Estimation of Geometric Brownian Motion Parameters: MLE and Bayesian Methods

Andrea Buffoli, Davide Rolfi

June 16, 2025

1 Introduction

This report presents the estimation of Geometric Brownian Motion (GBM) parameters using real financial data. We use the daily closing prices of Amazon (AMZN) stock, obtained from Yahoo Finance. The log-returns and the descriptive statistics are presented in Figure 1 and Table 1 respectively.

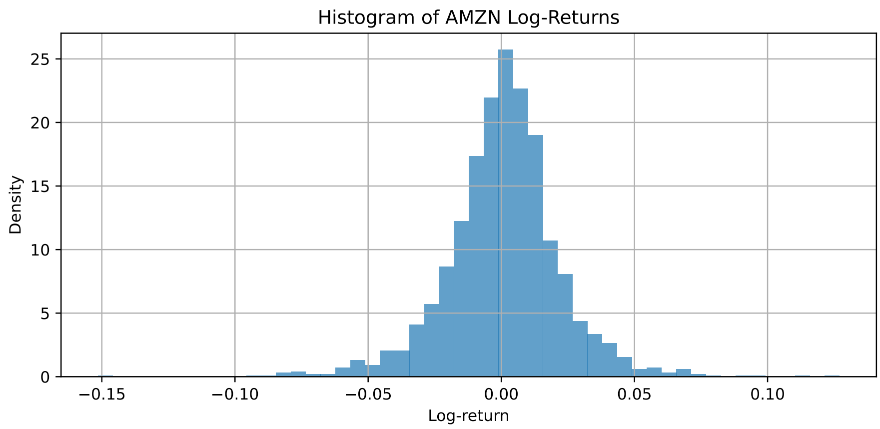

Figure 1: Histogram of daily log-returns for Amazon stock.

```
import yfinance as yf
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm
from google.colab import files
import pandas as pd
# Step 1: Download AMZN data
amzn = yf.download("AMZN", start="2018-01-01", end="2024-12-31")
amzn_close = amzn['Close'].dropna()
# Step 2: Compute log-returns
log_returns = np.log(amzn_close / amzn_close.shift(1)).dropna()
dt = 1 / 250 # Daily time

print(log_returns.describe())

# Step 4: Plot histogram of log-returns
```

```python
plt.figure(figsize=(8, 4))
plt.hist(log_returns, bins=50, density=True, alpha=0.7)
plt.title("Histogram of AMZN Log-Returns")
plt.xlabel("Log-return")
plt.ylabel("Density")
plt.grid(True)
plt.tight_layout()
plt.savefig("amazon_histogram_returns.pdf")
plt.show()
```

Table 1: Descriptive Statistics of Amazon Log-Returns (2018–2025)

Statistic	Value	Description
Count	1759	Number of daily observations
Mean	0.000747	Average daily log-return
Std Dev	0.021686	Standard deviation of daily log-returns
Min	-0.151398	Minimum observed log-return
25%	-0.009930	First quartile (Q1)
50%	0.001185	Median (Q2)
75%	0.012154	Third quartile (Q3)
Max	0.126949	Maximum observed log-return

2 Mathematical Formulation of the GBM

One of the most used models to describe the behavior of financial asset returns overtime is the Geometric Brownian Motion (GBM). It is defined by the following stochastic differential equation:

$$dS(t) = \mu S(t)\,dt + \sigma S(t)\,dW(t), \tag{1}$$

where μ represents the drift coefficient, i.e., the average rate of return of the stock, and σ denotes the volatility coefficient, i.e., the magnitude of the normally distributed shock $dW(t)$ affecting the infinitesimal change in the stock price $S(t)$.

To derive the analytical solution, we apply Itô's lemma to the logarithm of the process $\log S(t)$. This yields the following closed-form solution:

$$\log S(t) = \log S(0) + \left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W(t), \tag{2}$$

or alternatively

$$S\left(t\right) = S\left(0\right)e^{\left(\mu - \frac{1}{2}\sigma^2\right)t + \sigma W(t)} \tag{3}$$

which imply that the log-return over an interval of length Δt follows a normal distribution:

$$r(t) = \log\left(\frac{S(t+\Delta t)}{S(t)}\right) \sim \mathcal{N}\left(\left(\mu - \frac{\sigma^2}{2}\right)\Delta t,\ \sigma^2\Delta t\right). \tag{4}$$

From the analytical solution, we can directly observe that $S(t) > 0$ almost surely for all $t \geq 0$. This means that the asset price modeled by GBM can never become negative or reach zero.

3 Generalized Method of Moments (GMM)

One of the most straightforward approaches to estimate the parameters μ and σ of the Geometric Brownian Motion (GBM) is to approximate the continuous-time dynamics with a Euler–Maruyama discretization (or Euler scheme). In particular, by applying Itô's lemma to the GBM process in equation (1), we obtain the following dynamics for the logarithm of the asset price:

$$dlnS\left(t\right) = \left(\mu - \frac{1}{2}\sigma^2\right) dt + \sigma dW\left(t\right). \tag{5}$$

We can then discretize this process and apply the *Generalized Method of Moments (GMM)*, equating theoretical moments to their empirical counterparts. This leads to the following estimators:

$$\mu = \frac{\mathbb{E}\left(d\ln S(t)\right)}{dt} + \frac{1}{2}\sigma^2, \tag{6}$$

$$\sigma^2 = \frac{\mathbb{V}\left(d\ln S(t)\right)}{dt}, \tag{7}$$

where the expectations refer to the infinitesimal mean and variance of the logarithmic returns. In practice, these are approximated using discretized log-returns over intervals of length Δt, i.e., $r(t) = \log(S(t + \Delta t)/S(t))$. This allows us to estimate the parameters by matching the theoretical and empirical moments.

3.1 Python Code for Generalized Method of Moments with Real Amazon Data

```
import yfinance as yf
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm
from google.colab import files

# Step 1: Download AMZN data
amzn = yf.download("AMZN", start="2018-01-01", end="2024-12-31")
amzn_close = amzn['Close'].dropna()

# Step 2: Compute log-returns
log_returns = np.log(amzn_close / amzn_close.shift(1)).dropna()
dt = 1 / 250  # Daily time step

# Step 3: Moment-based estimation
r_bar = log_returns.mean().item()
r_var = log_returns.var(ddof=1).item()

# Estimators via method of moments
sigma_mom = np.sqrt(r_var / dt)
mu_mom = (r_bar / dt) + 0.5 * sigma_mom**2

print("Method of Moments estimates:")
print(" (mu): {:.4f}".format(mu_mom))
print(" (sigma): {:.4f}".format(sigma_mom))
```

```
# Step 4: Histogram + Normal Fit from GBM (moments)
mu_r = (mu_mom - 0.5 * sigma_mom**2) * dt
sigma_r = sigma_mom * np.sqrt(dt)
x = np.linspace(log_returns.min(), log_returns.max(), 200)

plt.figure(figsize=(8, 4))
plt.hist(log_returns, bins=50, density=True, alpha=0.6, label='Data')
plt.plot(x, norm.pdf(x, loc=mu_r, scale=sigma_r), 'r-', label='GBM Fit (Moments)')
plt.title("Method of Moments Fit of AMZN Log-Returns Using GBM")
plt.xlabel("Log-return")
plt.ylabel("Density")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.savefig("amazon_mom_gbm_fit.pdf")
plt.show()

# Step 5: Download the figure as PDFs
files.download("amazon_mom_gbm_fit.pdf")
```

Table 2: Generalized Method of Moments (GMM) for Amazon Log-Returns

Parameter	**Estimate**	**Description**
μ	0.2456	Drift (expected return per year)
σ	0.3429	Volatility (standard deviation per year)

Table 2 presents the method of moments estimates for the parameters of the Geometric Brownian Motion model, based on Amazon daily log-returns from 2018 to 2025. These values are consistent with the high-growth, high-risk nature of Amazon as a technology-driven company.

4 Maximum Likelihood Estimation (MLE)

We estimate the parameters μ and σ of the Geometric Brownian Motion model using the method of Maximum Likelihood Estimation (MLE), based on the observed log-returns of Amazon stock.

Given a sample of n independent log-returns $\{r(1), r(2), \dots, r(n)\}$, the MLE aims to find the parameter values that maximize the following log-likelihood function:

$$\log L = -\frac{n}{2}\log(2\pi\sigma^2\Delta t) - \frac{1}{2\sigma^2\Delta t}\sum_{i=1}^{n}\left(r(i) - \left(\mu - \frac{1}{2}\sigma^2\right)\Delta t\right)^2. \tag{8}$$

Maximizing this expression with respect to μ and σ^2 leads to the following closed-form estimators:

$$\hat{\sigma}^2 = \frac{1}{\Delta t}\cdot\frac{1}{n}\sum_{i=1}^{n}(r(i) - \bar{r})^2, \tag{9}$$

$$\hat{\mu} = \frac{\bar{r}}{\Delta t} + \frac{1}{2}\hat{\sigma}^2, \tag{10}$$

where $\bar{r} = \frac{1}{n}\sum_{i=1}^{n} r(i)$ is the sample mean of the log-returns. The estimator $\hat{\mu}$ accounts for the fact that the mean of the log-return is not $\mu\Delta t$ but rather $\left(\mu - \frac{1}{2}\sigma^2\right)\Delta t$ due to the Itô correction. This adjustment ensures consistency between the discretized log-return dynamics and the continuous-time GBM model. However, as well as the GMM 3, it provides only point estimates and does not convey information about the uncertainty or confidence intervals around these estimates. This issue can be addressed by Bayesian methods discussed in section 6.

4.1 Python Code for MLE with Real Amazon Data

```
import yfinance as yf
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm
from google.colab import files

# Step 1: Download AMZN data
amzn = yf.download("AMZN", start="2018-01-01", end="2024-12-31")
amzn_close = amzn['Close'].dropna()

# Step 2: Compute log-returns
log_returns = np.log(amzn_close / amzn_close.shift(1)).dropna()
dt = 1 / 250  # daily step

# Step 3: MLE estimation
log_returns_mean = log_returns.mean().item()
log_returns_var = log_returns.var(ddof=1).item()
mu_mle = (log_returns_mean + 0.5 * log_returns_var) / dt
sigma_mle = np.sqrt(log_returns_var / dt)

print("MLE mu: {:.4f}, sigma: {:.4f}".format(mu_mle, sigma_mle))

# Step 4: Plot histogram + GBM normal fit
mu_r = (mu_mle - 0.5 * sigma_mle**2) * dt
sigma_r = sigma_mle * np.sqrt(dt)
x = np.linspace(log_returns.min(), log_returns.max(), 200)

plt.figure(figsize=(8, 4))
plt.hist(log_returns, bins=50, density=True, alpha=0.6, label='Data')
plt.plot(x, norm.pdf(x, loc=mu_r, scale=sigma_r), 'r-', label='MLE GBM Fit')
plt.title("MLE Fit of AMZN Log-Returns Using GBM Parameters")
plt.xlabel("Log-return")
plt.ylabel("Density")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.savefig("amazon_mle_gbm_fit.pdf")
plt.show()

# Step 5: Download the PDFs
files.download("amazon_mle_gbm_fit.pdf")
```

Table 3: Maximum Likelihood Estimates (MLE) for Amazon Log-Returns

Parameter	Estimate	Description
μ	0.2456	Drift (expected return per year)
σ	0.3429	Volatility (standard deviation per year)

Table 3 presents the MLE estimates for the parameters of the Geometric Brownian Motion model, based on Amazon daily log-returns from 2018 to 2025. These values are the same of the ones estimated with the GMM method (see Table 2).

5 Empirical application of GMM and MLE

```
import yfinance as yf
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm
from google.colab import files
# Step 1: Download AMZN data
amzn = yf.download("AMZN", start="2018-01-01", end="2024-12-31")
amzn_close = amzn['Close'].dropna()
# Step 2: Compute log-returns
log_returns = np.log(amzn_close / amzn_close.shift(1)).dropna()
dt = 1 / 250 # Daily time step

# Step 3: Moment-based estimation
r_bar = log_returns.mean().item()
r_var = log_returns.var(ddof=1).item()
# Estimators via method of moments
sigma_mom = np.sqrt(r_var / dt)
mu_mom = (r_bar / dt) + 0.5 * sigma_mom**2
print("Method of Moments estimates:")
print(" (mu): {:.4f}".format(mu_mom))
print(" (sigma): {:.4f}".format(sigma_mom))


# Step 4: Histogram + Normal Fit from GBM (moments)
sigma_r = sigma_mom / np.sqrt(dt)
mu_r = (mu_mom - 0.5 * sigma_mom**2)
x = np.linspace(log_returns.min()/dt, log_returns.max()/dt, 200)


plt.figure(figsize=(8, 4))
plt.hist(log_returns/dt, bins=50, density=True, alpha=0.6, label='Data')
plt.plot(x, norm.pdf(x, loc=mu_r, scale=sigma_r), 'r-', label='GBM Fit (Moments)')
plt.title("GMM/MLE estimation of AMZN Annualized Log-Returns Using GBM")
plt.xlabel("Annualized Log-return")
plt.ylabel("Density")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.savefig("both_fit.pdf")
plt.show()
# Step 5: Download both figures as PDFs
#files.download("amazon_histogram_returns_mom.pdf")
```

```
files.download("both_fit.pdf")
```

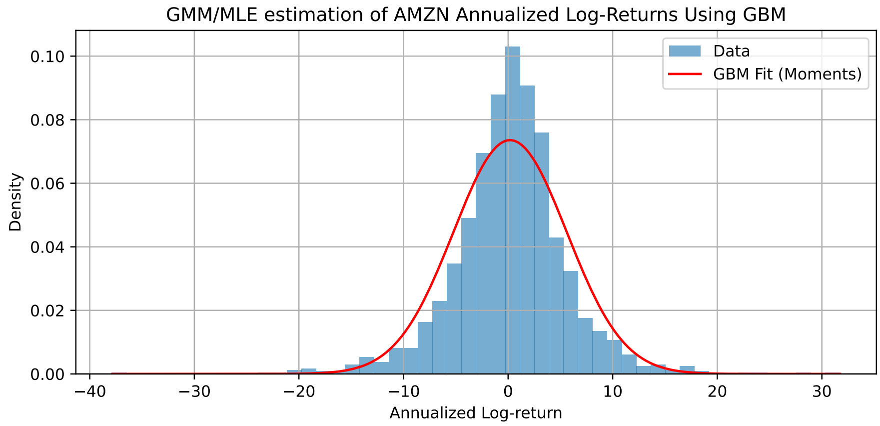

Figure 2: GMM and MLE normal fit to Annualized log-returns of Amazon stock.

The histogram 2 reveals a significant mismatch between empirical data and the assumptions underlying the Geometric Brownian Motion (GBM) model. While the normal distribution provides a reasonable fit near the center, it fails to capture the heavy tails and asymmetry present in the observed data. One can test the goodness of fit using different methods such as the Kolmogorov-Smirnov test or a qq-pp-plot.

In particular, financial returns are well-known to exhibit leptokurtosis, distributions with fat tails and sharp peaks relative to the normal distribution. This implies that extreme returns, both positive and negative, occur more frequently than predicted by Gaussian models. As a consequence, the GBM-based assumption underestimates the probability of large fluctuations, making it unreliable for modeling risk, pricing options, or estimating Value-at-Risk (VaR).

Moreover, empirical return distributions are often negatively skewed, meaning they exhibit a longer left tail. This skewness indicates that large losses are more frequent than large gains—another feature the GBM model fails to incorporate due to its inherent symmetry. This mischaracterization is especially dangerous in risk management, where the underestimation of downside risk can lead to inadequate capital buffers and poor hedging strategies.

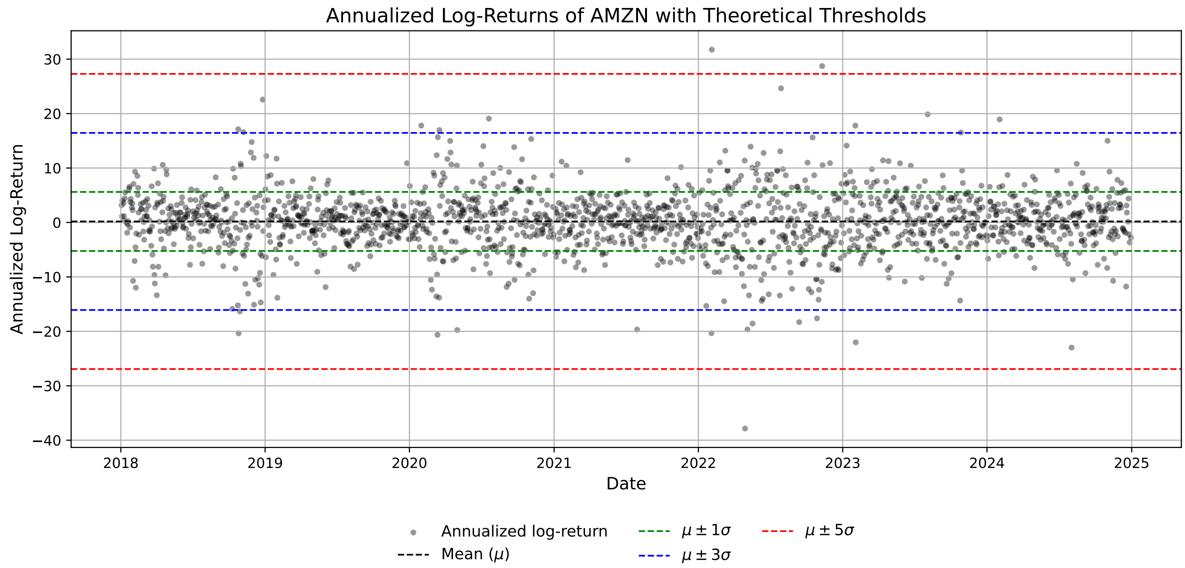

Figure 3: Scatter plot of annualized log-returns for Amazon (AMZN) from 2018 to 2025, overlaid with theoretical thresholds derived from the Geometric Brownian Motion (GBM) model. The dashed black line represents the estimated mean (μ). Colored dashed lines indicate symmetric thresholds at $\mu \pm 1\sigma$ (green), $\mu \pm 3\sigma$ (blue), and $\mu \pm 5\sigma$ (red). While most observations fall within $\pm 3\sigma$, a non-negligible number of extreme returns exceed this range, suggesting the presence of heavy tails in the empirical distribution.

Together, leptokurtosis and negative skewness contribute to substantial tail risk, the risk of rare but severe market movements. Since the GBM framework assumes normality, it effectively assigns negligible probability to these events, which in reality occur with non-trivial frequency. This leads to systemic underestimation of potential losses, particularly during periods of market stress.

In conclusion, while GBM remains a useful theoretical model for pedagogical purposes, it is not well-suited for capturing the empirical properties of financial returns. The observed deviation from normality, as highlighted by the MLE fit, calls for more sophisticated modeling approaches—such as jump-diffusion processes, stochastic volatility models, or distributions with heavier tails.

6 Bayesian Estimation

We now estimate the parameters μ and σ of the Geometric Brownian Motion model using a Bayesian approach. Unlike frequentist methods such as GMM or MLE, which provide point estimates based solely on observed data, the Bayesian framework treats model parameters as random variables and produces a full posterior distribution that reflects both prior beliefs and the likelihood of the observed data.

This approach is grounded in Bayes' theorem:

$$P(\theta \mid \text{data}) = \frac{P(\text{data} \mid \theta) \cdot P(\theta)}{P(\text{data})},$$

where θ represents the parameters of interest (here, μ and σ), $P(\theta)$ is the prior distribution encoding our beliefs before seeing the data, $P(\text{data} \mid \theta)$ is the likelihood, and $P(\theta \mid \text{data})$

is the posterior distribution that combines prior information with the evidence from the data.

Bayesian estimation is particularly useful when uncertainty quantification is important. Rather than producing single-point estimates, we obtain entire distributions for μ and σ, from which credible intervals and other uncertainty metrics can be derived. This is essential in financial contexts, where uncertainty is inherent and decision-making often depends on risk assessments.

In our case, we adopt a normal prior for μ, reflecting symmetry around zero with moderate spread, and a half-normal prior for σ, ensuring positivity while encoding reasonable bounds for volatility. The posterior distributions are obtained using modern Markov Chain Monte Carlo (MCMC) methods, which allow sampling from complex, high-dimensional distributions even when the normalizing constant $P(\text{data})$ is analytically intractable.

Compared to GMM and MLE, the Bayesian framework requires more computational effort but provides a richer and more flexible description of parameter uncertainty. This makes it especially appealing in scenarios with limited data, strong prior knowledge, or the need to propagate uncertainty in subsequent financial models or decision processes.

We assume a normal prior for the drift parameter μ, and a half-normal prior for the volatility parameter σ:

$$\begin{aligned}\mu &\sim \mathcal{N}(0, 1),\\ \sigma &\sim \text{HalfNormal}(0.2).\end{aligned}$$

The prior on μ encodes the belief that the expected return is centered around zero, allowing for significant deviations in both directions. This reflects an agnostic stance in the absence of strong prior knowledge, while still preventing extreme drift values.

The prior on σ enforces the natural positivity constraint on volatility, while expressing the belief that values above 100% annualized volatility are unlikely. The scale parameter of 0.2 is consistent with typical levels of volatility observed in equity markets.

Both priors are considered weakly informative: they regularize the inference process, help avoid unrealistic parameter estimates, and improve the stability of Markov Chain Monte Carlo sampling, while still allowing the observed data to dominate the posterior.

6.1 Python Code using PyMC

```
import yfinance as yf
import numpy as np
import matplotlib.pyplot as plt
import pymc as pm
import arviz as az

# Download AMZN price data
amzn = yf.download("AMZN", start="2018-01-01", end="2024-12-31")
amzn_close = amzn['Close'].dropna()

# Compute log-returns
log_returns = np.log(amzn_close / amzn_close.shift(1)).dropna().values
dt = 1 / 250  # daily time step

# Bayesian estimation using PyMC
with pm.Model() as model:
```

```
    mu = pm.Normal("mu", mu=0, sigma=1)
    sigma = pm.HalfNormal("sigma", sigma=0.2)
    likelihood = pm.Normal("r", mu=(mu - 0.5 * sigma**2) * dt,
                           sigma=sigma * np.sqrt(dt),
                           observed=log_returns)
    trace = pm.sample(2000, tune=1000, target_accept=0.95, random_seed=42)

# Plot posterior distributions
az.plot_posterior(trace, var_names=["mu", "sigma"], hdi_prob=0.95)
plt.tight_layout()
plt.savefig("amazon_bayes_posteriors.pdf")
plt.close()

# Download the PDF (if in Colab)
from google.colab import files
files.download("amazon_bayes_posteriors.pdf")
```

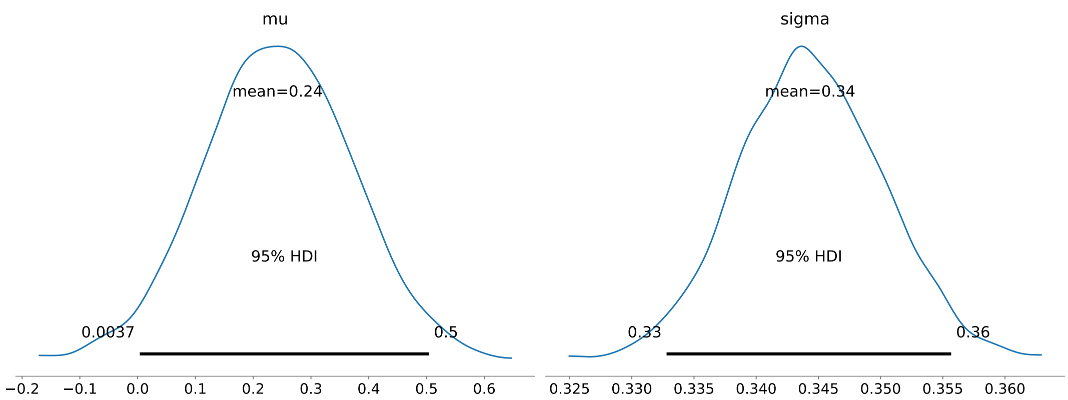

Figure 4: Posterior distributions of μ and σ for the GBM model (Bayesian estimation).

Figure 4 presents the posterior distributions of parameters for the Geometric Brownian Motion model, obtained through Bayesian inference using MCMC sampling.

The posterior for μ is centered around approximately 0.24, with a 95% highest density interval (HDI) that reflects uncertainty around this estimate. This suggests that, based on the observed log-returns and the chosen prior, the expected annualized return of Amazon stock is likely to lie within a plausible and data-informed range. Notably, the posterior is narrower than the prior, indicating that the data have significantly updated our belief about μ.

The posterior for σ is tightly concentrated around 0.34, again with a relatively narrow 95% HDI. This reflects high confidence in the estimated level of volatility. The posterior is shaped by both the data and the HalfNormal(0.2) prior, but the outcome shows that the observed returns strongly dominate the inference, resulting in a well-defined estimate.

Together, these posterior distributions illustrate the main advantage of the Bayesian approach: rather than providing point estimates alone (as in MLE), Bayesian inference yields full probability distributions for each parameter. This allows for more nuanced conclusions, incorporating parameter uncertainty directly into further decision-making or predictive modeling.

7 Conclusion

In this work, we estimated the parameters of a Geometric Brownian Motion (GBM) model using three distinct inferential approaches: Generalized Method of Moments (GMM), Maximum Likelihood Estimation (MLE) and Bayesian inference.

The GMM and MLE methods provide point estimates for the drift μ and volatility σ by maximizing the likelihood of the observed log-returns under the assumption of normality. However, it offers no direct information about parameter uncertainty and relies heavily on the assumption that the empirical data follow a Gaussian distribution.

The Bayesian approach, on the other hand, treats the parameters as random variables and combines prior beliefs with observed data through Bayes' theorem to generate posterior distributions. This allows for a more nuanced understanding of uncertainty and facilitates probabilistic reasoning about parameters. Nonetheless, it does not change the underlying modeling assumptions of GBM, it still assumes log-returns are normally distributed—and is therefore subject to the same structural limitations when the data deviate from Gaussianity.

Our empirical analysis using Amazon (AMZN) stock data reveals clear departures from normality in the log-return distribution. Specifically, we observe heavy tails (leptokurtosis) and negative skewness—both features that the GBM framework fails to capture. These distributional characteristics imply the presence of significant tail risk, meaning that extreme losses occur more frequently than predicted by the GBM model.

In conclusion, GMM, MLE and Bayesian methods offer valid tools for parameter estimation under Gaussian assumptions, the GBM model itself is not adequate for modeling real-world financial returns. Its inability to capture fat tails and skewness leads to a systematic underestimation of risk. For robust modeling in finance, more flexible frameworks that allow for jumps, stochastic volatility, or non-Gaussian innovations should be considered.

References

[1] Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3):637–654, 1973.

[2] Kevin Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2022.

[3] John Salvatier, Thomas V. Wiecki, and Christopher Fonnesbeck. Probabilistic programming in Python using PyMC. *PeerJ Computer Science*, 2:e55, 2016. `https://doi.org/10.7717/peerj-cs.55`

[4] Ravin Kumar, Colin Carroll, Osvaldo Martin, et al. ArviZ: A unified library for exploratory analysis of Bayesian models in Python. *Journal of Open Source Software*, 4(33):1143, 2019. `https://doi.org/10.21105/joss.01143`

[5] Matthew D. Hoffman and Andrew Gelman. The No-U-Turn Sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014.