Sensitivities in Finance

Stefano Ferioli, Gianfranco Arrighetti, Tommaso Fioratti

Stochastic Simulation - Fall semester 2024-2025

Introduction

In this project, we explore the application of Monte Carlo methods in financial computations to estimate the derivatives of quantities of interest. These derivatives are evaluated with respect to key model parameters such as volatility (σ), initial asset price (S_0), and interest rate (r). Accurately estimating these derivatives is fundamental for tasks such as assessing risk, pricing financial instruments, and designing effective hedging strategies.

We consider the following Stochastic Differential Equation,

$$\mathrm{d}S_t = rS_t\mathrm{d}t + \sigma S_t \mathrm{d}W_t,$$

for $t > 0$ and S_0 given, where W_t is a Brownian motion. One can show that the solution at time $t = T$ is a geometric Brownian motion that can be expressed as

$$S_T = S_0 \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)T + \sigma W_T\Big\}.$$

Given $f : \mathbb{R} \to \mathbb{R}$, we are interested in estimating the derivatives of

$$I = \mathbb{E}[f(S_T)] = \int_{\mathbb{R}} f\Big(S_0 \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)T + \sigma W_T\Big\}\Big) p_{W_T}(w)\mathrm{d}w = \int_{\mathbb{R}} f(s) p_{S_T}(s)\mathrm{d}s$$

where $p_{W_T}(w) = \frac{1}{\sqrt{2\pi T}} \exp\Big(-\frac{w^2}{2T}\Big)$ and $p_{S_T}(s) = \frac{1}{s\sigma\sqrt{2\pi T}} \exp\Big[-\frac{1}{2}\Big(\frac{\log(s/S_0) - \big(r - \frac{1}{2}\sigma^2\big)T}{\sigma\sqrt{T}}\Big)^2\Big]$.

Clearly the parameters S_0, σ, and r do, in general, affect the outcome of S_T and therefore of $I = \mathbb{E}[f(S_T)]$. The impact of a parameter change on I can be measured by the derivative of I with respect to the same parameter.

We present three techniques to estimate these derivatives.

First, the most straightforward approach to estimate the derivatives of I with respect to some parameter θ is to apply finite differences, that is to compute

$$\frac{\partial I}{\partial \theta} \approx \frac{I(\theta + h) - I(\theta - h)}{2h}$$

and

$$\frac{\partial^2 I}{\partial \theta^2} \approx \frac{I(\theta + h) - 2I(\theta) + I(\theta - h)}{h^2}.$$

where $I(\theta + h)$, $I(\theta)$, and $I(\theta - h)$ are to be estimated using Monte Carlo.

Second, assuming derivative and integral are interchangeable, which is not always true, we can apply a pathwise derivative, that is to compute

$$\frac{\partial}{\partial \theta}\mathbb{E}[f(S_T)] = \mathbb{E}\Big[\frac{\partial}{\partial \theta} f(S_T)\Big] = \mathbb{E}\Big[f'(S_T)\frac{\partial S_T}{\partial \theta}\Big] \tag{1}$$

and

$$\frac{\partial^2}{\partial\theta^2}\mathbb{E}[f(S_T)] = \mathbb{E}\Big[f''(S_T)\frac{\partial S_T}{\partial\theta} + f'(S_T)\frac{\partial^2 S_T}{\partial\theta^2}\Big],$$

where the expectations are to be calculated using Monte Carlo.

Last, assuming again derivative and integral are interchangeable, we can apply the likelihood ratio method, that is

$$\begin{aligned}\frac{\partial}{\partial\theta}\mathbb{E}[f(S_T)] &= \frac{\partial}{\partial\theta}\int_0^\infty f(s)p_{S_T}(s)\mathrm{d}s = \int_0^\infty f(s)\frac{\partial p_{S_T}}{\partial\theta}(s)\mathrm{d}s \\ &= \mathbb{E}\Big[f(S_T)\frac{\frac{\partial p_{S_T}}{\partial\theta}(S_T)}{p_{S_T}(S_T)}\Big] = \mathbb{E}\Big[f(S_T)\frac{\partial}{\partial\theta}\log p_{S_T}(S_T)\Big]\end{aligned} \tag{2}$$

and

$$\begin{aligned}\frac{\partial^2}{\partial\theta^2}\mathbb{E}[f(S_T)] &= \frac{\partial^2}{\partial\theta^2}\int_0^\infty f(s)p_{S_T}(s)\mathrm{d}s = \int_0^\infty f(s)\frac{\partial^2 p_{S_T}(s)}{\partial\theta^2}\mathrm{d}s = \\ = \mathbb{E}\Big[f(S_T)\frac{\frac{\partial^2 p_{S_T}}{\partial\theta^2}(S_T)}{p_{S_T}(S_T)}\Big] &= \mathbb{E}\Big[f(S_T)\Big(\frac{\partial^2}{\partial\theta^2}\log p_{S_T}(S_T) + \Big(\frac{\partial}{\partial\theta}\log p_{S_T}(S_T)\Big)^2\Big)\Big],\end{aligned}$$

where the expectations are to be calculated using Monte Carlo.

Fixed $f : \mathbb{R} \to \mathbb{R}$, we are interested in estimating the following quantities, known as "the Greeks" in finance.

Name	**Symbol**	**Definition**
Delta	δ	$\partial\,\mathbb{E}[f(S_T)]/\partial S_0$
Vega	ν	$\partial\,\mathbb{E}[f(S_T)]/\partial\sigma$
Gamma	γ	$\partial^2\,\mathbb{E}[f(S_T)]/\partial S_0^2$

1 European call option

An European call option is a financial derivative that gives the holder the right, but not the obligation, to buy an underlying asset at a predetermined strike price K on a specific maturity date T. The payoff of a European call option at maturity is defined as

$$f(S_T) = e^{-rT}[S_T - K]^+$$

where $[x]^+ = \max\{x, 0\}$. Here we take interest rate $r = 0.05$, maturity time $T = 1$, volatility $\sigma = 0.25$, initial asset price $S_0 = 100$, strike price $K = 120$.

Estimate Delta and Vega

We start by estimating δ and ν using finite differences, pathwise derivative and likelihood ratio. The estimates are plotted along with an asymptotic confidence interval as a function of the sample size $N \in [10^3, 10^6]$. For now, for the finite differences method, we choose $h = 10^{-5}$.

The confidence interval at the 0.95 level for the Monte Carlo estimator is obtained as

$$I_N = \Big[\hat{\mu}_N - c_{0.975}\frac{\hat{S}_N}{\sqrt{N}},\ \hat{\mu}_N + c_{0.975}\frac{\hat{S}_N}{\sqrt{N}}\Big],$$

where $\hat{\mu}_N$ is the sample mean (and the Monte Carlo estimator), $\hat{S}_N$ is the sample standard deviation, and $c_{0.975}$ is the 97.5% quantile of the standard normal distribution. Note that, given that N is at least 10^3, $\hat{\mu}_N$ can be very well approximated by a normal distribution and so we can use the asymptotic confidence interval with very little error.

Figure 1 displays the estimates of δ and ν obtained by the three methods, together with the asymptotic confidence intervals for different sample sizes N on a logarithmic scale.

In addition, we recall that the payoff of an European call option under the Black–Scholes model is known analytically and equal to

$$\mathbb{E}[e^{-rT}[S_T - K]^+] = \Phi(d_+)S_0 - \Phi(d_-)Ke^{-rT} \quad \text{with} \tag{3}$$
$$d_+ = \frac{1}{\sigma\sqrt{T}}\left[\ln\left(\frac{S_0}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)T\right] \quad \text{and} \quad d_- = d_+ - \sigma\sqrt{T},$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. This closed-form expression allows us to compute the analytical values of δ, ν, and γ, and plot them alongside the numerical estimates.

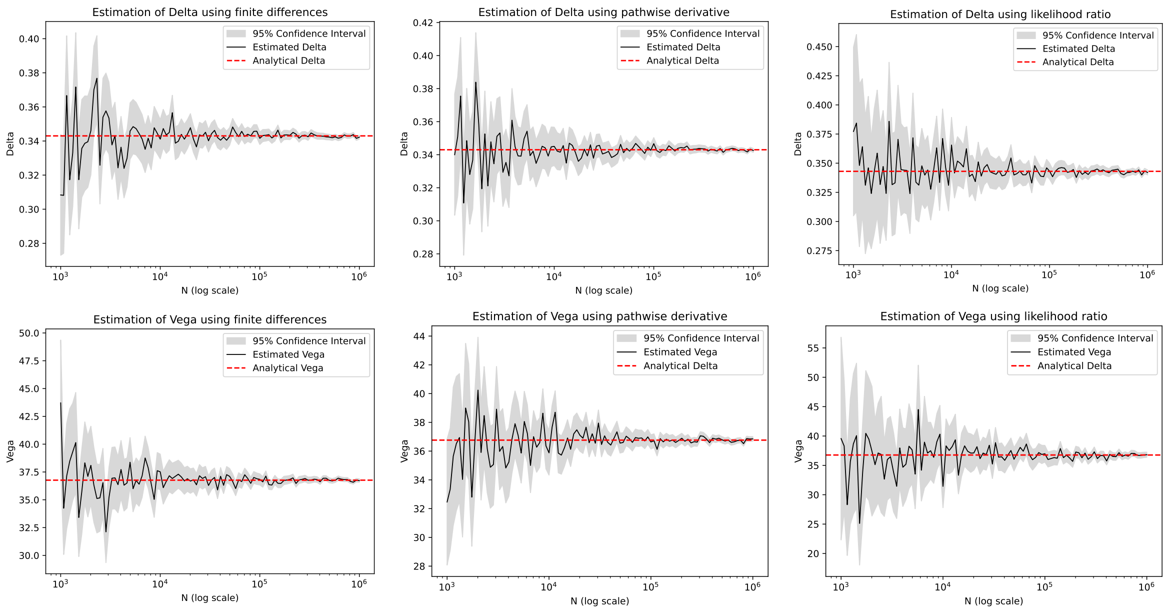

Figure 1

The three methods provide good approximations of δ and ν and (as expected) display the same asymptotic behavior. In order to have a better understanding of the performance of the estimators, in Figure 2, we compare the standard deviations of the estimators both when estimating δ and ν. We computed the standard deviations using a sample size of $N = 10^6$. The finite differences and pathwise derivative methods present comparable standard deviations in both cases, on the other hand, the likelihood ratio method displays substantially higher standard deviation.

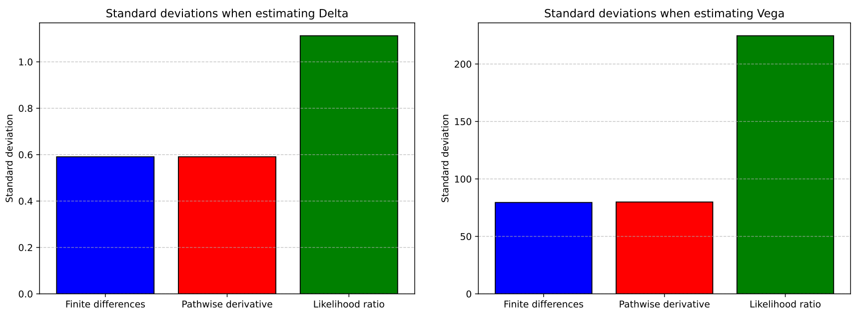

Figure 2

In addition, for the finite differences method, we investigate how the value of h affects the estimates of δ and ν. To do this, we fix $N = 10^6$ and, in Figure 3, we plot the estimates of the two parameters as a function of $h \in [10^{-8}, 10^{-0.5}]$.

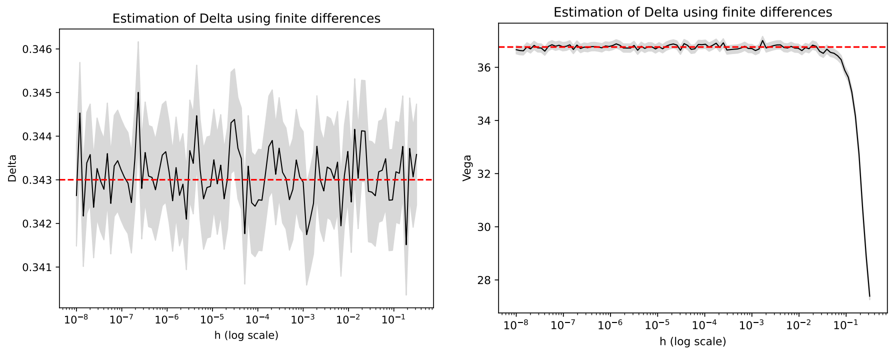

Figure 3

The value of h doesn't seem to impact much the result of the estimates. This is probably due to the fact that $\mathbb{E}[e^{-rT}[S_T - K]^+]$ is almost linear in S_0 and σ, respectively, in a neighborhood of the considered base values, and so the bias of the considered estimators is very close to 0. For δ, this is true provided that $h < 0.01$.

Finally, we consider the case in which we use different iid samples to estimate $I(\theta + \Delta\theta)$ and $I(\theta - \Delta\theta)$. Take $\theta = \sigma$ and define $\tilde{f}(\sigma, w) = f(S_0 \exp\{(r - \frac{1}{2}\sigma^2)T + \sigma w\})$, so that $I(\sigma) = \mathbb{E}[\tilde{f}(\sigma, W_T)]$, where $W_T \sim \mathcal{N}(0, T)$. Then, the two Monte Carlo estimators are

$$\hat{\mu}_N = \frac{1}{2Nh} \sum_{i=1}^{N} (\tilde{f}(\sigma + h, W_T^i) - \tilde{f}(\sigma - h, W_T^i))$$

and

$$\hat{\mu}_N^\star = \frac{1}{2Nh} \sum_{i=1}^{N} (\tilde{f}(\sigma + h, W_T^i) - \tilde{f}(\sigma - h, \tilde{W}_T^i))$$

where N is the sample size and $W_T^1, W_T^2, ..., W_T^N, \tilde{W}_T^1, \tilde{W}_T^2, ..., \tilde{W}_T^N \overset{\text{iid}}{\sim} \mathcal{N}(0, T)$.

Since the two estimators have the same expected value, the best one is the one with lower variance. We will show that the best estimator is $\hat{\mu}_N$.

First, we have

$$\mathbb{V}\mathrm{ar}(\hat{\mu}_N) = \frac{1}{4Nh^2} \mathbb{V}\mathrm{ar}(\tilde{f}(\sigma + h, W_T^1) - \tilde{f}(\sigma - h, W_T^1))$$

and

$$\mathbb{V}\mathrm{ar}(\hat{\mu}_N^\star) = \frac{1}{4Nh^2} \mathbb{V}\mathrm{ar}\big(\tilde{f}(\sigma + h, W_T^1) - \tilde{f}(\sigma - h, \tilde{W}_T^1)\big).$$

Since $\tilde{f}(\cdot, W_T^1)$ is almost surely twice differentiable in σ, we can write the first and second order Taylor expansions,

$$\begin{aligned}
&\mathbb{V}\mathrm{ar}(\tilde{f}(\sigma+h, W_T^1) - \tilde{f}(\sigma-h, W_T^1)) \\
&= \mathbb{V}\mathrm{ar}\Big(\big(\tilde{f}(\sigma, W_T^1) + h\tilde{f}'(\sigma, W_T^1) + \frac{h^2}{2}\tilde{f}''(\sigma, W_T^1) + \mathcal{O}(h^3)\big) \\
&\quad -\big(\tilde{f}(\sigma, W_T^1) - h\tilde{f}'(\sigma, W_T^1) + \frac{h^2}{2}\tilde{f}''(\sigma, W_T^1) + \mathcal{O}(h^3)\big)\Big) \\
&= \mathbb{V}\mathrm{ar}(2h\tilde{f}'(\sigma, W_T^1) + \mathcal{O}(h^3)) \\
&= \mathbb{V}\mathrm{ar}(2h\tilde{f}'(\sigma, W_T^1)) + \mathbb{V}\mathrm{ar}(\mathcal{O}(h^3)) + 2\mathbb{C}\mathrm{ov}(2h\tilde{f}'(\sigma, W_T^1),\, \mathcal{O}(h^3)) \\
&= 4h^2\mathbb{V}\mathrm{ar}(\tilde{f}'(\sigma, W_T^1)) + h^6\mathbb{V}\mathrm{ar}(\mathcal{O}(1)) + h^4\mathbb{C}\mathrm{ov}(\tilde{f}'(\sigma, W_T^1),\, \mathcal{O}(1)) \\
&= \mathcal{O}(h^2)
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{V}\mathrm{ar}\big(\tilde{f}(\sigma+h, W_T^1) - \tilde{f}(\sigma-h, \tilde{W}_T^1)\big) &\overset{\perp\!\!\!\perp}{=} \mathbb{V}\mathrm{ar}\big(\tilde{f}(\sigma+h, W_T^1)\big) + \mathbb{V}\mathrm{ar}\big(\tilde{f}(\sigma-h, \tilde{W}_T^1)\big) \\
&\overset{\text{iid}}{=} \mathbb{V}\mathrm{ar}\big(\tilde{f}(\sigma+h, W_T^1)\big) + \mathbb{V}\mathrm{ar}\big(\tilde{f}(\sigma-h, W_T^1)\big) \\
&= \mathbb{V}\mathrm{ar}\big(\tilde{f}(\sigma, W_T^1) + h\tilde{f}'(\sigma, W_T^1) + \mathcal{O}(h^2)\big) + \mathbb{V}\mathrm{ar}\big(\tilde{f}(\sigma, W_T^1) - h\tilde{f}'(\sigma, W_T^1) + \mathcal{O}(h^2)\big) \\
&= 2\mathbb{V}\mathrm{ar}\big(\tilde{f}(\sigma, W_T^1)\big) + 2h^2\mathbb{V}\mathrm{ar}\big(\tilde{f}'(\sigma, W_T^1)\big) + h^4\mathbb{V}\mathrm{ar}\big(\mathcal{O}(1)\big) \\
&\quad + h^2\mathbb{C}\mathrm{ov}\big(\tilde{f}(\sigma, W_T^1), \mathcal{O}(1)\big) + h^3\mathbb{C}\mathrm{ov}\big(\tilde{f}'(\sigma, W_T^1), \mathcal{O}(1)\big) \\
&= 2\mathbb{V}\mathrm{ar}\big(\tilde{f}(\sigma, W_T^1)\big) + \mathcal{O}(h^2).
\end{aligned}$$

This implies that

$$\mathbb{V}\mathrm{ar}(\hat{\mu}_N) = \mathcal{O}\Big(\frac{1}{N}\Big) \quad \text{and} \quad \mathbb{V}\mathrm{ar}(\hat{\mu}_N^\star) = \mathcal{O}\Big(\frac{1}{Nh^2}\Big),$$

showing that $\hat{\mu}_N$ has lower variance than $\hat{\mu}_N^\star$ asymptotically with respect to h and for h small enough.

Similarly, we can prove the same in the case where $\theta = S_0$.

Estimate Gamma

We continue by estimating $\gamma = \partial^2\,\mathbb{E}[f(S_T)]/\partial S_0^2$. Since there are two derivatives to compute, we potentially have four ways of doing it, by applying all possible combinations of pathwise derivative and likelihood ratio method. Our possibilities are:

- Use pathwise derivative twice (PW-PW),
- Use pathwise derivative first and likelihood ratio method second (PW-LR),
- Use pathwise derivative first and likelihood ratio method second (LR-PW),
- Use likelihood ratio method twice (LR-LR).

The goal is to determine which methods are not applicable and to verify the others through numerical experiments.

The only method that is not applicable in this context is PW-PW. In particular, we show that after computing the first derivative with a pathwise derivative, we are unable to compute another derivative using the same method.

Indeed, applying pathwise derivative method twice, we obtain the following formula,

$$\frac{\partial^2}{\partial S_0^2}\mathbb{E}[f(S_T)] \overset{\text{PW}}{=} \frac{\partial}{\partial S_0}\mathbb{E}\Big[f'(S_T)\frac{\partial S_T}{\partial S_0}\Big] \overset{\text{PW}}{=} \mathbb{E}\Big[f''(S_T)\frac{\partial S_T}{\partial S_0} + f'(S_T)\frac{\partial^2 S_T}{\partial S_0^2}\Big].$$

However, the second derivative of f is problematic since

$$\frac{\mathrm{d}^2}{\mathrm{d}s^2}f(s) = \frac{\mathrm{d}^2}{\mathrm{d}s^2}e^{-rT}[S_T - K]^+ = e^{-rT}\frac{\mathrm{d}}{\mathrm{d}s}\mathbf{I}\{S_T > K\},$$

which is the derivative of a (discontinuous) indicator function and thus it is not well-defined in the classical sense. As a result, in this scenario, it is not possiple to apply the pathwise derivative method twice to estimate γ.

On the other hand, in Appendix A, we formally prove that PW-LR, LR-PW, LR-LR are indeed applicable.

Figure 4 displays the estimates of γ obtained by the other three methods, together with the asymptotic confidence intervals for different sample sizes N on a logarithmic scale. In addition, using again Formula (3), we plot the analytical value of γ alongside the numerical estimates.

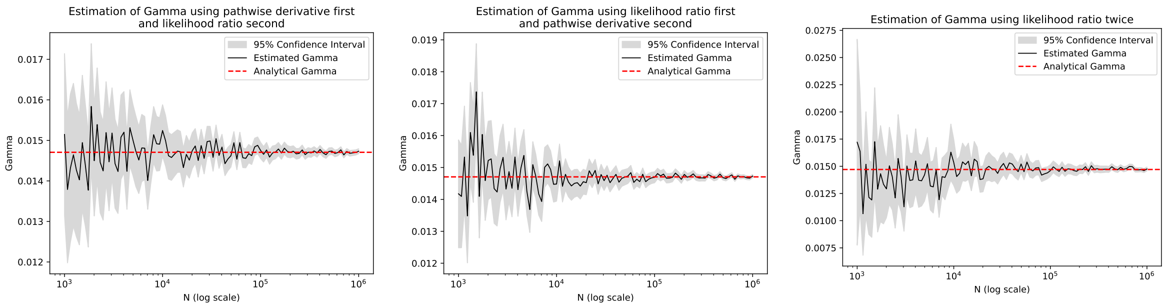

Figure 4

The three methods provide good approximations of γ and (as expected) display the same asymptotic behavior.

In order to have a better understanding of the performance of the estimators, in Figure 5, we compare the standard deviations of the three estimators. We computed the standard deviations using a sample size of $N = 10^6$. The mixed estimators present comparable standard deviations, on the other hand, the pure likelihood ratio method estimator displays substantially higher standard deviation.

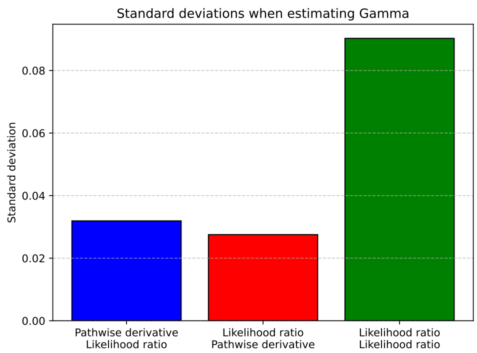

Figure 5

2 Digital call option

A Digital call option is a financial derivative that pays a fixed amount of money if the price of the underlying asset is above a predetermined strike price at expiration. Unlike the European call option, which provides a payoff proportional to the amount by which the underlying asset exceeds the strike price, a digital call option offers an all-or-nothing payout structure. The payoff of a Digital call option at maturity is defined as

$$f(S_T) = e^{-rT}\mathbf{I}\{S_T > K\}.$$

Here we take interest rate $r = 0.05$, maturity time $T = 0.25$, volatility $\sigma = 0.25$, initial asset price $S_0 = 100$, strike price $K = 100$.

In order to obtain an estimator for δ, we write

$$\mathbf{I}\{x > K\} = f_\varepsilon(x) + (\mathbf{I}\{x > K\} - f_\varepsilon(x)) = f_\varepsilon(x) + h_\varepsilon(x)$$

where $f_\varepsilon(x)$ is a continuous approximation of $\mathbf{I}\{x > K\}$ defined as $f_\varepsilon(x) = \min\{1, \max\{0, \frac{x-K+\varepsilon}{2\varepsilon}\}\}$. Then, we have

$$\frac{\partial}{\partial S_0}\mathbb{E}[e^{-rT}\mathbf{I}\{S_T > K\}] = e^{-rT}\frac{\partial}{\partial S_0}\mathbb{E}[f_\varepsilon(S_T)] + e^{-rT}\frac{\partial}{\partial S_0}\mathbb{E}[h_\varepsilon(S_T)],$$

so that we can apply pathwise differentiation for $\frac{\partial}{\partial S_0}\mathbb{E}[f_\varepsilon(S_T)]$ and the likelihood ratio method for $\frac{\partial}{\partial S_0}\mathbb{E}[h_\varepsilon(S_T)]$.

Given the sample size N, we want to split it optimally between the two parts of the estimator. That is, we want to find $\alpha \in (0, 1)$, that minimizes the variance of

$$\hat{\mu}_N = \frac{1}{\lfloor \alpha N \rfloor}\sum_{i=1}^{\lfloor \alpha N \rfloor} f'_\varepsilon(S_T)\frac{\partial S_T}{\partial S_0} + \frac{1}{N - \lfloor \alpha N \rfloor}\sum_{i=\lfloor \alpha N \rfloor+1}^{N} h_\varepsilon(S_T)\frac{\partial}{\partial S_0}\log p_{S_T}(S_T)$$

where $S_T^1, S_T^2, ..., S_T^N \overset{\text{iid}}{\sim} S_T$.

Let $\sigma_1^2 = \mathbb{V}\mathrm{ar}\big(f'_\varepsilon(S_T)\frac{\partial S_T}{\partial S_0}\big)$ and $\sigma_2^2 = \mathbb{V}\mathrm{ar}\big(h_\varepsilon(S_T)\frac{\partial}{\partial S_0}\log p_{S_T}(S_T)\big)$, then $\mathbb{V}\mathrm{ar}(\hat{\mu}_N) = \frac{\sigma_1^2}{\alpha N} + \frac{\sigma_2^2}{(1-\alpha)N}$, which is minimized when $\alpha = \sigma_1^2/(\sigma_1^2 + \sigma_2^2)$. As a consequence, we estimate σ_1^2 and σ_2^2 with a pilot run and compute the optimal split ratio as $\hat{\alpha} = \hat{S}_1^2/(\hat{S}_1^2 + \hat{S}_2^2) \approx 13.4\%$.

Figure 6 displays the estimate of δ together with the asymptotic confidence intervals for different sample sizes N on a logarithmic scale.

In addition, the payoff of a Digital call option under the Black–Scholes model can be found analytically as follow,

$$\mathbb{E}[e^{-rT}\mathbf{I}\{S_T > K\}] = e^{-rT}\mathbb{P}(S_T > K) = e^{-rT}\mathbb{P}\Big(S_0 \exp\big\{(r - \frac{1}{2}\sigma^2)T + \sigma W_T\big\} > K\Big)$$
$$= e^{-rT}\mathbb{P}\Big(W_T > \frac{\ln(K/S_0) - (r - \frac{1}{2}\sigma^2)T}{\sigma}\Big) = e^{-rT}\Big(1 - \Phi\Big(\frac{\ln(K/S_0) - (r - \frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}}\Big)\Big).$$

This allows us to compute the analytical value of δ and plot it alongside the numerical estimate.

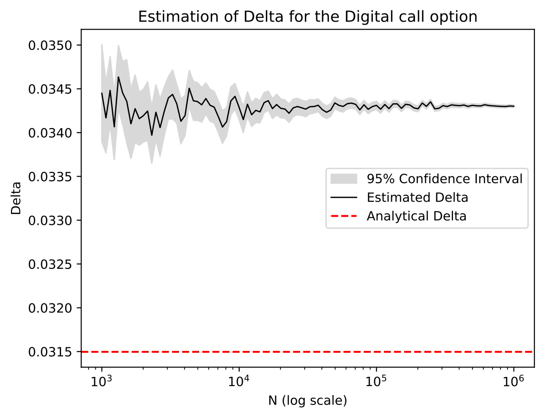

Figure 6

The method provides a good approximation of δ. While the estimator exhibits some bias, the small relative error (8.9%) indicates that the method is still very effective. Furthermore, the variance of the estimator is comparatively very low, also when using a small sample.

Finally, we investigate how the value of ε affects the variance of the estimator. To do this, we fix $N = 10^6$ and, in Figure 7, we plot the variance of the estimator as a function of $\varepsilon \in [0, 80]$.

We recall that setting $\varepsilon = 0$ is the same as using only the pathwise derivative method of $f(S_T)$. However, following a similar argument as for γ before, the pathwise derivative method cannot be applied directly to f, since f is not continuous at K. In addition, in this case, the estimator would be identically equal to 0 and have variance 0.

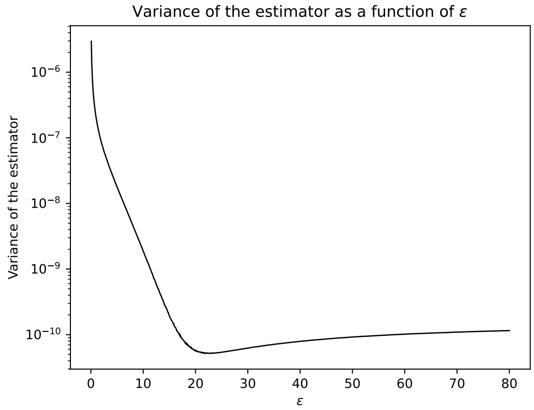

Figure 7

Excluding the case where $\varepsilon = 0$, the variance has unique minimum in $\varepsilon = 21.70$. This value has been found through a grid search over the interval $[0, 80]$ at 1000 uniformly spaced points within this range.

3 Asian call option

An Asian call option is a financial derivative that provides a payoff based on the average price of the underlying asset over a specified period, rather than its price at a single point in time. This feature reduces the impact of short-term volatility and makes the option less sensitive to price manipulation at expiration. The payoff of an Asian call option at maturity is defined as

$$f(S_{T/m}, S_{2T/m}, ..., S_T) = e^{-rT}[\overline{S}_T - K]^+.$$

where m is a fixed positive integer and $\overline{S}_T = \frac{1}{m}\sum_{i=1}^m S_{iT/m}$. For simplicity, in this section we use the notation $t_i = iT/m$ for $i = 0, 1, ..., m$.

Estimate Delta using pathwise derivative

In order to build a Monte Carlo estimator for δ using the pathwise derivative method, we need to compute f' and $\frac{\partial \overline{S}_T}{\partial S_0}$.

For the first, we have,

$$f'(s) = e^{-rT}\frac{\mathrm{d}}{\mathrm{d}s}[s - K]^+ = e^{-rT}\mathbf{I}\{s > K\}.$$

For the second, we start computing

$$\frac{\partial S_{t_i}}{\partial S_0} = \frac{\partial}{\partial S_0} S_0 \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)t_i + \sigma W_{t_i}\Big\} = \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)t_i + \sigma W_{t_i}\Big\} = \frac{S_{t_i}}{S_0},$$

so that, using linearity property of the differential operator,

$$\frac{\partial \overline{S}_T}{\partial S_0} = \frac{1}{m}\sum_{i=1}^m \frac{\partial S_{t_i}}{\partial S_0} = \frac{1}{m}\sum_{i=1}^m \frac{S_{t_i}}{S_0} = \frac{\overline{S}_T}{S_0}.$$

Finally, using (1), we have

$$\frac{\partial}{\partial S_0}\mathbb{E}[e^{-rT}[\overline{S}_T - K]^+] = \mathbb{E}\Big[e^{-rT}\mathbf{I}\{\overline{S}_T > K\}\frac{\overline{S}_T}{S_0}\Big]$$

which can be estimated via Monte Carlo.

Estimate Vega using pathwise derivative

In order to build a Monte Carlo estimator for ν using the pathwise derivative method, we only need to compute $\frac{\partial \overline{S}_T}{\partial \sigma}$, since f' is the same as above.

We start computing

$$\frac{\partial S_{t_i}}{\partial \sigma} = \frac{\partial}{\partial \sigma} S_0 \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)t_i + \sigma W_{t_i}\Big\} = S_0 \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)t_i + \sigma W_{t_i}\Big\}(-\sigma t_i + W_{t_i}),$$

so that, using linearity property of the differential operator,

$$\frac{\partial \overline{S}_T}{\partial \sigma} = \frac{1}{m}\sum_{i=1}^{m}\frac{\partial S_{t_i}}{\partial \sigma} = \frac{1}{m}\sum_{i=1}^{m} S_0 \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)t_i + \sigma W_{t_i}\Big\}(-\sigma t_i + W_{t_i}).$$

Finally, using (1), we have

$$\begin{aligned}\frac{\partial}{\partial \sigma}\mathbb{E}[e^{-rT}[\overline{S}_T - K]^+] &= \frac{1}{m}\mathbb{E}\Big[e^{-rT}\mathbf{I}\{\overline{S}_T > K\}\sum_{i=1}^{m} S_0 \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)t_i + \sigma W_{t_i}\Big\}(-\sigma t_i + W_{t_i})\Big] \\ &= \frac{1}{m}\mathbb{E}\Big[e^{-rT}\mathbf{I}\{\overline{S}_T > K\}\sum_{i=1}^{m} S_{t_i}(-\sigma t_i + W_{t_i})\Big]\end{aligned}$$

which can be estimated via Monte Carlo.

Estimate Delta using likelihood ratio method

In order to build a Monte Carlo estimator for δ using the likelihood ratio method, we have two alternatives. We can either try to characterize the distribution of $\overline{S}_T$, or directly use the joint density of $S_{t_1}, \dots, S_{t_m}$. Here, we use the second approach.

The first step is to find the joint density of $S_{t_1}, \dots, S_{t_m}$. We notice that

$$\begin{aligned}S_{t_i} &= S_0 \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)t_i + \sigma W_{t_i}\Big\} \\ &= S_0 \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)t_{i-1} + \sigma W_{t_{i-1}}\Big\}\exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)(t_i - t_{i-1}) + \sigma(W_{t_i} - W_{t_{i-1}})\Big\} \\ &= S_{t_{i-1}} \exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)(t_i - t_{i-1}) + \sigma(W_{t_i} - W_{t_{i-1}})\Big\},\end{aligned} \tag{4}$$

so that, using the independence of increments for the Brownian Motion, S_{t_i} depends on $S_{t_1}, \dots, S_{t_{i-1}}$ only through $S_{t_{i-1}}$. This means that we can factor the joint density as

$$g_{S_{t_1},\dots,S_{t_m}}(s_1, \dots, s_m) = g_{S_{t_1}}(s_1)\prod_{i=2}^{m} g_{S_{t_i}|S_{t_{i-1}},\dots,S_{t_1}}(s_i \mid s_{i-1}, \dots, s_1) = g_{S_{t_1}}(s_1)\prod_{i=2}^{m} g_{S_{t_i}|S_{t_{i-1}}}(s_i \mid s_{i-1})$$

where

$$g_{S_{t_1}}(s_1) = \frac{1}{s_1\sigma\sqrt{2\pi t_1}}\exp\Big[-\frac{1}{2}\Big(\frac{\log(s_1/S_0) - \big(r - \frac{1}{2}\sigma^2\big)t_1}{\sigma\sqrt{t_1}}\Big)^2\Big]$$

and

$$g_{S_{t_i}|S_{t_{i-1}}}(s_i \mid s_{i-1}) = \frac{1}{s_i\sigma\sqrt{2\pi(t_i - t_{i-1})}}\exp\Big[-\frac{1}{2}\Big(\frac{\log(s_i/s_{i-1}) - \big(r - \frac{1}{2}\sigma^2\big)(t_i - t_{i-1})}{\sigma\sqrt{t_i - t_{i-1}}}\Big)^2\Big],$$

for $i = 2, \dots, m$. Note that these are probability density function of log-normal random variables, that is

$$S_{t_i} \mid S_{t_{i-1}} \sim \text{log-normal}\Big(\log(S_{t_{i-1}}) + \big(r - \frac{1}{2}\sigma^2\big)(t_i - t_{i-1}),\, \sigma^2(t_i - t_{i-1})\Big),$$

for $i = 1, \ldots, m$, using the convention $S_{t_1} \mid S_{t_0} \equiv S_{t_1}$.

We continue by computing $\frac{\partial}{\partial S_0} \log g_{S_{t_1},\ldots,S_{t_m}}(S_{t_1}, \ldots, S_{t_m})$. Since S_0 is a parameter of $g_{S_{t_1}}$ only, we obtain

$$\begin{aligned}\frac{\partial}{\partial S_0} \log g_{S_{t_1},\ldots,S_{t_m}}(S_{t_1}, \ldots, S_{t_m}) &= \frac{\partial}{\partial S_0}\Big(\log g_{S_{t_1}}(S_{t_1}) + \sum_{i=2}^{m} \log g_{S_{t_i}|S_{t_{i-1}}}(S_{t_i} \mid S_{t_{i-1}})\Big) \\ &= \frac{\partial}{\partial S_0} \log g_{S_{t_1}}(S_{t_1}) = \frac{\log(S_{t_1}/S_0) - \big(r - \frac{1}{2}\sigma^2\big)t_1}{\sigma^2 t_1 S_0}.\end{aligned}$$

Finally, substituting $S_{t_1} = S_0 \exp\big((r - \frac{1}{2}\sigma^2)t_1 + \sigma W_{t_1}\big)$, we obtain

$$\frac{\partial}{\partial S_0} \log g_{S_{t_1},\ldots,S_{t_m}}(S_{t_1}, \ldots, S_{t_m}) = \frac{W_{t_1}}{\sigma t_1 S_0}$$

so that, using (2),

$$\frac{\partial}{\partial S_0}\mathbb{E}[e^{-rT}[\overline{S}_T - K]^+] = \mathbb{E}\Big[e^{-rT}[\overline{S}_T - K]^+ \frac{W_{t_1}}{\sigma t_1 S_0}\Big]$$

which can be estimated via Monte Carlo.

Estimate Vega using likelihood ratio method

In order to build a Monte Carlo estimator for ν using the likelihood ratio method, we only need to compute $\frac{\partial}{\partial \sigma} \log g_{S_{t_1},\ldots,S_{t_m}}(S_{t_1}, \ldots, S_{t_m})$.

We start computing, for $i = 1, \ldots, m$,

$$\begin{aligned}&\frac{\partial}{\partial \sigma} \log g_{S_{t_i}|S_{t_{i-1}}}(S_{t_i} \mid S_{t_{i-1}}) \\ &= -\frac{1}{\sigma} + \frac{\log(S_{t_i}/S_{t_{i-1}}) - \big(r - \frac{1}{2}\sigma^2\big)(t_i - t_{i-1})}{\sigma\sqrt{t_i - t_{i-1}}}\Big(\frac{\log(S_{t_i}/S_{t_{i-1}}) - \big(r - \frac{1}{2}\sigma^2\big)(t_i - t_{i-1})}{\sigma\sqrt{t_i - t_{i-1}}} - \sqrt{t_i - t_{i-1}}\Big).\end{aligned} \tag{5}$$

Substituting (4) in (5), we obtain

$$\frac{\partial}{\partial \sigma} \log g_{S_{t_i}|S_{t_{i-1}}}(S_{t_i} \mid S_{t_{i-1}}) = \frac{(W_{t_i} - W_{t_{i-1}})^2}{\sigma(t_i - t_{i-1})} - (W_{t_i} - W_{t_{i-1}}) - \frac{1}{\sigma}$$

Using linearity property of the differential operator,

$$\begin{aligned}\frac{\partial}{\partial \sigma} \log g_{S_{t_1},\ldots,S_{t_m}}(S_{t_1}, \ldots, S_{t_m}) &= \frac{\partial}{\partial \sigma}\Big(\log g_{S_{t_1}}(S_{t_1}) + \sum_{i=2}^{m} \log g_{S_{t_i}|S_{t_{i-1}}}(S_{t_i} \mid S_{t_{i-1}})\Big) \\ &= \frac{\partial}{\partial \sigma} \log g_{S_{t_1}}(S_{t_1}) + \sum_{i=2}^{m} \frac{\partial}{\partial \sigma} \log g_{S_{t_i}|S_{t_{i-1}}}(S_{t_i} \mid S_{t_{i-1}}) \\ &= \sum_{i=1}^{m}\Big(\frac{(W_{t_i} - W_{t_{i-1}})^2}{\sigma(t_i - t_{i-1})} - (W_{t_i} - W_{t_{i-1}}) - \frac{1}{\sigma}\Big) = -W_T + \sum_{i=1}^{m} \frac{(W_{t_i} - W_{t_{i-1}})^2 - (t_i - t_{i-1})}{\sigma(t_i - t_{i-1})}\end{aligned}$$

so that, using (2),

$$\frac{\partial}{\partial \sigma}\mathbb{E}[e^{-rT}[\overline{S}_T - K]^+] = \mathbb{E}\Big[e^{-rT}[\overline{S}_T - K]^+\Big(-W_T + \sum_{i=1}^{m} \frac{(W_{t_i} - W_{t_{i-1}})^2 - (t_i - t_{i-1})}{\sigma(t_i - t_{i-1})}\Big)\Big]$$

which can be estimated via Monte Carlo.

Appendix

A Applicability of mixed methods when estimating Gamma for the European call option

As an extra, in this section, we prove that the methods PW-LR, LR-PW, LR-LR are indeed applicable.

For simplicity, we assume that the first derivative is always well-defined and computable. This assumption is reasonable because, with respect to the first derivative, all functions involved are continuous and sufficiently smooth. Furthermore, our numerical experiments in Section 1 have shown that the various methods for the first derivative converge to the correct value.

In order to verify which methods are applicable, we will make use of the following theorem.

Theorem A (Differentiation under the Integral Sign)

Let $J \subset \mathbb{R}$ be an interval. Suppose $g : \mathbb{R} \times J \to [-\infty, \infty]$ satisfies the following conditions:

(i) For each $t \in J$, $g(x,t)$ is a integrable function of x on X.

(ii) For almost every $x \in \mathbb{R}$, $g(x,t)$ is differentiable with respect to t for all $t \in J$.

(iii) There exists an integrable function $h : \mathbb{R} \to [-\infty, \infty]$ such that for almost every $x \in \mathbb{R}$ and for all $t \in J$,

$$\Big|\frac{\partial g(x,t)}{\partial t}\Big| \leq h(x).$$

Then, the function $G : J \to \mathbb{R}$, defined as $G(t) = \int_{\mathbb{R}} g(x,t)\,\mathrm{d}x$, is differentiable on J, and

$$G'(t) = \int_{\mathbb{R}} \frac{\partial g(x,t)}{\partial t}\,\mathrm{d}x.$$

Since we are interested in computing the derivatives in $S_0 = 100$, we can assume $S_0 \in J := [S_0^{\text{min}}, S_0^{\text{max}}]$, with $S_0^{\text{min}} = 100 - \varepsilon$ and $S_0^{\text{max}} = 100 + \varepsilon$ for $\varepsilon > 0$ and $0 < S_0^{\text{min}} < 100 < S_0^{\text{max}}$.

Use pathwise derivative first and likelihood ratio second

We show that we can use the likelihood ratio method to estimate the second derivative.

We start by constructing our Monte Carlo estimator. We have

$$\begin{aligned}\frac{\partial^2}{\partial S_0^2}\mathbb{E}[f(S_T)] &\overset{\text{PW}}{=} \frac{\partial}{\partial S_0}\mathbb{E}\Big[f'(S_T)\frac{\partial S_T}{\partial S_0}\Big] = \frac{\partial}{\partial S_0}\mathbb{E}\Big[f'(S_T)\frac{S_T}{S_0}\Big] = \frac{\partial}{\partial S_0}\frac{1}{S_0}\mathbb{E}[f'(S_T)S_T] \\ &= -\frac{1}{S_0^2}\mathbb{E}[f'(S_T)S_T] + \frac{1}{S_0}\frac{\partial}{\partial S_0}\mathbb{E}[f'(S_T)S_T]\end{aligned}$$

where

$$\begin{aligned}\frac{\partial}{\partial S_0}\mathbb{E}[f'(S_T)S_T] &= \frac{\partial}{\partial S_0}\int_0^\infty s\,f'(s)p_{S_T}(s,S_0)\mathrm{d}s \overset{\text{LR}}{=} \int_0^\infty s\,f'(s)\frac{\partial}{\partial S_0}p_{S_T}(s,S_0)\mathrm{d}s \\ &= \mathbb{E}\Big[S_T f'(S_T)\frac{\partial}{\partial S_0}\log p_{S_T}(S_T,S_0)\Big].\end{aligned}$$

In order to justify $\overset{\text{LR}}{=}$, we must show that

$$g(s,S_0) = s\,f'(s)p_{S_T}(s,S_0) = \frac{e^{-rT}}{\sigma\sqrt{2\pi T}}\mathbf{I}\{s > K\}\exp\Big[-\frac{1}{2}\Big(\frac{\log(s/S_0) - \big(r - \frac{1}{2}\sigma^2\big)T}{\sigma\sqrt{T}}\Big)^2\Big]$$

satisfies properties (i), (ii), and (iii) from Theorem A.

Condition (i) is satisfied because, for all $S_0 \in J$,

$$\int_0^\infty g(s, S_0)\mathrm{d}s = e^{-rT}\int_0^\infty s\,\mathbf{I}\{s > K\}p_{S_T}(s, S_0)\mathrm{d}s \le e^{-rT}\int_0^\infty s\,p_{S_T}(s, S_0)\mathrm{d}s = e^{-rT}\mathbb{E}[S_T] < \infty,$$

since S_T is a log-normal random variable.

Condition (ii) is satisfied because

$$\begin{aligned}\frac{\partial}{\partial S_0}g(s, S_0) &= \frac{e^{-rT}}{\sigma\sqrt{2\pi T}}\mathbf{I}\{s > K\}\exp\Big[-\frac{1}{2}\Big(\frac{\log(s/S_0) - \big(r - \frac{1}{2}\sigma^2\big)T}{\sigma\sqrt{T}}\Big)^2\Big]\frac{\log(s/S_0) - \big(r - \frac{1}{2}\sigma^2\big)T}{\sigma^2 T S_0} \\ &= e^{-rT}\mathbf{I}\{s > K\}s\,\frac{\log(s/S_0) - \big(r - \frac{1}{2}\sigma^2\big)T}{\sigma^2 T S_0}p_{S_T}(s, S_0)\end{aligned}$$

which is defined for every $S_0 > 0$.

Condition (iii) is satisfied because, for all $S_0 \in J$,

$$\Big|\frac{\partial}{\partial S_0}g(s, S_0)\Big| \le \Big|e^{-rT}\mathbf{I}\{s > K\}s\,\frac{\log(s) - \big(r - \frac{1}{2}\sigma^2\big)T}{\sigma^2 T S_0^{\mathsf{min}}}\max_{S_0\in J}p_{S_T}(s, S_0)\Big| = h(s).$$

Since J is bounded and p_{S_T} is continuous in both parameters,

$$\max_{S_0\in J}p_{S_T}(s, S_0) \sim \frac{1}{s}\exp\Big[-\frac{\log^2(s)}{2\sigma^2 T}\Big] \quad \text{and} \quad h(s) \sim \log(s)\exp\Big[-\frac{\log^2(s)}{2\sigma^2 T}\Big]$$

for $s \to \infty$. Then, since h is continuous, it follows that

$$\int_0^\infty h(x)\mathrm{d}x < \infty,$$

concluding the proof.

Use likelihood ratio method twice

We show that we can use the likelihood ratio method to estimate the second derivative.

We start by constructing our Monte Carlo estimator. We have

$$\begin{aligned}\frac{\partial^2}{\partial S_0^2}\mathbb{E}[f(S_T)] &= \frac{\partial^2}{\partial S_0^2}\int_0^\infty f(s)p_{S_T}(s)\mathrm{d}s \overset{\mathsf{LR}}{=} \frac{\partial}{\partial S_0}\int_0^\infty f(s)\frac{\partial p_{S_T}(s)}{\partial S_0}\mathrm{d}s \overset{\mathsf{LR}}{=} \int_0^\infty f(s)\frac{\partial^2 p_{S_T}(s)}{\partial S_0^2}\mathrm{d}s \\ &= \mathbb{E}\Big[f(S_T)\frac{\frac{\partial^2 p_{S_T}}{\partial S_0^2}(S_T)}{p_{S_T}(S_T)}\Big] = \mathbb{E}\Big[f(S_T)\Big(\frac{\partial^2}{\partial S_0^2}\log p_{S_T}(S_T) + \Big(\frac{\partial}{\partial S_0}\log p_{S_T}(S_T)\Big)^2\Big)\Big],\end{aligned}$$

In order to justify the second $\overset{\mathsf{LR}}{=}$, we must show that

$$g(s, S_0) = f(s)\frac{\partial}{\partial S_0}p_{S_T}(s, S_0) = e^{-rT}[s - K]^+\frac{\log(s/S_0) - \big(r - \frac{1}{2}\sigma^2\big)T}{\sigma^2 T S_0}p_{S_T}(s, S_0)$$

satisfies properties (i), (ii), and (iii) from Theorem A.

Condition (i) is satisfied because, for all $S_0 \in J$,

$$p_{S_T}(s, S_0) \sim \frac{1}{s}\exp\Big[-\frac{\log^2(s)}{2\sigma^2 T}\Big] \quad \text{and} \quad g(s, S_0) \sim \log(s)\exp\Big[-\frac{\log^2(s)}{2\sigma^2 T}\Big]$$

for $s \to \infty$.

Condition (ii) is satisfied because

$$\begin{aligned}&\frac{\partial}{\partial S_0}g(s, S_0) \\ &= e^{-rT}[s - K]^+\Big(\Big(\frac{\log(s/S_0) - \big(r - \frac{1}{2}\sigma^2\big)T}{\sigma^2 T S_0}\Big)^2 - \frac{\log(s/S_0) - \big(r - \frac{1}{2}\sigma^2\big)T + 1}{\sigma^2 T S_0}\Big)p_{S_T}(s, S_0)\end{aligned}$$

which is defined for every $S_0 > 0$.

Condition (iii) is satisfied because, for all $S_0 \in J$,

$$\Big|\frac{\partial}{\partial S_0} g(s, S_0)\Big|$$
$$\leq \Big|e^{-rT}[s-K]^+\Big(\Big(\frac{\log(s) - \big(r - \frac{1}{2}\sigma^2\big)T}{\sigma^2 T S_0^{\mathsf{min}}}\Big)^2 - \frac{\log(s) - \big(r - \frac{1}{2}\sigma^2\big)T + 1}{\sigma^2 T S_0^{\mathsf{min}}}\Big) \max_{S_0 \in J} p_{S_T}(s, S_0)\Big| = h(s).$$

Since J is bounded and p_{S_T} is continuous in both parameters,

$$\max_{S_0 \in J} p_{S_T}(s, S_0) \sim \frac{1}{s}\exp\Big[-\frac{\log^2(s)}{2\sigma^2 T}\Big] \quad \text{and} \quad h(s) \sim \log^2(s)\exp\Big[-\frac{\log^2(s)}{2\sigma^2 T}\Big]$$

for $s \to \infty$. Then, since h is continuous, it follows that

$$\int_0^\infty h(x)\mathrm{d}x < \infty,$$

concluding the proof.

Use likelihood ratio first and pathwise derivative second

We show that we can use the pathwise derivative method to estimate the second derivative.

We start by constructing our Monte Carlo estimator. We have

$$\frac{\partial^2}{\partial S_0^2}\mathbb{E}[f(S_T)] \overset{\mathsf{LR}}{=} \frac{\partial}{\partial S_0}\mathbb{E}[f(S_T)\frac{\partial}{\partial S_0}\log p_{S_T}(S_T, S_0)]$$
$$\overset{\mathsf{PW}}{=} \mathbb{E}\Big[\Big(f'(S_T)\frac{\partial}{\partial S_0}\log p_{S_T}(S_T, S_0) + f(S_T)\frac{\partial^2}{\partial S_0 \partial S_T}\log p_{S_T}(S_T, S_0)\Big)\frac{\partial S_T}{\partial S_0} + \frac{\partial^2}{\partial S_0^2}\log p_{S_T}(S_T, S_0)\Big]$$
$$= \mathbb{E}\Big[e^{-rT}\frac{W_T}{\sigma T S_0^2}\big(\mathbf{I}\{S_T > K\}S_T - [S_T - K]^+\big)\Big].$$

In order to justify $\overset{\mathsf{PW}}{=}$, we must show that

$$g(s, S_0) = f(S_T)\frac{\partial}{\partial S_0}\log p_{S_T}(S_T, S_0)p_{W_T}(s) = e^{-rT}\Big[S_0\exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)T + \sigma s\Big\} - K\Big]^+ \frac{s}{\sigma T S_0}p_{W_T}(s)$$

satisfies properties (i), (ii), and (iii) from Theorem A.

Condition (i) is satisfied because, for all $S_0 \in J$,

$$p_{W_T}(s) \sim \exp\Big(-\frac{s^2}{2T}\Big) \quad \text{and} \quad g(s, S_0) \sim s\,\exp(\sigma s)\,\exp\Big(-\frac{s^2}{2T}\Big)$$

for $s \to \infty$.

Condition (ii) is satisfied because

$$\frac{\partial}{\partial S_0}g(s, S_0)$$
$$= e^{-rT}\frac{s}{\sigma T S_0}\Big(\mathbf{I}\Big\{S_0\exp\big\{\big(r - \frac{1}{2}\sigma^2\big)T + \sigma s\big\} > K\Big\} - \frac{1}{S_0}\Big[S_0\exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)T + \sigma s\Big\} - K\Big]^+\Big)p_{W_T}(s)$$

which is defined for every $S_0 > 0$.

Condition (iii) is satisfied because, for all $S_0 \in J$,

$$\Big|\frac{\partial}{\partial S_0}g(s, S_0)\Big| \leq \Big|e^{-rT}\frac{s}{\sigma T S_0^{\mathsf{min}}}\Big(\mathbf{I}\Big\{S_0^{\mathsf{max}}\exp\big\{\big(r - \frac{1}{2}\sigma^2\big)T + \sigma s\big\} > K\Big\}$$
$$-\frac{1}{S_0^{\mathsf{min}}}\Big[S_0^{\mathsf{max}}\exp\Big\{\big(r - \frac{1}{2}\sigma^2\big)T + \sigma s\Big\} - K\Big]^+\Big)p_{W_T}(s)\Big| = h(s).$$

Since h is continuous and

$$h(s) \sim s\, \exp(\sigma s)\, \exp\left(-\frac{s^2}{2T}\right)$$

for $s \to \infty$, it follows that

$$\int_0^\infty h(x)\mathrm{d}x < \infty,$$

concluding the proof.

B An extract of the code

In this section, we report an extract of the code we used for our numerical experiments. Since this is just an extract, we do not report the definition of all the functions utilized in the code, but just some of them, as examples.

European call option

```python
# We initialize a random number generator for reproducibility
rng = np.random.default_rng(42)

# Set parameters
r = 0.05
T = 1
sig = 0.25
S0 = 100
K = 120

num_NN = 100
NN = np.logspace(3,6,num_NN).astype(int)
NN = np.array([x for (i,x) in enumerate(NN) if i==0 or x>NN[i-1]])

h = 0.00001


# Define S and f for European call option
def S(WT, r = 0.05, T = 1, sig = 0.25, S0=100):
    return S0 * np.exp((r-0.5*sig**2)*T + sig*WT)
def f(S, r = 0.05, T = 1, K = 120):
    return np.exp(-r*T)*np.maximum(S - K, 0)

# Define derivarive of f and derivative of S wrt S0
def de_f(S, r = 0.05, T = 1, K = 120):
    return np.exp(-r*T) * np.array([1. if x-K >= 0 else 0. for x in S])
def de_S0_S(WT, r = 0.05, T = 1, sig = 0.25, S0=100):
    return np.exp((r-0.5*sig**2)*T + sig*WT)

# Define mixed derivative of the score function wrt S0 and S
def de_S_de_S0_logp(S, r = 0.05, T = 1, sig = 0.25, S0=100):
    return 1./(sig**2 * T * S0 * S)


# Estimate Delta using finite differences
deltas_fd = np.empty(num_NN, dtype=float)
deltas_fd_std = np.empty(num_NN, dtype=float)
for i, N in enumerate(NN):
    WT = rng.normal(0, T, size=N)

    sample = (f(S(WT, S0 = S0 + h)) - f(S(WT, S0 = S0 - h)))/(2*h)

    deltas_fd[i] = np.mean(sample)
    deltas_fd_std[i] = np.std(sample)/np.sqrt(N)

# Estimate Delta using pathwise derivative
deltas_pw = np.empty(num_NN, dtype=float)
deltas_pw_std = np.empty(num_NN, dtype=float)
for i, N in enumerate(NN):
```

```
    WT = rng.normal(0, T, size=N)

    sample = de_f(S(WT)) * de_S0_S(WT)

    deltas_pw[i] = np.mean(sample)
    deltas_pw_std[i] = np.std(sample)/np.sqrt(N)

# Estimate Vega using likelihood ratio method
vegas_lr = np.empty(num_NN, dtype=float)
vegas_lr_std = np.empty(num_NN, dtype=float)
for i, N in enumerate(NN):
    WT = rng.normal(0, T, size=N)

    sample = f(S(WT)) * de_sig_logp(S(WT))

    vegas_lr[i] = np.mean(sample)
    vegas_lr_std[i] = np.std(sample)/np.sqrt(N)

# Estimate Gamma using likelihood ratio first and pathwise derivative second
gammas_lr_pw = np.empty(num_NN, dtype=float)
gammas_lr_pw_std = np.empty(num_NN, dtype=float)
for i, N in enumerate(NN):
    WT = rng.normal(0, T, size=N)

    sample = (de_f(S(WT))*de_S0_logp(S(WT)) + f(S(WT))*de_S_de_S0_logp(S(WT))) * de_S0_S
        (WT) + f(S(WT))*de2_S0_logp(S(WT))

    gammas_lr_pw[i] = np.mean(sample)
    gammas_lr_pw_std[i] = np.std(sample)/np.sqrt(N)

# Estimate Gamma using likelihood ratio twice
gammas_lr_lr = np.empty(num_NN, dtype=float)
gammas_lr_lr_std = np.empty(num_NN, dtype=float)
for i, N in enumerate(NN):
    WT = rng.normal(0, T, size=N)

    sample = f(S(WT)) * (de2_S0_logp(S(WT)) + de_S0_logp(S(WT))**2)

    gammas_lr_lr[i] = np.mean(sample)
    gammas_lr_lr_std[i] = np.std(sample)/np.sqrt(N)
```

Digital call option

```
# Set parameters
r = 0.05
T = 0.25
sig = 0.25
S0 = 100
K = 100

num_NN = 100
NN = np.logspace(3,6,num_NN).astype(int)
NN = np.array([x for (i,x) in enumerate(NN) if i==0 or x>NN[i-1]])

eps = 20

# Define S and f for Digital call option
def S(WT, r = 0.05, T = 0.25, sig = 0.25, S0=100):
    return S0 * np.exp((r-0.5*sig**2)*T + sig*WT)
def f(S, r = 0.05, T = 0.25, K = 100):
    return np.exp(-r*T) * np.array([1. if x > K else 0. for x in S])

# Approximate f as fe + he
def fe(S, r = 0.05, T = 0.25, K = 100, eps=20):
    if eps == 0:
        return f(S, r=r, T=T, K=K)
    else:
        return np.exp(-r*T) * np.clip((S-K+eps)/(2*eps), 0, 1)
def he(S, r = 0.05, T = 0.25, K = 100, eps=20):
```

```python
    return f(S, r=r, T=T, K=K) - fe(S, r=r, T=T, K=K, eps=eps)

# Define derivarive of fe, derivative of S wrt S0, and derivative of the score function
    wrt S0
def de_fe(S, r = 0.05, T = 0.25, K = 100, eps=20):
    return np.exp(-r*T) * np.array([1./(2*eps) if -eps < x-K <= eps else 0. for x in S])
def de_S0_S(WT, r = 0.05, T = 0.25, sig = 0.25, S0=100):
    return np.exp((r-0.5*sig**2)*T + sig*WT)
def de_S0_logp(S, r = 0.05, T = 0.25, sig = 0.25, S0=100):
    return (np.log(S/S0)-(r-0.5*sig**2)*T)/(sig**2*T*S0)


# Use a pilot run to determine the optimal split ratio
N = 10000000

WT1 = rng.normal(0, T, size=N)
WT2 = rng.normal(0, T, size=N)

sample1 = de_fe(S(WT1)) * de_S0_S(WT1)
sample2 = he(S(WT2)) * de_S0_logp(S(WT2))

var1 = np.var(sample1)
var2 = np.var(sample2)

alpha = var1/(var1+var2)

print(f"Optimal split ratio: {100*alpha:.2f}%")


# Estimate Delta
deltas_N = np.empty(num_NN, dtype=float)
deltas_N_std = np.empty(num_NN, dtype=float)
for i, N in enumerate(NN):
    N1 = int(N*alpha)
    N2 = N - N1

    WT1 = rng.normal(0, T, size=N1)
    WT2 = rng.normal(0, T, size=N2)

    sample1 = de_fe(S(WT1)) * de_S0_S(WT1)
    sample2 = he(S(WT2)) * de_S0_logp(S(WT2))

    deltas_N[i] = np.mean(sample1) + np.mean(sample2)
    deltas_N_std[i] = np.sqrt(np.var(sample1)/N1 + np.var(sample2)/N2)

print(f"The relative error is {(deltas_N[-1]/delta_analytical(S0, K, r, sig, T) - 1)
    *100:.03f}%")


# Investigate how Epsilon impacts the variance of the estimator
N = 1000000

eeps_num = 1000
eeps = np.linspace(0,80,eeps_num)

deltas_eps = np.empty(eeps_num, dtype=float)
deltas_eps_var = np.empty(eeps_num, dtype=float)
for i, eps in enumerate(eeps):
    N1 = int(N*alpha)
    N2 = N - N1

    WT1 = rng.normal(0, T, size=N1)
    WT2 = rng.normal(0, T, size=N2)

    sample1 = de_fe(S(WT1), eps=eps) * de_S0_S(WT1)
    sample2 = he(S(WT2), eps=eps) * de_S0_logp(S(WT2))

    deltas_eps[i] = np.mean(sample1) + np.mean(sample2)
    deltas_eps_var[i] = np.var(sample1)/N1 + np.var(sample2)/N2

```

```python
eps_opt = eeps[np.argmin(deltas_eps_var[1:])]
print(f"The minimal variance is achieved when eps = {eps_opt:.02f}")
```