Hamilton Trust Summer Internship Project Report

An Investigation of Markov Chain Monte-Carlo Methods

Ruaidhrí Campion
Supervised by Prof. Stefan Sint

7th July 2022

Contents

1 Abstract

In this project, the application of Markov Chain Monte Carlo (MCMC) methods is studied. Random variables are generated using an MCMC algorithm, and a relationship between the algorithm's internal parameter and the random variable's parameter is established by investigating visual distributions, thermalisation, and integrated autocorrelation times. A similar approach is used for observables of a physical model, where two algorithms are directly compared. Finally, methods on how to obtain more accurate results using Monte Carlo integration are briefly discussed.

2 Introduction

2.1 Monte Carlo Methods

Monte Carlo methods are a type of computer algorithm that involve random sampling. While in theory these methods can be used to solve simple problems, they are far outmatched by other deterministic methods, such as Simpson's rule. [1] It is only for complicated problems such as statistical mechanical systems where Monte Carlo methods become useful.

The validity of Monte Carlo methods is based upon the law of large numbers, namely

$$\lim_{N\to\infty} \sum_{i=1}^{N} \frac{g(X_i)}{N} = \mathrm{E}[g(X)],$$

where X_i, X are identically distributed random variables. By using Monte Carlo methods to generate a sufficiently large number of points, this expected value can be approximated. If the generated points are independent of one another, then the mean squared error of the data is $\frac{\sigma^2}{N}$, where σ^2 is the variance of the random variable.

2.2 Markov Chains

A Markov chain is a sequence of data points where each state is solely determined by the previous state. More explicitly,

$$P\{X_{n+1} = x_i \,|\, X_n = x_j, X_{n-1} = x_{j_{n-1}}, \ldots, X_0 = x_{j_0}\} = P\{X_{n+1} = x_i \,|\, X_n = x_j\} \,\forall\, i, j, j_k, n.$$

An extra condition can be imposed in the case of a homogeneous Markov chain, where the probabilities do not depend on the location of the event in the chain. In practice, when one wants to sample from a probability distribution, a Markov process whose equilibrium distribution is the desired distribution is constructed, generating a sequence of sampled points.

2.3 Thermalisation

One hurdle that must be overcome when using Markov Chain Monte Carlo (MCMC) methods is thermalisation (also called stabilisation or equilibration). If the initial point in the chain is far from equilibrium, then it might take a considerable amount of time until the algorithm sufficiently thermalises, i.e. reaches a point close to equilibrium. When using MCMC methods, it is often wise to generate more samples than required and discard the a certain number of the first points.

2.4 Autocorrelation

By definition, MCMC methods do not generate independent data points. Thus, the mean squared error of the data is not the "naive" $\frac{\sigma^2}{N}$, but rather $\frac{\sigma^2}{N_{\text{eff}}}$, where N_{eff} is the number of effectively independent points.

One approach to calculating the true mean squared error is the binning method. [2] This method involves grouping bins of data of size b and taking its average as a single point in a new chain, generating

$N_b \equiv \frac{N}{b}$ points. Repeating this will lead to the value of $\frac{\sigma^2}{N_b}$ converging to the true mean squared error of the data. A useful quantity called the integrated correlation time τ_{int} is given by

$$\tau_{\text{int}} = \lim_{N_b \to \infty} \frac{\frac{\sigma^2}{N_b}}{\frac{\sigma^2}{N}},$$

and can be interpreted as the number of points needed to be generated equivalent to a single independent point.

2.5 Algorithms

There are many different types of MCMC algorithms used in practice for sampling from a distribution $p(x)$. One such algorithm is the Metropolis-Hastings algorithm. [3] Given a point x in the Markov chain, a new point y is generated from the uniform distribution $[x - \delta, x + \delta]$, where δ is the step size parameter of the algorithm. If $p(y) \geqslant p(x)$, then y is accepted as the next point in the chain. Otherwise, y is accepted with a probability $\frac{p(y)}{p(x)}$.

3 Continuous Distributions

One straightforward application of MCMC methods is the generation of random variables. In this project, the Metropolis-Hastings algorithm was employed to generate samples from normal distributions ($\mathcal{N}(0, \sigma)$) and exponential distributions ($\text{Exp}(\lambda)$) to establish a relationship between the distribution parameters and the optimal choice of step size δ.

3.1 Sampling

From generating normal and exponential distributions varying both σ/λ and δ (Figure 1), the choice of step size was easily seen to strongly influence the validity of the generated points.

3.2 Thermalisation

To determine which choice of step size resulted in the quickest thermalisation, the time for the algorithm to reach certain points was calculated for various starting points. In particular, the time (i.e. number of points) $t_<$ to pass from one side of the mean to the other and the times t_1, t_2, t_3 to be within one, two, and three standard deviations of the mean were calculated (Figure 2). For each choice of σ and λ, it was found that a step size of $\delta \approx \sqrt{10}\,\text{Var}(X)$ minimised these times, where $\text{Var}(X)$ represents the variance of the distribution in question.

3.3 Autocorrelation

Finally, using the binning method (Figure 3), the integrated autocorrelation times for the mean of each distribution and the integrals f_1, f_2 were estimated (Figure 4), where

$$f_1 = \int_{-\infty}^{\infty} \frac{e^{-|x|}}{2} \frac{e^{-\frac{x^2}{2}}}{\sqrt{2\pi}}\, dx = \text{E}_{\mathcal{N}(0,1)}\left[\frac{e^{-|x^2|}}{2}\right] = \frac{\sqrt{e}}{2}\,\text{erf}\left(\frac{1}{\sqrt{2}}\right),$$

$$f_2 = \int_{0}^{\infty} e^{-x} \frac{2e^{-\frac{x^2}{2}}}{\sqrt{2\pi}}\, dx = \text{E}_{\text{Exp}(1)}\left[\frac{2e^{-\frac{x^2}{2}}}{\sqrt{2\pi}}\right] = \sqrt{e}\,\text{erf}\left(\frac{1}{\sqrt{2}}\right).$$

As was the case with thermalisation, the choice of $\delta \approx \sqrt{10}\,\text{Var}(X)$ was found to be optimal for reducing integrated correlation times.

4 Ising Model

While MCMC methods can be used to generate random variables, they are often not used in this fashion as there are other methods that can generate independent samples much quicker. A more reasonable approach is to use these methods to calculate observables of physical models. One such example is the Ising model, a grid made up of spin sites which interact with their nearest neighbours and a magnetic field. While the Ising model may consist of a grid of any dimension and a magnetic field, this project deals with the case of the two-dimensional Ising model with no external field, governed by the Hamiltonian [4]

$$H = -g \sum_{i,j} s_{i,j} \left(s_{i+1,j} + s_{i,j+1} \right),$$

where g determines the strength of the interactions and $s_{i,j} = \pm 1$ represents a value of ± 1 at site (i, j). The probability mass function of the model is given by

$$p(S) \propto e^{-\beta H(S)},$$

where S is a configuration of spins, and $\beta = (k_B T)^{-1}$ is the thermodynamic beta at temperature T. The critical temperature T_c, i.e. the temperature at which a change of phase occurs, is given by [5]

$$T_c = \frac{2|g|}{k_B \ln(1 + \sqrt{2})}.$$

Observables of interest for the Ising model include energy, magnetisation, and heat capacity at constant volume, given by

$$U = H(S), \qquad M = \sum_{i,j} s_{i,j}, \qquad c_V = k_B \beta^2 \mathrm{Var}(U).$$

4.1 Algorithms

For generating random variables, it was reasonable to study the effects of varying the step size parameter δ. For the Ising model, however, there is no such parameter, as MCMC methods are used to attempt spin flips for each site. Instead, the Metropolis-Hastings and heat bath algorithms were used to calculate the energy U, magnetisation M, and heat capacity c_V of the system for a range of temperatures in the cases of $g > 0$ and $g < 0$.

The Metropolis-Hastings algorithm for the Ising model is similar to that for the continuous distributions, accepting a spin flip if it results in a lower energy or with a probability $e^{-\beta(E'-E)}$. The heat bath algorithm, however, is slightly different, accepting a spin flip with probability $\frac{e^{-\beta E'}}{e^{-\beta E} + e^{-\beta E'}}$. The notable difference between these two is that the heat bath algorithm does not always accept a spin flip that results in a lower energy.

4.2 Modifications

When calculating the heat capacity of the system using the heat bath algorithm, it was noticed that the resulting graph contained some unexpected sharp peaks for low temperatures (Figure 5). This was due to the fact that the algorithm had not sufficiently thermalised once measurements began, and so the algorithm was re-run for low temperatures beginning at the configuration that was previously reached by the algorithm, i.e. the point assumed to be close to the equilibrium configuration.

4.3 Autocorrelation

Using the binning method, the integrated correlation times for energy and magnetisation were calculated for both algorithms (Figure 6). The integrated correlation times for the heat capacity were unable to be calculated using the binning method, as many of the graphs were not useful (Figure 7). This is most likely due to the fact that the mean squared error of the heat capacity involves calculating

the variance of the variance of the energy, which can be difficult to calculate, and so many more points would need to be generated to overcome this.

For the energy calculations, the Metropolis-Hastings algorithm was far superior in terms of a lower integrated correlation time, however an interesting result is that for $g = -1$, both algorithms generate points with an integrated correlation time less than 1, which corresponds to each point in the Markov chain being multiple effectively independent points.

4.4 Observables

In each case, the true error of each observable was negligible, as can be seen from the graphs of energy and magnetisation (Figure 8). While the Metropolis-Hastings algorithm is much preferable in terms of integrated correlation times, there is little to no difference in the calculations of the observables.

5 Variance Reduction

Two popular methods of variance reduction are antithetic variables and control variates. [3]

For a random variable $X = h(U_1, \ldots, U_m)$ that depends on m independent uniform distributions $U_1, \ldots, U_m$ on (0,1), then $X' = h(1-U_1, \ldots, 1-U_m)$ will have the same distribution as X. In the case that h is a monotonic function of each of its variables, then X and X' will be negatively correlated. Thus, given a sample point X_i, generating X_i' will result in a more accurate estimate than generating X_j. This is the antithetic variables approach, and is very straightforward to implement.

The control variate approach, however, does not require X to be a monotone function of random numbers. This approach uses the fact that X and $X' = X + c(Y - \mu_y)$ have the same distribution, where Y is a distribution with known expected value μ_y, and c is any constant. The variance of X' is minimised for $c = -\frac{\mathrm{Cov(X,Y)}}{Var(Y)}$, leading to $\mathrm{Var}(X') = \mathrm{Var}(X) - \frac{[\mathrm{Cov}(X,Y)]^2}{\mathrm{Var}(Y)}$. Thus, when X and Y are strongly correlated, the variance of X' is much smaller than that of X.

These methods were used to calculate various one-dimensional and ten-dimensional integrals (Figure 9). There seemed to be little to no difference between the two for the one-dimensional integrals, however the antithetic approach was comparable to no variance reduction for the ten-dimensional integrals.

6 Conclusion

This project focused on the different types of parameters and algorithms in Markov Chain Monte Carlo methods, and how an optimal choice can improve calculations dramatically. As these methods are widely used in areas such as statistical physics and quantum field theory, it is of the upmost importance that such algorithms are chosen and studied carefully when being employed.

7 References

1: A. D. Sokal; Monte Carlo Methods in Statistical Mechanics...; Cargèse Summer School Lecture Series; 1996.
2: B. Berg; MCMC III; Florida State University Lecture Series.
3: S. M. Ross; Simulaton; Academic Press, Cambridge; 5th ed.; 2013.
4: I. Sachs, S. Sen, J. C. Sexton; Elements of Statistical Mechanics...; Cambridge University Press, Cambridge; 1st ed.; 2006.
5: L. Onsager; "Crystal statistics. I. A two-dimensional model with an order-disorder transition"; *Physical Review*, Series II, **65** (3–4): 117–149; 1944.
The code and graphs produced for this project can be found at `https://github.com/campioru/HamiltonJS`

8 Appendix

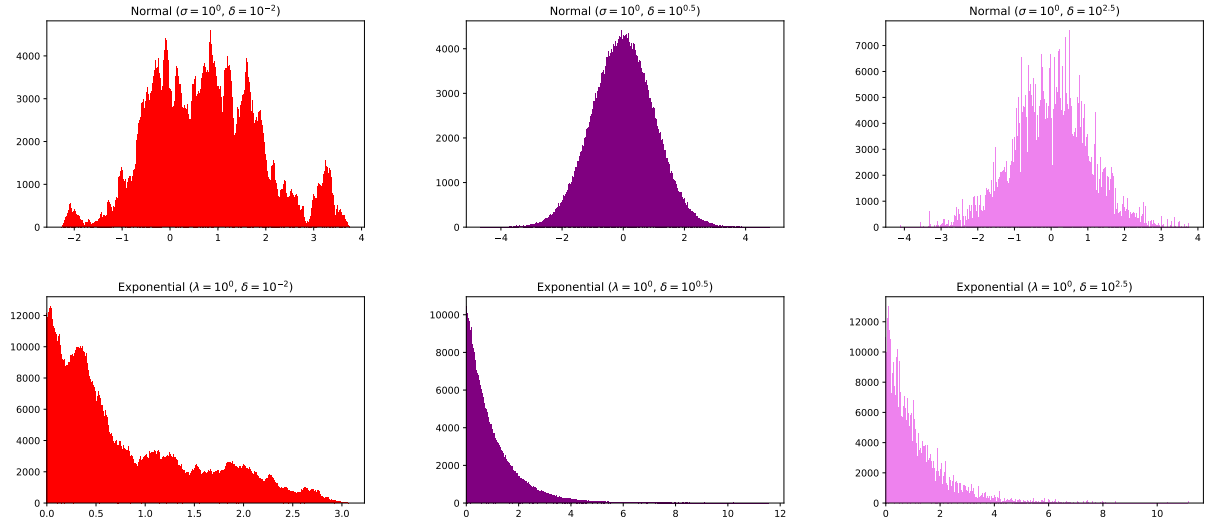

Figure 1: $\mathcal{N}(0,1)$ and Exp(1) distributions generated from Metropolis-Hastings algorithms with step sizes $\delta = 10^{-2}, 10^{\frac{1}{2}}, 10^{\frac{5}{2}}$ and $N = 2^{20}$.

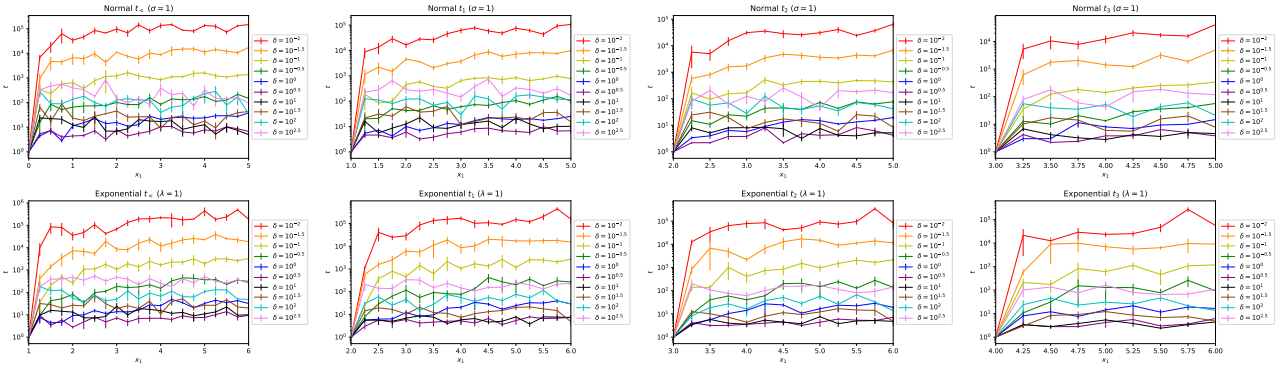

Figure 2: $t_<, t_1, t_2, t_3$ for $\mathcal{N}(0,1)$ and Exp(1) for step sizes $\delta = 10^{-2}, 10^{-\frac{3}{2}}, \ldots, 10^{2}, 10^{\frac{5}{2}}$.

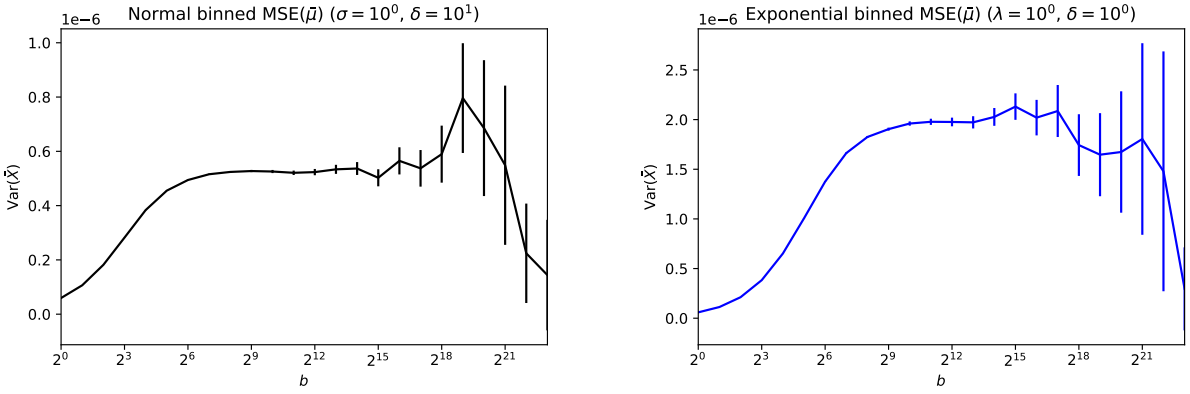

Figure 3: Binning method employed for $\mathcal{N}(0,1)$, $\delta = 10$ and Exp(1), $\delta = 1$ with $N = 2^{24}$.

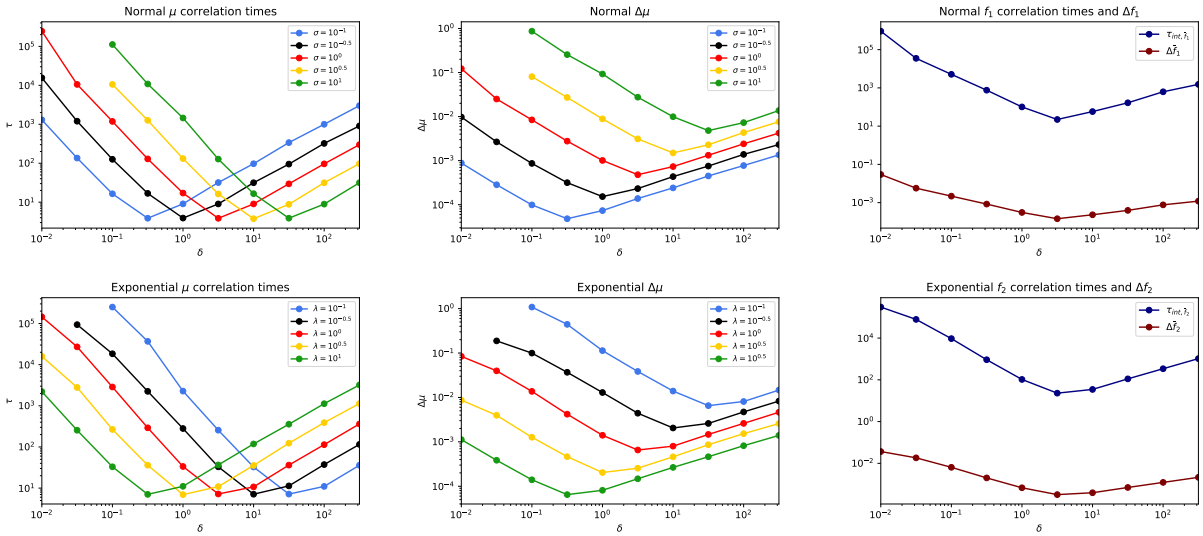

Figure 4: Integrated correlation time estimates for $\mathrm{E}[\mathcal{N}(0,\sigma)]$ and $\mathrm{E}[\mathrm{Exp}(\lambda)]$ for varying step size δ, and for f_1 and f_2.

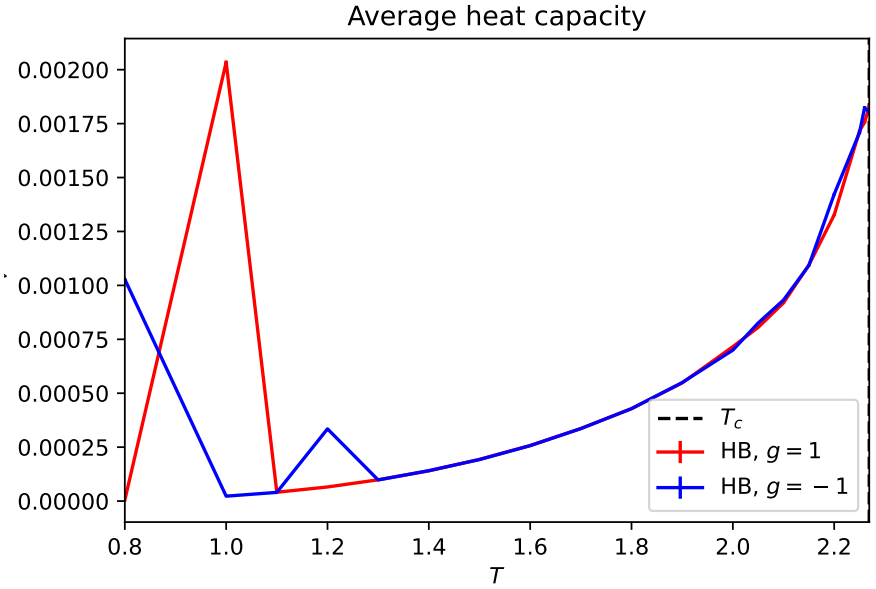

Figure 5: Calculated values of heat capacity using the heat bath algorithm for $g = \pm 1$.

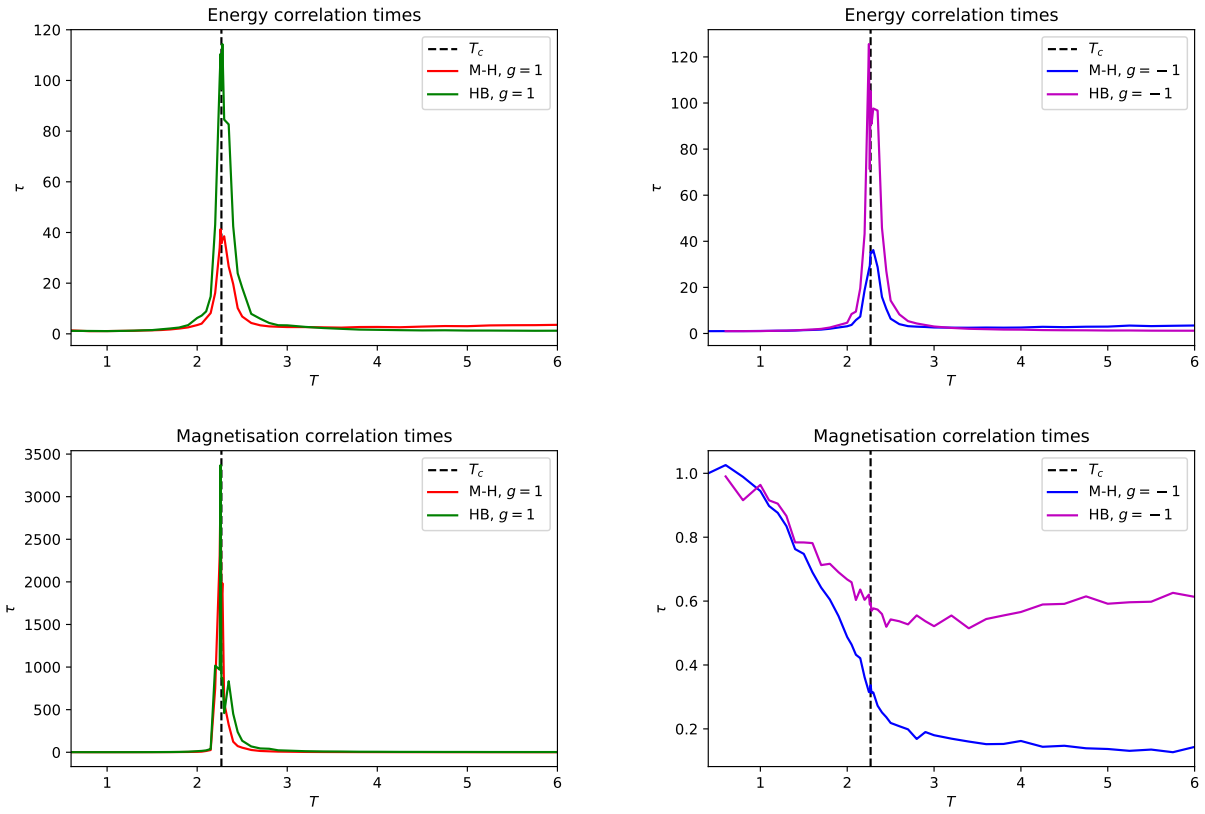

Figure 6: Calculated values of heat capacity using the heat bath algorithm for $g = \pm 1$.

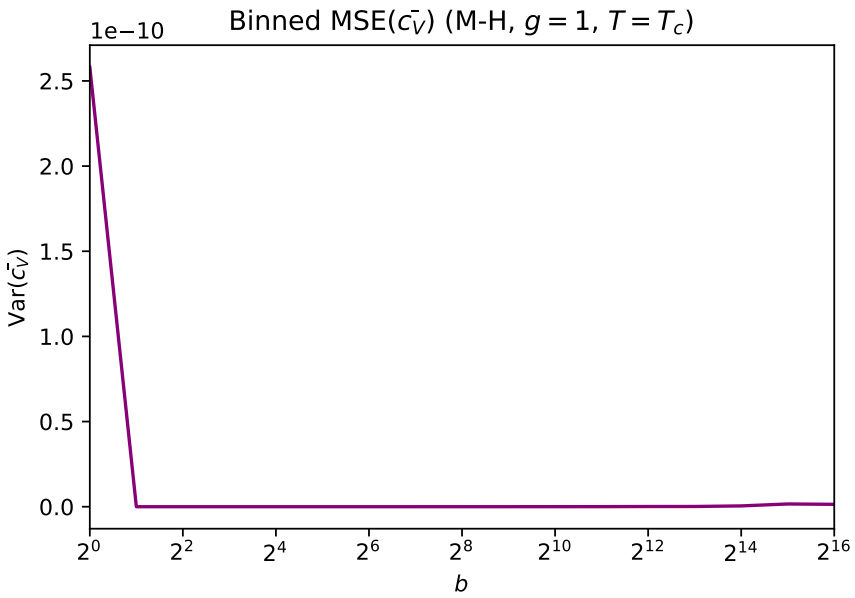

Figure 7: Attempt of the binning method for the $g = 1$ Metropolis-Hastings algorithm at the critical temperature.

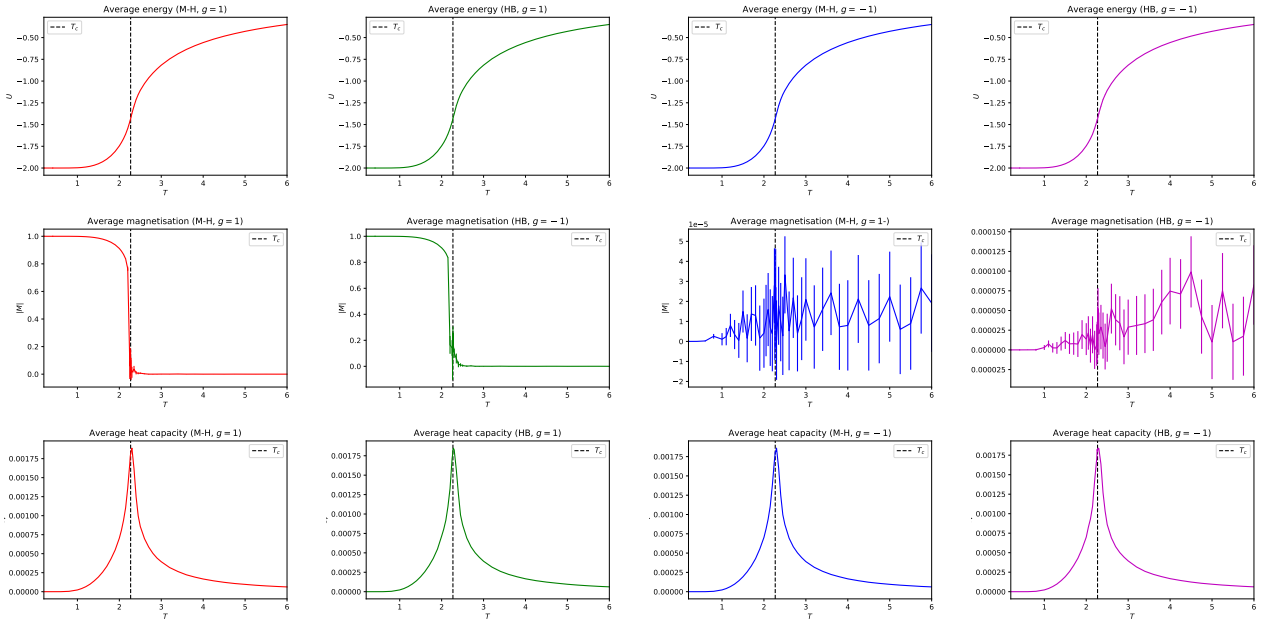

Figure 8: Plots of energy and magnetisation (and their respective true errors) and heat capacity for each algorithm.

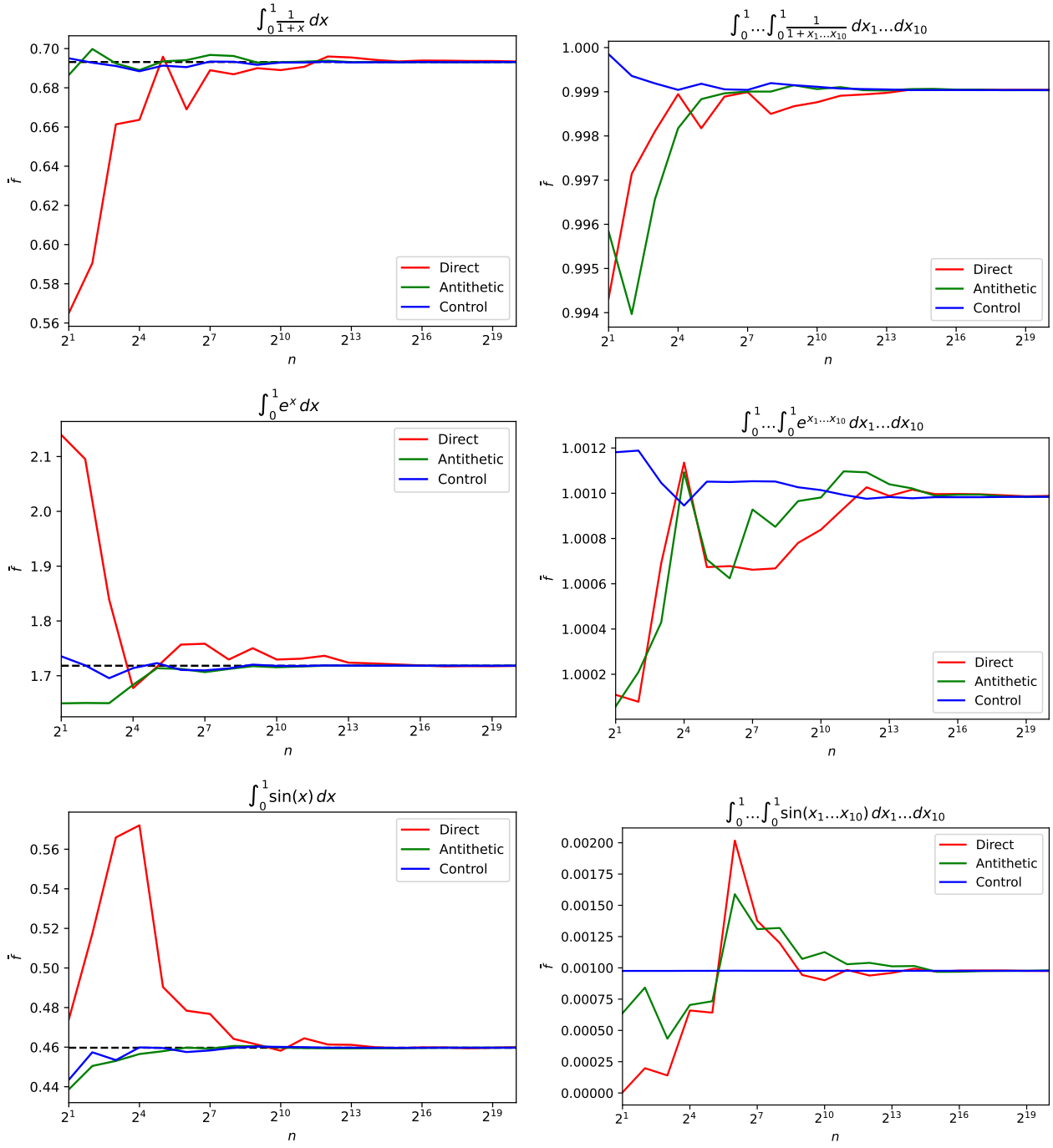

Figure 9: Integrals approximated with Monte Carlo simulations using no variance reduction, antithetic variables, and control variates.