demo_bs

August 30, 2025

1 *Reminder of Itô's lemma*

1.1 The part that is Brownian Specific

Kiyoshi Itô (1940s) was building a new **integration theory** for stochastic processes.

- Ordinary calculus: $\int f(t)\, dt$
- Stochastic calculus: $\int f(t, W_t)\, dW_t$, where $dW_t^{\mathbb{P}} \equiv W_{t+dt}^{\mathbb{P}} - W_t^{\mathbb{P}}, \quad dW_t^{\mathbb{P}} \sim \mathcal{N}(0, dt)$.

But Brownian motion W_t is nowhere differentiable, so classical integration does not work. Itô's main achievement was to define a new type of integral.

1.2 2. The "main theorem"

The central result is the **existence and properties of the Itô integral**.

For a square-integrable process f, the integral

$$\int_0^t f(s, W_s)\, dW_s \tag{1}$$

is well-defined, and it satisfies:

- **Martingale property**
  The stochastic integral is a martingale.

  – 1. $\mathbb{E}[|X_t|] < \infty$ for all t. – 2. For all $s < t$, $\mathbb{E}[X_t \mid \mathcal{F}_s] = X_s$.
- **Itô isometry** – For Brownian Motion

$$\mathbb{E}\left[\left(\int_0^t f(s, W_s)\, dW_s\right)^2\right] = \mathbb{E}\left[\int_0^t f(s, W_s)^2\, ds\right]. \tag{2}$$

$$\int_a^b f(t)\, dW_t = \lim_{\|\Pi\| \to 0} \sum_{i=0}^{n-1} f(t_i)\,(W_{t_{i+1}} - W_{t_i}), \tag{3}$$

  The integral is the sum of all the $f(t)$, weighted by the incremental (infinitesimally small) of the random part, similar to the chain rule where $dW_t = dW_t/dt \times dt$ not the possible W_t at time t, but rather $W_t = W(t)$.

This theorem is the foundation of stochastic calculus.

1.3 3. Where Itô's lemma fits

Once the integral is defined, consider an SDE

$$dX_t = a(X_t, t)\, dt + b(X_t, t)\, dW_t. \tag{4}$$

If $f = f(X_t, t)$, what dynamics does f satisfy?

Itô's lemma gives the answer:

$$df = f_t\, dt + f_x\, dX_t + \tfrac{1}{2} f_{xx}\, (dX_t)^2. \tag{5}$$

This is the **chain rule** in stochastic calculus, derived from the Itô integral and the quadratic variation property of Brownian motion.

- In real analysis:
  - **Main theorem**: the Lebesgue integral exists with convergence theorems.

  - **Chain rule**: a lemma built upon it.
- In stochastic calculus:
  - **Main theorem**: the Itô integral is well-defined with isometry and martingale properties. – 1. $\mathbb{E}[|X_t|] < \infty$ for all t. – 2. For all $s < t$, $\mathbb{E}[X_t \mid \mathcal{F}_s] = X_s$.
  - **Itô's lemma**: the chain rule for SDEs.

$$dX_t = a\, dt + b\, dW_t. \tag{6}$$

- The deterministic part $a\, dt$ is order dt. - The random part $b\, dW_t$ is order $\sqrt{dt}$. This can be seen in two ways:

1.4 Stories in our story

First:

$$\mathbb{E}[|dW_t|] = \sqrt{dt}\, \mathbb{E}[|Z|] = \sqrt{dt}\, \sqrt{\tfrac{2}{\pi}} \tag{7}$$

Second, Because $dW_t/\sqrt{dt} \Rightarrow \mathcal{N}(0,1)$ (tight, non-degenerate), we write $dW_t = O_p(\sqrt{dt})$. That is, for any $\varepsilon > 0$ there exists M s.t.

$$\Pr\left(\frac{|dW_t|}{\sqrt{dt}} > M \right) < \varepsilon \tag{8}$$

for all sufficiently small dt. This formalizes "order $\sqrt{dt}$."

Quick flashback:

$$\begin{aligned} \text{Deterministic:} \forall \delta > 0, \exists M:\ 0 < h < \delta \implies |f(h)| \le M h^{\alpha} \implies f(h) = \mathcal{O}(h^{\alpha}). \\ \text{Stochastic:} \forall \varepsilon > 0, \exists M:\ \Pr(|X_h| > M h^{\alpha}) < \varepsilon \quad (h \text{ small}) \implies X_h = \mathcal{O}(h^{\alpha}). \end{aligned} \tag{9}$$

(It is crucial to find the smallest α to correctly to define the infinitesimal behavior) So the dominant scaling of dX_t is $\sqrt{dt}$, not dt.

---

1. Classical Taylor expansion

For a smooth $f(x,t)$:

$$df = f_t\,dt + f_x\,dX_t + \tfrac{1}{2} f_{xx}(dX_t)^2 + o\big((dX_t)^2\big). \tag{10}$$

In ordinary calculus, if $dX_t \sim dt$, then - $(dX_t)^2 \sim (dt)^2$, which is negligible compared to dt. - So we drop both $\frac{1}{2} f_{xx}(dX_t)^2$ and the remainder $o((dX_t)^2)$.

Substituting the stochastic process (GBM) Square it:

$$(dX_t)^2 = (a\,dt + b\,dW_t)^2 = a^2(dt)^2 + 2ab\,dt\,dW_t + b^2(dW_t)^2. \tag{11}$$

Apply Itô rules "Itô's lemma = Taylor expansion truncated at second order" - $(dt)^2 = 0$, - $dt\,dW_t = 0$, - $(dW_t)^2 = dt$.

Itô's lemma is similar: it looks like a truncation trick, but it's actually the rigorous foundation that allows us to define stochastic integrals, PDE connections, risk-neutral pricing, etc.

Proof of the third one:

Let $\Delta W := W_{t+\Delta t} - W_t \sim \mathcal{N}(0, \Delta t)$. Then

$$\mathbb{E}\big[(\Delta W)^2\big] = \Delta t, \qquad \mathrm{Var}\big((\Delta W)^2\big) = \mathbb{E}[(\Delta W)^4] - \big(\mathbb{E}(\Delta W)^2\big)^2 = 3(\Delta t)^2 - (\Delta t)^2 = 2(\Delta t)^2. \tag{12}$$

Hence, with $R_{\Delta t} := (\Delta W)^2 - \Delta t$,

$$\mathbb{E}[R_{\Delta t}] = 0, \qquad \mathrm{SD}(R_{\Delta t}) = \sqrt{2}\Delta t. \tag{13}$$

So $R_{\Delta t} = O_p(\Delta t)$ (indeed also $O_{L^2}(\Delta t)$). Conclusion: for a single increment,

$$(\Delta W)^2 = \Delta t + O_p(\Delta t), \tag{14}$$

not $O(\Delta t^2)$.

Alternatively,

Let $\Pi = \{0 = t_0 < t_1 < \cdots < t_n = T\}$ be a partition, and define

$$Q(\Pi) = \sum_{i=0}^{n-1} \big(W_{t_{i+1}} - W_{t_i}\big)^2. \tag{15}$$

For Brownian motion,

$$Q(\Pi) \xrightarrow[|\Pi|\to 0]{\text{a.s.}} T, \tag{16}$$

Where $\|\Pi\| := \max_i (t_{i+1} - t_i)$. so the quadratic variation is $[W]_t = t$. Heuristically this is the integral identity

$$\sum (dW_t)^2 = \int_0^T dt, \quad \text{i.e.} \quad (dW_t)^2 = dt. \tag{17}$$

2 The Lemma Beyond Brownian

Finance quickly saw that real markets are not pure Brownian: - Heavy tails → models with jumps (Poisson, Lévy). - Volatility clustering → stochastic volatility (Heston model). - Long memory → fractional Brownian motion and rough volatility.

In these cases, Itô's lemma still works, but in generalized forms (with quadratic variation and jump terms).

3 Black–Scholes PDE: step-by-step derivation

3.1 1) Modeling assumptions

- Risky asset S_t follows GBM (Geometric Brownian Motion):

$$dS_t = \mu S_t \, dt + \sigma S_t \, dW_t, \quad \sigma > 0 \tag{18}$$

- Money market account B_t grows at constant risk-free rate r:

$$dB_t = r B_t \, dt \tag{19}$$

- No arbitrage, continuous trading, no dividends.

- Derivative value $V = V(S,t)$ is smooth enough ($C^{2,1}$ in (S,t)).
- Where $dW_t \sim \mathcal{N}(0, dt)$

3.2 2) Apply Itô's lemma to $V(S_t,t)$ – the equation for the derivatives – total value(time value + intrinsic value)

$$dV = \frac{\partial V}{\partial t} \, dt + \frac{\partial V}{\partial S} \, dS_t + \tfrac{1}{2} \frac{\partial^2 V}{\partial S^2} (dS_t)^2 \tag{20}$$

Since $(dW_t)^2 = dt$ and $(dS_t)^2 = \sigma^2 S_t^2 dt$:

$$dV = \left(\frac{\partial V}{\partial t} + \mu S \frac{\partial V}{\partial S} + \tfrac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} \right) dt + \sigma S \frac{\partial V}{\partial S} \, dW_t \tag{21}$$

3.3 3) Construct hedged portfolio

Portfolio:

$$\Pi_t = V(S_t,t) - \Delta S_t \tag{22}$$

- $V(S_t,t)$ is the value of the option. - Π_t is the hedged portfolio (long the option, short Δ shares). - Δ is the hedge ratio (also called option delta in Greeks)

Self-financing change:

$$d\Pi = dV - \Delta \, dS \tag{23}$$

Choose $\Delta = \frac{\partial V}{\partial S}$ so the dW_t term cancels:

$$d\Pi = \left(\frac{\partial V}{\partial t} + \tfrac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} \right) dt \tag{24}$$

3.4 4) No-arbitrage condition

Riskless Π must earn risk-free rate:

$$d\Pi = r\Pi\, dt = r\left(V - S\frac{\partial V}{\partial S}\right) dt \tag{25}$$

Equating both:

$$\frac{\partial V}{\partial t} + \tfrac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} = r\left(V - S\frac{\partial V}{\partial S}\right) \tag{26}$$

3.5 5) We arrive at Black–Scholes PDE

$$\frac{\partial V}{\partial t} + \tfrac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0 \tag{27}$$

3.6 6) Terminal & boundary conditions – European Style

European call (K, T):

$$V(S, T) = \max(S - K, 0) \tag{28}$$

$$V(0, t) = 0, \quad V(S, t) \sim S - Ke^{-r(T-t)} \text{ as } S \to \infty \tag{29}$$

European put:

$$V(S, T) = \max(K - S, 0) \tag{30}$$

$$V(0, t) = Ke^{-r(T-t)}, \quad V(S, t) \to 0 \text{ as } S \to \infty \tag{31}$$

3.7 7) Risk-neutral representation

the stock dynamics under the risk-neutral measure $\mathbb{Q}$:

$$dS_t = rS_t\, dt + \sigma S_t\, dW_t^{\mathbb{Q}} \tag{32}$$

$dW_t^{\mathbb{Q}} = dW_t^{\mathbb{P}} + \theta\, dt$, where $\theta = \frac{\mu - r}{\sigma}$. The reason for using $\mathbb{Q}-$ martingale is because the risk-neutral hypothesis, and we move all the premiums inside the measures/probability density $\mathbb{Q}$.

$W^{\mathbb{P}}$ are $W^{\mathbb{Q}}$ two different process they are both $\mathcal{N}(0, t)$ in their measure. analoge to coin toss: - $\mathbb{P}$: $p(H) = 0.6$ $W_n = \sum_i^n (X_i - 0.2)$ - $\mathbb{Q}$: $p(H) = 0.5$ $W_n = \sum_i^n X_i$, $\mathbb{E}^{\mathbb{P}}[W_n] \neq 0$, non martingale, but $\mathbb{E}^{\mathbb{Q}}[W_n] = 0$, martingale - ($\mathbb{P}$, $\mathbb{Q}$: measure/probability distribution)

By Feynman–Kac:

$$V(S, t) = e^{-r(T-t)}\, \mathbb{E}^{\mathbb{Q}}[\Phi(S_T) \mid S_t = S] \tag{33}$$

- Φ Payoff function

3.8 8) Closed-form Black–Scholes formula

1) Risk-neutral distribution of S_T

$$\ln S_T \sim \mathcal{N}\Big(\ln S + (r - \tfrac{1}{2}\sigma^2)\tau,\ \sigma^2\tau\Big), \tag{34}$$

equivalently

$$S_T = S\exp\Big((r - \tfrac{1}{2}\sigma^2)\tau + \sigma\sqrt{\tau}\,Z\Big), \quad Z \sim \mathcal{N}(0,1). \tag{35}$$

2) Call option value as discounted expectation

$$C = e^{-r\tau}\,\mathbb{E}^{\mathbb{Q}}[(S_T - K)^+]\,. \tag{36}$$

Split into two parts:

$$C = e^{-r\tau}\left(\mathbb{E}[S_T\mathbf{1}_{\{S_T>K\}}] - K\,\mathbb{Q}(S_T > K)\right). \tag{37}$$

3) Compute probabilities

$$\mathbb{Q}(S_T > K) = \mathbb{Q}\left(Z > \frac{\ln K - \ln S - (r - \frac{1}{2}\sigma^2)\tau}{\sigma\sqrt{\tau}}\right) = N(d_2), \tag{38}$$

with

$$d_2 = \frac{\ln(S/K) + (r - \frac{1}{2}\sigma^2)\tau}{\sigma\sqrt{\tau}}. \tag{39}$$

$$\mathbb{E}[S_T\mathbf{1}_{\{S_T>K\}}] = Se^{r\tau}N(d_1), \tag{40}$$

where

$$d_1 = \frac{\ln(S/K) + (r + \frac{1}{2}\sigma^2)\tau}{\sigma\sqrt{\tau}} = d_2 + \sigma\sqrt{\tau}. \tag{41}$$

4) Call price

$$C = SN(d_1) - Ke^{-r\tau}N(d_2). \tag{42}$$

5) Put option via put–call parity

$$P = Ke^{-r\tau}N(-d_2) - SN(-d_1). \tag{43}$$

4 Now we do: Black–Scholes Validation: Analytic vs Monte Carlo

- Risk-neutral GBM simulation (from src/gbm.py)
- BS closed-form (inline)
- Confidence interval check
- Convergence $O(1/\sqrt(N))$
- Distribution sanity check for S_T

[1]:
```
import math
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
import sys, os
project_root = os.path.abspath(os.path.join(os.getcwd(), '..'))
sys.path.append(project_root)
```

[2]:
```
%cd $project_root
```

```
/Users/zhaoyub/Documents/Tradings/option-mini-lab
```

```
/Users/zhaoyub/Library/Python/3.12/lib/python/site-
packages/IPython/core/magics/osm.py:417: UserWarning: This is now an optional
IPython functionality, setting dhist requires you to install the `pickleshare`
library.
  self.shell.db['dhist'] = compress_dhist(dhist)[-100:]
```

[3]:
```
%load_ext autoreload
```

[4]:
```
from src.gbm import (
    sample_terminal_risk_neutral,
    simulate_risk_neutral_paths,
    gbm_terminal_moments,
    bs_call,
    bs_put
)
# Reproducibility
rng = np.random.default_rng(42)
```

4.1 Parameters

[5]:
```
S0, K = 100.0, 100.0
r, sigma = 0.05, 0.20
T = 1.0
```

4.2 Black–Scholes closed-form (inline)

Call:

$$C = S_0 N(d_1) - K e^{-rT} N(d_2), \quad d_1 = \frac{\ln(S_0/K) + (r + \frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}}, \; d_2 = d_1 - \sigma\sqrt{T}. \tag{44}$$

Put:

$$P = K e^{-rT} N(-d_2) - S_0 N(-d_1). \tag{45}$$

[6]:
```
true_call = bs_call(S0, K, r, sigma, T)
true_put  = bs_put (S0, K, r, sigma, T)
true_call, true_put
```

[6]: (10.450583572185565, 5.573526022256971)

4.3 Monte Carlo estimator

Discounted payoff:

$$\hat{C}_{\text{MC}} = e^{-rT}\,\frac{1}{N}\sum_{i=1}^{N}(S_T^{(i)} - K)^+. \tag{46}$$

Standard error and 95% CI:

$$\text{SE} = \frac{s}{\sqrt{N}}, \quad \text{CI}_{95\%} = \hat{C} \pm 1.96\,\text{SE}. \tag{47}$$

[7]:
```
def mc_euro_call(S0, K, r, sigma, T, n, rng=None):
    ST = sample_terminal_risk_neutral(S0, r, sigma, T, n_paths=n, rng=rng)
    payoff = np.maximum(ST - K, 0.0)
    disc_payoff = np.exp(-r * T) * payoff
    price = disc_payoff.mean()
    se = disc_payoff.std(ddof=1) / math.sqrt(n)
    return price, se

# Quick smoke test
mc_price, mc_se = mc_euro_call(S0, K, r, sigma, T, n=200_000, rng=rng)
mc_price, mc_se, true_call
```

[7]: (10.463413536194789, 0.033117555441239864, 10.450583572185565)

4.4 CI coverage check

Does the 95% CI include the analytic price?

[8]:
```
lo, hi = mc_price - 1.96 * mc_se, mc_price + 1.96 * mc_se
covered = (lo <= true_call <= hi)
print(f"MC={mc_price:.6f}, SE={mc_se:.6f}, 95% CI=({lo:.6f}, {hi:.6f}), "
      f"analytic={true_call:.6f}, covered={covered}")
```

```
MC=10.463414, SE=0.033118, 95% CI=(10.398503, 10.528324), analytic=10.450584,
covered=True
```

4.5 Convergence ($O(1/\sqrt(N))$)

We expect:

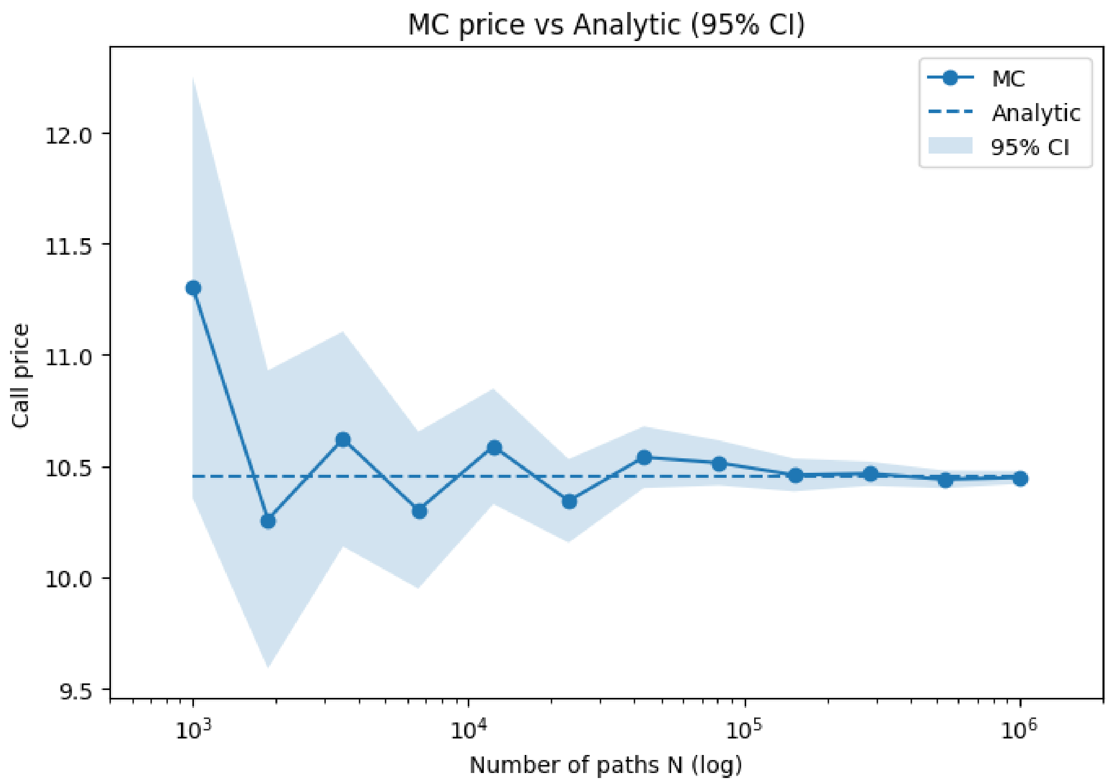
$$|\hat{C}_N - C| = \mathcal{O}(N^{-1/2}). \tag{48}$$

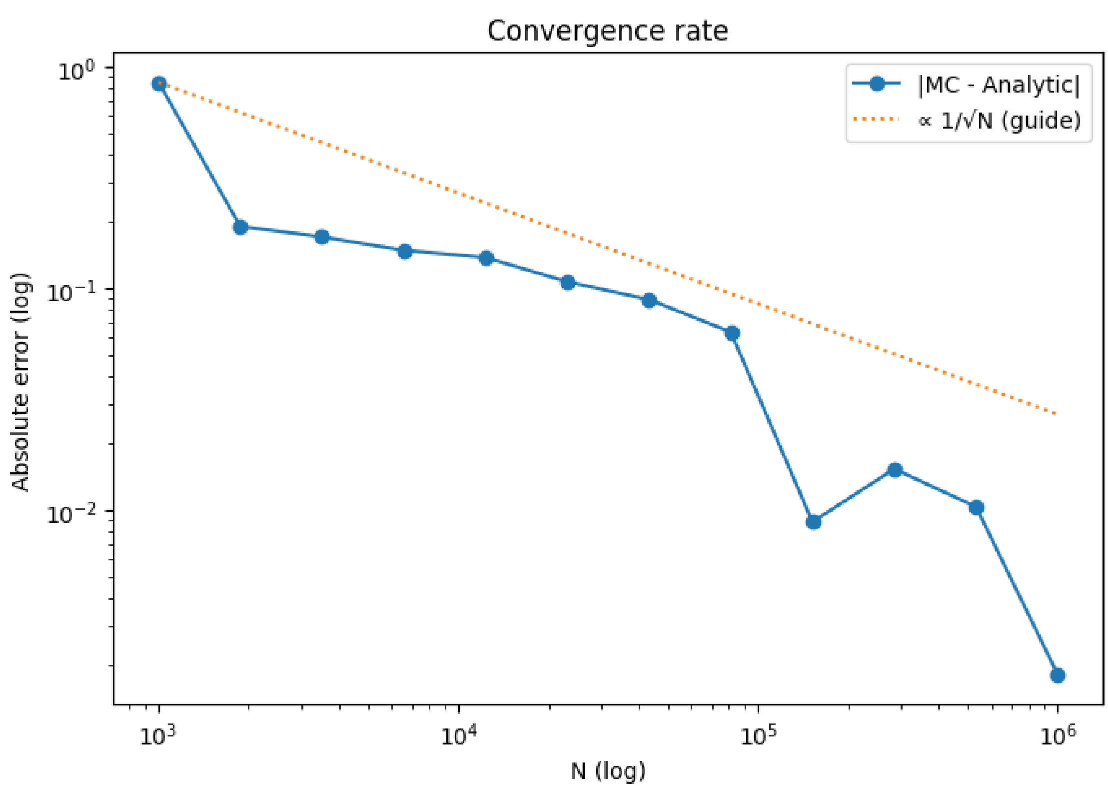
[9]:
```
Ns = np.unique(np.round(np.logspace(3, 6, 12)).astype(int))
est, se, err = [], [], []
```

```python
# fresh RNG for fair scaling across N
base_rng = np.random.default_rng(123)

for n in Ns:
    # use independent streams (split by SeedSequence)
    child_rng = np.random.default_rng(base_rng.integers(0, 2**32 - 1))
    p, s = mc_euro_call(S0, K, r, sigma, T, n=n, rng=child_rng)
    est.append(p)
    se.append(s)
    err.append(abs(p - true_call))

est, se, err = map(np.asarray, (est, se, err))
```

[10]:

```python
# %%
plt.figure(figsize=(7,5))
plt.xscale("log")
plt.plot(Ns, est, marker="o", label="MC")
plt.hlines(true_call, xmin=Ns.min(), xmax=Ns.max(), linestyles="dashed",␣
 ↪label="Analytic")
plt.fill_between(Ns, est-1.96*se, est+1.96*se, alpha=0.2, label="95% CI")
plt.xlabel("Number of paths N (log)")
plt.ylabel("Call price")
plt.xlim(Ns.min()/2, Ns.max()*2)
plt.title("MC price vs Analytic (95% CI)")
plt.legend()
plt.tight_layout()
plt.show()

# %%
plt.figure(figsize=(7,5))
plt.xscale("log"); plt.yscale("log")
plt.plot(Ns, err, marker="o", label="|MC - Analytic|")
# 1/sqrt(N) guide (scaled to first point)
guide = err[0] * math.sqrt(Ns[0]) / np.sqrt(Ns)
plt.plot(Ns, guide, linestyle="dotted", label="  1/√N (guide)")
plt.xlabel("N (log)"); plt.ylabel("Absolute error (log)")
plt.title("Convergence rate")
plt.legend()
plt.tight_layout()
plt.show()
```


MC price vs Analytic (95% CI)
MC
Analytic
95% CI
Call price
Number of paths N (log)


Convergence rate
|MC - Analytic|
∝ 1/√N (guide)
Absolute error (log)
N (log)

4.6 Distribution sanity check for S_T

Theoretical:

$$\mathbb{E}[S_T] = S_0 e^{rT}, \quad \mathrm{Var}[S_T] = S_0^2 e^{2rT}(e^{\sigma^2 T} - 1). \tag{49}$$

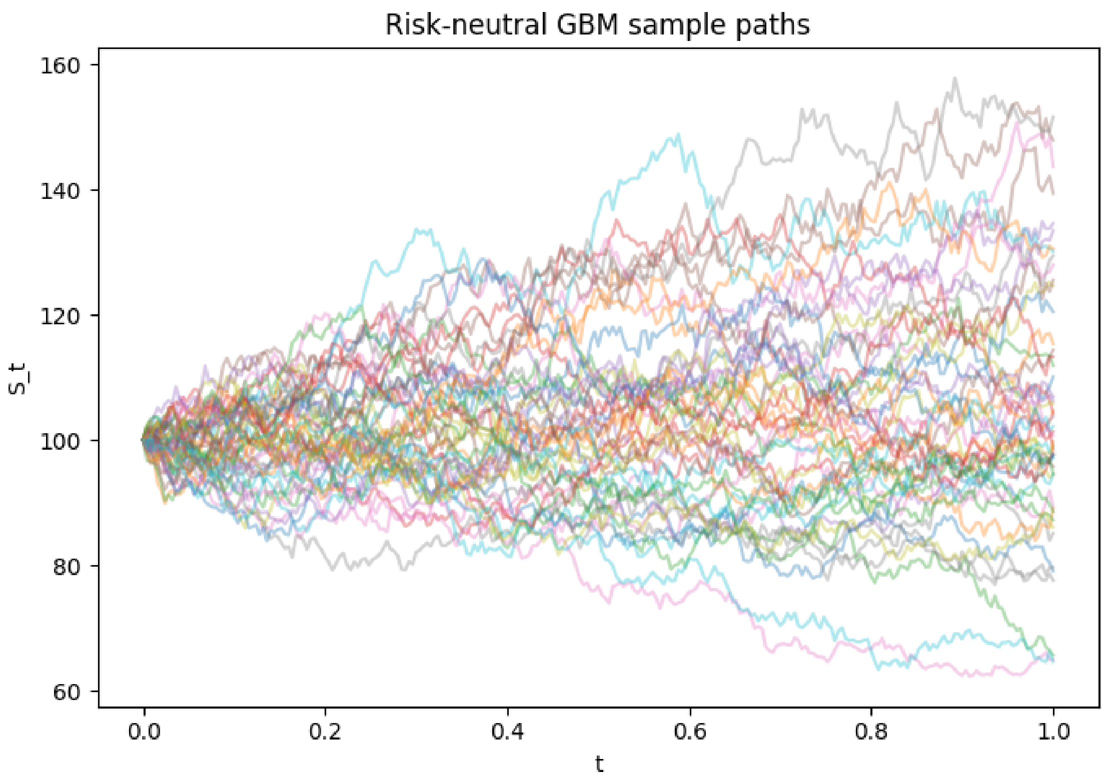
[11]:
```
m_theory, v_theory = gbm_terminal_moments(S0, r, sigma, T)

ST = sample_terminal_risk_neutral(S0, r, sigma, T, n_paths=1_000_000, rng=np.
↪random.default_rng(7))
m_emp, v_emp = ST.mean(), ST.var(ddof=1)

print(f"E[ST] theory={m_theory:.6f}, empirical={m_emp:.6f}")
print(f"Var[ST] theory={v_theory:.6f}, empirical={v_emp:.6f}")
```

```
E[ST] theory=105.127110, empirical=105.123602
Var[ST] theory=451.028808, empirical=450.675339
```

4.7 Optional: path visualization (intuition)

Just to see risk-neutral drift vs randomness.

[12]:
```
t, paths = simulate_risk_neutral_paths(S0, r, sigma, T, n_steps=250,␣
↪n_paths=50, rng=np.random.default_rng(9))
plt.figure(figsize=(7,5))
for i in range(paths.shape[0]):
    plt.plot(t, paths[i], alpha=0.35)
plt.xlabel("t"); plt.ylabel("S_t")
plt.title("Risk-neutral GBM sample paths")
plt.tight_layout()
plt.show()
```



4.8 Results table (handy if exporting)

Columns: N, MC price, SE, 95% CI, |error|

[13]:
```
df = pd.DataFrame({
    "N": Ns,
    "MC Price": est,
    "SE": se,
    "CI Lower": est - 1.96*se,
    "CI Upper": est + 1.96*se,
    "Abs Error": err,
})
df.round(6)
```

[13]:

	N	MC Price	SE	CI Lower	CI Upper	Abs Error
0	1000	11.305226	0.483816	10.356947	12.253506	0.854643
1	1874	10.259393	0.341999	9.589074	10.929712	0.191190
2	3511	10.621709	0.246756	10.138069	11.105350	0.171126
3	6579	10.301729	0.180001	9.948927	10.654532	0.148854
4	12328	10.588732	0.132434	10.329161	10.848303	0.138149
5	23101	10.343139	0.095757	10.155455	10.530823	0.107445
6	43288	10.539772	0.070836	10.400932	10.678611	0.089188

7	81113	10.514315	0.051616	10.413149	10.615482	0.063732
8	151991	10.459428	0.037716	10.385505	10.533350	0.008844
9	284804	10.465923	0.027588	10.411851	10.519995	0.015340
10	533670	10.440230	0.020145	10.400746	10.479715	0.010353
11	1000000	10.448774	0.014709	10.419946	10.477603	0.001809

[ ]:

```
! jupyter nbconvert --to pdf notebooks/demo_bs.ipynb
```

[ ]:

```
```