Aalto University
School of Science
Degree programme in Engineering Physics and Mathematics

Estimating the dynamics of a geometric Brownian motion using MCMC

Bachelor's Thesis
02.07.2020

Matias Linnankoski

Aalto University
School of Science

Aalto University, P.O. BOX 11000, 00076 AALTO
www.aalto.fi

Abstract of bachelor's thesis

Author Matias Linnankoski	
Title of thesis Estimating the dynamics of a geometric Brownian motion using MCMC	
Degree programme Engineering Physics and Mathematics	
Major Mathematics and System Sciences	**Code of major** SCI3029
Supervisor Asst.Prof Pauliina Ilmonen	
Thesis advisor(s) M.Sc. Niko Lietzén	
Date 02.07.2020	**Number of pages** 19 + 2 **Language** English

Abstract

This study examines how Metropolis algorithm can be applied to estimate the parameters of a geometric Brownian motion. The Metropolis algorithm is a Markov Chain Monte Caro method, which can be used for solving computational inverse problems. The geometric Brownian motion is a stochastic process which is used for modelling asset prices. It is used in well-recognized models, such as the Black-Scholes option pricing model. The geometric Brownian motion includes two parameters, drift and volatility, from which the first one will be estimated.

The data is generated by simulation and the objective is to measure the performance of the algorithm. The geometric Brownian motion is simulated by using a normally distributed drift and a constant volatility. The Metropolis algorithm is set up such, that it uses the log-normally distributed returns of the geometric Brownian motion. This is accomplished by a change of variable during the Metropolis algorithm. The distribution of the drift is estimated by assuming a known volatility. The estimated distribution for the drift parameter is then compared to the one from which it was generated from. The comparison is made by comparing the means and variances of the distributions which are also communicated through visuals.

The results show that the algorithm performs well for estimating the mean of the distribution of the drift, but the estimated variance differed from the real variance. The thesis shows results of the estimation for one price path, which decreases the creditability of the results. However, the method was tested for multiple different paths and gave reasonably consistent results.

Keywords Markov chain, Monte Carlo, MCMC, geometric Brownian motion, drift, Bayes, parameter estimation

Aalto-yliopisto
Perustieteiden
korkeakoulu

Aalto-yliopisto, PL 11000, 00076 AALTO
www.aalto.fi

Tekniikan kandidaatintyön tiivistelmä

Tekijä Matias Linnankoski

Työn nimi Geometrisen Brownin liikkeen parametrin estimointi MCMC avulla

Koulutusohjelma Teknillinen fysiikka ja matematiikka

Pääaine Matematiikka ja systeemitieteet **Pääaineen koodi** SCI3029

Vastuuopettaja Apulaisprofessori Pauliina Ilmonen

Työn ohjaaja(t) DI Niko Lietzén

Päivämäärä 02.07.2020 **Sivumäärä** 19 + 2 **Kieli** Englanti

Tiivistelmä

Työssä tutkitaan, miten Metropolis algoritmia pystytään hyödyntämään geometrisen Brownin liikkeen parametrin estimoimiseen. Metropolis algoritmi on Markovin ketju Monte Carlo simulointi menetelmä, jota pystyään käyttämään laskennallisten inversio-ongelmien ratkaisemiseen. Geometrinen Brownin liike on stokastinen prosessi, jota käytetään rahoituksessa osakkeiden hintojen mallinnukseen. Geometrinen Brownin liike koostuu kahdesta parametrista, virtaus ja volatiliteetti, joista ensimmäinen pyritään estimoimaan.

Työssä käytetty data generoidaan simuloimalla. Työn tarkoituksena on mitata, miten Metropolis algoritmi suoriutuu virtaus parametrin estimoinnissa. Virtaus parametrin estimointi tehdään asetelmassa, jossa volatiliteetti on tunnettu. Geometrinen Brownin liike simuloidaan, siten että, virtaus noudattaa normaalijakaumaa ja volatiliteetti pysyy vakiona. Metropolis algoritmi hyödyntää parametrin estimoimisessa geometrisen Brownin liikkeen lisäyksiä, jotka ovat log-normaalisti jakautuneita. Tulokset validoidaan vertailemalla estimoidun ja todellisen jakauman keskiarvoja ja varianssseja.

Algoritmi suoriutui hyvin jakauman keskiarvon estimoimisessa, mutta sen varianssin estimaatti oli kaksinkertainen todelliseen verrattuna. Parametrin estimointi esitetään työssä vain yhdelle geometrisen Brownin liikkeen polulle, joka heikentää tulosten luottavuutta. Estimointia kuitenkin kokeiltiin työn aikana usealle eri polulle, ja tulokset olivat melko tasaisia.

Avainsanat Markov ketju, Monte Carlo, MCMC, geometrinen Brownin liike, Bayes, parametrin estimointi

Contents

1 Introduction

In 1946 a mathematician named Stanislaw Ulam tried to figure out the probability of winning a game of solitaire. This appeared to be a challenging problem to be solved analytically, and hence he came up with an alternative method to solve the problem: play solitaire numerous times and calculate the percentage of games which turned out successful. He figured that this approach could be used for solving other complex problems by transforming uncertain events into random operations and simulating them on a computer, which was then later named as Monte Carlo simulation.[14]

During the last decades technology has developed rapidly and the increased computing capacity has become a significant asset in modern problem solving [17]. This has allowed the creation of new modelling techniques and further development of the ones created earlier. By exploiting Markov Chains, the main idea behind Monte Carlo simulation has been applied in order to sample from different kinds of probability distributions. These kinds of algorithms are denoted as Markov Chain Monte Carlo (MCMC) methods, which consists of multiple of different algorithms with some common principles [3]. The most recognized algorithm from the MCMC class is the Metropolis-Hastings algorithm, which has been referred to as one of the most influential algorithms during the recent centuries [5].

In this thesis, the Metropolis-Hastings method will be used to estimate the parameters of a Geometric Brownian Motion (GBM). GBM, which will be introduced more thoroughly later, is a stochastic process which is used in finance to model asset prices [7]. Although it can be considered as a simplified way of modelling an asset price, it is applied in well-recognized models, such as the Black-Scholes option pricing model [4].

The geometric Brownian motion used in this study is simulated, making the thesis a simulation study. Simulation studies are crucial for valuating the performance of new algorithms and statistical methods. They are also useful for finding wrong assumptions in methods, and testing the methods sensitivity to perturbed data. [13]

This thesis begins by covering the theory behind the methods used. The methods will be introduced in a separate section as well as the implementation of them. Finally, the results of the thesis will be covered and concluded.

2 Background

2.1 Theory

Throughout, let $(\Omega, \mathcal{F}, P)$ be a common probability space for the random variables considered in this thesis.

A stochastic process has the Markov property, if its future state depends only on its current state. A discrete time stochastic process which satisfies the Markov property is called a Markov chain, whereas a continuous time process satisfying the property is called a Markov process [10]. This section will start by defining a Markov chain and some of its properties, which are required by the MCMC algorithm.

Definition 2.1. A sequence of random variables $(X_0, X_1, ...X_t)$ taking values in the countable state space S is a *Markov chain*, if it satisfies the following

$$\mathsf{P}(X_{t+1} = x_{t+1}|X_t = x_t, X_{t-1} = x_{t-1}, ..., X_0 = x_0) = \mathsf{P}(X_{t+1} = x_{t+1}|X_t = x_t)\,.$$

Since the transition probabilities depend only on the previous states, the transition probabilities of a Markov Chain, taking values in the countable state space S, can be expressed with a matrix. This kind of a matrix is called a *transition matrix*. In a transition matrix P, the element P_{ij} corresponds to the probability of the process moving from state i to j

$$P_{ij} = P(i,j) = \mathsf{P}(X_{t+1} = j|X_t = i)\ ,$$

where $i, j \in S$.

The Metropolis-Hastings algorithm, which will be used in the parameter estimation, requires that the underlying Markov Chain is aperiodic and irreducible in order to have an unique stationary distribution [19]. The definitions for these requirements are presented below.

Definition 2.2. Let $X_0, X_1, ..., X_n$ be a Markov Chain, taking values in the countable state space S, and having the transition matrix P. The period of state i, denoted as $d(i)$, is defined as the greatest common divisor (gcd)

$$d(i) = \mathrm{gcd}(n : P^n(i,i) > 0)\ ,$$

where P^n denotes the $n:th$ power of P. A Markov Chain is called *aperiodic*, if $d(i) = 1, \forall i \in S$.

Definition 2.3. Let $X_0, X_1, ..., X_n$ be a Markov Chain, taking values in the countable state space S and having the transition matrix P. The Markov

Chain is called *irreducible*, if for all states $i, j \in S$ there exists a $t \geq 1$, for which $P^t(i, j) > 0$.

The stationary distribution describes the distribution for the states over the long run of the Markov chain. The mathematical definition for this goes as follows.

Definition 2.4. Let π be a probability distribution. The probability distribution π is called the *stationary distribution* of a Markov Chain, taking values in the countable state space S and having the transition matrix P, if

$$\sum_{i \in S} \pi(i) P(i, j) = \pi(j) \ , j \in S \ .$$

The central limit theorem is relevant for the convergence of the underlying Markov chain in the MCMC algorithm [9]. The Central Limit Theorem states (CLT) that the sample mean of independently and identically distributed observations converges in distribution to a normal distribution, when $n \to \infty$.

Theorem 2.1 (Central Limit Theorem)**.** *Let $X_1, X_2, ..., X_n$ be a sequence of independently and identically distributed (iid) random variables, each having the expected value $E(X_i) = \mu$ and variance $\operatorname{Var}(X_i) = \sigma^2 < \infty$. Let $\overline{X}_n$ be the sample mean, defined as,*

$$\overline{X}_n = \frac{1}{n} \sum_{i=1}^{n} X_i \ .$$

The central limit theorem states that,

$$\sqrt{n}(\overline{X}_n - \mu) \xrightarrow[n \to \infty]{\mathcal{D}} N(0, \sigma^2) \ ,$$

where $\xrightarrow{\mathcal{D}}$ denotes convergence in distribution.

The MCMC algorithm has numerous iterations in order to obtain a large sample size for the results. The Weak Law of Large Numbers introduces the benefits for the large sample size.

Theorem 2.2 (Weak Law of Large Numbers)**.** *Let $X_1, X_2, ..., X_n$ be a sequence of independently and identically distributed (iid) random variables, each having the expected value $E(X_i) = \mu < \infty$ and variance $\operatorname{Var}(X_i) = \sigma^2$. Let $\overline{X}_n$ be the sample mean, defined as,*

$$\overline{X}_n = \frac{1}{n} \sum_{i=1}^{n} X_i \ .$$

The Weak Law of Large Numbers states that,

$$\overline{X}_n \xrightarrow[n\to\infty]{\mathsf{P}} \mu \ ,$$

where $\xrightarrow{\mathsf{P}}$ *denotes convergence in probability.*

More detailed information about convergence in probability and distribution can be found in reference [1].

Calculating expected values may require solving integrals which are difficult to compute. These kinds of integrals can be approximated by using Monte Carlo techniques [15].

Assume that X is a random variable with a probability density function $p(x)$. The expected value of $f(X)$ can now be approximated according to the Weak Law of Large numbers by

$$E[f(X)] = \int f(x)d(x)\,dx \approx \frac{1}{n}\sum_{i=1}^{n} f(x_i) \ ,$$

where x_i are observed or sampled values from the distribution $p(x)$ and $n \to \infty$.

The MCMC algorithm applies Bayes' Theorem for obtaining the posterior distribution for the distribution being sampled.

Theorem 2.3 (Bayes' Theorem)**.** *Let* $(\Omega, \mathcal{F}, \mathsf{P})$ *be a probability space, where* Ω *is a sample space,* $\mathcal{F}$ *is a sigma-algebra on* Ω *and* P *a probability measure on* $(\Omega, \mathcal{F})$*. Assume that* $A, B \in \mathcal{F}$ *are events such that* $\mathsf{P}(A) > 0$ *and* $\mathsf{P}(B) > 0$*. The Bayes' theorem states that:*

$$\mathsf{P}(A|B) = \frac{\mathsf{P}(A \cap B)}{\mathsf{P}(B)} = \frac{\mathsf{P}(A)\mathsf{P}(A|B)}{\mathsf{P}(B)} \ .$$

Assume that $f(x|\theta)$ is a density function with the parameter θ, which generates independent random variables. The observers belief of the parameter θ is described with the prior distribution $p(\theta)$. Given the observation x_1, generated from the density function $f(x|\theta)$, the posterior distribution can be calculated as

$$p(\theta|x_1) = \frac{p(\theta)f(x_1|\theta)}{\sum_{\theta'} p(\theta')f(x_1|\theta')} \ .$$

More generally, when observing the data point x_n, the distribution $p(\theta|x_{n-1}, ..., x_1)$ is used as the prior distribution and it can be updated into a posterior distribution in a similar way

$$p(\theta|x_n, ..., x_1) = \frac{p(\theta|x_{n-1}, ..., x_1) f(x_n, ..., x_1|\theta)}{\sum_{\theta'} p(\theta'|x_{n-1}, ..., x_1) f(x_n, ..., x_1|\theta')} \ .$$

2.2 Asset Price Dynamics

This study uses an asset price model, which focuses on two features in the asset prices: drift and volatility. The drift describes the deterministic part of the changes in the asset price, happening in each infinitesimal timestep. Since the asset price is a stochastic process, it also contains randomness. This randomness creates fluctuations in the price producing stochastic variance, or in this context, volatility.

2.2.1 Volatility

Volatility plays an important role when pricing financial derivative securities [18]. It describes the variation in the returns of a financial instrument and is often associated with risk. An asset's historical volatility σ_r can be calculated by exploiting its historical daily spot prices $(S_0, S_1, ..., S_n)$ with

$$\sigma_r^2 = A \sum_{i=1}^{n} \frac{\ln(S_i/S_{i-1})^2}{n} \ ,$$

where A is an annualization factor. However, this value is based on the variation of prices during some historical time interval. In fact, the volatility of an asset changes within time and thus describing an asset's volatility as a time independent constant is a harsh simplification [8].

A volatility which varies over time and contains randomness is often referred with the term stochastic volatility. The effects of this time-varying volatility in financial instruments can be described using stochastic volatility models. For simplicity this thesis considers only constant volatility, but for further reading about the topic see reference [2].

2.2.2 Drift

The drift describes the mean returns of an asset. If there were no volatility present in the asset, the drift μ would correspond to the continuously compounding risk free rate r. In our model, volatility decreases the expected returns of an asset in a longer time horizon. A short explanation for this goes followingly: The volatility is generated by an underlying Brownian motion, which has normally distributed increments. If the price drops within a timestep with 20%, it should increase by 25% to reach the original price. However, the 20% drop is more probable than the 25% rise.

To adjust this, a volatility drag coefficient is subtracted from the drift. In a risk-neutral economy, the following equality should hold

$$\mu = r + \frac{1}{2}\sigma^2$$

For further details about risk neutrality, see reference [11].

3 Methods

3.1 Simulation

In this thesis, we will generate the data by simulation. This enables us to validate the results of the parameter estimation reliably. Thus, the study covers the basics in the simulation of financial asset returns. The randomness in the GBM asset price model is created by a Brownian Motion. Discrete time Brownian motion is a stochastic process, which can be interpreted as a random walk, where the increments are normally distributed with a mean of zero and their variance corresponds to the size of the time-step.

Definition 3.1. A stochastic process $\{B_t\}_{t\geq 0}$ in the probability space $(\Omega, \mathcal{F}, \mathsf{P})$ is a standard Brownian motion if it satisfies:
i) $\mathsf{P}(B_0 = 0) = 1$
ii) $B_{t_n} - B_{t_{n-1}}, B_{t_{n-1}} - B_{t_{n-2}}, ..., B_{t_1} - B_{t_0}$ are mutually independent for any $0 = t_0 < t_1 < ... < t_n$.
iii) For any $0 \leq s < t$, the increment $B_t - B_s \sim N(0, t-s)$
iv) Process B_t has almost surely continuous paths.

We assume that the stock price S_t follows a Geometric Brownian Motion:

$$dS_t = \mu S_t \, dt + \sigma S_t \, dB_t \ , \tag{1}$$

where μ is a constant drift term, σ constant volatility and X_t a standard Brownian Motion. This stochastic differential equation can be represented in the following discrete form

$$\Delta S_t = \mu S_t \Delta t + \sigma S_t \epsilon \sqrt{\Delta t} \ ,$$

where ϵ follows the distribution $N(0,1)$.

This stochastic differential equation (1) can be solved by applying Itô's lemma. The derivation of the equation is presented in appendix section A, and has the following analytical solution

$$S_t = S_0 e^{(\mu - \frac{1}{2}\sigma^2)t + \sigma B_t} ,$$

where the Brownian Motions B_t expected value is 0 and variance t.

Thus, the logarithm of the returns $\frac{S_t}{S_0}$ is normally distributed:

$$\ln(\frac{S_t}{S_0}) \sim N((\mu - \frac{1}{2}\sigma^2)t, \sigma^2 t) \tag{2}$$

This enables us to to simulate a path for our asset price. By splitting the time period $[0,T]$ into a desired amount of time-steps
$(0 = t_0 < t_1 < ... < t_n = T)$, we can simulate an asset's price path as:

$$S_{t_{i+1}} = S_{t_i} e^{\epsilon_i} ,$$

where ϵ_i is a sample generated from the distribution $N((\mu - \frac{1}{2}\sigma^2)(t_{i+1} - t_i), \sigma^2(t_{i+1} - t_i))$ and $i = 0, 1, ..., n-1$.

A few examples of GBM paths are visualised in Figure 1. Note that these paths are not the ones used in this thesis.

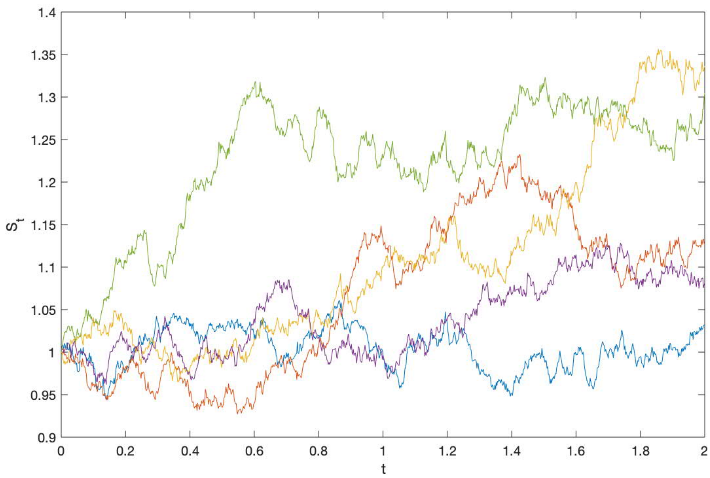

Figure 1: Paths for simulated stock prices, where $\mu = 0.1, \sigma^2 = 0.1$ and $S_0 = 1$.

3.2 Metropolis-Hastings

The objective of the M-H algorithm is to to generate samples from the target distribution π, from which conventional sampling may be difficult. To simulate the target distribution, a proposal density $q(y|x)$ where the next state y depends on the previous state x is introduced. The candidates generated by the proposal distribution are accepted with the probability ratio

$$\alpha(y, x) = \min(\frac{\pi(y)q(y|x)}{\pi(x)q(x|y)}, 1) \ . \tag{3}$$

In the M-H algorithm, the target distribution has to be known to some proportionality since in the acceptance rate (3) the normalization constant gets cancelled. By generating a sample between $[0, 1]$ from a uniform distribution and comparing it to the acceptance rate, we are able to accept the candidate samples with a correct frequency.

Assuming that the proposal distribution is chosen appropriately, the acceptance rate guides the algorithm to converge to the stationary distribution, satisfying

$$\pi(y)q(x|y) = \pi(x)q(y|x) \ .$$

The convergence may be sensitive to the initialization of the parameters and they should be selected carefully. Even if the parameters are initialized appropriately, it takes time for the algorithm to reach the stationary distribution. Thus, some amount of the first output samples are discarded when evaluating the results. These discarded samples are often referred to as burn in samples. For further details of the requirements of the algorithm, see reference [12].

The general algorithm is presented in pseudo code below in Algorithm 1. Note that x and y are not necessarily single values and that they can present, e.g., a vector of parameters.

Algorithm 1: Metropolis Hastings

Result:
$P(x)$, target distribution;
$q(x|y)$, proposal density function;
x_1, intial guess;
$\alpha(x, y)$, acceptance rate;
for i = *1,2,...,n* **do**
    $y \leftarrow q(y|x_i)$;
    $u \leftarrow U(0, 1)$;
    if $u > \alpha(y, x_i)$ **then**
        $x_{i+1} = y$;
    else
        $x_{i+1} = x_i$;
    end
end
return $(x_1, x_2, ..., x_n)$

This thesis applies M-H to a fairly simple problem, and does not show the full capability of the algorithm. Usually the algorithm is used for sampling from more complex distributions.

4 Implementation

The parameters for the GBM were chosen as follows

$$\begin{aligned} \mu &\sim N(0.3, 0.05^2) \\ \sigma^2 &= 0.1^2 \\ T &= 10 \\ n &= 255 \cdot T \\ dt &= \frac{T}{n} = \frac{1}{100} \ . \end{aligned}$$

Since the parameter estimation exploits only the returns of the price, it would be sufficient to only generate the returns and store them. Thus, the problem could be rephrased as an estimation of the parameters of Gaussian data, with a Gaussian mean.

The price path used for this study is shown in Figure 2.

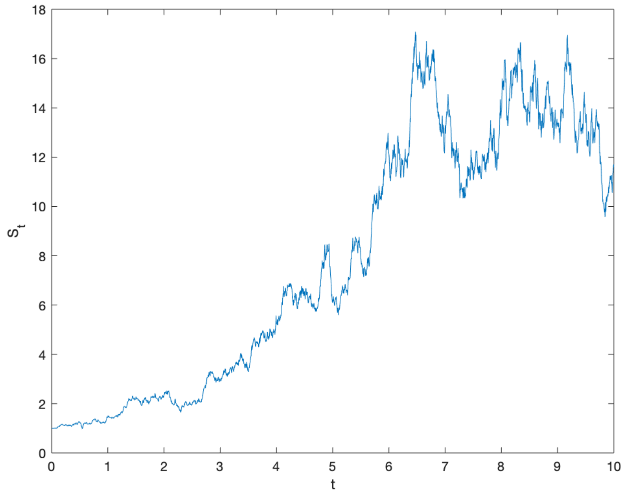

Figure 2: Price path which is used for the MCMC algorithm.

The returns Y_t for a price path S_t are calculated as

$$Y_{t_i} = \ln(\frac{S_{t_i}}{S_{t_{i-1}}}) \ ,$$

where $0 = t_0 < t_i \leq t_n$. Since the increments in the price between timesteps are mutually independent, the distribution for each Y_{t_i} can be calculated with equation (2). Thus, Y_{t_i} follows the distribution $N((\mu - \frac{1}{2}\sigma^2)dt, \sigma^2 dt)$, where $dt = t_i - t_{i-1}$. Figure 3 illustrates the returns of the simulated price path.

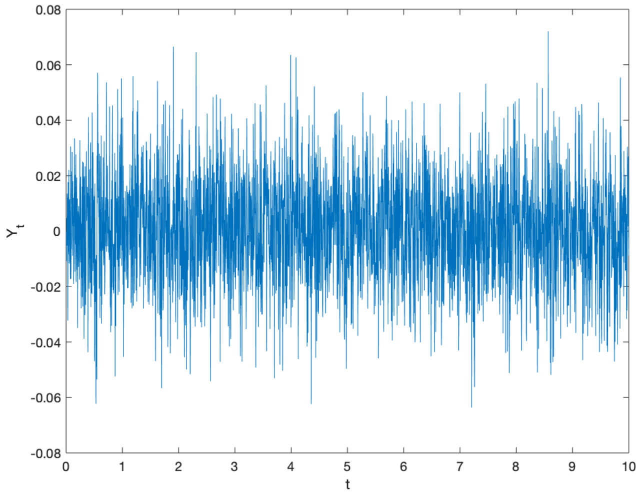

Figure 3: Returns of the price path which is used for the MCMC algorithm.

The likelihood for Y as a univariate function of μ is visualised in Figure 4. We see how the likelihood function reaches its maximum roughly between 0.2 and 0.3, as expected. This includes the volatility drag coefficient $-\frac{1}{2}\sigma^2$, which will be cancelled out when calculating the results.

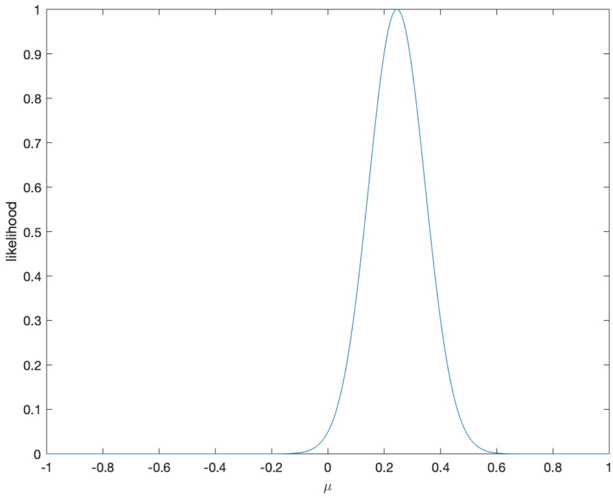

Figure 4: Likelihood function for Y as a function of μ, when $\sigma^2 = 0.1$. The likelihood function is scaled such that it takes values between 0 and 1.

The likelihood for Y as a univariate function of σ^2 is visualised in Figure 5.

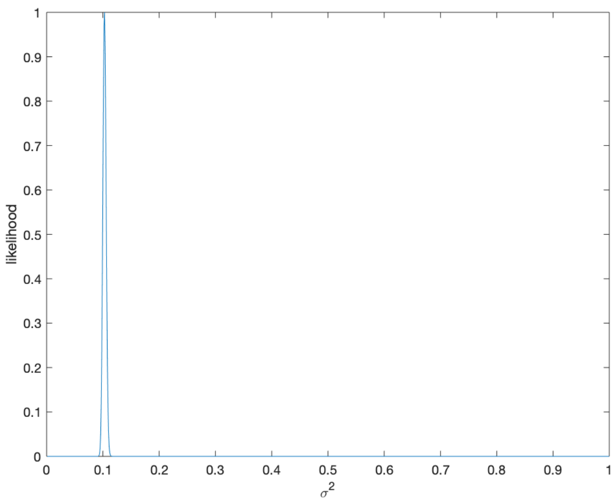

Figure 5: Likelihood function for Y as a function of σ^2, when $\mu = 0.3$. The likelihood function is scaled such that it takes values between 0 and 1.

We see how the likelihood function as a function of σ^2 peaks at slightly over 0.1. This might be due to the deviation caused by the distribution of μ. The true variation for the returns can be derived as

$$\begin{aligned} Y_i &\sim (N(\mu, \sigma_\mu^2) - \frac{1}{2}\sigma^2)dt + N(0, \sigma^2 dt) \\ Y_i &\sim N(\mu dt, \sigma_\mu^2 dt^2) + N(0, \sigma^2 dt) - \frac{1}{2}\sigma^2 dt \\ Y_i &\sim N(\mu dt, \sigma_\mu^2 dt^2 + \sigma^2 dt) - \frac{1}{2}\sigma^2 dt \ , \end{aligned}$$

which yields $\mathrm{Var}(Y_i) = \sigma_\mu^2 dt^2 + \sigma^2 dt$.

However, since the distribution of μ is treated at this stage as unknown, the likelihood function will not exploit its variance. Thus, the likelihood function is conditioned only with the known variance.

The conditional likelihood for the vector Y is

$$p(Y|\mu, \sigma^2) = (2\pi\sigma^2 dt)^{-\frac{n}{2}} \exp(-\frac{\sum_{i=0}^{n} (Y_i - (\mu - \frac{1}{2}\sigma^2)dt)^2}{2\sigma^2 dt}) \ .$$

By making the variable changes

$$\hat{\mu} = (\mu - \frac{1}{2}\sigma^2)dt \tag{4}$$

$$\hat{\sigma} = \sigma^2 dt \ , \tag{5}$$

the likelihood function can be expressed in a more simple form

$$p(Y|\mu, \sigma^2) = (2\pi\hat{\sigma}^2)^{-\frac{n}{2}} \exp(-\frac{\sum_{i=0}^{n} (Y_i - \hat{\mu})^2}{2\hat{\sigma}^2}) \ .$$

The objective is to sample from the posterior distribution $p(\mu|Y, \sigma^2)$. To accomplish this, we will first sample from the distribution $p(\hat{\mu}|Y, \hat{\sigma}^2)$, and transform the samples with the closed form equation (4).

First, we will assume the following prior distribution

$$p(\hat{\mu}) \sim N \ .$$

Bayes Theorem implies that the posterior distribution for $\hat{\mu}$ is

$$p(\hat{\mu}|Y, \hat{\sigma}^2) = \frac{p(Y|\hat{\mu}, \hat{\sigma}^2)p(\hat{\mu})}{\int p(Y|\hat{\mu}, \hat{\sigma}^2)p(\hat{\mu})d\hat{\mu}} \ ,$$

having the proportionality

$$p(\hat{\mu}|Y, \hat{\sigma}^2) \propto p(Y|\hat{\mu}, \hat{\sigma}^2)p(\hat{\mu}) \ .$$

Since the prior distribution for $\hat{\mu}$ is normal, its transition kernel will also be normal. The new candidates $\hat{\mu}_p$ will be drawn from

$$\hat{\mu}_p \sim N(\hat{\mu}_i, \sigma_0^2) \ ,$$

where $\hat{\mu}_i$ is the current state and σ_0^2 a predefined variance for the distribution. The variance will be adjusted by testing different values and looking at the acceptance percent of the algorithm. For example, a too large variance in the proposal distribution will result in a small acceptance rate, which is undesirable. The importance of the proposal distribution's variance is discussed in reference [16].

Since the proposal distribution is symmetric, the algorithm used is more precisely a Metropolis algorithm. The Metropolis-Hastings algorithm is a generalized version of the Metropolis algorithm, where the proposal distribution is symmetric. The symmetric proposal density gets cancelled out from the acceptance rate, making the acceptance rate depend only on the target distribution. [5]

The algorithm used in the study is presented in Algorithm 2.

Algorithm 2: Metropolis

Result:
$x_i = 0.01$;
for i = *1,2,...,n* **do**
  $x_p \leftarrow N(x_i, \sigma_0^2)$;
  $\alpha \leftarrow \max(\frac{P(Y|\mu_p,\sigma^2)}{P(Y|\mu_i,\sigma^2)}, 1)$;
  $u \leftarrow U(0,1)$;
  if $u > \alpha$ **then**
    $x_{i+1} = x_p$;
  else
    $x_{i+1} = x_i$;
  end
end
return $(x_1, x_2, ..., x_n)$

When using large sample sizes the likelihood functions takes large values which results in numerical overflows in computation. To mitigate this problem the probabilities and their ratios were calculated in log scale.

After the simulation, the samples $\hat{\mu} = (\hat{\mu}_1, \hat{\mu}_2, ..., \hat{\mu}_m)$ were transformed as

$$\mu = \frac{\hat{\mu}}{dt} + \frac{1}{2}\sigma^2 \ ,$$

and the transformations were then used for evaluating the performance. This transformation is done for all data used in the results section.

5 Results

The amount of iterations used in the Metropolis algorithm was 10000. The algorithm converged quickly, depending on the chosen initial value and the variance of the proposal distribution. The 2000 first samples were considered as burn in samples, and thus were discarded. Figure 6 presents the trace plot for all generated samples.

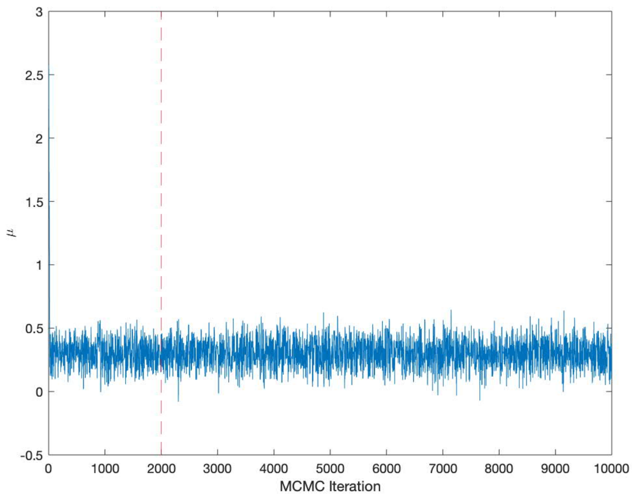

Figure 6: The trace of the sampled values of μ. The red line describes the burn in limit.

We see how the iterations converge quickly from the initial value and create a consistent pattern. The autocorrelation, which is presented in Figure 7, also supports the convergence.

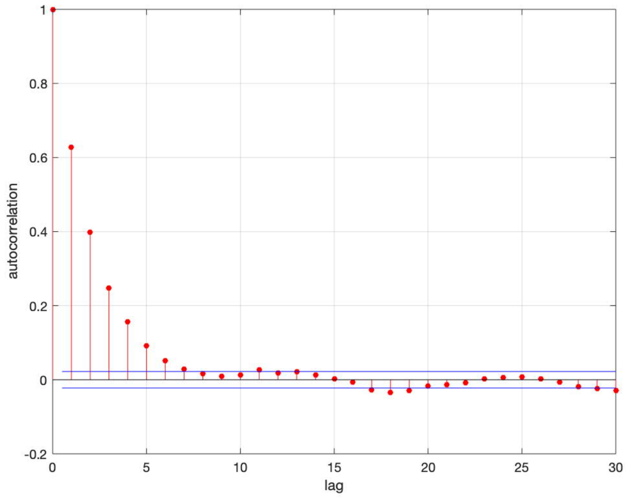

Figure 7: Autocorrelations of the samples as a function of lag.

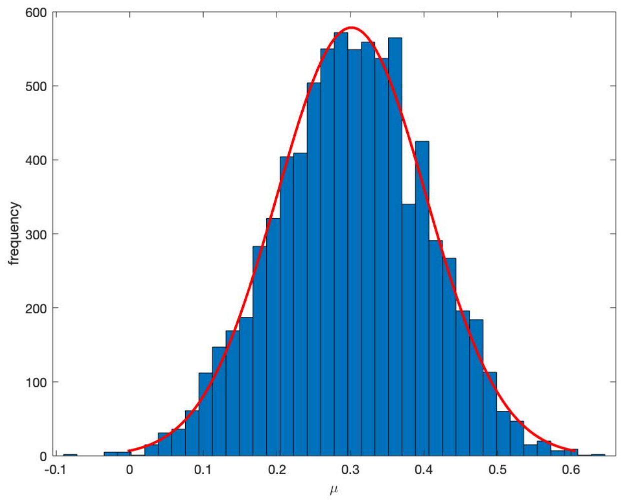

Figure 8: Histogram and a normal distribution fit for the MCMC samples.

The histogram in Figure 8 shows the estimated posterior distribution of μ. The fitted normal distribution is $N(0.30, 0.10)$. More detailed statistics of the estimated posterior distribution are presented in Table 1.

Table 1: Comparison of the estimated and known values for μ. The known values are presented in a bold format. The percents present confidence intervals.

Parameter	Mean	SD	2.5%	Median	97.5%
μ	0.302	0.102	0.100	0.303	0.495
$\boldsymbol{\mu}$	**0.3**	**0.05**	**0.202**	**0.3**	**0.398**

The estimation of the mean performed very well, the error being under 1% compared to the real value. However, the standard deviation of the estimated distribution differed considerably from the real one. This thesis lacks the estimation of the variance, and therefore the estimated variance of μ is not reliable. Although the study was only made for one sample, the results were fairly consistent also for other simulated data samples.

6 Conclusions

The study introduced how geometric Brownian motion can be used to model an asset's price. This theory was then used to simulate a price path, where the drift followed a normal distribution and the variance was set as a constant. The mean of this simulated price path was estimated using a Metropolis algorithm to validate how the algorithm works for this kind of data.

The results were good regarding the mean of the drift, however the variance of the drift differed from the real, which might be a result of a relatively small sample size or a bias in the model. This implies that the algorithm could be used to model the drift of a geometric Brownian motion. However, the variation of the drift was significantly smaller than the variance caused by the Brownian motion, which gave a good basis for the study.

The asset price model used assumes that the dynamics of the asset remains constant during the time series. Earlier studies imply, that the drift rate is sensitive to changes in the economic variables, and news stories related to the asset [6]. In addition, asset prices often include jumps, which are not taken into account by the geometric Brownian motion [20]. The first step in continuing the study would be including the estimation of the variance. After this, the geometric Brownian motion could be replaced with a more sophisticated model for the asset price.

References

[1] Patrick Billingsley. *Convergence of probability measures.* John Wiley & Sons, 2013.

[2] Tim Bollerslev and Hans Ole Mikkelsen. Modeling and pricing long memory in stock market volatility. *Journal of Econometrics*, 73(1):151–184, 1996.

[3] Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of Markov chain Monte Carlo.* CRC press, 2011.

[4] Marc Chesney and Louis Scott. Pricing European currency options: a comparison of the modified Black-Scholes model and a random variance model. *Journal of Financial and Quantitative Analysis*, 24(3):267–284, 1989.

[5] Siddhartha Chib and Edward Greenberg. Understanding the Metropolis-Hastings algorithm. *The American Statistician*, 49(4):327–335, 1995.

[6] David M Cutler, James M Poterba, and Lawrence H Summers. What moves stock prices? Technical report, National Bureau of Economic Research, 1988.

[7] Hans U Gerber and Elias SW Shiu. Geometric Brownian motion models for assets and liabilities: from pension funding to optimal dividends. *North American Actuarial Journal*, 7(3):37–51, 2003.

[8] Eric Ghysels, Andrew C Harvey, and Eric Renault. 5 stochastic volatility. *Handbook of Statistics*, 14:119–191, 1996.

[9] Galin L Jones et al. On the Markov chain central limit theorem. *Probability Surveys*, 1(299-320):5–1, 2004.

[10] N.G. Van Kampen. Markov process - an overview.
`https://www.sciencedirect.com/topics/mathematics/markov-process`.
(Online, accessed 01.06.2020).

[11] Stephen F LeRoy and Charles J LaCivita. Risk aversion and the dispersion of asset prices. *Journal of Business*, pages 535–547, 1981.

[12] Faming Liang. On the use of stochastic approximation Monte Carlo for Monte Carlo integration. *Statistics & Probability Letters*, 79(5):581–587, 2009.

[13] Tim P Morris, Ian R White, and Michael J Crowther. Using simulation studies to evaluate statistical methods. *Statistics in Medicine*, 38(11):2074–2102, 2019.

[14] Christian Robert and George Casella. A short history of Markov chain Monte Carlo: subjective recollections from incomplete data. *Statistical Science*, pages 102–115, 2011.

[15] Christian P Robert and George Casella. The Metropolis-Hastings algorithm. In *Monte Carlo Statistical Methods*, pages 231–283. Springer, 1999.

[16] Gareth O Roberts, Jeffrey S Rosenthal, et al. Optimal scaling for various Metropolis-Hastings algorithms. *Statistical Science*, 16(4):351–367, 2001.

[17] Stuart Russell and Peter Norvig. Artificial intelligence: a modern approach. 2002.

[18] Louis O Scott. Pricing stock options in a jump-diffusion model with stochastic volatility and interest rates: applications of Fourier inversion methods. *Mathematical Finance*, 7(4):413–426, 1997.

[19] Adrian FM Smith and Gareth O Roberts. Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Methodological)*, 55(1):3–23, 1993.

[20] Stephen J Taylor. *Asset price dynamics, volatility, and prediction.* Princeton university press, 2011.

A Derivation of Asset Price Formula

Ito's Lemma

Let X_t be a process that satisfies the following stochastic differential equation

$$dX_t = \mu_t X_t dt + \sigma_t X_t dW_t \ . \tag{6}$$

For a function $f = f(t, X_t)$ the Taylor expansion gives the following

$$df = \frac{\partial f}{\partial t}dt + \frac{\partial f}{\partial X_t}dX_t + \frac{1}{2}\frac{\partial^2 f}{\partial t^2}(dt)^2 + \frac{1}{2}\frac{\partial^2 f}{\partial X_t^2}(dX_t)^2 + \frac{\partial^2 f}{\partial X_t \partial t}(dt\, dX_t) \ ...$$

By substituting X_t with the equation (6), we have

$$\begin{aligned} df =& \frac{\partial f}{\partial t}dt + \frac{\partial f}{\partial X_t}(\mu_t dt + \sigma_t dW_t) + \frac{1}{2}\frac{\partial^2 f}{\partial t^2}(dt)^2 + \\ & \frac{1}{2}\frac{\partial^2 f}{\partial X_t^2}(\mu_t^2 (dt)^2 + \mu_t \sigma_t dt dW_t + \sigma_t^2 (dW_t)^2) + \frac{\partial^2 f}{\partial X_t \partial t}((\mu_t dt + \sigma_t dW_t)dt) ... \ , \end{aligned}$$

where $\langle dt, dt \rangle = 0$, $\langle dt, dW_t \rangle = 0$ and $\langle dW_t, dW_t \rangle = dt$. Thus,

$$df = (\frac{\partial f}{\partial t} + \mu_t \frac{\partial f}{\partial X_t} + \frac{1}{2}\sigma_t^2 \frac{\partial^2 f}{\partial X_t^2})dt + \sigma_t \frac{\partial f}{\partial X_t} dW_t \ . \tag{7}$$

Stock Price Formula

In this thesis, we model the stock price S_t as a geometric Brownian motion:

$$dS_t = \mu_t dt + \sigma_t dW_t \ ,$$

where $\mu_t = \mu S_t$ and $\sigma_t = \sigma S_t$.
Let $f = f(t, S_t) = \ln S_t$, for which

$$\frac{\partial f}{\partial t} = 0, \ \frac{\partial f}{\partial S_t} = \frac{1}{S_t}, \ \frac{\partial^2 f}{\partial S_t^2} = -\frac{1}{S_t^2} \ .$$

Now, by equation (7) we have

$$d(\ln S_t) = (\mu - \frac{1}{2}\sigma)dt + \sigma dW_t$$

By integrating both sides

$$\int_0^t d(\ln S_t) = \int_0^t (\mu - \frac{1}{2}\sigma)dt + \int_0^t \sigma dW_t \ ,$$

we obtain that

$$\ln S_t - \ln S_0 = (\mu - \frac{1}{2}\sigma)t + \sigma(W_t - W_0) \ ,$$

where $W_0 = 0$. By rearranging the equation, the stock price S_t can be expressed as

$$S_t = S_0 e^{(\mu - \frac{1}{2}\sigma)t + \sigma W_t} \ .$$