the data D is given by Bayes' Theorem:

$$p(w|D) = \frac{p(D|w)p(w)}{p(D)} \tag{4.2}$$

where $P(D)$ is the evidence, which is a normalization constant. If the data consists of input-output pairs (x, y), the likelihood function can be expressed as:

$$p(D|w) = \prod_i p(y_i|x_i, w) \tag{4.3}$$

where $p(y_i|x_i, w)$ is the probability of observing output y_i given input x_i and weights w.

However, computing the posterior distribution over weights is intractable for most neural networks due to the high dimensionality of the weight space. To address this, Hortúa et al. (2020) employ variational inference, which approximates the true posterior $p(w|D)$ with a simpler distribution $q(w|\theta)$

parameterized by θ. The parameters θ are optimized by minimizing the Kullback-Leibler (KL) divergence between the true posterior and the approximate distribution:

$$\mathrm{KL}(q(w|\theta)||p(w|D)) = \int q(w|\theta) \ln \frac{q(w|\theta)}{p(w|D)} dw \tag{4.4}$$

Once the posterior distribution over weights is obtained, predictions for new inputs x^* can be estimated by marginalizing over the weights:

$$p(y^*|x^*, D) = \int p(y^*|x^*, w)p(w|D)dw \tag{4.5}$$

The posterior predictive distribution $p(y^*|x^*, D)$ can be estimated by Monte Carlo sampling. This allows the BNN to produce both a mean prediction and credible intervals that reflect epistemic (model-based) and aleatoric (data-based) uncertainties.

To train the networks, 50,000 synthetic full-sky CMB realizations were generated using the Boltzmann solver CLASS and HEALPY. Each realization was projected into 20×20 degree patches of 256×256 pixels. The model varied three ΛCDM parameters: baryon density $\Omega_b h^2$, cold dark matter density $\Omega_c h^2$, and the amplitude of the primordial power spectrum A_s. Other parameters were fixed to values from Planck Collaboration et al. (2020).

BNNs were able to recover the posterior distributions of cosmological parameters with correlations comparable to MCMC methods. Using the best-performing configuration, the network generated thousands of posterior samples in seconds, achieving speedups to the order of 10^4 over MCMC while maintaining realistic uncertainty estimates.

5. OTHER CASE STUDIES

Beyond the applications covered in Section III and Section IV, MCMC methods have been successfully applied to a variety of other astrophysical parameter estimation problems. A non-exhaustive list of such applications includes work in Gravitational Wave paremeter estimation, Dark Matter detection, and Black Hole modeling.

5.1. *Gravitational Wave Parameter Estimation*

Gravitational waves (GWs) are ripples in spacetime caused by accelerating massive objects, such as merging black holes or neutron stars, first detected by (Abbott et al. 2016). The detection and analysis of GWs provide a new window into the universe, allowing for the study of extreme astrophysical phenomena and tests of general relativity (Abbott et al. 2016). MCMC methods have been extensively used in GW data analysis to estimate the parameters of GW sources from noisy detector data. These parameters include the masses, spins, sky locations, and distances of the merging objects (Christensen & Meyer 2022). The likelihood function for GW data is typically complex, involving the convolution of theoretical waveform models with detector noise characteristics (Veitch et al. 2015). MCMC algorithms, such as the Metropolis-Hastings and Hamiltonian Monte Carlo, have been employed to sample the posterior distributions of source parameters, including masses, spins, and sky locations (Christensen & Meyer 2022; Veitch et al. 2015).

The key challenge MCMC methods face is the high dimensionality of the parameter space, often exceeding 15 dimensions (Christensen & Meyer 2022). Like CMB parameter estimation (see Section

IV), traditional MCMC methods can struggle to efficiently explore such high-dimensional spaces (Veitch et al. 2015). This complexity requires efficient sampling techniques and careful tuning of proposal distributions to ensure convergence and adequate exploration of the posterior.

These challenges have been addressed by optimizing MCMC algorithms for GW data analysis. For example, Wong et al. (2023) developed a high-performance Bayesian inference framework by integrating likelihood heterodyning and a gradient-based MCMC sampler to optimize parameter estimation. Likelihood heterodyning reduces the computational cost of evaluating the likelihood function by taking advantage of the fact that waveforms corresponding to nearby points in parameter space are very similar, meaning that their ratio varies smoothly with frequency. By decomposing the likelihood into a fixed, highly oscillatory reference term and a smooth ratio term, the likelihood can be computed accurately using far fewer frequency bins (Zackay et al. 2018). This dramatically accelerates likelihood evaluations without loss of precision. The framework also employs a gradient-based MCMC sampler: the Metropolis-adjusted Langevin algorithm (MALA) (Grenander & Miller 1994). MALA uses the gradient of the posterior probability to propose new samples more efficiently. Instead of exploring parameter space through random jumps, it moves preferentially toward regions of higher posterior probability.

In injection-recovery tests of 1,200 synthetic GW signals, the Wong et al. (2023) pipeline successfully recovered the injected parameters with statistically consistent credible intervals. When applied to real events, the method produced posterior distributions consistent with previous estimations, but with speedup to the order of 10^3.

5.2. *Dark Matter Detection*

Dark matter comprises approximately 85% of the matter in the universe, yet its nature remains one of the most fundamental unsolved problems in physics (Cushman et al. 2013). Direct detection experiments aim to observe the scattering of dark matter particles, such as weakly interacting massive particles (WIMPs), off atomic nuclei in deep underground detectors such as XENON and LUX (Arina 2014). Parameter estimation in these experiments involves constraining dark matter particle model parameters such as the WIMP-nucleon scattering cross section, dark matter mass, and interaction type using nuclear recoil energy spectra and event rates (Cushman et al. 2013).

The primary challenge in dark matter direct detection analysis, as with many other fields, is the computational expense of likelihood function evaluation combined with the high dimensionality of particle physics parameter spaces (Cerdeno et al. 2025). To address these challenges, Cerdeno et al. (2025) developed the Truncated Marginal Neural Ratio Estimation (TMNRE) approach. TMNRE avoids explicit likelihood calculations by training neural networks on simulated dark matter scattering data to estimate the likelihood ratio from the data alone, similar to the BNN approach discussed in Section IV. In validation studies comparing TMNRE with conventional MCMC for WIMP dark matter with spin-dependent and spin-independent interactions with protons and neutrons in xenon experiments, the machine learning approach recovered identical posterior distributions to traditional MCMC but achieved speedups of several orders of magnitude.

5.3. *Black Hole Modeling*

Black holes are among the most extreme objects in the universe, and measuring their fundamental properties provides crucial tests of general relativity and constraints on black hole formation and

evolution mechanisms. X-ray reflection spectroscopy has emerged as the leading method for measuring black hole spins in both stellar-mass black holes in X-ray binaries and supermassive black holes in active galactic nuclei (AGN) (Reynolds et al. 2012). This technique exploits relativistically broadened iron emission lines from black hole accretion disks, where frame-dragging effects depend sensitively on black hole spin (Reynolds et al. 2012).

Black hole spin measurement via X-ray reflection spectroscopy faces significant challenges from degeneracies between model parameters (Reynolds et al. 2012). The analysis requires fitting complex spectral models with numerous parameters including black hole spin, accretion disk inclination angle, iron abundance, reflection fraction, and coronal geometry. A critical degeneracy exists between the iron abundance of the innermost accretion disk and the black hole spin parameter: supersolar iron abundances can mimic the spectral signatures of rapid spin, or alternatively, rapid spin can appear less extreme when paired with subsolar iron abundance assumptions.

Reynolds et al. (2012) employed MCMC-based spectral analysis to address this degeneracy through analysis of high-quality X-ray spectra from the Suzaku satellite for the AGN NGC 3783. The MCMC analysis revealed a strong correlation between iron abundance and black hole spin, demonstrating that assuming solar iron abundance led to biased spin estimates. By allowing iron abundance to vary freely in the MCMC fitting, they obtained a more accurate and reliable measurement of the black hole spin parameter. The analysis concluded that NGC 3783 hosts a rapidly spinning black hole with spin parameter $a > 0.88$, while also constraining the iron abundance to be supersolar at $Z_{\mathrm{Fe}} = 2 - 4 Z_\odot$.

Sisk-Reynés et al. (2022) further applied MCMC-based X-ray reflection spectroscopy to measure the spin of the supermassive black hole in the quasar H1821+643 using data from the Chandra X-ray Observatory. Their MCMC analysis yielded a the first well-defined spin measurement of $a = 0.62^{+0.22}_{-0.37}$, an inclination angle of $i = 44.6^{+3.3}_{-3.3}$, and an iron abundance of $Z_{\mathrm{Fe}} = 1.02^{+1.33}_{-0.38} Z_\odot$.

5.4. *Further Case Studies*

Many more applications of Bayesian inference and MCMC methods exist in astrophysics beyond those covered in this review. A few notable examples are listed, along with their references for further reading:

1. Creation of pulsar timing arrays (Lai & Li 2025)

2. Quasar light curve analysis (Donnan et al. 2021)

3. Asteroid mass estimation (Siltala & Granvik 2020)

Overall, MCMC methods have become indispensable tools across a wide range of astrophysical problems, enabling robust inference in the presence of complex models, high-dimensional parameter spaces, and challenging data characteristics. Despite the computational challenges involved, ongoing methodological advancements continue to expand the applicability and efficiency of MCMC techniques in astrophysics for application to further research.

6. DISCUSSION

6.1. *Optimization versus Transformation*

The case studies reveal a fundamental pattern: Bayesian-MCMC frameworks are primarily being leveraged to optimize and refine existing research methodologies rather than to fundamentally transform the underlying scientific questions being asked.

In the direct imaging of exoplanets (Section III), Bayesian inference is applied to enhance the rigor of detection thresholds and to prioritize targets for follow-up observations. Rather than replacing traditional PSF subtraction techniques, Bayesian methods augment these approaches by providing uncertainty quantification and hypothesis testing frameworks. Similarly, in CMB parameter analysis (Section IV), the introduction of CosmoHammer and ensemble MCMC methods does not change the core physics of parameter estimation. The approaches serve to accelerate computation while remaining faithful to the underlying Boltzmann solvers. Section V further illustrates this pattern: sophisticated MCMC implementations and advanced likelihood engineering reduce computational barriers but preserve the essential physical models and methodologies.

This observation reflects the maturity of the field. In domains where physical theories are well-developed and observational techniques are proven, the role of statistical methods is to extract maximum precision from existing frameworks. The case studies demonstrate that modern astrophysics has largely solved the methodological question of how to do Bayesian inference at scale; the remaining focus is on doing it faster and more efficiently.

As of the time of writing, MCMC methods such as Metropolis-Hastings variants and ensemble samplers remain the standard for Bayesian parameter estimation across astrophysics. This dominance is well-earned: MCMC methods are widely understood and validated against decades of astronomical data. Their robustness in handling complex likelihood functions, multi-modal posteriors, and high-dimensional parameter spaces makes them a reliable tool. Furthermre, tuning MCMC for a well-understood problem is a lower-risk path than adopting untested methods. Researchers can leverage existing codebases, validation protocols, and community expertise. The payoff is immediate and quantifiable—wall-clock time reduced by orders of magnitude translates directly to increased scientific productivity.

6.2. *Competing Paradigms*

Beneath the optimization narrative, a competing methodological paradigm is emerging. The case studies containing Bayesian Neural Networks (Section 4.4) and neural ratio estimation (Section 5.2) represent a departure from traditional MCMC. These approaches do not merely accelerate the existing framework, but replace it with a learned emulator or surrogate model, fundamentally altering how statistical inference is performed.

The results are striking. Hortúa et al. (2020) achieved 10,000-fold speedups for CMB parameter estimation while maintaining realistic uncertainty quantification. Cerdeno et al. (2025) demonstrated that neural ratio estimation recovers identical posterior distributions to MCMC but at speedups of several orders of magnitude. These are far from incremental gains. Instead, Bayesian Neural Networks represent a different regime of computational feasibility.

Yet, these methods have yet to reach a point of general use in astrophysics. The majority of active research continues to refine MCMC pipelines rather than exploring neural network alternatives. This lag may be a result of both institutional inertia and technical questions about the reliability of machine learning methods for novel problems. Hortúa et al. (2020) outlines a number of concerns about the application of deep learning models in physics, including the models being prone to overfitting and

generating unreliable data. Though Hortúa et al. (2020) studied a Bayesian treatment of these models as a possible solution to these concerns, the BNNs still required large training datasets of simulated data. When confronted with observational regimes that differ substantially from the training domain, the question of these method's applicability remains.

The existence of these competing methodologies, however, creates a decision point for practitioners: when is it scientifically and computationally justified to adopt advanced machine learning approaches over optimized MCMC?

Based on the case studies, MCMC tended to be the optimal choice for when:

1. The posterior was well-understood:
   When prior physics strongly constrained the parameter space (as in CMB cosmology with the six ΛCDM parameters), MCMC could be tuned to exploit this structure. The dimensionality remained manageable, and validation was straightforward.

2. Likelihood evaluations were not expensive:
   If the bottleneck was not likelihood computation but rather posterior exploration, MCMC techniques provided substantial gains without the overhead of training machine learning models.

3. Interpretability and explainability were critical:
   When a clear understanding was required of why particular parameters were inferred, MCMC was the better chouce due to its transparent sampling history. Neural network predictions, on the other hand, acted as a black box.

On the contrary, machine learning approached were used when likelihood evaluation was the computational bottleneck. When thousands or millions of likelihood evaluations were required, amortizing the cost through pre-trained emulators became optimal. CMB parameter estimation is an example of this: thousands of Boltzmann solver calls justify the upfront investment in training a neural network.

Looking forward, the most likely scenario is not the wholesale replacement of MCMC but rather methodological differentiation based on problem characteristics. Established, well-understood research programs with stable parameter spaces will increasingly adopt neural network-based methods, trading some interpretability for computational efficiency. Frontier research, exploratory studies, and problem domains with evolving theoretical frameworks will retain MCMC as the preferred tool for its transparency and generality.

The field will ultimately benefit from both traditions existing in productive tension. MCMC methods will continue to improve through algorithmic refinement and better proposal engineering. Machine learning approaches will mature through larger training datasets and improved uncertainty quantification. The division of labor will reflect the nature of the science: MCMC for precision measurement and validation; neural networks for throughput and scale.

7. CONCLUSION

Bayesian inference and MCMC methods have become essential infrastructure for contemporary astrophysics. However, these methods are optimizing the existing scientific basis rather than fundamentally transforming it. Whether applied to refining PSF subtraction in direct imaging or accelerating Boltzmann solver calls in cosmology, the statistical machinery operates within well-established

theoretical frameworks. The advancement manifests itself by extracting more precision and speed from known physics.

This maturation reflects the field's sophistication. Modern astrophysics has largely resolved the technical question of how to perform Bayesian inference at the scales demanded by contemporary observations. The remaining challenge is methodological: choosing among competing approaches that increasingly offer comparable accuracy with vastly different computational profiles.

The case studies reveal clear patterns in this choice. MCMC retains dominance in established, well-understood research programs where interpretability and generality are valued, and where likelihood evaluations, while expensive, are not the primary bottleneck. Machine learning alternatives emerge as compelling when amortizing computational cost becomes advantageous—in high-dimensional problems. The question is not which method is universally superior, but rather which method best matches the scientific problem at hand.

7.1. *Future Work*

Despite the maturity of MCMC and the promise of machine learning alternatives, the case studies reveal several gaps that neither paradigm has fully addressed.

Firstly, Neither MCMC nor BNNs naturally address scenarios where the physical model itself may be incomplete or incorrect. Both methods assume the model is true and focus on inferring parameters. Developing Bayesian frameworks that simultaneously infer parameters and model uncertainty remains an open frontier.

Furthermore, as surveys like LSST and Rubin Observatory generate unprecedented data volumes, the question becomes not just how to estimate parameters for individual sources but how to allocate computational resources across millions of sources, some of which may be faint or unusual. This problem of combining statistical efficiency with practical resource management has not been comprehensively addressed by either MCMC or neural network approaches.

Bayesian inference through MCMC has established itself as a cornerstone methodology in astrophysics: reliable across diverse applications, adaptable to novel problems, and increasingly augmented by machine learning alternatives. While these methods will not fundamentally alter the underlying physics being studied, they have transformed how that physics is evaluated. MCMC provides a principled computational framework that converts complex observational data into quantitative constraints on physical models, enabling discoveries that observations alone cannot reveal.

Future astrophysics will demand both the interpretability and robustness of MCMC and the scalability and efficiency of machine learning approaches. The most productive path forward lies in recognizing these as complementary rather than competing paradigm. As astronomical surveys and physical simulations grow increasingly sophisticated, the ability to fluently deploy multiple inference strategies will distinguish the most effective research programs.

APPENDIX

A. RADIAL VELOCITY BRUTE-FORCE APPROACH

```
1  #%%
2  # --- Radial Velocity Example ---
```

```python
# This script demonstrates simple Bayesian parameter estimation for a
# sinusoidal radial velocity model of the form:
# v(t) = K * sin(2 * pi * t / P + phi)

import numpy as np
import matplotlib.pyplot as plt

np.random.seed(42)  # The meaning of life for reproducibility

#%%
# --- True Parameter Values ---
K = 10.0                 # Velocity semi-amplitude [m/s]
P = 21.4                 # Period of the signal [days]
phi = 0.25 * np.pi  # Phase offset [radians]
sig_ob = 3.0         # Observational uncertainty [m/s]

#%%
# --- Generate Synthetic Data ---
# Simulate 34 observations across 42 days
t = np.sort(42 * np.random.rand(34))

# Observed velocities = true model + Gaussian noise
v_ob = K * np.sin((2 * np.pi * t / P) + phi) + np.random.normal(0, sig_ob,
    size=len(t))

#%%
# --- Plot Observations Only ---
plt.figure(figsize=(6, 4))
plt.errorbar(
    t, v_ob, yerr=sig_ob, fmt='o', color='black',
    ecolor='gray', elinewidth=1, capsize=2,
    label='Observations', markersize=4
)
plt.xlabel("Time [days]", fontsize=12)
plt.ylabel("Radial velocity [m/s]", fontsize=12)
plt.grid(True, linestyle='--', alpha=0.5)
plt.tight_layout()
plt.show()

#%%
# --- Plot Observations and True Model ---
t_dense = np.linspace(0, 42, 1000)
plt.figure(figsize=(6, 4))
plt.errorbar(
    t, v_ob, yerr=sig_ob, fmt='o', color='black',
    ecolor='gray', elinewidth=1, capsize=2,
    label='Observations', markersize=4
)
```

```python
50  plt.plot(t_dense, K * np.sin(2 * np.pi * (t_dense / P) + phi),
51          'r--', linewidth=2, label='True model')
52  plt.xlabel("Time [days]", fontsize=12)
53  plt.ylabel("Radial velocity [m/s]", fontsize=12)
54  plt.grid(True, linestyle='--', alpha=0.5)
55  plt.tight_layout()
56  plt.show()
57
58  #%%
59  # --- Brute-Force Bayesian Parameter Estimation ---
60  # Define prior, likelihood, and posterior functions
61
62  # Uniform prior over plausible parameter ranges
63  def log_prior(K, P, phi):
64      if 5 < K < 15 and 20 < P < 30 and 0 < phi < 2 * np.pi:
65          return -np.log(10 * 10 * 2 * np.pi)
66      else:
67          return -np.inf
68
69  # Gaussian likelihood assuming constant sigma
70  def log_likelihood(K, P, phi):
71      v_model = K * np.sin((2 * np.pi * t / P) + phi)
72      return -0.5 * np.sum(
73          ((v_ob - v_model)/sig_ob)**2 + np.log(2*np.pi*sig_ob**2)
74      )
75
76  #%%
77  # --- Evaluate Posterior on a Grid ---
78  K_vals = np.linspace(5, 15, 100)
79  P_vals = np.linspace(20, 30, 100)
80  phi_vals = np.linspace(0, 2 * np.pi, 100)
81  posterior_vals = np.zeros((len(K_vals), len(P_vals), len(phi_vals)))
82
83  norm = 0
84  for i in range(len(K_vals)):
85      for j in range(len(P_vals)):
86          for k in range(len(phi_vals)):
87              posterior_vals[i, j, k] = np.exp(log_prior(K_vals[i], P_vals[j
                  ], phi_vals[k]) + log_likelihood(K_vals[i], P_vals[j],
                  phi_vals[k]))
88              norm += posterior_vals[i, j, k]
89  posterior_vals /= norm
90
91  #%%
92  # --- Compute Marginal Posteriors and Statistics ---
93  posterior_K = np.sum(np.sum(posterior_vals, axis=2), axis=1)
94  posterior_P = np.sum(np.sum(posterior_vals, axis=0), axis=1)
95  posterior_phi = np.sum(np.sum(posterior_vals, axis=0), axis=0)
```

```python
96
97  mean_K = np.sum(K_vals * posterior_K) / np.sum(posterior_K)
98  std_K  = np.sqrt(np.sum((K_vals - mean_K) ** 2 * posterior_K) / np.sum(
       posterior_K))
99  mean_P = np.sum(P_vals * posterior_P) / np.sum(posterior_P)
100 std_P  = np.sqrt(np.sum((P_vals - mean_P) ** 2 * posterior_P) / np.sum(
       posterior_P))
101 mean_phi = np.sum(phi_vals * posterior_phi) / np.sum(posterior_phi)
102 std_phi  = np.sqrt(np.sum((phi_vals - mean_phi) ** 2 * posterior_phi) / np
       .sum(posterior_phi))
103
104 #%%
105 # --- Plot Marginal Posterior Distributions ---
106 plt.figure(figsize=(4, 4))
107 plt.plot(K_vals, posterior_K, color='blue')
108 plt.axvline(K, color='r', ls='--', label='True K')
109 plt.xlabel("K [m/s]")
110 plt.title("Posterior of K")
111 plt.legend()
112 plt.grid(True, linestyle='--', alpha=0.5)
113 plt.show()
114 print(f"mu={mean_K:.2f}, sigma={std_K:.2f}")
115
116 plt.figure(figsize=(4, 4))
117 plt.plot(P_vals, posterior_P, color='green')
118 plt.axvline(P, color='r', ls='--', label='True T')
119 plt.xlabel("P [days]")
120 plt.title("Posterior of T")
121 plt.legend()
122 plt.grid(True, linestyle='--', alpha=0.5)
123 plt.show()
124 print(f"mu={mean_P:.2f}, sigma={std_P:.2f}")
125
126 plt.figure(figsize=(4, 4))
127 plt.plot(phi_vals, posterior_phi, color='orange')
128 plt.axvline(phi, color='r', ls='--', label='True phi')
129 plt.xlabel("phi [radians]")
130 plt.title("Posterior of phi")
131 plt.legend()
132 plt.grid(True, linestyle='--', alpha=0.5)
133 plt.show()
134 print(f"mu={mean_phi:.2f}, sigma={std_phi:.2f}")
135
136 #%%
137 # --- Compare True Model vs Posterior Predictions ---
138 plt.figure(figsize=(8, 5))
139 N_samples = 500
140
```

```
141  for i in range(N_samples):
142      K_draw = np.random.choice(K_vals, p=posterior_K)
143      P_draw = np.random.choice(P_vals, p=posterior_P)
144      phi_draw = np.random.choice(phi_vals, p=posterior_phi)
145      plt.plot(t_dense, K_draw * np.sin(2 * np.pi * (t_dense / P_draw) +
             phi_draw),
146              color='lightblue', alpha=0.2)
147
148  plt.plot(t_dense, K * np.sin(2 * np.pi * (t_dense / P) + phi),
149          'r--', lw=2, label='True model')
150  plt.plot(t_dense, mean_K * np.sin(2 * np.pi * (t_dense / mean_P) +
         mean_phi),
151          'b-', lw=2, label='Posterior mean prediction')
152
153  plt.errorbar(t, v_ob, yerr=sig_ob, fmt='o', color='black',
154              ecolor='gray', elinewidth=1, capsize=2,
155              label='Observations', markersize=4)
156  plt.xlabel("Time [days]")
157  plt.ylabel("Radial velocity [m/s]")
158  plt.title("Posterior Mean and Uncertainty vs True Model")
159  plt.legend()
160  plt.grid(True, linestyle='--', alpha=0.4)
161  plt.tight_layout()
162  plt.show()
163  # %%
```

B. RADIAL VELOCITY MCMC APPROACH

```
1   #%%
2   # --- Radial Velocity Example ---
3   # This script demonstrates simple MCMC parameter estimation for a
4   # sinusoidal radial velocity model of the form:
5   # v(t) = K * sin(2 * pi * t / P + phi)
6
7   import numpy as np
8   import matplotlib.pyplot as plt
9
10  np.random.seed(42)  # The meaning of life for reproducibility
11  #%%
12  # --- True Parameter Values ---
13  K = 10.0            # Velocity semi-amplitude [m/s]
14  P = 21.4            # Period of the signal [days]
15  phi = 0.25 * np.pi  # Phase offset [radians]
16  sig_ob = 3.0        # Observational uncertainty (standard deviations)
17
18  #%%
19  # --- Generate Synthetic Data ---
20  # Simulate 34 observations across 42 days
```

```python
21  t = np.sort(42 * np.random.rand(34))
22
23  # Observed velocities = true model + Gaussian noise
24  v_ob = K * np.sin((2 * np.pi * t / P) + phi) + np.random.normal(0, sig_ob,
        size=len(t))
25
26  #%%
27  # --- MCMC Parameter Estimation ---
28  # Define prior, likelihood, and posterior functions
29
30  # Uniform prior over plausible parameter ranges
31  def log_prior(K, P, phi):
32      if 5 < K < 15 and 20 < P < 30 and 0 < phi < 2 * np.pi:
33          return -np.log(10 * 10 * 2 * np.pi)
34      else:
35          return -np.inf
36
37  # Gaussian likelihood assuming constant sigma
38  def log_likelihood(K, P, phi):
39      v_model = K * np.sin((2 * np.pi * t / P) + phi)
40      return -0.5 * np.sum(
41        ((v_ob - v_model)/sig_ob)**2 + np.log(2*np.pi*sig_ob**2)
42      )
43
44  #%%
45  # Define proposal distribution
46  def newProposal(current):
47    step_size = [0.25, 0.25, 0.1] # Standard deviation for new proposals
48    return [
49      np.random.normal(current[0], step_size[0]),
50      np.random.normal(current[1], step_size[1]),
51      np.random.normal(current[2], step_size[2])
52    ]
53
54  #%%
55  # Metropolis Algorithm
56  walks = 10
57  burn_steps = 1000
58  steps = 12000
59  chain = np.zeros((walks, burn_steps + steps + 1,3))
60
61  for w in range(walks):
62    curr_state = [14,25,np.pi] # Initial guess, should be a good guess
        within the parameter space
63
64    curr_prior = log_prior(*curr_state)
65    curr_likelihood = log_likelihood(*curr_state)
66    chain[w,0] = curr_state
```

```python
67
68    accept_count = 0
69
70    for i in range(burn_steps + steps):
71      new_proposal = newProposal(curr_state)
72
73      new_prior = log_prior(*new_proposal)
74      new_likelihood = -np.inf
75      # Only calculate likelihood if the new proposal is in the parameter
          space
76      if (new_prior != -np.inf):
77        new_likelihood = log_likelihood(*new_proposal)
78
79      # Accept or reject the proposal?
80      new_log_post = new_prior + new_likelihood
81      curr_log_post = curr_prior + curr_likelihood
82      alpha = np.exp(new_log_post - curr_log_post)
83      if alpha > 1:
84        alpha = 1
85
86      if np.random.rand() < alpha:
87        curr_state = new_proposal
88        curr_prior = new_prior
89        curr_likelihood = new_likelihood
90        accept_count += 1
91
92      chain[w,i+1] = curr_state
93    print(accept_count/(burn_steps + steps))
94
95  # Remove burn steps
96  chain = chain[:,burn_steps+1:]
97
98  #%%
99  # Plot the results of the algorithm
100 K_chain = chain[:,:,0]
101 P_chain = chain[:,:,1]
102 phi_chain = chain[:,:,2]
103 chain_range = np.arange(burn_steps,burn_steps+steps)
104
105 fig, axes = plt.subplots(3, 1, figsize=(10, 8), sharex=True)
106
107 # K trace
108 for p in range(len(chain)):
109   axes[0].plot(chain_range, K_chain[p])
110 axes[0].axhline(y=K, color='r', ls='--', label='True K')
111 axes[0].set_ylabel('K [m/s]')
112 axes[0].set_title('Trace of K')
113 axes[0].legend()
```

```python
114
115  # P trace
116  for p in range(len(chain)):
117    axes[1].plot(chain_range, P_chain[p])
118  axes[1].axhline(y=P, color='r', ls='--', label='True T')
119  axes[1].set_ylabel('Period [days]')
120  axes[1].set_title('Trace of T')
121  axes[1].legend()
122
123  # phi trace
124  for p in range(len(chain)):
125    axes[2].plot(chain_range, phi_chain[p])
126  axes[2].axhline(y=phi, color='r', ls='--', label='True phi')
127  axes[2].set_ylabel('Phase [rad]')
128  axes[2].set_xlabel('Step')
129  axes[2].set_title('Trace of phi')
130  axes[2].legend()
131
132  plt.tight_layout()
133  plt.show()
134  # %%
135  # Plot histograms
136  plt.figure(figsize=(4, 4))
137  plt.hist(K_chain.flatten(), bins=50, color='tab:blue', alpha=0.7)
138  plt.axvline(K, color='r', ls='--', label='True K')
139  plt.ylabel('Count')
140  plt.xlabel('K [m/s]')
141  plt.title('Posterior of K')
142  plt.legend()
143  plt.show()
144
145  plt.figure(figsize=(4, 4))
146  plt.hist(P_chain.flatten(), bins=50, color='tab:blue', alpha=0.7)
147  plt.axvline(P, color='r', ls='--', label='True T')
148  plt.ylabel('Count')
149  plt.xlabel('Period [days]')
150  plt.title('Posterior of T')
151  plt.legend()
152  plt.show()
153
154  plt.figure(figsize=(4, 4))
155  plt.hist(phi_chain.flatten(), bins=50, color='tab:blue', alpha=0.7)
156  plt.axvline(phi, color='r', ls='--', label='True phi')
157  plt.ylabel('Count')
158  plt.xlabel('Phase [rad]')
159  plt.title('Posterior of phi')
160  plt.legend()
161  plt.show()
```

```python
162  # %%
163  # Calculate expectation values and standard deviations
164  K_mean = np.mean(K_chain)
165  K_std = np.std(K_chain)
166
167  P_mean = np.mean(P_chain)
168  P_std = np.std(P_chain)
169
170  phi_mean = np.mean(phi_chain)
171  phi_std = np.std(phi_chain)
172
173  print(f"K = {K_mean:.2f} +/- {K_std:.2f}")
174  print(f"P = {P_mean:.2f} +/- {P_std:.2f}")
175  print(f"phi = {phi_mean:.2f} +/- {phi_std:.2f}")
176  # %%
```

REFERENCES

Abbasi, R., Ackermann, M., Adams, J., et al. 2023, The Astrophysical Journal, 959, 96, doi: 10.3847/1538-4357/aceefc

Abbott, B. P., Abbott, R., Abbott, T. D., et al. 2016, Physical Review Letters, 116, 061102, doi: 10.1103/PhysRevLett.116.061102

Akeret, J., Seehars, S., Amara, A., Refregier, A., & Csillaghy, A. 2013, Astronomy and Computing, 2, 27, doi: 10.1016/j.ascom.2013.06.003

Arico', G., Angulo, R., & Zennaro, M. 2022, Open Research Europe, 1, 152, doi: 10.12688/openreseurope.14310.2

Arina, C. 2014, Physics of the Dark Universe, 5–6, 1, doi: 10.1016/j.dark.2014.03.003

Bixel, A., & Apai, D. 2019, The Astronomical Journal, 159, 3, doi: 10.3847/1538-3881/ab5222

Borne, K. D. 2009, Astroinformatics: A 21st Century Approach to Astronomy, arXiv, doi: 10.48550/arXiv.0909.3892

Brewer, B. J. 2018, in Bayesian Astrophysics, ed. A. A. Ramos & I. Arregui (Cambridge: Cambridge university press)

Cannon, A. J., & Pickering, E. C. 1918, The Henry Draper Catalogue (The Observatory)

Carter, A. L., Hinkley, S., Kammerer, J., et al. 2023, The Astrophysical Journal Letters, 951, L20, doi: 10.3847/2041-8213/acd93e

Cerdeno, D., de los Rios, M., & Perez, A. D. 2025, Journal of Cosmology and Astroparticle Physics, 2025, 038, doi: 10.1088/1475-7516/2025/01/038

Chauvin, G. 2023, Comptes Rendus. Physique, 24, 129, doi: 10.5802/crphys.139

Christensen, N., & Meyer, R. 2022, Reviews of Modern Physics, 94, 025001, doi: 10.1103/RevModPhys.94.025001

Ciuca, R., Hernández, O. F., & Wolman, M. 2019, Monthly Notices of the Royal Astronomical Society, 485, 1377, doi: 10.1093/mnras/stz491

Collaboration, S., Pallathadka, G. A., Aghakhanloo, M., et al. 2025, The Nineteenth Data Release of the Sloan Digital Sky Survey, arXiv, doi: 10.48550/arXiv.2507.07093

Comrie, L. J. 1932, Monthly Notices of the Royal Astronomical Society, 92, 694, doi: 10.1093/mnras/92.7.694

Cox, R. T. 1946, American Journal of Physics, 14, 1, doi: 10.1119/1.1990764

Currie, T., Biller, B., Lagrange, A.-M., et al. 2023, Direct Imaging and Spectroscopy of Extrasolar Planets, arXiv, doi: 10.48550/arXiv.2205.05696

Cushman, P., Galbiati, C., McKinsey, D. N., Robertson, H., & Tait, T. M. P. 2013

Donnan, F. R., Horne, K., & Santisteban, J. V. H. 2021, Monthly Notices of the Royal Astronomical Society, 508, 5449, doi: 10.1093/mnras/stab2832

Duncombe, R. L. 1988, Celestial Mechanics, 45, 1, doi: 10.1007/BF01228969

Dunkley, J., Bucher, M., Ferreira, P. G., Moodley, K., & Skordis, C. 2005, Monthly Notices of the Royal Astronomical Society, 356, 925, doi: 10.1111/j.1365-2966.2004.08464.x

Efstathiou, G., & Bond, J. R. 1999, Monthly Notices of the Royal Astronomical Society, 304, 75, doi: 10.1046/j.1365-8711.1999.02274.x

Feigelson, E. D., & Babu, G. J. 2004, Statistical Challenges in Modern Astronomy, arXiv, doi: 10.48550/arXiv.astro-ph/0401404

Fischer, D. A., Howard, A. W., Laughlin, G. P., et al. 2014, in Protostars and Planets VI (University of Arizona Press)

Follette, K. B. 2023, Publications of the Astronomical Society of the Pacific, 135, 093001, doi: 10.1088/1538-3873/aceb31

Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. 2013, Publications of the Astronomical Society of the Pacific, 125, 306, doi: 10.1086/670067

Gaia Collaboration, Vallenari, A., Brown, A. G. A., et al. 2023, Astronomy & Astrophysics, 674, A1, doi: 10.1051/0004-6361/202243940

Gal, Y., & Ghahramani, Z. 2016, Dropout as a Bayesian Approximation: Appendix, arXiv, doi: 10.48550/arXiv.1506.02157

Gelman, A., Gilks, W. R., & Roberts, G. O. 1997, The Annals of Applied Probability, 7, 110, doi: 10.1214/aoap/1034625254

Goodman, J., & Weare, J. 2010, Communications in Applied Mathematics and Computational Science, 5, 65, doi: 10.2140/camcos.2010.5.65

Gregory, P. C. 2005, The Astrophysical Journal, 631, 1198, doi: 10.1086/432594

Grenander, U., & Miller, M. I. 1994, Journal of the Royal Statistical Society. Series B (Methodological), 56, 549

Guimond, C. M., & Cowan, N. B. 2018, The Astronomical Journal, 155, 230, doi: 10.3847/1538-3881/aabb02

Gurney, K. 2018, An Introduction to Neural Networks (London: CRC Press), doi: 10.1201/9781315273570

Guth, A. H. 1981, Physical Review D, 23, 347, doi: 10.1103/PhysRevD.23.347

Guyon, O., Pluzhnik, E. A., Galicher, R., et al. 2005, The Astrophysical Journal, 622, 744, doi: 10.1086/427771

Hastings, W. K. 1970, Biometrika, 57, 97, doi: 10.1093/biomet/57.1.97

Holmberg, E. 1940, The Astrophysical Journal, 92, 200, doi: 10.1086/144212

Hortúa, H. J., Volpi, R., Marinelli, D., & Malagò, L. 2020, Physical Review D, 102, 103509, doi: 10.1103/PhysRevD.102.103509

Hubble, E. P. 1930, The Astrophysical Journal, 71, 231, doi: 10.1086/143250

Huijse, P., Estevez, P. A., Protopapas, P., Principe, J. C., & Zegers, P. 2014, IEEE Computational Intelligence Magazine, 9, 27, doi: 10.1109/MCI.2014.2326100

Kaushik, M., Mattoo, A., & Rastogi, R. 2024, Exoplanet Detection : A Detailed Analysis, arXiv, doi: 10.48550/arXiv.2404.09143

Kaushik, M., Mattoo, A., Rastogi, R., & Khare, M. D. 2025, in Big Data Analytics in Astronomy, Science, and Engineering, ed. S. Sachdeva, Y. Watanobe, & S. Bhalla (Cham: Springer Nature Switzerland), 138–153, doi: 10.1007/978-3-031-86193-2_9

Lafreniere, D., Marois, C., Doyon, R., Nadeau, D., & Artigau, E. 2007, The Astrophysical Journal, 660, 770, doi: 10.1086/513180

Lagrange, A.-M., Bonnefoy, M., Chauvin, G., et al. 2010, Science, 329, 57, doi: 10.1126/science.1187187

Lai, J., & Li, C. 2025, Accelerated Bayesian Inference for Pulsar Timing Arrays: Normalizing Flows for Rapid Model Comparison Across Stochastic Gravitational-Wave Background Sources, arXiv, doi: 10.48550/arXiv.2504.04211

Leavesley, S., & Tárnok, A. 2018, Cytometry Part A, 93, 977, doi: 10.1002/cyto.a.23637

Lewis, A., & Bridle, S. 2002, Physical Review D, 66, 103511, doi: 10.1103/PhysRevD.66.103511

Lewis, A., Challinor, A., & Lasenby, A. 2000, The Astrophysical Journal, 538, 473, doi: 10.1086/309179

Lovis, C., & Fischer, D. 2010, in Exoplanets, 2–4

Lynden-Bell, D. 1967, Monthly Notices of the Royal Astronomical Society, 136, 101, doi: 10.1093/mnras/136.1.101

Lyot, B. 1939, Monthly Notices of the Royal Astronomical Society, 99, 580, doi: 10.1093/mnras/99.8.580

Macintosh, B., Graham, J. R., Barman, T., et al. 2015, Science, 350, 64, doi: 10.1126/science.aac5891

Madhusudhan, N. 2019, Annual Review of
    Astronomy and Astrophysics, 57, 617,
    doi: 10.1146/annurev-astro-081817-051846

Marois, C., Lafreniere, D., Doyon, R., Macintosh,
    B., & Nadeau, D. 2006, The Astrophysical
    Journal, 641, 556, doi: 10.1086/500401

Marois, C., Macintosh, B., Barman, T., et al.
    2008, Science, 322, 1348,
    doi: 10.1126/science.1166585

Merriman, M. 1877, The Analyst, 4, 33,
    doi: 10.2307/2635472

Meshkat, T., Bailey, V., Rameau, J., et al. 2013,
    The Astrophysical Journal Letters, 775, L40,
    doi: 10.1088/2041-8205/775/2/L40

Metropolis, N., Rosenbluth, A. W., Rosenbluth,
    M. N., Teller, A. H., & Teller, E. 1953, The
    Journal of Chemical Physics, 21, 1087,
    doi: 10.1063/1.1699114

Neal, R. 1993, Probabilistic Inference Using
    Markov Chain Monte Carlo Methods
    (Department of Computer Science, University
    of Toronto)

Nisenson, P., & Papaliolios, C. 2001, The
    Astrophysical Journal, 548, L201,
    doi: 10.1086/319110

Olley, A. 2018, Revue de Synthèse, 139, 267,
    doi: 10.1163/19552343-13900014

Peebles, P. J. E. 1973, The Astrophysical Journal,
    185, 413, doi: 10.1086/152431

Planck Collaboration, Ade, P. A. R., Aghanim,
    N., et al. 2016, Astronomy & Astrophysics, 594,
    A24, doi: 10.1051/0004-6361/201525833

Planck Collaboration, Aghanim, N., Akrami, Y.,
    et al. 2020, Astronomy & Astrophysics, 641, A6,
    doi: 10.1051/0004-6361/201833910

Reddy, A. N. K., & Hashemi, M. 2018, Journal of
    Physics: Conference Series, 1096, 012140,
    doi: 10.1088/1742-6596/1096/1/012140

Reynolds, C. S., Brenneman, L. W., Lohfink,
    A. M., et al. 2012, The Astrophysical Journal,
    755, 88, doi: 10.1088/0004-637X/755/2/88

Ruane, G., Mawet, D., Kastner, J., et al. 2017,
    The Astronomical Journal, 154, 73,
    doi: 10.3847/1538-3881/aa7b81

Ruffio, J.-B., Mawet, D., Czekala, I., et al. 2018,
    The Astronomical Journal, 156, 196,
    doi: 10.3847/1538-3881/aade95

Schafer, C. M. 2015, Annual Review of Statistics
    and Its Application, 2, 141,
    doi: 10.1146/annurev-statistics-022513-115538

Schmelzle, J., Lucchi, A., Kacprzak, T., et al.
    2017, Cosmological Model Discrimination with
    Deep Learning, arXiv,
    doi: 10.48550/arXiv.1707.05167

Seager, S., Bains, W., & Petkowski, J. 2016,
    Astrobiology, 16, 465,
    doi: 10.1089/ast.2015.1404

Siltala, L., & Granvik, M. 2020, Astronomy &
    Astrophysics, 633, A46,
    doi: 10.1051/0004-6361/201935608

Sisk-Reynés, J., Reynolds, C. S., Matthews, J. H.,
    & Smith, R. N. 2022, Monthly Notices of the
    Royal Astronomical Society, 514, 2568,
    doi: 10.1093/mnras/stac1389

Smoot, G. F., Bennett, C. L., Kogut, A., et al.
    1992, The Astrophysical Journal, 396, L1,
    doi: 10.1086/186504

Staggs, S., Dunkley, J., & Page, L. 2018, Reports
    on Progress in Physics, 81, 044901,
    doi: 10.1088/1361-6633/aa94d5

Stigler, S. M. 1975, Biometrika, 62, 503,
    doi: 10.1093/biomet/62.2.503

—. 1981, The Annals of Statistics, 9, 465

Tegmark, M., Taylor, A., & Heavens, A. 1997,
    The Astrophysical Journal, 480, 22,
    doi: 10.1086/303939

Toomre, A., & Toomre, J. 1972, The Astrophysical
    Journal, 178, 623, doi: 10.1086/151823

Trotta, R. 2008, Contemporary Physics, 49, 71,
    doi: 10.1080/00107510802066753

van Ravenzwaaij, D., Cassey, P., & Brown, S. D.
    2018, Psychonomic Bulletin & Review, 25, 143,
    doi: 10.3758/s13423-016-1015-8

Veitch, J., Raymond, V., Farr, B., et al. 2015,
    Physical Review D, 91, 042003,
    doi: 10.1103/PhysRevD.91.042003

Von Toussaint, U. 2011, Reviews of Modern
    Physics, 83, 943,
    doi: 10.1103/RevModPhys.83.943

Wandelt, B. D. 2003

Wei, J. 2018, A Survey of Exoplanetary Detection
    Techniques, arXiv,
    doi: 10.48550/arXiv.1805.02771

Wong, K. W. K., Isi, M., & Edwards, T. D. P.
    2023, Fast Gravitational Wave Parameter
    Estimation without Compromises, arXiv,
    doi: 10.48550/arXiv.2302.05333

Zackay, B., Dai, L., & Venumadhav, T. 2018,
    Relative Binning and Fast Likelihood
    Evaluation for Gravitational Wave Parameter
    Estimation, arXiv,
    doi: 10.48550/arXiv.1806.08792

Zanoni, C. A., & Hill, H. A. 1965, JOSA, 55,
    1608, doi: 10.1364/JOSA.55.001608

Zwicky, F. 1937, The Astrophysical Journal, 86,
    217, doi: 10.1086/143864