SDEs Coursework

Fat Tailed Bears (was S&P 500)
William Zheng, Antao Lou, Daniel Zhang, Uathesh Ranjan
02045918, 01857746, 02044064, 02031296

March 2024

Q1

This bear risk reversal (BRR) option consists of the investor going long on an out-of-the-money put with strike K_2, and also shorting an out-of-the-money call with strike K_1. From the graph, we can see that the payoff decreases as the stock price rises. Therefore, the investor would only take this BRR position if they expect the stock price to drop below K_2 at maturity time T, allowing them to make a profit. Alternatively, if the stock price increases, they expect that it won't exceed K_1, because if the stock price S_T falls between K_2 and K_1, then the investor breaks even. If the price exceeds K_1, the payoff would be negative. The payoff increases if the stock price drops below K_2, and the further it drops, the more money the investor makes. The maximum value of the payoff would be K_2, which would occur if the stock price is zero at maturity. On the other hand, if the stock price becomes larger than K_2, then the put is worth nothing but the short call gives a negative payoff of $K_1 - S_T$. Note that the loss is potentially unlimited as there is no bound to which the stock price can grow. This implies that the investor is fairly confident in their investment approach with a high-risk tolerance, as they have implemented no measure to mitigate this. However, they have limited some risk by having a short call of strike price K_1 - this provides a region of 0 payout if S_T falls between K_2 and K_1, thus allowing for a margin of error if they are slightly incorrect about the direction that the stock will move.

Q2

First, we note that the payoff Y is written as an addition of a call and a put option payoff. In lectures, we have covered that under the Black Scholes model, the prices of the calls and puts are as follows (note we have substituted t=0 into these equations already):

$$V_{BS}^{CALL}(0, K_1) = S_0\Phi(d_1) - K_1 e^{-rT}\Phi(d_2)$$

$$V_{BS}^{PUT}(0, K_2) = K_2 e^{-rT}\Phi(-d_2) - S_0\Phi(-d_1)$$

Note that $d_1(0) = \frac{\ln\frac{S_0}{K_1} + \left(r + \frac{\sigma^2}{2}\right)T}{\sigma\sqrt{T}}$, $\quad d_2(0) = d_1(0) - \sigma\sqrt{T}$. Again, we have substituted t=0 already.

So to price the bear risk reversal under the BS model, we need to first work out the prices of the call and the put option using the numbers provided to us (we will use rounding to 4 significant figures from now on). We obtain $d_1 = 0.9892$ and $d_2 = -0.2757$. Note, to work out $\Phi(d_2)$, we use the symmetry of the normal distribution and use $\Phi(d_2) = 1 - \Phi(-d_2)$. Plugging these values, we get $V_{BS}^{CALL} = \$58.95$ and $V_{BS}^{PUT} = \$19.03$. Therefore the price of the bear risk reversal would be $\$19.03 - \$58.95 = -\$39.91$. The subtraction seems to be off by 1p but that's due to rounding.

Q3

We will now present a Python code that will calculate the VaR and ES of the bear risk reversal position over a risk horizon H=1y at 95% confidence level, running 50,000 scenarios.

```python
# import necessary packages
import numpy as np
import math
import statistics
from scipy.stats import norm
import matplotlib.pyplot as plt

# Stock data
S0=100; Sig=0.4; r=0.05; miu=0.1
# Option strikes and maturity
k1=105; k2=95; T=10
# Number of simulations, confidence level, risk horizon
n=50000; confidence=0.95; h=1

# Option prices at time 0
d1c=(math.log(S0/k1)+(r+0.5*Sig**2)*T)/(Sig*T**0.5);
d1p=(math.log(S0/k2)+(r+0.5*Sig**2)*T)/(Sig*T**0.5);
c0=S0*norm.cdf(d1c)-k1*math.exp(-r*T)*norm.cdf(d1c-Sig*T**0.5);
p0=-S0*norm.cdf(-d1p)+k2*math.exp(-r*T)*norm.cdf(-d1p+Sig*T**0.5);
v0=-c0+p0

"""Computing the prices at time h"""
# Adjust T and initialise scenarios and stock prices
T=T-h; Zt = np.zeros(n); St = np.zeros(n);

# Seed random number generator
from random import seed
from random import gauss
seed(1)

# Generate some random Gaussian values and the stock scenarios at time h
for j in range(n):
 Zt[j] = gauss(0,1)
 St[j]=S0*math.exp((miu-0.5*Sig**2)*h)*math.exp(Zt[j]*Sig*(h**0.5))
# End for loop

# Initialise call and put prices for each scenario
ct=np.zeros(n); pt=np.zeros(n);

# Generating call and put scenarios at time h
for i in range(n):
     d1cnew=(math.log(St[i]/k1)+(r+0.5*Sig**2)*T)/(Sig*(T**0.5));
     d1pnew=(math.log(St[i]/k2)+(r+0.5*Sig**2)*T)/(Sig*(T**0.5));
     ct[i]=St[i]*norm.cdf(d1cnew)-k1*math.exp(-r*T)*norm.cdf(d1cnew-Sig*T**(0.5));
     pt[i]=-St[i]*norm.cdf(-d1pnew)+k2*math.exp(-r*T)*norm.cdf(-d1pnew+Sig*T**(0.5))
# End for loop

# Value of portfolio at time t in all scenarios
vt = -ct+pt

# Loss scenarios
vvar = v0-vt; vvar = np.sort(vvar)

# Extracting VaR from the loss at the right confidence level
ivar = round((confidence)*n); var = vvar[ivar]
```

```
# Calculating ES
ESv=statistics.mean(vvar[range(math.floor((confidence)*n),n)])

# Output VaR and ES values
print("VaR:", var); print("ES:", ESv)
```

Running the code gives us VaR = 92.75003 and ES = 131.01592 approximately.

Q4

Now we produce a histogram of the density of the loss distribution at 1 year, indicating the VaR and ES points in the loss graph.

```
# Code for plot
plt.hist(vvar, bins = 100)
plt.axvline(x=var, color='b', label='VaR={}'.format(round(var,5)))
plt.axvline(x=ESv, color='r', label='ES={}'.format(round(ESv,5)))
plt.legend()
plt.show()
```

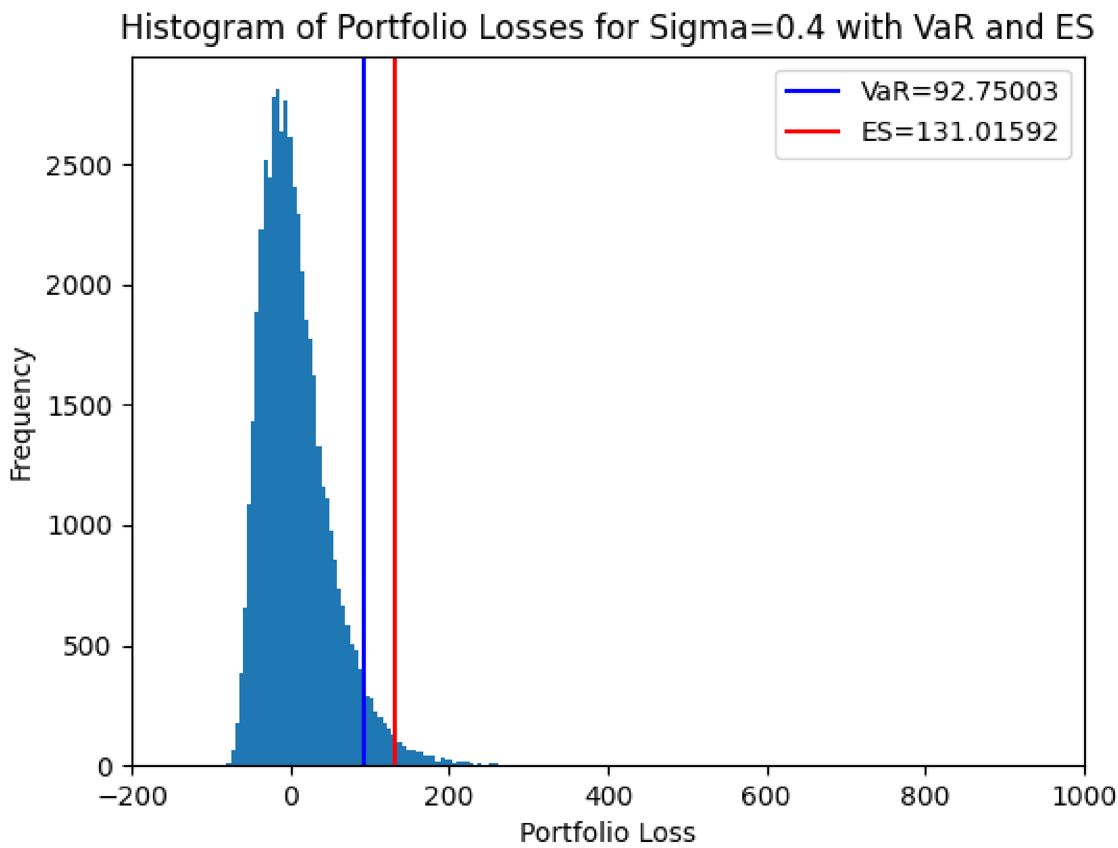

Figure 1: Histogram of density of loss distribution with $\sigma = 0.4$

Q5

In this section, we illustrate the impact on VaR and ES as we increase the volatility of the stock.

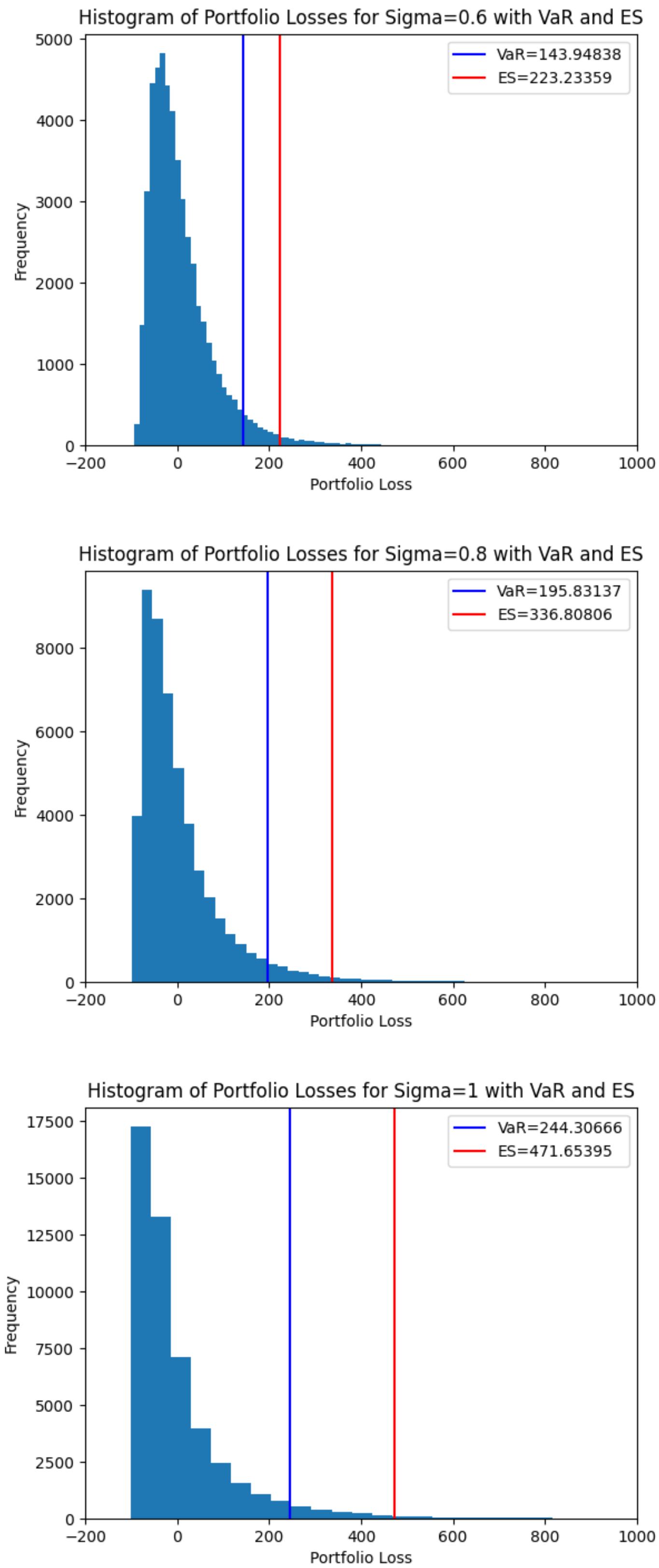

Figure 2: Histograms of density of loss distribution with (a) $\sigma = 0.6$ (b) $\sigma = 0.8$ (c) $\sigma = 1.0$

Q6

Evidently, as the volatility increases, the risk of loss with this contract increases. We can observe that VaR and ES increase with σ. This makes sense because higher volatility means that the distribution will be more spread out, so for a fixed confidence level α, you would need to move further along the x axis to reach the α quantile, hence VaR and ES will be higher. We see from the graph below that VaR and ES both increase non-linearly with respect to the volatility, with ES growing faster than VaR:

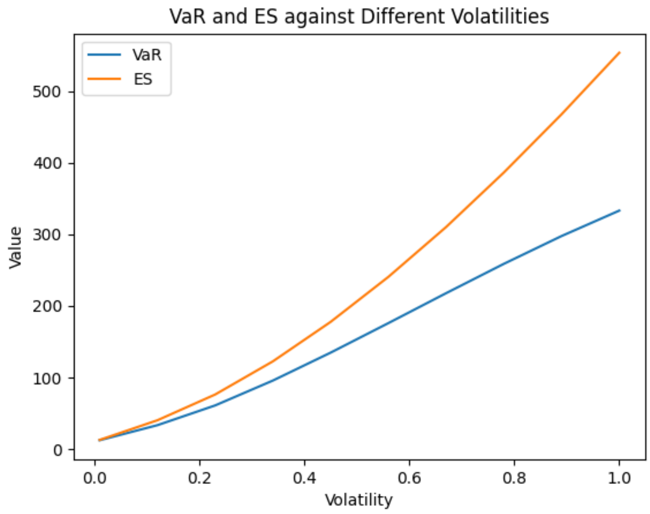

Figure 3: Graph of VaR and ES against volatility

Intuitively, we expect this nonlinear behaviour because we know from the theory and the histograms in the previous question that the returns follow a lognormal distribution, so as volatility increases, the right tail grows longer more than proportionally, which means that the VaR and ES will also increase more than proportionally with volatility. We expect ES to grow more rapidly than VaR because it is the weighted average of losses beyond the VaR level.