Hand-In Exercise #3

Continuous Time Finance 2 (FinKont2)

Yasin Baysal (cmv882)

Mathematics-Economics
University of Copenhagen
Block 3, 2025

Portfolio insurance, static hedging, stochastic volatility

We first consider a Black-Scholes model with a constant interest rate r and stock price process

$$dS(t) = \mu S(t)dt + \sigma S(t)dW^P(t),$$

where W^P is a Brownian motion under the real-world probability measure P, while Q as usual will denote the martingale measure in this model.

Question 1.a

We consider a self-financing strategy that has the fixed fraction a invested in the stock and let A_t^a denote its value process. We choose to use results from section 6.4 in Björk (2019) to express the portfolio in relative terms (portfolio weights) and then invoke the self-financing condition.

First, by Definition 6.11, the portfolio weights are defined by

$$w_t^S = \frac{h_t^S S_t}{A_t^a} \quad \text{and} \quad w_t^B = \frac{h_t^B B_t}{A_t^a},$$

where h_t^S and h_t^B are the numbers of units held in the stock and bond, and $B_t = e^{rt}$ is the bank account. However, since the strategy invests a fixed fraction a in the stock, we can simply write

$$w_t^S = a \quad \text{and} \quad w_t^B = 1 - a.$$

According to the self-financing condition from Lemma 6.12, the dynamics of the value process A_t^a, in the absence of any consumption and dividends, is given by

$$dA_t^a = A_t^a \left(w_t^S \frac{dS(t)}{S(t)} + w_t^B \frac{dB(t)}{B(t)} \right) = A_t^a \left(a \frac{dS(t)}{S(t)} + (1-a) \frac{dB(t)}{B(t)} \right).$$

By Definition 7.2 in Björk (2019), the dynamics of the stock S_t and risk-free asset (bank account) B_t in the Black-Scholes model under the risk-neutral measure Q are given by

$$dS(t) = rS(t)dt + \sigma S(t)dW^Q(t) \Leftrightarrow \frac{dS(t)}{S(t)} = rdt + \sigma dW^Q(t)$$

$$dB(t) = rB(t)dt \Leftrightarrow \frac{dB(t)}{B(t)} = rdt,$$

where r is the constant interest rate r, σ the volatility and W_t^Q a standard Brownian motion under Q. We can then plug these dynamics into the self-financing condition, such that

$$\begin{aligned}
dA_t^a &= A_t^a\left(a\frac{dS_t}{S_t} + (1-a)\frac{dB_t}{B_t}\right) \\
&= A_t^a\left(a\left(rdt + \sigma dW_t^Q\right) + (1-a)rdt\right) \\
&= A_t^a\left(ardt + a\sigma dW_t^Q + (1-a)rdt\right) \\
&= A_t^a\left(ardt + a\sigma dW_t^Q + rdt - ardt\right) \\
&= A_t^a\left(rdt + a\sigma dW_t^Q\right) \\
&= A_t^a rdt + A_t^a a\sigma dW_t^Q
\end{aligned}$$

which is the dynamics of the value process A_t^a. By equation (5.10) in Björk (2019), we recognize this as a geometric Brownian motion (GBM) with drift rate r and volatility $a\sigma$. The explicit solution to this GBM follows directly from Proposition 5.2 in Björk (2019), which states that

$$A_t^a = A_0^a \exp\left\{\left(r - \frac{(a\sigma)^2}{2}\right)t + a\sigma W_t^Q\right\}.$$

Next, note that the stock price under Q in the Black-Scholes model raised to the power a is

$$S_t = S_0 \exp\left\{\left(r - \frac{\sigma^2}{2}\right)t + \sigma W^Q(t)\right\} \Leftrightarrow S_t^a = S_0^a \exp\left\{a\left(r - \frac{\sigma^2}{2}\right)t + a\sigma W_t^Q\right\}.$$

If we then compare this with the solution for A_t^a (i.e., form the ratio A_t^a/S_t^a), we get that

$$\begin{aligned}
\frac{A_t^a}{S_t^a} &= \frac{A_0^a}{S_0^a}\exp\left\{\left(r - \frac{(a\sigma)^2}{2}\right)t + a\sigma W_t^Q - \left(a\left(r - \frac{\sigma^2}{2}\right)t + a\sigma W_t^Q\right)\right\} \\
&= \frac{A_0^a}{S_0^a}\exp\left\{\left(r - \frac{(a\sigma)^2}{2}\right)t + a\sigma W_t^Q - a\left(r - \frac{\sigma^2}{2}\right)t - a\sigma W_t^Q\right\} \\
&= \frac{A_0^a}{S_0^a}\exp\left\{\left(r - \frac{(a\sigma)^2}{2}\right)t - a\left(r - \frac{\sigma^2}{2}\right)t\right\} \\
&= \frac{A_0^a}{S_0^a}\exp\left\{\left(\left(r - \frac{(a\sigma)^2}{2}\right) - a\left(r - \frac{\sigma^2}{2}\right)\right)t\right\} \\
&= \frac{A_0^a}{S_0^a}\exp\left\{\left(r - \frac{1}{2}a^2\sigma^2 - ar + \frac{\sigma^2}{2}\right)t\right\} \\
&= \frac{A_0^a}{S_0^a}\exp\left\{\left((1-a)r + \frac{1}{2}a\sigma^2(1-a)\right)t\right\} \\
&= \frac{A_0^a}{S_0^a}\exp\left\{(1-a)\left(r + \frac{1}{2}a\sigma^2\right)t\right\}.
\end{aligned}$$

Thus, we define the deterministic function g by

$$g(t) = \frac{A_0^a}{S_0^a} \exp\left\{(1-a)\left(r + \frac{1}{2}a\sigma^2\right)t\right\},$$

which depends on time and model parameters. Using this function, we have that $A_t^a = g(t)S_t^a$. To verify this explicitly, we substitute the expression for S_t^a and combine the terms, such that

$$\begin{aligned}
g(t)S_t^a &= \frac{A_0^a}{S_0^a} \exp\left\{(1-a)\left(r + \frac{1}{2}a\sigma^2\right)t\right\} S_0^a \exp\left\{a\left(r - \frac{\sigma^2}{2}\right)t + a\sigma W_t^Q\right\} \\
&= A_0^a \exp\left\{(1-a)\left(r + \frac{1}{2}a\sigma^2\right)t + a\left(r - \frac{\sigma^2}{2}\right)t + a\sigma W_t^Q\right\} \\
&= A_0^a \exp\left\{(1-a)tr + \frac{1}{2}a\sigma^2(1-a)t + art - \frac{1}{2}a\sigma^2 t + a\sigma W_t^Q\right\} \\
&= A_0^a \exp\left\{rt - \frac{1}{2}a^2\sigma^2 t + a\sigma W_t^Q\right\} \\
&= A_0^a \exp\left\{\left(r - \frac{1}{2}a^2\sigma^2\right)t + a\sigma W_t^Q\right\} \\
&= A_0^a \exp\left\{\left(r - \frac{(a\sigma)^2}{2}\right)t + a\sigma W_t^Q\right\} \\
&= A_t^a
\end{aligned}$$

as desired under Q. Lastly, note that under the real-world measure P, the stock dynamics are

$$\frac{dS_t}{S_t} = \mu dt + \sigma dW_t^P$$

while the bank account still is $\frac{dB_t}{B_t} = rdt$. Then, by the self-financing condition (Lemma 6.12),

$$\begin{aligned}
dA_t^a &= A_t^a \left(a\frac{dS_t}{S_t} + (1-a)\frac{dB_t}{B_t}\right) \\
&= A_t^a \left(a\left(\mu dt + \sigma dW_t^P\right) + (1-a)rdt\right) \\
&= A_t^a \left(\left(a\mu + (1-a)r\right)dt + a\sigma dW_t^P\right).
\end{aligned}$$

Thus, under P, A_t^a is still a GBM with drift $a\mu + (1-a)r$ and volatility $a\sigma$ with explicit solution

$$A_t^a = A_0^a \exp\left\{\left(a\mu + (1-a)r - \frac{(a\sigma)^2}{2}\right)t + a\sigma\, W_t^P\right\},$$

where it can be shown that the function $g(t)$ turns out to be identical in form to the one we obtained under Q, because when we take ratio A_t^a/S_t^a, then we get that

$$\begin{aligned}
\frac{A_t^a}{S_t^a} &= \frac{A_0^a}{S_0^a}\exp\left\{\left(a\mu + (1-a)r - \frac{(a\sigma)^2}{2} - a\left(\mu - \frac{\sigma^2}{2}\right)\right)t\right\} \\
&= \frac{A_0^a}{S_0^a}\exp\left\{\left((1-a)r + \frac{1}{2}a\sigma^2(1-a)\right)t\right\} \\
&= \frac{A_0^a}{S_0^a}\exp\left\{(1-a)\left(r + \frac{1}{2}a\sigma^2\right)t\right\},
\end{aligned}$$

and by choosing this as the deterministic function $g(t)$, we also get $A_t^a = g(t)S_t^a$ under P.

Question 1.b

Now, a pension company invests its client savings according to the fixed fraction strategy described in the previous question over the interval $[0, T]$. The company is considering offering portfolio insurance to its clients through a contract that pays $(K - A_T^a)^+$ at T for some K.

To find out what would it cost for the pension company to replicate a portfolio insurance contract, we first recall from Question 1.a that A_t^a under the risk-neutral measure Q is given by

$$A_t^a = A_0^a e^{\left(r - \frac{(a\sigma)^2}{2}\right)t + a\sigma W_t^Q}$$

In other words, A_t^a follows a GBM with drift r and volatility $a\sigma$. Thus, at time T we have

$$A_T^a = A_0^a e^{\left(r - \frac{(a\sigma)^2}{2}\right)T + a\sigma W_T^Q}$$

Since the pension company wishes to offer a portfolio insurance contract $\mathcal{X}$ that pays $\mathcal{X} = (K - A_T^a)^+$, the cost for replicating this claim is determined by the unique arbitrage-free (or replicating) price at time 0; otherwise, an arbitrage possibility would exist. By Proposition 7.11 with the risk-neutral valuation formula in Björk (2019), the arbitrage-free time-t of $\mathcal{X}$ is

$$\Pi_t(\mathcal{X}) = e^{-r(T-t)}\mathbb{E}_t^Q[(K - A_t^a)^+].$$

But, since A_T^a is lognormally distributed under Q, this expectation can be seen as the price of a European put option on the value process A_T^a with strike K, maturity T, risk-free interest rate r and volatility $a\sigma$. From Proposition 10.2 in Björk (2019) with the put-call parity, for a put

$P_{BS}(t,s)$ and call $C_{BS}(t,s)$ on the same underlying s with strike K and maturity T at time t,

$$P_{BS}(t,s) = Ke^{-r(T-t)} - s + C_{BS}(t,s)$$

Also, from Proposition 7.13 in Björk (2019), the call price under the Black–Scholes model at time t is given by

$$C_{BS}(t,s) = s\Phi[d_1(t,s)] - Ke^{-r(T-t)}\Phi[d_2(t,s)],$$

where $\Phi(\cdot)$ denotes the cumulative distribution function for the standard normal distribution $\mathcal{N}(0,1)$. So, by first plugging $C_{BS}(t,s)$ into the put–call parity, we obtain that

$$\begin{aligned}
P_{BS}(t,s) &= Ke^{-r(T-t)} - s + C_{BS}(t,s) \\
&= Ke^{-r(T-t)} - s + \Big[s\Phi(d_1(t,s)) - Ke^{-r(T-t)}\Phi(d_2(t,s))\Big] \\
&= Ke^{-r(T-t)}[1-\Phi(d_2(t,s))] - s[1-\Phi(d_1(t,s))].
\end{aligned}$$

Recalling that $1-\Phi(x) = \Phi(-x)$ (symmetry of the normal distribution), we get $1-\Phi(d_2(t,s)) = \Phi(-d_2(t,s))$ and $1-\Phi(d_1(t,s)) = \Phi(-d_1(t,s))$. Applying these identities, we write the price as

$$P_{BS}(t,s) = Ke^{-r(T-t)}\Phi(-d_2(t,s)) - s\Phi(-d_1(t,s)).$$

So, by using this Black–Scholes formula for a European put in our case, we obtain

$$\Pi_t(\mathcal{X}) = Ke^{-r(T-t)}\Phi(-d_2(t,A_t^a)) - A_t^a\Phi(-d_1(t,A_t^a))$$

as the arbitrage-free price of the portfolio insurance contract $\mathcal{X}$ at time t, where

$$\begin{aligned}
d_1(t,A_t^a) &= \frac{1}{a\sigma\sqrt{T-t}}\left\{\log\left(\frac{A_t^a}{K}\right) + \left(r + \frac{1}{2}(a\sigma)^2\right)(T-t)\right\} \\
d_2(t,A_t^a) &= d_1(t,A_t^a) - a\sigma\sqrt{T-t}.
\end{aligned}$$

Next, we have to find out how the contract can be replicated by trading dynamically in the stock and the risk-free asset. First, we compute the delta of the contract $\Delta_{\mathcal{X}}$. Since $\Delta_{\mathcal{X}}$ is the derivative of the contract value with respect to S cf. Defintion 10.4 in Björk (2019), we have

$$\Delta_{\mathcal{X}} = \frac{\partial \Pi_t(\mathcal{X})}{\partial S} = \frac{\partial \Pi_t(\mathcal{X})}{\partial A}\frac{\partial A}{\partial S},$$

and by using the price of the insurance contract and definition of A_t^a, we obtain

$$\begin{aligned}\Delta_{\mathcal{X}} = \frac{\partial \Pi_t(\mathcal{X})}{\partial A}\frac{\partial A}{\partial S} &= -\Phi(-d_1(t, A_t^a))g(t)aS_t^{a-1} \\ &= -\Phi(-d_1(t, A_t^a))a\frac{g(t)S_t^a}{S_t} \\ &= -\Phi(-d_1(t, A_t^a))a\frac{A_t^a}{S_t}.\end{aligned}$$

To replicate the contract, we employ a dynamic delta-hedging strategy using the underlying stock. First, the time horizon from 0 to T is divided into discrete intervals, and multiple paths of the stock price S_t – and thus A_t^a – are simulated at each step. At the initial time $t = 0$, the contract's theoretical price is used to construct the hedging portfolio by purchasing Δ shares of the stock and depositing any leftover capital into in the risk-free bank account. As time progresses, the portfolio is rebalanced at each step so that the holding in the underlying stock matches the updated delta. This procedure is repeated until the final date T (maturity), when the portfolio is unwound. The hedge error is then calculated as the difference between the terminal value of the hedging portfolio and the contract's payoff at maturity.

For our simulations, we use the parameter values specified in the exercise text, which are:

Parameter	S_0	A_0^a	T	K	r	a	σ	μ	N	n
Value	1	1	30	$e^{rT} = e^{0.02\cdot 30} = e^{0.6}$	0.02	0.5	0.20	0.02	1000	7560

Table 1: Parameter values used for the simulations in the discrete hedge experiment.

We set the time horizon to $T = 30$ years and assume 252 trading days per year. Thus, selecting $n = 30 \cdot 252 = 7560$ corresponds to daily rebalancing over the entire 30-year period. In the top plot of Figure X, we show the terminal value of the hedging portfolio as a function of the stock price at time T. Each dot represents a single simulated path from our experiment, while the black curve depicts the payoff of the insurance contract for comparison. We refer to Appendix 1 for the full implementation, including the code used to generate the forthcoming plots.

In the top plot of Figure 1, the hedge portfolio (red dots) closely aligns with the insurance contract payoff (black line) across the range of final portfolio values A_T^a. This shows that the replication strategy performs extremely well over a wide range of outcomes. The histogram of final hedge errors (bottom-left plot) is tight around zero, confirming that the discrete delta hedging produces negligible replication error. Also, the bottom-right plot shows that the standard deviation of the discounted error converges to zero as the number of hedge points increases.

Thus, as $n \to \infty$, the strategy perfectly replicates the payoff of the portfolio insurance contract.

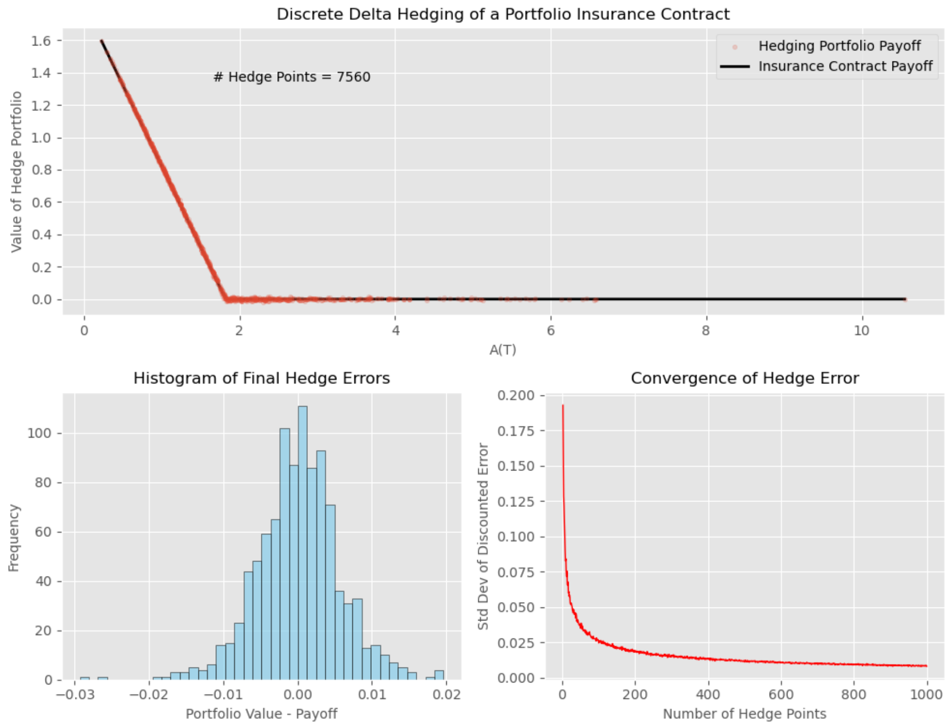

Figure 1: Discrete hedging experiment of a portfolio insurance option with 7560 hedge points (daily hedging over $T \in [0, 30]$), showing the hedge portfolio vs. portfolio insurance put payoff (top), distribution of hedge errors (bottom-left), and convergence of the discounted error (bottom-right).

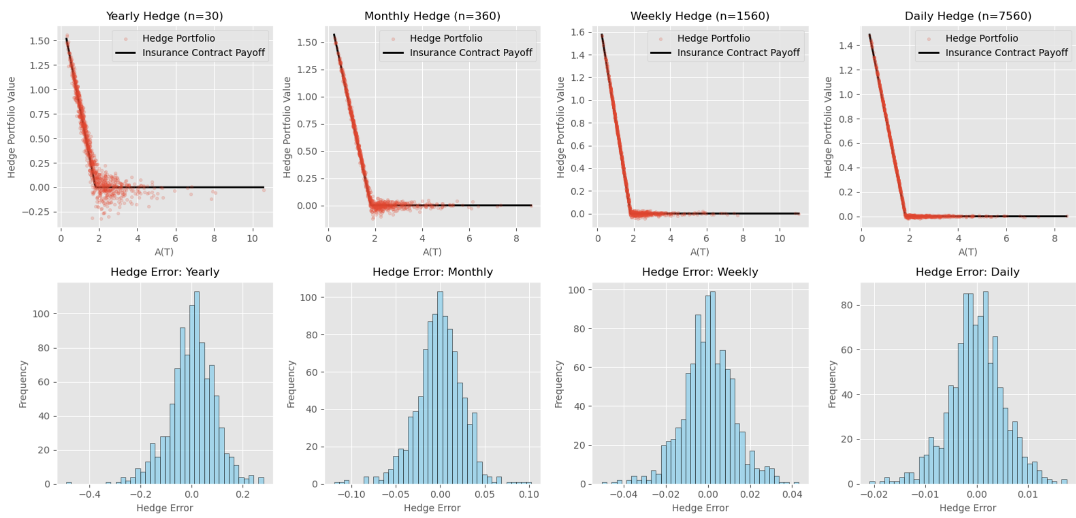

Figure 2: Discrete delta hedging outcomes under different hedging frequencies over a 30-year horizon.

Additionally, we have performed the same discrete hedging experiment for four different hedging frequencies — yearly ($n = 30$), monthly ($n = 360$), weekly ($n = 1560$), and daily ($n = 7560$). The top row in Figure 2 above shows the final hedge portfolio values (red dots) plotted against the simulated final portfolio values A_t^a, with the insurance contract payoff indicated by the black line. The bottom row displays histograms of the final hedge errors for each hedging frequency. As the hedging frequency increases, the hedge portfolio payoff aligns more closely with the insurance payoff, and the hedge errors cluster more tightly around zero.

Question 1.c

We have to compute (i.e., give formulas) for three different quantities, where the first one is

$$e^{-rT}\mathbb{E}^P((K - A_T^a)^+).$$

Since A_t^a cf. Question 1.a is a geometric Brownian motion under both the P and Q measures, then we can write the value process at maturity T under both measures as

$$\begin{aligned} &P\text{-measure:} \quad A_T^a = A_0^a e^{\left(a(\mu-r)+r-\frac{(a\sigma)^2}{2}\right)T+a\sigma W_T^P} \\ &Q\text{-measure:} \quad A_T^a = A_0^a e^{\left(r-\frac{(a\sigma)^2}{2}\right)T+a\sigma W_T^Q}, \end{aligned}$$

and here we note that the only difference is $\mu = r$ under the Q-measure.

To compute the first quantity above under the P-measure, we first rewrite the expectation as

$$\mathbb{E}^P((K - A_T^a)^+) = \mathbb{E}^P((K - A_T^a)1_{\{K>A_T^a\}}) = K\mathbb{E}^P(1_{\{K>A_T^a\}}) - \mathbb{E}^P(A_T^a 1_{\{K>A_T^a\}}).$$

Then, applying the standard result that the expectation of an indicator function equals the probability of the corresponding event, we can express the expectation as a P-probability by

$$K\mathbb{E}^P(1_{\{K>A_T^a\}}) = KP(K > A_T^a) = KP(\log(K) > \log(A_T^a)).$$

Next, we substitute in the expression for $\log(A_T^a)$ under the P-measure, using the geometric Brownian motion representation given by

$$\log(A_T^a) = \log(A_0^a) + \left(a(\mu - r) + r - \frac{1}{2}(a\sigma)^2\right)T + a\sigma W_T^P,$$

such that we get

$$
\begin{aligned}
K\mathbb{E}^P(1_{\{K>A_T^a\}}) &= KP(\log(K) > \log(A_T^a)) \\
&= KP\left(\frac{\log(K) - \log(A_0^a) - \left(a(\mu - r) + r - \frac{1}{2}(a\sigma)^2\right)T}{a\sigma\sqrt{T}} > Y\right), \quad Y \sim \mathcal{N}(0,1) \\
&= KP\left(\frac{\log(K) - \log(A_0^a) - \left(a(\mu - r) + r - \frac{1}{2}(a\sigma)^2\right)T}{a\sigma\sqrt{T}}\right) \\
&= K\Phi\left(\frac{\log(K) - \log(A_0^a) - \left(a(\mu - r) + r - \frac{1}{2}(a\sigma)^2\right)T}{a\sigma\sqrt{T}}\right) \\
&= K\Phi(-d_2), \quad \text{where} \quad d_2 = \frac{\log\left(\frac{A_0^a}{K}\right) + \left(a(\mu - r) + r - \frac{1}{2}(a\sigma)^2\right)T}{a\sigma\sqrt{T}},
\end{aligned}
$$

and where $\Phi(\cdot)$ denotes the cumulative distribution function for the standard normal distribution $\mathcal{N}(0,1)$. To then compute the second term $\mathbb{E}^P(A_T^a 1_{\{K>A_T^a\}})$, we use this rule from MathFin: Let $Y \sim \mathcal{N}(0,1)$ be a random variable, and let λ and c be real constants. Then, we have that

$$
\mathbb{E}(e^{\lambda Y} 1_{\{Y<c\}}) = e^{\lambda^2/2}\Phi(c - \lambda).
$$

So, by applying this rule to the second term, we obtain

$$
\begin{aligned}
\mathbb{E}^P(A_T^a 1_{\{K>A_T^a\}}) &= \mathbb{E}^P\left(A_0^a e^{\left(a(\mu-r)+r-\frac{(a\sigma)^2}{2}\right)T + a\sigma W_T^P} 1_{\{K>A_T^a\}}\right) \\
&= A_0^a e^{\left(a(\mu-r)+r-\frac{(a\sigma)^2}{2}\right)T} \mathbb{E}^P(e^{a\sigma\sqrt{T}Y} 1_{\{-d_2>Y\}}),
\end{aligned}
$$

where we have used that $W_T^a \sim \mathcal{N}(0,T) \Rightarrow W_T^P = \sqrt{T}Y$ for $Y \sim \mathcal{N}(0,1)$ and also the same transformation that we did for $K\mathbb{E}^P(1_{K>A_T^a})$ above. By then recognizing $c = -d_2$ and $\lambda = a\sigma\sqrt{T}$, we use the above stated rule from MathFin, which gives us that

$$
\begin{aligned}
\mathbb{E}^P(A_T^a 1_{\{K>A_T^a\}}) &= A_0^a e^{\left(a(\mu-r)+r-\frac{(a\sigma)^2}{2}\right)T} e^{\frac{(a\sigma)^2 T}{2}} \Phi(-d_2 - a\sigma\sqrt{T}) \\
&= A_0^a e^{(a(\mu-r)+r)T} \Phi(-d_1),
\end{aligned}
$$

where

$$
d_1 = \frac{\log\left(\frac{A_0^a}{K}\right) + \left(a(\mu - r) + r - \frac{1}{2}(a\sigma)^2\right)T}{a\sigma\sqrt{T}},
$$

since $d_1 = d_2 + a\sigma\sqrt{T} \Leftrightarrow -d_1 = -d_2 - a\sigma\sqrt{T}$. Then, we finally get the first quantity by

$$\begin{aligned}
e^{-rT}\mathbb{E}^P((K - A_T^a)^+) &= e^{-rT}(K\mathbb{E}^P(1_{\{K>A_T^a\}}) - \mathbb{E}^P(A_T^a 1_{\{K>A_T^a\}})) \\
&= e^{-rT}\left(K\Phi(-d_2) - A_0^a e^{(a(\mu-r)+r)T}\Phi(-d_1)\right) \\
&= e^{-rT}K\Phi(-d_2) - A_0^a e^{(a\mu-ar)T}\Phi(-d_1).
\end{aligned}$$

For the next quantity $e^{-rT}\mathbb{E}^Q((K - S_T^a)^+)$, we first note that S_t is a geometric Brownian motion under the Q-measure, and the explicit solution to this GBM follows directly from Proposition 5.2 in Björk (2019), which states that

$$S_T = S_0 e^{\left(r-\frac{\sigma^2}{2}\right)T + \sigma W_T^Q} \Leftrightarrow S_T^a = S_0^a e^{a\left(r-\frac{\sigma^2}{2}\right)T + a\sigma W_T^Q}.$$

Then, by splitting the expectation using indicator functions, we get

$$\mathbb{E}^Q((K - S_T^a)^+) = K\mathbb{E}^Q(1_{\{K>S_T^a\}}) - \mathbb{E}^Q(S_T^a 1_{\{K>S_T^a\}}),$$

and for the first expectation, we use the same approach as before, such that

$$\begin{aligned}
K\mathbb{E}^P(1_{\{K>S_T^a\}}) &= KQ(K > S_T^a) \\
&= KQ(\log(K) > \log(S_T^a)) \\
&= K\Phi\left(\frac{\log\left(\frac{K}{S_0^a}\right) - a\left(r - \frac{\sigma^2}{2}\right)T}{a\sigma\sqrt{T}}\right) \\
&= K\Phi(-d_2), \quad \text{where} \quad d_2 = \frac{\log\left(\frac{S_0^a}{K}\right) + a\left(r - \frac{\sigma^2}{2}\right)T}{a\sigma\sqrt{T}}.
\end{aligned}$$

By applying the same rule from MathFin to the second expectation, it gives us that

$$\begin{aligned}
\mathbb{E}^Q(S_T^a 1_{\{K>S_T^a\}}) &= S_0^a e^{a\left(r-\frac{\sigma^2}{2}\right)T}\mathbb{E}^Q\left(e^{a\sigma\sqrt{T}Y} 1_{\{Y<(\log(K/S_0^a)-a(r-\sigma^2/2)T)/a\sigma\sqrt{T}\}}\right) \\
&= S_0^a e^{a\left(r-\frac{1}{2}\sigma^2\right)T}\mathbb{E}^Q\left(e^{a\sigma\sqrt{T}Y} 1_{\{Y<-d_2\}}\right).
\end{aligned}$$

By then again recognizing $c = -d_2$ and $\lambda = a\sigma\sqrt{T}$, we obtain

$$\begin{aligned}
\mathbb{E}^Q\left(S_T^a 1_{\{K>S_T^a\}}\right) &= S_0^a e^{a\left(r-\frac{\sigma^2}{2}\right)T} e^{\frac{(a\sigma)^2}{2}T}\Phi(-d_2 - a\sigma\sqrt{T}) \\
&= S_0^a e^{a\left(r-\frac{\sigma^2}{2}+\frac{a\sigma^2}{2}\right)T}\Phi(-d_1), \quad \text{where} \quad d_1 = d_2 + a\sigma\sqrt{T}.
\end{aligned}$$

since again $d_1 = d_2 + a\sigma\sqrt{T} \Leftrightarrow -d_1 = -d_2 - a\sigma\sqrt{T}$. Then, we get the second quantity by

$$\begin{aligned}
e^{-rT}\mathbb{E}^Q((K - S_T^a)^+) &= e^{-rT}(K\mathbb{E}^Q(1_{\{K>S_T^a\}}) - \mathbb{E}^Q(S_T^a 1_{\{K>S_T^a\}})) \\
&= e^{-rT}\left(K\Phi(-d_2) - S_0^a e^{a\left(r-\frac{\sigma^2}{2}+\frac{a\sigma^2}{2}\right)T}\Phi(-d_1)\right) \\
&= e^{-rT}K\Phi(-d_2) - S_0^a e^{\left(-r+ar-\frac{a\sigma^2}{2}+\frac{(a\sigma)^2}{2}\right)T}\Phi(-d_1).
\end{aligned}$$

We now observe that the final quantity $ae^{-rT}\mathbb{E}^Q((K - S_T)^+)$ corresponds to the time-zero price of a European put option on the underlying asset S_T, scaled by the factor a. Applying the Black–Scholes pricing formula from Proposition 7.13 in Björk (2019) and using the put–call parity relationship from Proposition 10.2 in Björk (2019), this can be written as

$$ae^{-rT}\mathbb{E}^Q\left[(K - S_T)^+\right] = aKe^{-rT}\Phi(-d_2) - aS_0\Phi(-d_1)$$

where the terms d_1 and d_2 are given by

$$d_1 = \frac{\log\left(\frac{S_0}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)T}{\sigma\sqrt{T}}, \quad \text{and} \quad d_2 = d_1 - \sigma\sqrt{T}.$$

We now compare the replication price of the portfolio insurance contract from Question 1.b to three related pricing expressions introduced here. These prices differ in either the probability measure used or choice of underlying. The parameter values used for numerical calculations are:

Parameter	S_0	A_0^a	T	K	r	a	σ	μ
Value	1	1	30	$e^{rT} = e^{0.02\cdot 30} = e^{0.6}$	0.02	0.5	0.20	0.07

Table 2: Parameter values used for the simulations in the discrete hedge experiment.

Using these, we compute the replication price and three quantities (see Appendix 2). The table below summarizes the values with a brief comparison of each with the replication price:

Quantity		**Value**	**Comparison**
Replication Price	(1b)	**0.2158**	✓ Baseline (correct price)
$e^{-rT}\mathbb{E}^P((K - A_T^a)^+)$	(1c.1)	0.0304	↓ Much lower: higher drift under $\mu > r$
$e^{-rT}\mathbb{E}^Q((K - (S_T)^a)^+)$	(1c.2)	0.4119	↑ Much higher: wrong underlying
$ae^{-rT}\mathbb{E}^Q((K - S_T)^+)$	(1c.3)	0.2081	↓ Slightly lower: fails to capture rebalancing

Table 3: Comparison of replication price from Question 1.b with the three quantities in 1.c.

The replication price from Question 1.b is the correct, arbitrage-free price for the portfolio insurance contract based on the actual value process A^a_T. The P-measure price (1c.1) underestimates the cost, because it uses the real-world drift $0.07 = \mu > r = 0.02$, thus expecting higher returns and lower put payoff likelihood. The other two quantities, (1c.2) and (1c.3), deviate because they are based on different underlyings that do not accurately reflect the payoff structure of the insurance contract. Specifically, (1c.2) overestimates the price by using S^a_T instead of A^a_T. While S^a_T may seem related, it ignores the continuous rebalancing that defines the fixed-fraction strategy, and its greater volatility leads to a higher option value. In contrast, (1c.3) uses the correct risk-neutral measure but applies a simple scaling to a standard Black–Scholes put on S_T. Although this results in a value close to the correct price, it slightly underestimates the true cost, as it also fails to account for the dynamic nature of the portfolio strategy.

Question 1.d

We have to prove equation (1) – the so-called spanning formula – from Carr and Madan (2001).

The equation implies that any twice continuously differentiable function $f(S)$ of the terminal stock price S can be replicated by a uniquely defined initial portfolio. This portfolio consists of $f(S_0) - f'(S_0)S_0$ units of discount bonds, $f'(S_0)$ shares of the stock, and a continuum of out-of-the-money options for all strikes K, each weighted by $f''(K)dK$, i.e., $f(S)$ is given by

$$f(S) = [f(S_0) - f'(S_0)S_0] + f'(S_0)S + \int_0^{S_0} f''(K)(K-S)^+ dK + \int_{S_0}^{\infty} f''(K)(S-K)^+ dK,$$

where $(K-S)^+$ and $(S-K)^+$ are the payoffs of a put and call option at strike K, respectively.

First, let f' be a real-valued function on the closed interval $[F, S]$, and let f be the antiderivative of f', a continuous function on $[F, S]$. Then, by the Fundamental Theorem of Calculus[1],

$$f(S) - f(F) = \int_F^S f'(u)\mathrm{d}u \Leftrightarrow f(S) = f(F) + \int_F^S f'(u)\mathrm{d}u. \qquad (*)$$

Note that the integral can be broken into two distinct parts for $S > F$ and $S < F$, such that

$$\begin{aligned}\int_F^S f'(u)\mathrm{d}u &= 1_{\{S>F\}} \int_F^S f'(u)\mathrm{d}u + 1_{\{S<F\}} \int_F^S f'(u)\mathrm{d}u \\ &= 1_{\{S>F\}} \int_F^S f'(u)\mathrm{d}u - 1_{\{S<F\}} \int_S^F f'(u)\mathrm{d}u,\end{aligned}$$

[1] See e.g. Theorem E.18 in Schilling (2005).

where we used the general rule for reversing the limits of integration, i.e., $\int_a^b f(x)\mathrm{d}x = -\int_b^a f(x)\mathrm{d}x$. Next, we reapply the Fundamental Theorem of Calculus to the integrals above, such that

$$f'(u) - f'(F) = \int_F^u f''(v)\mathrm{d}v \Leftrightarrow f'(u) = f'(F) + \int_F^u f''(v)\mathrm{d}v$$

for the first integral, while we for the second get that

$$f'(u) - f'(K) = -\int_u^F f''(v)\mathrm{d}v \Leftrightarrow f'(u) = f'(F) - \int_u^F f''(v)\mathrm{d}v,$$

and then the expression for $\int_S^F f'(u)\mathrm{d}u$ from before can be rewritten as

$$\int_F^S f'(u)\mathrm{d}u = 1_{\{S>F\}} \int_F^S \left[f'(F) + \int_F^u f''(v)\mathrm{d}v \right] \mathrm{d}u - 1_{\{S<F\}} \int_S^F \left[f'(F) - \int_u^F f''(v)\mathrm{d}v \right] \mathrm{d}u.$$

By rearranging and splitting these two integrals, we obtain two pair of integrals given by

$$\begin{aligned}\int_F^S f'(u)\mathrm{d}u =& 1_{\{S>F\}} \int_F^S f'(F)\mathrm{d}u - 1_{\{S<F\}} \int_S^F f'(F)\mathrm{d}u + \\ & 1_{\{S>F\}} \int_F^S \int_F^u f''(v)\mathrm{d}v\mathrm{d}u + 1_{\{S<F\}} \int_S^F \int_u^F f''(v)\mathrm{d}v\mathrm{d}u. \end{aligned} \tag{**}$$

We now consider each of the two pairs of integrals. For the first pair of integrals (i.e., the two first integrals above), we note that $f'(F)$ does not depend on u. Thus, we simply obtain

$$\begin{aligned} 1_{\{S>F\}} \int_F^S f'(F)\mathrm{d}u - 1_{\{S<F\}} \int_S^F f'(F)\mathrm{d}u &= 1_{\{S>F\}} \int_F^S f'(F)\mathrm{d}u + 1_{\{S<F\}} \int_F^S f'(F)\mathrm{d}u \\ &= \int_F^S f'(F)\mathrm{d}u \\ &= f'(F) \int_F^S \mathrm{d}u \\ &= f'(F)(S-F). \end{aligned}$$

For the next pair of integrals (i.e., the two last integrals in the equation above the previous one), we apply Fubini's theorem[2] to swap the order of integration, making sure to preserve the same domain of integration. For the first integral in the pair

$$\int_F^S \int_F^u f''(F)\mathrm{d}v\mathrm{d}u,$$

we note that it has domain of integration given by $\left\{ \substack{F<v<u,\\ F<u<S} \right\}$, which is equivalent to $\left\{ \substack{v<u<S,\\ F<v<S} \right\}$

[2]See e.g. Corollary 13.9 in Schilling (2005).

when swapping the order of integration. Hence, we can write the integral as

$$\int_F^S \int_F^u f''(F)\mathrm{d}v\mathrm{d}u = \int_F^S \int_v^S f''(v)\mathrm{d}u\mathrm{d}v = \int_F^S f''(v)(S-v)\mathrm{d}v.$$

Similarly, the other integral

$$\int_S^F \int_u^F f''(v)\mathrm{d}v\mathrm{d}u,$$

has domain of integration given by $\left\{ \substack{u<v<F,\\ S<u<F} \right\}$, which is equivalent to $\left\{ \substack{S<u<v,\\ S<v<F} \right\}$ when swapping the order. Hence, we can – by performing the integral over u like before – write the integral as

$$\int_S^F \int_u^F f''(v)\mathrm{d}v\mathrm{d}u = \int_S^F \int_S^v f''(v)\mathrm{d}u\mathrm{d}v = \int_S^F f''(v)(v-S)\mathrm{d}v.$$

If we then substitute these integrals back in equation (**) and then directly back in (*), we get

$$f(S) = f(F) + f'(F)(S-F) + 1_{\{S>F\}} \int_F^S f''(v)(S-v)\mathrm{d}v + 1_{\{S<F\}} \int_S^F f''(v)(v-S)\mathrm{d}v.$$

Hereafter, we observe that the indicator function $1_{\{S>F\}}$ takes the value 1 only when $S > F$ and likewise, $1_{\{S<F\}}$ is 1 only when $S < F$. Thus, in the linear terms of the integrals in the equation above, we can replace these indicator functions by applying the positive-part operator, $(\cdot)^+$. This reformulation directly yields

$$f(S) = f(F) + f'(F)(S-F) + \int_F^\infty f''(v)(S-v)^+\mathrm{d}v + \int_0^F f''(v)(v-S)^+\mathrm{d}v.$$

By setting $F = S_0$ (i.e., the initial stock price) and $v = K$ (i.e., the strike price), we get

$$\begin{aligned} f(S) &= f(S_0) + f'(S_0)(S-S_0) + \int_{S_0}^\infty f''(K)(S-K)^+\mathrm{d}K + \int_0^{S_0} f''(K)(K-S)^+\mathrm{d}K \\ &= f(S_0) + f'(S_0)S - f'(S_0)S_0 + \int_{S_0}^\infty f''(K)(S-K)^+\mathrm{d}K + \int_0^{S_0} f''(K)(K-S)^+\mathrm{d}K \\ &= [f(S_0) - f'(S_0)S_0] + f'(S_0)S + \int_{S_0}^\infty f''(K)(S-K)^+\mathrm{d}K + \int_0^{S_0} f''(K)(K-S)^+\mathrm{d}K \end{aligned}$$

which exactly matches equation (1) in Carr and Madan (2001) as desired.

Question 1.e

Now, we have to use the spanning formula (in a suitably discretized version) to calculate the composition of a (static) portfolio of put options with expiry T on the stock that replicates the portfolio insurance contract. From Question 1.d, the spanning formula is given by

$$f(S) = [f(S_0) - f'(S_0)S_0] + f'(S_0)S + \int_{S_0}^{\infty} f''(K)(S-K)^+ \mathrm{d}K + \int_0^{S_0} f''(K)(K-S)^+ \mathrm{d}K$$

for a twice continuously differentiable function $f(S)$ of the stock price S. Since we are only interested in portfolios made up of European puts, we adjust the formula to reflect that by

$$f(S) = \int_0^{\infty} f''(K)(K-S)^+ \mathrm{d}K,$$

which shows that any payoff $f(S)$ can be replicated using only put options across all strike prices.

Note that in the general decomposition from Question 1.d, the payoff was split into two parts: A put component for strike prices $K < S_0$, and a call component for strike prices $K > S_0$. However, when using exclusively puts, we must account for the entire payoff – covering both the parts that would normally be replicated by puts and those that would normally be replicated by calls. This is why the integration extends over all strike prices from 0 to ∞ now.

Referring back to Question 1.a, where we established that the value process A_t^a at maturity T is given by $A_T^a = g(T)S_T^a$, we can express the payoff of the portfolio insurance contract as

$$f(S) = (K - A_T^a)^+ = (K - g(T)S_T^a)^+,$$

which is the payoff function we are aiming to replicate. By recalling that the notation $(\cdot)^+ = \max\{x, 0\}$ denotes the positive part of x, the payoff function $f(S) = (K - g(T)S_T^a)^+$ is defined in a piecewise manner: it produces a positive payoff only when the stock price S is below a specified threshold K, and otherwise when it exceeds the threshold, it returns zero.

Therefore, our aim is to find the specific point $S^\star$ at which the payoff becomes zero. To do this, we analyze the second derivative of the payoff function, which is given by

$$f'(S) = \begin{cases} -g(T)aS^{a-1} & \text{if } S < S^\star \\ 0 & \text{if } S > S^\star. \end{cases} \Rightarrow f''(S) = \begin{cases} -g(T)a(a-1)S^{a-2} & \text{if } S < S^\star \\ 0 & \text{if } S > S^\star. \end{cases}$$

Thus, as S approaches $S^\star$ from below, $f'(S)$ tends to $-g(T)a(S^\star)^{a-1}$. As S approaches $S^\star$ from above, $f'(S)$ is identically 0. Hence, $f'(S)$ jumps at $S = S^\star$. So, the second derivative is not defined in the usual sense at $S = S^\star$, because $f'(S)$ is discontinuous there. However, within the spanning formula we handle this discontinuity using the generalized second derivative. Therefore, we introduce a Dirac delta function at $S^\star$ that is weighted by the size of the jump in $f'(S)$.

By Section 1 in Pedersen (2024), the jump size is given by

$$\begin{aligned}\Delta f'(S^\star) &= f'_+(S^\star) - f'_-(S^\star)\\ &= 0 - (-g(T)a(S^\star)^{a-1})\\ &= g(T)a(S^\star)^{a-1},\end{aligned}$$

which means that the generalized second derivative is given by

$$f''(K) = -g(T)a(a-1)K^{a-2} \cdot 1_{\{K<S^\star\}} + g(T)a(S^\star)^{\alpha-1} \cdot \delta(K - S^\star),$$

where $\delta(K - S^\star)$ denotes the Dirac delta function[3] that is given by

$$\delta(K-S^\star) = \begin{cases} \infty & \text{for } K - S^\star = 0 \Leftrightarrow K = S^\star \\ 0 & \text{elsewhere} \end{cases} \qquad \text{with} \qquad \int_{-\infty}^{\infty} \delta(K-S^\star)\, dK = 1.$$

If we then insert the expression for $f''(K)$ into the spanning formula for solely puts, we get that

$$\begin{aligned}f(S) &= \int_0^\infty f''(K)(K-S)^+ \mathrm{d}K\\ &= \int_0^\infty \left(-g(T)a(a-1)K^{a-2}1_{\{K<S^\star\}} + g(T)aK^{a-1}\delta(K-S^\star)\right)(K-S)^+ \mathrm{d}K\\ &= \int_0^\infty g(T)a(1-a)K^{a-2}1_{\{K<S^\star\}}(K-S)^+ \mathrm{d}K + \int_0^\infty g(T)aK^{a-1}\delta(K-S^\star)(K-S)^+ \mathrm{d}K.\end{aligned}$$

The indicator function in the first integral restricts the integration domain to $K < S^\star$, so that

$$\int_0^\infty g(T)a(1-a)K^{a-2}1_{\{K<S^\star\}}(K-S)^+ \mathrm{d}K = \int_0^{S^\star} g(T)a(1-a)K^{a-2}(K-S)^+ \mathrm{d}K.$$

For the second integral, we make use of the Dirac delta identity[4] (sifting property) given by

$$\int_{-\infty}^{\infty} \delta(x-x_0)f(x)\mathrm{d}x = f(x_0),$$

and when we apply that to the function $f(K) = g(T)aK^{a-1}(K-S)^+$, then we obtain

$$\int_0^\infty g(T)aK^{a-1}\delta(K-S^\star)(K-S)^+ \mathrm{d}K = g(T)a(S^\star)^{a-1}(S^\star - S)^+.$$

[3]See https://en.wikipedia.org/wiki/Dirac_delta_function.
[4]See https://en.wikipedia.org/wiki/Dirac_delta_function#Translation

By then substituting these two expressions into the equation for $f(S)$, we get

$$\begin{aligned} f(S) &= \int_0^\infty g(T)a(1-a)K^{a-2}1_{\{K<S^\star\}}(K-S)^+ \mathrm{d}K + \int_0^\infty g(T)aK^{a-1}\delta(K-S^\star)(K-S)^+ \mathrm{d}K \\ &= g(T)a(1-a)\int_0^{S^\star} K^{a-2}(K-S)^+ \mathrm{d}K + g(T)a(S^\star)^{a-1}(S^\star - S)^+, \end{aligned}$$

which is the final composition of a (static) portfolio of put options with expiry T on the stock that replicates the portfolio insurance contract.

Next, we examine how the price of the portfolio insurance contract behaves as the discretization becomes finer and finer. To do so, we numerically approximate the integral and Dirac delta terms in the static hedge representation using a uniform grid of strike prices $K \in [0, S^\star]$, where $S^\star = \left(\frac{K}{g(T)}\right)^{1/a}$. For each strike K_i, we compute the Black-Scholes price of a European put option and weight it by $g(T)\,a(1-a)\,K_i^{a-2}$. The integral is then approximated as a Riemann sum with spacing ΔK. The Dirac delta contribution at $K = S^\star$ is added separately and weighted by $g(T)\,a(S^\star)^{a-1}$. We compute the total static hedge price for varying numbers of strikes $n \in [50, 10000]$, in steps of 25, and plot the results to see how the approximation converges as the discretization becomes finer, which is seen in Figure 3 below. See Appendix 3 for code.

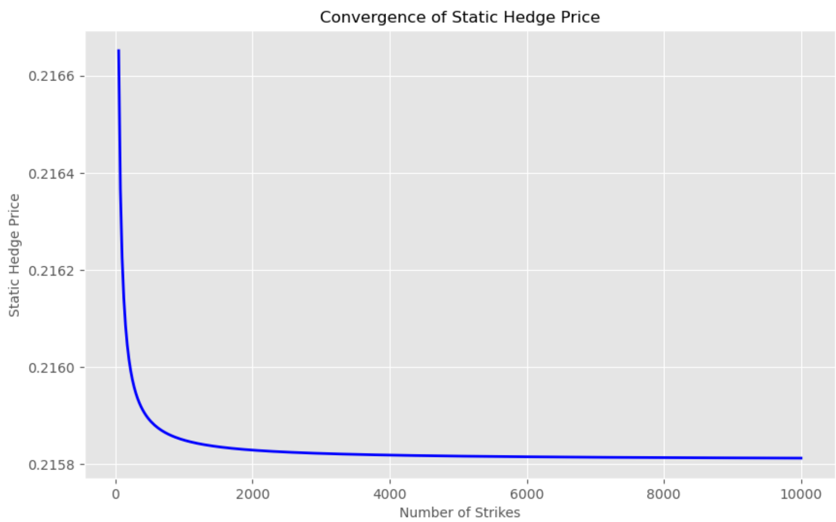

Figure 3: Convergence of the static hedge price with increasing number of strikes.

As we see, the static hedge price converges to approximately 0.2158 as the number of strikes increases. This is exactly what we expect, since in Question 1.c we calculated the arbitrage-free price of the portfolio insurance contract to be 0.2158. Thus, the discretized spanning formula using put options provides an accurate replication of the portfolio insurance payoff as desired.

Question 1.f

We now change to a market where S has Heston dynamics, which we for simplicity give directly under a martingale measure Q,

$$\begin{aligned} dS(t) &= rS(t)dt + \sqrt{V(t)}S(t)dW_1^Q(t) \\ dV(t) &= \kappa(\theta - V(t))dt + \epsilon\sqrt{V(t)}dW_2^Q(t), \end{aligned}$$

where $dW_1^Q(t)dW_2^Q(t) = \rho dt$. We first have to determine the dynamics of A_t^a (the value process), as we did earlier but now in the Heston framework.

As in Question 1.a, we thus use the self-financing condition from Lemma 6.12 in Björk (2019), where the dynamics of A_t^a, in the absence of consumption and dividends, is given by

$$dA_t^a = A_t^a\left(w_t^S\frac{dS(t)}{S(t)} + w_t^B\frac{dB(t)}{B(t)}\right) = A_t^a\left(a\frac{dS(t)}{S(t)} + (1-a)\frac{dB(t)}{B(t)}\right).$$

The dynamics $dS(t)$ of the stock $S(t)$ are given above, and we still have that $dB(t) = rB(t)dt$ from before. Thus, we now get

$$dS(t) = rS(t)dt + \sqrt{V(t)}S(t)dW_1^Q(t) \Leftrightarrow \frac{dS(t)}{S(t)} = rdt + \sqrt{V(t)}dW_1^Q(t)$$

$$dB(t) = rB(t)dt \Leftrightarrow \frac{dB(t)}{B(t)} = rdt,$$

where r is the constant interest rate r, σ the volatility and W_1^Q a standard Brownian motion under Q. We can then plug these dynamics into the self-financing condition, such that

$$\begin{aligned} dA_t^a &= A_t^a\left(a\frac{dS(t)}{S(t)} + (1-a)\frac{dB(t)}{B(t)}\right) \\ &= A_t^a\left(a\left(rdt + \sqrt{V(t)}dW_1^Q(t)\right) + (1-a)rdt\right) \\ &= A_t^a\left(ardt + a\sqrt{V(t)}dW_1^Q(t) + rdt - ardt\right) \\ &= A_t^a\left(rdt + a\sqrt{V(t)}dW_1^Q(t)\right) \\ &= A_t^a\left(rdt + \sqrt{a^2V(t)}dW_1^Q(t)\right), \end{aligned}$$

which is the dynamics of A_t^a. It is a stochastic volatility model with drift r and volatility $\sqrt{a^2V(t)}$, where $V(t)$ follows the Heston dynamics $dV(t) = \kappa(\theta - V(t))dt + \epsilon\sqrt{V(t)}dW_2^Q(t)$.

Hereafter, we have to explain how the price $e^{-rT}\mathbb{E}^Q((K - A_T^a)^+)$ can be calculated, and we remember that this – just like in Question 1.b – can be interpreted as the price of a European put option on the value process A_T^a with strike K, maturity T and risk-free interest rate r.

Even though A_t^a not an actual traded stock, under the risk-neutral measure Q its evolution mirrors that of a Heston-type asset: it grows at the risk-free rate r and exhibits stochastic volatility $\sqrt{a^2V(t)}$. So, we can treat A_t^a as the underlying in a Heston framework and employ the corresponding pricing formula for a European call option with strike K, maturity T and initial value A_0^a. Thus, by using Equation (10) in Heston (1993), we denote the call price by

$$\text{Call}(0, A_0^a, \nu, T, K) = A_0^a P_1 - Ke^{-rT}P_2$$

with $\nu = (\kappa, \theta, \varepsilon, \rho, V(0))$ being the vector of the Heston parameters. The probabilities P_1 and P_2 are by Equation (18) in Heston (1993) given by

$$P_j = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \text{Re}\left[\frac{e^{-i\phi\log K} f_j(x,\nu,\phi)}{i\phi}\right] \mathrm{d}\phi$$

for $j = 1, 2$, and the functions $f_j(x, \nu, \phi)$ are the characteristic functions that by Equation (17) in Heston (1993) given by

$$f_j(x,\nu,\phi) = \exp(C_j(\tau,\phi) + D_j(\tau,\phi)\nu + i\phi x),$$

where

$$\begin{aligned}
C_j(\tau,\phi) &= r\phi_i\tau + \frac{\kappa^*\theta^*}{\sigma^2}\left[(b_j - \rho\sigma\phi_i + d)\tau - 2\ln\left(\frac{1-ge^{d\tau}}{1-g}\right)\right] \\
D_j(\tau,\phi) &= \frac{b_j - \rho\sigma\phi_i + d}{\sigma^2}\left[\frac{1-e^{d\tau}}{1-ge^{d\tau}}\right] \\
g &= \frac{b_j - \rho\sigma\phi_i + d}{b_j - \rho\sigma\phi - d} \\
d &= \sqrt{(\rho\sigma\phi - b_j)^2 - \sigma^2(2u_j\phi_i - \phi^2)}
\end{aligned}$$

with

$$u_1 = \frac{1}{2}, \quad u_2 = -\frac{1}{2}, \quad b_1 = \kappa^* - \rho\sigma, \quad b_2 = \kappa^*.$$

Since our goal is to price the European put option with payoff $(K - A_T^a)^+$, we can obtain its price via the put-call parity from Proposition 10.2 in Björk (2019), which gives us

$$\begin{aligned} e^{-rT}\mathbb{E}^Q((K - A_T^a)^+) &= \text{Put}(0, A_0^a, \nu, T, K) \\ &= Ke^{-rT} + \text{Call}(0, A_0^a, \nu, T, K) - A_0^a \\ &= Ke^{-rT} + A_0^a P_1 - Ke^{-rT} P_2 - A_0^a. \end{aligned}$$

as the arbitrage-free price of the portfolio insurance contract under the risk-neutral measure Q.

Question 1.g

Using the result obtained in the previous Question 1.f, we have to calculate the value of the arbitrage-free price $e^{-rT}\mathbb{E}^Q((K - A_T^a)^+)$ using the following parameter values:

Parameter	**Description**	**Value**
V_0, θ	Initial variance and long-term mean of variance	0.20^2
κ	Mean reversion speed	2.0
ε	Volatility of variance (vol of vol)	1.0
ρ	Correlation (asset vs. variance)	-0.5
S_0, A_0^a	Initial spot and initial value process	1
T	Time to maturity	30
K	Strike price	e^{rT}
a	Fixed fraction invested in stock	0.5
r	Risk-free interest rate	0.02

Table 4: Parameter values used for calculations in the Heston model.

Since $K = e^{rT}$ and $A_0^a = 1$, we can insert these values in the arbitrage-free price of the portfolio insurance contract under the risk-neutral measure Q that we obtained in Question 1.f, such that

$$\begin{aligned} e^{-rT}\mathbb{E}^Q((K - A_T^a)^+) &= Ke^{-rT} + A_0^a P_1 - Ke^{-rT} P_2 - A_0^a \\ &= e^{rT}e^{-rT} + 1P_1 - e^{rT}e^{-rT}P_2 - 1 \\ &= 1 + P_1 - P_2 - 1 \\ &= P_1 - P_2 \\ &= \text{Call}(0, A_0^a, \nu, T, K) \end{aligned}$$

Thus, the price reduces to the difference $P_1 - P_2$, which, as shown in Question 1.f, is equivalent to the price of a Heston call option with interest r, maturity T, strike $K = e^{rT}$ and initial asset value $A_0^a = 1$, and where the Heston volatility parameters are given as in Table 4.

In the original Heston model presented in Question 1.f, volatility is treated as its own stochastic process, which makes it harder to price options analytically. On top of that, Heston's formula can be quite slow when dealing with larger simulation tasks. To get around this, we use a reformulation by Alexander Lipton in Lipton (2002), which rewrites the Heston pricing formula in a more practical way. With this approach, call option prices in the Heston model can be expressed using a one-dimensional integral, also making the computation faster.

By Equation (6) in Lipton (2002), we can express call option prices in the Heston model as

$$
\begin{aligned}
\mathrm{Call}^{(SV)}(0, A_0^a, \nu, T, K) =& e^{-r^1 T} S \\
&- \frac{e^{-r^0 T} K}{2\pi} \int_{-\infty}^{\infty} \frac{\exp\left[(-ik + \frac{1}{2})X + \alpha^{(SV)}(T, k) + (k^2 + \frac{1}{4})\beta^{(SV)}(T, k)\nu\right]}{k^2 + \frac{1}{4}} \mathrm{d}k
\end{aligned}
$$

at time $t = 0$, where the functions $\alpha^{(SV)}$ and $\beta^{(SV)}$ by Equation (7) in Lipton (2002) are

$$
\begin{aligned}
\alpha^{(SV)} &= -\frac{\kappa\theta}{\varepsilon^2}\left[\psi_+ \tau + 2\ln\left(\frac{\psi_- + \psi_+ e^{-\xi\tau}}{2\xi}\right)\right] \\
\beta^{(SV)} &= -\frac{1 - e^{-\xi\tau}}{\psi_- + \psi_+ e^{-\xi\tau}}
\end{aligned}
$$

with $\tau = T - t = T - 0 = T$, since we look at time 0. Furthermore, we have that

$$
\begin{aligned}
X &= \log\left(\frac{S_t}{K}\right) + r(T - t) \\
\psi_+ &= -\left(ik\rho\varepsilon + \hat{\kappa}\right) + \xi \\
\psi_- &= \left(ik\rho\varepsilon + \hat{\kappa}\right) + \xi \\
\xi &= \sqrt{k^2\varepsilon^2(1 - \rho^2) + 2ik\varepsilon\rho\hat{\kappa} + \hat{\kappa}^2 + \frac{\varepsilon^2}{4}} \\
\hat{\kappa} &= \kappa - \frac{\rho\varepsilon}{2}.
\end{aligned}
$$

From Question 1.f, we know that $dA_t^a = A_t^a\left(rdt + \sqrt{a^2 V(t)}dW_1^Q(t)\right)$, which implies that the portfolio behaves similarly to a stock under the Heston model, but with its volatility scaled by the investment fraction a. Thus, the effective variance process for the portfolio becomes

$$
\widetilde{V}(t) = a^2 V(t),
$$

which allows us to interpret A_t^a as a Heston-type asset with rescaled parameters, where

- $V_0^{\mathrm{eff}} = a^2 V(0) = 0.25 \cdot 0.04 = 0.01$

- $\theta^{\text{eff}} = a^2\theta = 0.25 \cdot 0.04 = 0.01$
- $\varepsilon^{\text{eff}} = a\varepsilon = 0.5 \cdot 1 = 0.5$

To implement this method numerically (see implementation and code in Appendix 4), we define the integrand based on the call option pricing formula. The integration is then carried out over the domain $k \in [-100, 100]$. After that, we substitute the rescaled parameters into the formula.

Recall from Question 1.f that

$$e^{-rT}\mathbb{E}^Q\left[(K - A_T^a)^+\right] = \text{Call}^{(SV)}(0, A_0^a, \nu, T, K)$$

with Heston parameters $\nu = (\kappa, \theta, \varepsilon, \rho, V_0^{\text{eff}})$, where $\text{Call}^{(SV)}$ denotes the call price under the stochastic volatility model with the rescaled parameters stated above. Using this method, and plugging in the correct parameter values for $a = 0.5$, we obtain the following numerical result for the price of the portfolio insurance contract:

$$e^{-rT}\mathbb{E}^Q\left((K - A_T^a)^+\right) \approx 0.20391.$$

Question 1.h

We consider the static put option hedge portfolio from Question 1.e and have to calculate its time 0-price assuming that put options are priced with the Q-parameters given in Question 1.f.

Recall that the spanning formula for the payoff can be written as

$$f(S_T) = g(T)a(1-a)\int_0^{S^\star} K^{a-2}(K - S_T)^+\,\mathrm{d}K \;+\; g(T)a(S^\star)^{a-1}(S^\star - S_T)^+.$$

Under Q, the time-0 price of a European put option with strike K and maturity T is $\text{Put}(S_0, K, T) = e^{-rT}\,\mathbb{E}^Q\left[(K - S_T)^+\right]$. So in our case, the time-0 price Π_0 of the replicating portfolio is given by

$$\Pi_0 = g(T)a(S^\star)^{a-1}\,\text{Put}(S_0, S^\star, T) + g(T)a(1-a)\int_0^{S^\star} K^{a-2}\,\text{Put}(S_0, K, T)\mathrm{d}K,$$

where the first term corresponds to the contribution from the Dirac delta (jump) at $K = S^\star$, and the integral represents the continuous component from the range $K \in [0, S^\star]$. We approximate the continuous integral by a Riemann sum, where we do the following (see Appendix 5 for code):

1. Partition the interval $[0, S^\star]$ into n subintervals with grid points $\{K_i\}_{i=1}^n$ and spacing ΔK.

2. For each strike K_i, compute the European put price $\mathrm{Put}(S_0, K_i, T)$ using the Heston model under the Q-measure (with parameters from Table X).

3. Weight each put price by the corresponding weight $f''(K_i)\Delta K = g(T)a(1-a)K_i^{a-2}\Delta K$.

Thus, the discretized approximation for the hedge portfolio price becomes

$$\Pi_0 \approx g(T)a(S^\star)^{a-1}\,\mathrm{Put}(S_0, S^\star, T) + g(T)a(1-a)\sum_{i=1}^{n} K_i^{a-2}\,\mathrm{Put}(S_0, K_i, T)\,\Delta K.$$

Using the numerical procedure outlined in Question 1.g, the time-0 hedge portfolio value is

$$\Pi_0 \approx 0.2078.$$

Although the spanning formula provides a theoretical decomposition of the payoff, in practice the continuous integral is approximated by a finite sum. This introduces discretization error, so while the hedge value converges very close to the arbitrage-free price as the number of strikes increases, the replication is not perfect in practice due to these numerical and practical constraints.

When $a = 1$, the entire pension portfolio is invested in the stock, so $A_T^1 = g(T)S_T$. From Question 1.a, one finds that $g(T) = 1$ whenever $a = 1$. Hence, the payoff simplifies to

$$f(S_T) = (K - A_T^1)^+ = (K - S_T)^+,$$

which is exactly the payoff of a standard European put option on S_T with strike K. Recall that

$$S^\star = \left(\frac{K}{g(T)}\right)^{\frac{1}{a}},$$

so for $a = 1$ (with $g(T) = 1$) we have $S^\star = K$. Substituting $a = 1$ into the spanning formula

$$\begin{aligned}
f(S_T) &= g(T)a(1-a)\int_0^{S^\star} K^{a-2}(K - S_T)^+\,\mathrm{d}K + g(T)a\,(S^\star)^{a-1}(S^\star - S_T)^+ \\
&\overset{a=1}{=} g(T)\cdot 1\cdot(1-1)\int_0^{S^\star} K^{1-2}(K - S_T)^+\mathrm{d}K + g(T)\cdot 1\cdot (S^\star)^0(S^\star - S_T)^+ \\
&= 0 + g(T)(S^\star - S_T)^+
\end{aligned}$$

the factor $a(1-a)$ vanishes, leaving

$$f(S_T) = g(T)(S^\star - S_T)^+ = (S^\star - S_T)^+ = \mathrm{Put}(S_T, S^\star, T).$$

Thus, when $a = 1$ the static hedge reduces to purchasing a single European put on S_T with strike $S^\star$, yielding an exact replication without the need to integrate over multiple strikes.

Question 1.i

We now assume that $a = 1$ (with all other things as before in the Heston model), such that the portfolio insurance contract is just a put option on the stock. We then first have to find the Δ of the put price. By Definition 10.4 in Björk (2019), using the results from Question 1.g, the delta is the derivative of the price w.r.t. the stock price given by

$$\Delta_{\text{put}} = \frac{\partial P^{(SV)}}{\partial s}.$$

Because a closed-form solution for the call price is available through the Lipton method cf. Question 1.g, then the corresponding put price can be obtained via put–call parity, i.e.,

$$P^{(SV)} = C^{(SV)} - S + Ke^{-rT},$$

and by differentiating both sides with respect to S, we get that

$$\frac{P^{(SV)}}{\partial s} = \frac{C^{(SV)}}{\partial s} - 1 \Leftrightarrow \Delta_{\text{put}} = \Delta_{\text{call}} - 1.$$

From earlier, we have that the call price in the Lipton Heston model is given by

$$\text{Call}^{(SV)}(0, S, \nu, T, K) = e^{-r^1 T} S - \frac{e^{-r^0 T} K}{2\pi} \int_{-\infty}^{\infty} \Phi\big(X(S), k\big)\, dk, \quad \text{with} \quad X(S) = \ln\left(\frac{S}{K}\right) + rT,$$
$$\Phi(X, k) = \exp\left[\left(\tfrac{1}{2} - ik\right)X + \alpha^{(SV)}(T, k) + \left(k^2 + \tfrac{1}{4}\right)\beta^{(SV)}(T, k)\nu\right] \frac{1}{k^2 + \frac{1}{4}},$$

and then we obtain Δ_{call} by

$$\begin{aligned}
\Delta_{\text{call}}(S) &= \frac{\partial \text{Call}^{(SV)}(0, S, \nu, T, K)}{\partial S} \\
&= \frac{\partial}{\partial S}\left[e^{-r^1 T} S - \frac{e^{-r^0 T} K}{2\pi} \int_{-\infty}^{\infty} \Phi\big(X(S), k\big)\,\mathrm{d}k\right] \\
&= e^{-r^1 T} - \frac{e^{-r^0 T} K}{2\pi} \frac{\mathrm{d}}{\mathrm{d}S} \int_{-\infty}^{\infty} \Phi\big(X(S), k\big)\,\mathrm{d}k \\
&= e^{-r^1 T} - \frac{e^{-r^0 T} K}{2\pi} \int_{-\infty}^{\infty} \frac{\partial \Phi\big(X(S), k\big)}{\partial S}\,\mathrm{d}k \\
&= e^{-r^1 T} - \frac{e^{-r^0 T} K}{2\pi} \int_{-\infty}^{\infty} \frac{\partial \Phi\big(X(S), k\big)}{\partial X} \frac{\partial X}{\partial S}\,\mathrm{d}k \\
&= e^{-r^1 T} - \frac{e^{-r^0 T} K}{2\pi\, S} \int_{-\infty}^{\infty} \left(\tfrac{1}{2} - ik\right) \Phi\big(X(S), k\big)\mathrm{d}k.
\end{aligned}$$

If we use the equation $\Delta_{\text{put}} = \Delta_{\text{call}} - 1$, we then obtain that the delta of the put is

$$\begin{aligned}
\Delta_{\text{put}}(S) &= \Delta_{\text{call}}(S) - 1 \\
&= \left[e^{-r^1 T} - \frac{e^{-r^0 T} K}{2\pi\, S} \int_{-\infty}^{\infty} \left(\tfrac{1}{2} - ik\right) \Phi\big(X(S), k\big)\, dk\right] - 1,
\end{aligned}$$

Next, we have to demonstrate by simulation that a strategy holding $\Delta(t)$ in the stock at time t and is kept self-financing with the risk-free asset does not perfectly replicate the put option payoff. We use the same parameters as in Table 4 with the only difference being $a = 1$.

The implementation is found in Appendix 6 with a quick overview here: The simulation starts by setting the initial conditions, calculating the option price and its delta (adjusted for the put) via the Heston Fourier method, and establishing the initial hedging portfolio with appropriate positions in the stock and the risk-free asset. Time to maturity is discretized into $N_{\text{hedge}} = 7560$ steps, and due to computational complexity, only $n = 100$ simulation repetitions are done.

At each step, the stock price and variance are updated using the stochastic model, and the portfolio is rebalanced by recomputing the delta to maintain a self-financing strategy. At the final time step T, the portfolio value is compared to the theoretical put payoff, with a plot displaying the payoffs (see Figure X) to show the replication error introduced by discrete hedging.

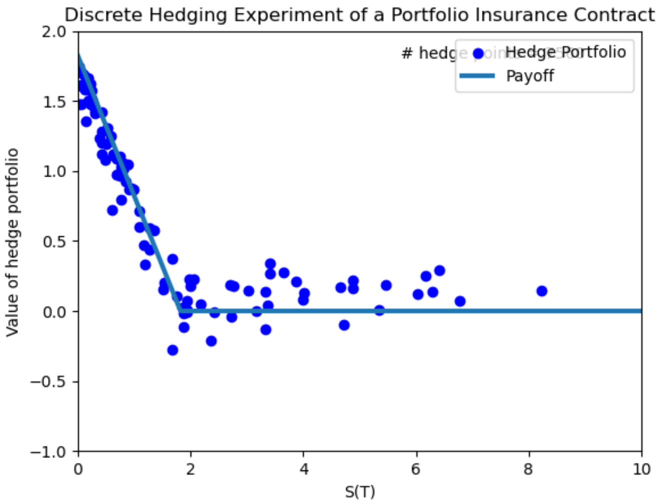

Figure 4: Final portfolio values (blue dots) vs. theoretical put payoff (solid line) with 7560 hedge points.

As seen in Figure 4, the discrete hedging strategy does not perfectly replicate the put payoff, since the scatter of portfolio values around the payoff curve highlights great replication error.

Question 1.j

In this last question, we have to explain how to perfectly replicate an expiry T, strike K_1-put option in the Heston model by trading dynamically in the stock, the risk-free asset and a put option with strike K_0 (and expiry T). We follow the argumentation from page 5 in Gatheral (2006):

We know that the stock price S and its variance V satisfy the SDEs given by

$$dS_t = rS_t dt + \sqrt{V_t} S_t dW_{1,t}^Q,$$
$$dV_t = \kappa(\theta - V_t)dt + \epsilon\sqrt{V_t} dW_{2,t}^Q,$$

with $\langle dW_{1,t}^Q dW_{2,t}^Q \rangle = \rho dt$.

Let a portfolio Π consist of the option desired to be hedged with value $V_t = V(t, S_t, v_t)$, a short position of Δ units of the underlying stock, and a short position of Δ_1 units of a second volatility sensitive derivative $U_t = U(t, S_t, v_t)$. Then, the portfolio value is given by

$$\Pi_t = V_t - \Delta S_t - \Delta_1 U_t.$$

Under the assumption that the portfolio is self-financing, the dynamics of Π_t are given by

$$d\Pi_t = dV_t - \Delta dS_t - \Delta_1 dU_t.$$

By applying a two-dimensional version of Itô's formula from Theorem 4.19 in Björk (2019) to derive the diffusion processes for V_t and U_t, we obtain that

$$\begin{aligned}
d\Pi_t &= \frac{\partial V_t}{\partial t}\,dt + \frac{\partial V_t}{\partial S}\,dS_t + \frac{\partial V_t}{\partial v}\,dv_t + \frac{1}{2}\left(\frac{\partial^2 V_t}{\partial S_t^2}\,(dS_t)^2 + \frac{\partial^2 V_t}{\partial v_t^2}\,(dv_t)^2 + 2\,\frac{\partial^2 V_t}{\partial S_t\,\partial v_t}\,dS_t\,dv_t\right) \\
&\quad - \Delta\,dS_t \\
&\quad - \Delta_1\left(\frac{\partial U_t}{\partial t}\,dt + \frac{\partial U_t}{\partial S_t}\,dS_t + \frac{\partial U_t}{\partial v_t}\,dv_t + \frac{1}{2}\left(\frac{\partial^2 U_t}{\partial S_t^2}\,(dS_t)^2 + \frac{\partial^2 U_t}{\partial v_t^2}\,(dv_t)^2 + 2\,\frac{\partial^2 U_t}{\partial S_t\,\partial v_t}\,dS_t\,dv_t\right)\right).
\end{aligned}$$

Then, using the Heston model dynamics above, we get that

$$\begin{aligned}
d\Pi_t = &\left(\frac{\partial V_t}{\partial t} + \frac{1}{2}v_t S_t^2\frac{\partial^2 V_t}{\partial S_t^2} + \rho\,\epsilon v_t S_t\frac{\partial^2 V_t}{\partial S_t \partial v_t} + \frac{1}{2}\epsilon^2 v_t\frac{\partial^2 V_t}{\partial v_t^2}\right) dt \\
&- \Delta_1\left(\frac{\partial U_t}{\partial t} + \frac{1}{2}v_t S_t^2\frac{\partial^2 U_t}{\partial S_t^2} + \rho\,\epsilon v_t S_t\frac{\partial^2 U_t}{\partial S_t \partial v_t} + \frac{1}{2}\epsilon^2 v_t\frac{\partial^2 U_t}{\partial v_t^2}\right) dt \\
&+ \left(\frac{\partial V_t}{\partial S_t} - \Delta_1\,\frac{\partial U_t}{\partial S_t} - \Delta\right) dS_t \\
&+ \left(\frac{\partial V_t}{\partial v_t} - \Delta_1\frac{\partial U_t}{\partial v_t}\right) dv_t.
\end{aligned}$$

Since the sources of risk lie in the dS_t and dv_t terms, the portfolio can be made instantaneously risk-free by eliminating exposure to these components, which is achieved by setting

$$\begin{aligned}
\frac{\partial V_t}{\partial v_t} - \Delta_1\frac{\partial U_t}{\partial v_t} &= 0 \\
\frac{\partial V_t}{\partial S_t} - \Delta_1\,\frac{\partial U_t}{\partial S_t} - \Delta &= 0.
\end{aligned}$$

This leads to a system of two equations in two unknowns, which we solve as follows:

$$\begin{aligned}
\frac{\partial V_t}{\partial v_t} - \Delta_1\frac{\partial U_t}{\partial v_t} = 0 \quad &\Leftrightarrow \quad \Delta_1 = \frac{\partial V_t/\partial v_t}{\partial U_t/\partial v_t} = \frac{\frac{\partial V_t}{\partial\sqrt{v_t}}\cdot\frac{\partial\sqrt{v_t}}{\partial v_t}}{\frac{\partial U_t}{\partial\sqrt{v_t}}\cdot\frac{\partial\sqrt{v_t}}{\partial v_t}} = \frac{\mathcal{V}(V_t)}{2\sqrt{v_t}}\cdot\frac{2\sqrt{v_t}}{\mathcal{V}(U_t)} = \frac{\mathcal{V}(V_t)}{\mathcal{V}(U_t)}, \\
\frac{\partial V_t}{\partial S_t} - \Delta_1\frac{\partial U_t}{\partial S_t} - \Delta = 0 \quad &\Leftrightarrow \quad \Delta = \frac{\partial V_t}{\partial S_t} - \frac{\mathcal{V}(V_t)}{\mathcal{V}(U_t)}\cdot\frac{\partial U_t}{\partial S_t},
\end{aligned}$$

where $\mathcal{V}(X)$ is the derivative of a contract X w.r.t. $\sigma = \sqrt{v_t}$, i.e. the vega of a derivative X as its sensitivity to $\sqrt{v_t}$ cf. Definition 10.4 in Björk (2019).

By letting V_t and U_t represent put options with strikes K_1 and K_0 respectively, then we write their deltas explicitly as Δ_{put,K_1} and Δ_{put,K_0}. Substituting in, we get

$$\Delta = \Delta_{\text{put},K_1} - \frac{\mathcal{V}(V_t)}{\mathcal{V}(U_t)}\Delta_{\text{put},K_0}.$$

Thus, we can construct an instantaneously risk-free portfolio by holding $\mathcal{V}(V_t)/\mathcal{V}(U_t)$ units of the put option with strike K_0 and a short position in the stock equal to $\Delta_{\text{put},K_1} - \mathcal{V}(V_t)/\mathcal{V}(U_t)\Delta_{\text{put},K_0}$. These conditions leaves us with

$$\begin{aligned} d\Pi_t &= \left(\frac{\partial V_t}{\partial t} + \frac{1}{2}v_t S_t^2 \frac{\partial^2 V_t}{\partial S_t^2} + \frac{1}{2}\epsilon^2 v_t \frac{\partial^2 V_t}{\partial v_t^2} + \rho\epsilon v_t S_t \frac{\partial^2 V_t}{\partial S_t \partial v_t}\right)dt \\ &\quad - \Delta_1\left(\frac{\partial U_t}{\partial t} + \frac{1}{2}v_t S_t^2 \frac{\partial^2 U_t}{\partial S_t^2} + \frac{1}{2}\epsilon^2 v_t \frac{\partial^2 U_t}{\partial v_t^2} + \rho\epsilon v_t S_t \frac{\partial^2 U_t}{\partial S_t \partial v_t}\right)dt \\ &= r\Pi_t\, dt \\ &= r(V_t - \Delta S_t - \Delta_1 U_t)\, dt \end{aligned}$$

since any risk-free portfolio must yield the risk-free rate r, otherwise an arbitrage opportunity would arise, and it follows that the value of our hedged portfolio must grow at this rate. Hence, we can replicate the value of the strike-K_1 put option by continuously adjusting our positions in the underlying stock and the auxiliary strike-K_0 put option as described above, and allocating any remaining value to the risk-free asset (i.e., the bank account).

If, instead of the strike-K_0 put, we trade a variance swap with expiry T, we can still replicate the strike-K_1 put using a similar approach. In this case, the auxiliary option is replaced by a variance swap, which is directly sensitive to the stochastic variance v_t in the Heston model. Since the payoff of a variance swap is linked to the realized variance over the period $[0, T]$, its time-t value is approximately linear in the instantaneous variance v_t. This linear dependency makes the variance swap particularly useful for offsetting the vega exposure of the put.

Let V_t denote the value of the K_1-put, and let VS_t denote the value of the variance swap. We construct the replication portfolio as follows:

$$\Pi_t = V_t - \Delta\, S_t - \phi\, \text{VS}_t,$$

where the coefficients Δ and ϕ are selected to neutralize the sensitivities with respect to both S_t (delta risk) and v_t (vega risk). The delta hedge condition is given by

$$\Delta = \frac{\partial V_t}{\partial S_t},$$

while the vega hedge condition is

$$\phi = \frac{\partial V_t/\partial v_t}{\partial \, \mathrm{VS}_t/\partial v_t}.$$

Here, ϕ is the number of variance swap units required to offset the vega of the K_1-put. Since the swap's value under the Heston model depends linearly on the expected realized variance, its vega is constant and can be computed explicitly. By dynamically adjusting Δ and ϕ as S_t and v_t evolve, and by investing any residual funds in the risk-free asset, we form a self-financing, hedged portfolio that eliminates both delta and vega risks. Thus, the portfolio becomes locally riskless and grows at the risk-free rate, replicating the payoff of the K_1-put at expiration.

Appendix

1. Code for Question 1b

```
import numpy as np
from scipy.stats import norm
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec

class PortfolioInsuranceHedgeBase:
    def __init__(self, r, S0, A0, T, K, a, sigma, Nhedges, Nreps):
        self.r = r
        self.S0 = S0
        self.A0 = A0
        self.T = T
        self.K = K
        self.a = a
        self.sigma = sigma
        self.Nhedges = Nhedges
        self.Nreps = Nreps
        self.dt = T / Nhedges
        self.exp_r_dt = np.exp(r * self.dt)
        self.g0 = A0 / (S0 ** a)

        self.S = None
        self.A = None
        self.V = None
        self.hedge_error = None

    def compute_g(self, t):
        return self.g0 * np.exp((1 - self.a) * (self.r + 0.5 * self.a * self.sigma ** 2) * t)
```

```python
def A_from_S(self, S, t):
    return self.compute_g(t) * (S ** self.a)

def black_scholes_put_price(self, A, T_rem):
    if T_rem <= 0:
        return np.maximum(self.K - A, 0)
    vol = self.a * self.sigma
    d1 = (np.log(A / self.K) + (self.r + 0.5 * vol ** 2) * T_rem) / (vol * np.sqrt(T_rem))
    d2 = d1 - vol * np.sqrt(T_rem)
    price = self.K * np.exp(-self.r * T_rem) * norm.cdf(-d2) - A * norm.cdf(-d1)
    return price

def delta(self, A, S, T_rem):
    if T_rem <= 0:
        return 0
    vol = self.a * self.sigma
    d1 = (np.log(A / self.K) + (self.r + 0.5 * vol ** 2) * T_rem) / (vol * np.sqrt(T_rem))
    delta = -norm.cdf(-d1) * self.a * A / S
    return delta

def hedge_experiment(self):
    self.S = np.full(self.Nreps, self.S0, dtype=float)
    self.A = np.full(self.Nreps, self.A0, dtype=float)

    price0 = self.black_scholes_put_price(self.A0, self.T)
    delta0 = self.delta(self.A0, self.S0, self.T)
    self.V = np.full(self.Nreps, price0, dtype=float)
    current_delta = np.full(self.Nreps, delta0, dtype=float)
    cash = self.V - current_delta * self.S

    for i in range(1, self.Nhedges + 1):
        t = i * self.dt
        T_rem = self.T - t
        Z = np.random.normal(size=self.Nreps)
        self.S *= np.exp((self.r - 0.5 * self.sigma ** 2) * self.dt + self.sigma * np.sqrt(self.dt) *
        ↪  Z)
        self.A = self.A_from_S(self.S, t)

        cash = cash * np.exp(self.r * self.dt)
        self.V = current_delta * self.S + cash

        new_delta = self.delta(self.A, self.S, T_rem) if T_rem > 0 else 0
        cash = self.V - new_delta * self.S
        current_delta = new_delta

    payoff = np.maximum(self.K - self.A, 0)
    self.hedge_error = self.V - payoff
    return self.hedge_error

def final_payoff_error(self):
    return self.hedge_error
```

```
    @classmethod
    def convergence_of_hedge_error(cls, ax, r, S0, A0, T, K, a, sigma, Nreps, max_hedges=1000):
        std_devs = []
        for Nhedges in range(2, max_hedges + 1):
            hedge = cls(r, S0, A0, T, K, a, sigma, Nhedges, Nreps)
            hedge.hedge_experiment()
            error = hedge.final_payoff_error()
            discounted_error = np.exp(-r * T) * error
            std_devs.append(np.std(discounted_error))
        ax.plot(range(2, max_hedges + 1), std_devs, linewidth=1, color='red')
        ax.set_xlabel("Number of Hedge Points", fontsize=10)
        ax.set_ylabel("Std Dev of Discounted Error", fontsize=10)
        ax.set_title("Convergence of Hedge Error", fontsize=12)
        ax.grid(True)

    @classmethod
    def plot_hedge_experiment(cls, r, S0, A0, T, K, a, sigma, Nreps, Nhedges, save_path=None):
        hedge = cls(r, S0, A0, T, K, a, sigma, Nhedges, Nreps)
        hedge.hedge_experiment()
        final_errors = hedge.final_payoff_error()
        S_final = hedge.S
        A_final = hedge.A
        V_final = hedge.V
        payoff = np.maximum(K - A_final, 0)

        fig = plt.figure(figsize=(10, 8))
        plt.style.use('ggplot')
        gs = gridspec.GridSpec(2, 2, height_ratios=[1, 1])
        ax1 = fig.add_subplot(gs[0, :])
        ax2 = fig.add_subplot(gs[1, 0])
        ax3 = fig.add_subplot(gs[1, 1])

        ax1.scatter(A_final, V_final, s=10, alpha=0.2, label="Hedge Portfolio Payoff", zorder=2)
        A_range = np.linspace(np.min(A_final), np.max(A_final), 200)
        payoff_range = np.maximum(K - A_range, 0)
        ax1.plot(A_range, payoff_range, 'k-', linewidth=2, label="Insurance Contract Payoff", zorder=1)
        ax1.set_title("Discrete Delta Hedging of a Portfolio Insurance Contract", fontsize=12)
        ax1.text(0.35, 0.85, f"# Hedge Points = {Nhedges}", transform=ax1.transAxes, fontsize=10,
                 verticalalignment='top', horizontalalignment='right')
        ax1.set_xlabel("A(T)", fontsize=10)
        ax1.set_ylabel("Hedge Portfolio Value", fontsize=10)
        ax1.legend()

        ax2.hist(final_errors, bins=40, color='skyblue', edgecolor='black', alpha=0.7)
        ax2.set_title("Histogram of Final Hedge Errors", fontsize=12)
        ax2.set_xlabel("Hedge Error", fontsize=10)
        ax2.set_ylabel("Frequency", fontsize=10)

        cls.convergence_of_hedge_error(ax=ax3, r=r, S0=S0, A0=A0, T=T, K=K, a=a, sigma=sigma, Nreps=Nreps,
        ↪   max_hedges=1000)

        plt.tight_layout(rect=[0, 0, 1, 0.96])
        if save_path:
```

```python
            plt.savefig(save_path)
        else:
            plt.show()

    @classmethod
    def plot_multiple_hedge_experiments(cls, r, S0, A0, T, K, a, sigma, Nreps, hedge_dict, save_path=None):
        ncols = len(hedge_dict)
        fig, axes = plt.subplots(2, ncols, figsize=(4 * ncols, 8))
        plt.style.use('ggplot')

        for i, (label, Nhedges) in enumerate(hedge_dict.items()):
            hedge = cls(r, S0, A0, T, K, a, sigma, Nhedges, Nreps)
            hedge.hedge_experiment()
            A_final = hedge.A
            V_final = hedge.V
            hedge_error = hedge.final_payoff_error()

            ax_scatter = axes[0, i]
            ax_scatter.scatter(A_final, V_final, s=10, alpha=0.2, label="Hedge Portfolio", zorder=2)
            A_range = np.linspace(np.min(A_final), np.max(A_final), 200)
            payoff_range = np.maximum(K - A_range, 0)
            ax_scatter.plot(A_range, payoff_range, 'k-', linewidth=2, label="Insurance Contract Payoff",
            ↪  zorder=1)
            ax_scatter.set_title(f"{label} Hedge (n={Nhedges})", fontsize=12)
            ax_scatter.set_xlabel("A(T)", fontsize=10)
            ax_scatter.set_ylabel("Hedge Portfolio Value", fontsize=10)
            ax_scatter.legend()

            ax_hist = axes[1, i]
            ax_hist.hist(hedge_error, bins=40, color='skyblue', edgecolor='black', alpha=0.7)
            ax_hist.set_title(f"Hedge Error: {label}", fontsize=12)
            ax_hist.set_xlabel("Hedge Error", fontsize=10)
            ax_hist.set_ylabel("Frequency", fontsize=10)

        plt.tight_layout(rect=[0, 0, 1, 0.96])
        if save_path:
            plt.savefig(save_path)
        else:
            plt.show()

if __name__ == "__main__":
    S0 = 1
    A0 = 1
    T = 30
    K = np.exp(0.6)
    r = 0.02
    a = 0.5
    sigma = 0.20
    Nreps = 1000

    Nhedges_daily = 7560
    PortfolioInsuranceHedgeBase.plot_hedge_experiment(r, S0, A0, T, K, a, sigma, Nreps, Nhedges_daily,
    ↪  save_path="Assignment_3/question_1b_portfolio_insurance_hedge_experiment.png")
```

```
    hedge_dict = {
        "Yearly": 30,
        "Monthly": 360,
        "Weekly": 1560,
        "Daily": 7560
    }
    PortfolioInsuranceHedgeBase.plot_multiple_hedge_experiments(r, S0, A0, T, K, a, sigma, Nreps,
    ↪ hedge_dict, save_path="Assignment_3/multiple_hedge_experiments.png")
```

2. Code for Question 1c

```
import numpy as np
from scipy.stats import norm

r = 0.02
mu = 0.07
sigma = 0.20
S0 = 1
A0 = 1
T = 30
a = 0.5
K = np.exp(r * T)

a_sigma = a * sigma
d1_q1b = (np.log(A0 / K) + (r + 0.5 * a_sigma**2) * T) / (a_sigma * np.sqrt(T))
d2_q1b = d1_q1b - a_sigma * np.sqrt(T)
price_q1b = K * np.exp(-r * T) * norm.cdf(-d2_q1b) - A0 * norm.cdf(-d1_q1b)

d2_1c_1 = (np.log(A0 / K) + (a * (mu - r) + r - 0.5 * a_sigma**2) * T) / (a_sigma * np.sqrt(T))
d1_1c_1 = d2_1c_1 + a_sigma * np.sqrt(T)
price_1c_1 = np.exp(-r * T) * K * norm.cdf(-d2_1c_1) - A0 * np.exp((a * mu - a * r) * T) *
↪ norm.cdf(-d1_1c_1)

d2_1c_2 = (np.log(S0**a / K) + a * (r - 0.5 * sigma**2) * T) / (a * sigma * np.sqrt(T))
d1_1c_2 = d2_1c_2 + a * sigma * np.sqrt(T)
price_1c_2 = np.exp(-r * T) * (K * norm.cdf(-d2_1c_2) - S0**a * np.exp((a * r - 0.5 * a * sigma**2 + 0.5 *
↪ a**2 * sigma**2) * T) * norm.cdf(-d1_1c_2))

d1_1c_3 = (np.log(S0 / K) + (r + 0.5 * sigma**2) * T) / (sigma * np.sqrt(T))
d2_1c_3 = d1_1c_3 - sigma * np.sqrt(T)
price_1c_3 = a * (K * np.exp(-r * T) * norm.cdf(-d2_1c_3) - S0 * norm.cdf(-d1_1c_3))

round(price_q1b, 4), round(price_1c_1, 4), round(price_1c_2, 4), round(price_1c_3, 4)
```

3. Code for Question 1e

```
import numpy as np
from scipy.stats import norm
import matplotlib.pyplot as plt

class StaticHedgePricer:
```

```python
    def __init__(self, a, S0, r, T, sigma):
        self.a = a
        self.S0 = S0
        self.r = r
        self.T = T
        self.sigma = sigma
        self.K = np.exp(r * T)
        self.gT = np.exp(((1 - a) * r + 0.5 * a * (1 - a) * sigma**2) * T)
        self.S_star = (self.K / self.gT)**(1 / a)

    def black_scholes_put(self, S0, strike, T, r, sigma):
        d1 = (np.log(S0 / strike) + (r + 0.5 * sigma**2) * T) / (sigma * np.sqrt(T))
        d2 = d1 - sigma * np.sqrt(T)
        put_price = strike * np.exp(-r * T) * (1 - norm.cdf(d2)) + S0 * (norm.cdf(d1) - 1)
        return put_price

    def compute_static_hedge(self, n):
        K_grid = np.linspace(0.001, self.S_star, n)
        dK = K_grid[1] - K_grid[0]

        weights = self.a * (1 - self.a) * self.gT * (K_grid ** (self.a - 2))
        put_prices = np.array([
            self.black_scholes_put(self.S0, Ki, self.T, self.r, self.sigma)
            for Ki in K_grid
        ])

        integral_contribution = np.sum(weights * put_prices * dK)

        dirac_weight = self.a * self.gT * (self.S_star ** (self.a - 1))
        dirac_put = self.black_scholes_put(self.S0, self.S_star, self.T, self.r, self.sigma)
        dirac_contribution = dirac_weight * dirac_put

        hedge_price = integral_contribution + dirac_contribution
        return hedge_price

    def convergence_plot(self, n_min=50, n_max=10000, step=25, save_path=None):
        n_values = np.arange(n_min, n_max + 1, step)
        hedge_prices = [self.compute_static_hedge(n) for n in n_values]

        plt.style.use('ggplot')

        fig, ax = plt.subplots(figsize=(10, 6))
        fig.patch.set_facecolor('white')
        ax.set_facecolor('#E5E5E5')

        ax.plot(n_values, hedge_prices, linewidth=2, color='blue')
        ax.set_xlabel("Number of Strikes", fontsize=10)
        ax.set_ylabel("Static Hedge Price", fontsize=10)
        ax.set_title("Convergence of Static Hedge Price", fontsize=12, color='black')

        ax.grid(True)

        if save_path:
```

```
            plt.savefig(save_path, facecolor=fig.get_facecolor(), edgecolor='none')
        else:
            plt.show()

if __name__ == "__main__":
    a = 0.5
    S0 = 1
    r = 0.02
    T = 30
    sigma = 0.2

    pricer = StaticHedgePricer(a, S0, r, T, sigma)
    pricer.convergence_plot(save_path="Assignment_3/question_1e_static_hedge_convergence.png")
```

4. Code for Question 1g

```
import numpy as np
from scipy.integrate import quad

def Heston_Fourier(spot, timetoexp, strike, r, divyield, V, theta, kappa, epsilon, rho, greek=1):
    X = np.log(spot/strike) + (r - divyield) * timetoexp
    kappahat = kappa - 0.5 * rho * epsilon
    xiDummy = kappahat**2 + 0.25 * epsilon**2

    def integrand(k):
        xi = np.sqrt(k**2 * epsilon**2 * (1 - rho**2) + 2j * k * epsilon * rho * kappahat + xiDummy)
        Psi_P = - (1j * k * rho * epsilon + kappahat) + xi
        Psi_M = (1j * k * rho * epsilon + kappahat) + xi
        arg_log = (Psi_M + Psi_P * np.exp(-xi * timetoexp)) / (2 * xi)
        alpha = -kappa * theta * (Psi_P * timetoexp + 2 * np.log(arg_log)) / epsilon**2
        beta = - (1 - np.exp(-xi * timetoexp)) / (Psi_M + Psi_P * np.exp(-xi * timetoexp))
        numerator = np.exp(( -1j * k + 0.5 ) * X + alpha + (k**2 + 0.25) * beta * V)

        if greek == 1:
            dummy = np.real(numerator / (k**2 + 0.25))
        elif greek == 2:
            dummy = np.real((0.5 - 1j*k) * numerator / (spot * (k**2 + 0.25)))
        elif greek == 3:
            dummy = -np.real(numerator / spot**2)
        elif greek == 4:
            dummy = np.real(numerator * beta)
        else:
            raise ValueError("Invalid greek value. Use 1, 2, 3, or 4.")
        return dummy

    integral_value, _ = quad(integrand, -100, 100, limit=200)

    if greek == 1:
        dummy = np.exp(-divyield * timetoexp) * spot - strike * np.exp(-r * timetoexp) * integral_value /
        ↪  (2 * np.pi)
    elif greek == 2:
```

```
        dummy = np.exp(-divyield * timetoexp) - strike * np.exp(-r * timetoexp) * integral_value / (2 *
        ↪  np.pi)
    elif greek == 3:
        dummy = -strike * np.exp(-r * timetoexp) * integral_value / (2 * np.pi)
    elif greek == 4:
        dummy = -strike * np.exp(-r * timetoexp) * integral_value / (2 * np.pi)

    return dummy

if __name__ == '__main__':
    spot = 1.0
    T = 30.0
    r = 0.02
    divyield = 0.0
    strike = np.exp(r * T)
    a = 0.5
    V_eff = a**2 * 0.04
    theta_eff = a**2 * 0.04
    kappa = 2.0
    epsilon_eff = a * 1.0
    rho = -0.5

    price = Heston_Fourier(spot, T, strike, r, divyield, V_eff, theta_eff, kappa, epsilon_eff, rho,
    ↪  greek=1)
    print(f"Call Price: {price:.4f}")
```

5. Code for Question 1h

```
import numpy as np
from Assignment_3.question_1g_code import Heston_Fourier

a = 0.5
S0 = 1.0
A0 = 1.0
r = 0.02
T = 30.0
sigma = 0.2
K = np.exp(r * T)
gT = np.exp(((1 - a) * r + 0.5 * a * (1 - a) * sigma**2) * T)
S_star = (K / gT) ** (1 / a)

theta = sigma**2
kappa = 2.0
epsilon = 1.0
rho = -0.5
v0 = sigma**2

n = 1000
K_grid = np.linspace(0.001, S_star, n)
dK = K_grid[1] - K_grid[0]

def fpp(K):
```

```
    return a * (1 - a) * gT * K ** (a - 2)

weights = a * (1 - a) * gT * K_grid ** (a - 2)

put_prices = np.array([
    Heston_Fourier(S0, T, Ki, r, 0, v0, theta, kappa, epsilon, rho, greek=1) + Ki * np.exp(-r * T) - S0
    for Ki in K_grid
])

total_price = np.sum(weights * put_prices * dK)
dirac_weight = a * gT * S_star ** (a - 1)
dirac_put_price = (Heston_Fourier(S0, T, S_star, r, 0, v0, theta, kappa, epsilon, rho, greek=1) +
                   S_star * np.exp(-r * T) - S0)
hedge_price = total_price + dirac_weight * dirac_put_price

round(hedge_price, 4)
```

6. Code for Question 1i

```
import numpy as np
import matplotlib.pyplot as plt
from Assignment_3.question_1g_code import Heston_Fourier

def simulate_hedging(r, sigma, spot, rho, S0, capT, V0, theta, kappa, epsilon, Nhedge, Nrep):
    dt = capT / Nhedge
    strike = np.exp(r * capT)
    S = np.full(Nrep, S0)
    V = np.full(Nrep, V0)
    initial_outlay = Heston_Fourier(spot, capT, strike, r, 0, V0, theta, kappa, epsilon, rho, greek=1)
    Vpf = np.full(Nrep, initial_outlay)
    delta_initial = Heston_Fourier(S0, capT, strike, r, 0, V0, theta, kappa, epsilon, rho, greek=2) - 1
    delta = np.full(Nrep, delta_initial)
    a_pos = delta.copy()
    b_pos = Vpf - a_pos * S

    for i in range(1, Nhedge):
        print(f"Step {i}/{Nhedge}")
        timetomat = capT - i * dt
        dW1 = np.sqrt(dt) * np.random.randn(Nrep)
        dW2 = rho * dW1 + np.sqrt(1 - rho**2) * np.sqrt(dt) * np.random.randn(Nrep)
        X = np.log(S)
        X = X + (r - 0.5 * np.maximum(V, 0)) * dt + np.sqrt(np.maximum(V, 0)) * dW1
        S = np.exp(X)
        V = V + kappa * (theta - np.maximum(V, 0)) * dt + epsilon * np.sqrt(np.maximum(V, 0)) * dW2
        Vpf = a_pos * S + b_pos * np.exp(r * dt)
        new_delta = np.empty(Nrep)
        for j in range(Nrep):
            new_delta[j] = Heston_Fourier(S[j], timetomat, strike, r, 0, V[j], theta, kappa, epsilon, rho,
            ↪ greek=2) - 1
        delta = new_delta.copy()
        a_pos = delta.copy()
        b_pos = Vpf - a_pos * S
```

```
    dW1 = np.sqrt(dt) * np.random.randn(Nrep)
    dW2 = rho * dW1 + np.sqrt(1 - rho**2) * np.sqrt(dt) * np.random.randn(Nrep)
    X = np.log(S)
    X = X + (r - 0.5 * np.maximum(V, 0)) * dt + np.sqrt(np.maximum(V, 0)) * dW1
    S = np.exp(X)
    V = V + kappa * (theta - np.maximum(V, 0)) * dt + epsilon * np.sqrt(np.maximum(V, 0)) * dW2
    Vpf = a_pos * S + b_pos * np.exp(r * dt)

    return S, Vpf, strike, Nhedge

def plot_results(S, Vpf, strike, Nhedge, save_path=None):
    fig, ax = plt.subplots()
    ax.scatter(S, Vpf, color='blue', label='Hedge Portfolio')
    ax.set_xlabel("S(T)")
    ax.set_ylabel("Value of hedge portfolio")
    ax.set_xlim(0, 10)
    ax.set_ylim(-1, 2)
    ax.set_title("Discrete Hedging Experiment of a Portfolio Insurance Contract")
    ax.text(9, 1.8, f"# hedge points = {Nhedge}", horizontalalignment='right')
    x_vals = np.linspace(0, 10, 400)
    payoff = np.maximum(strike - x_vals, 0)
    ax.plot(x_vals, payoff, linewidth=3, label='Payoff')
    ax.legend()

    if save_path:
        plt.savefig(save_path, facecolor=fig.get_facecolor(), edgecolor='none')  # Use fig.get_facecolor()
    else:
        plt.show()

if __name__ == "__main__":
    r = 0.02
    sigma = 0.2
    spot = 1.0
    rho = -0.5
    S0 = 1.0
    capT = 30.0
    V0 = 0.2 ** 2
    theta = 0.2 ** 2
    kappa = 2.0
    epsilon = 1.0
    Nhedge = int(252 * 30)
    Nrep = 100
    S, Vpf, strike, Nhedge = simulate_hedging(r, sigma, spot, rho, S0, capT, V0, theta, kappa, epsilon,
    ↪  Nhedge, Nrep)
    plot_results(S, Vpf, strike, Nhedge, save_path="Assignment_3/question_1i_hedging_results.png")
```

References

Björk, T. (2019). *Arbitrage Theory in Continuous Time.* Oxford Academic, 4th edition.

Carr, P. and Madan, D. (2001). Optimal positioning in derivative securities. *Quantitative Finance*, 1:19–37.

Gatheral, J. (2006). *The Volatility Surface.* John Wiley Sons, Inc., 1st edition.

Heston, S. L. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency. *The Review of Financial Studies*, 6(2):327–343.

Lipton, A. (2002). The vol smile problem. *Masterclass with Deutsche Bank*, pages 61–65.

Pedersen, J. L. (2024). Functions of finite variaition.

Schilling, R. L. (2005). *Measures, Integrals and Martingales.* Cambridge University Press, 1st edition.