Reassuranse - del 3

Arne Bang Huseby

Universitetet i Oslo

STK 1500

— Løsninger på oppgavene

Oppgave 1

Anta at $X \sim \mathcal{N}(\mu, \sigma^2)$, der $\mu = 120$ og $\sigma = 15$, og la $\alpha = 0.2$.

a) Finn $V_\alpha(X)$ ved å benytte metoden beskrevet i Subseksjon 3.1.

b) Finn $V_\alpha(X)$ ved å lage et python-script som benytter funksjonen `norm.isf` beskrevet i Subseksjon 3.4.

Løsning:

a) $V_{0.2}(X) = \sigma \cdot y_{0.2} + \mu = 15 \cdot 0.84162 + 120 = 132.624$

Oppgave 1

Anta at $X \sim \mathcal{N}(\mu, \sigma^2)$, der $\mu = 120$ og $\sigma = 15$, og la $\alpha = 0.2$.

a) Finn $V_\alpha(X)$ ved å benytte metoden beskrevet i Subseksjon 3.1.

b) Finn $V_\alpha(X)$ ved å lage et python-script som benytter funksjonen `norm.isf` beskrevet i Subseksjon 3.4.

Løsning:

a) $V_{0.2}(X) = \sigma \cdot y_{0.2} + \mu = 15 \cdot 0.84162 + 120 = 132.624$

Oppgave 1 (forts.)

b) Et python-script som benytter funksjonen `norm.isf`:

```
#############################################################
#                                                           #
#    Bestem VaR for alpha = 0.2 for en risiko               #
#    som er normalfordelt med parametre 120 og 15.          #
#                                                           #
#############################################################

from scipy.stats import norm

alpha = 0.2

V_alpha = norm.isf(alpha, loc=120, scale=15)

print(f"V_alpha: {V_alpha:.3f}")
```

Output:

```
V_alpha: 132.624
```

Oppgave 2

Anta at $X \sim \log \mathcal{N}(\mu, \sigma^2)$, $\mu = 3$ og $\sigma = 2$, og la $\alpha = 0.2$.

a) Finn $V_\alpha(X)$ ved å benytte metoden beskrevet i Subseksjon 3.2.

b) Finn $V_\alpha(X)$ ved å lage et python-script som benytter funksjonen `lognorm.isf` beskrevet i Subseksjon 3.4.

Løsning:

a) $V_{0.2}(X) = \exp(\sigma \cdot y_{0.2} + \mu) = \exp(2 \cdot 0.84162 + 3) = 108.120$

Oppgave 2

Anta at $X \sim \log \mathcal{N}(\mu, \sigma^2)$, $\mu = 3$ og $\sigma = 2$, og la $\alpha = 0.2$.

a) Finn $V_\alpha(X)$ ved å benytte metoden beskrevet i Subseksjon 3.2.

b) Finn $V_\alpha(X)$ ved å lage et python-script som benytter funksjonen `lognorm.isf` beskrevet i Subseksjon 3.4.

Løsning:

a) $V_{0.2}(X) = \exp(\sigma \cdot y_{0.2} + \mu) = \exp(2 \cdot 0.84162 + 3) = 108.120$

Oppgave 2 (forts.)

b) Et python-script som benytter funksjonen `lognorm.isf`:

```
#############################################################
#                                                           #
#    Bestem VaR for alpha = 0.2 for en risiko               #
#    som er lognormalfordelt med parametre 3 og 2.          #
#                                                           #
#############################################################

from math import exp
from scipy.stats import lognorm

alpha = 0.2

V_alpha = lognorm.isf(alpha, s=2, scale=exp(3))

print(f"V_alpha: {V_alpha:.3f}")
```

Output:

```
V_alpha: 108.120
```

Beregning av VaR – Paretofordelt risiko

Paretofordelingen har en kumulativ fordelingsfunksjon F_X gitt ved:

$$F_X(x) = 1 - \left(\frac{x_{\min}}{x}\right)^{\beta}, \quad \text{for } x > x_{\min} > 0.$$

Overlevelsesfunksjonen $S_X(x) = P(X > x) = 1 - P(X \leq x) = 1 - F_X(x)$ blir dermed:

$$S_X(x) = \left(\frac{x_{\min}}{x}\right)^{\beta}, \quad \text{for } x > x_{\min} > 0.$$

Vi ser her at overlevelsesfunksjonen er kontinuerlig. Dermed kan vi benytte Proposisjon (VaR 3) og finne $V_\alpha(X)$ som løsning x_α av ligningen:

$$S_X(x_\alpha) = \left(\frac{x_{\min}}{x_\alpha}\right)^{\beta} = \alpha$$

I dette tilfellet er det lett å løse denne ligningen, og vi får:

$$V_\alpha(X) = x_\alpha = \frac{x_{\min}}{\alpha^{1/\beta}}$$

Oppgave 3

Anta at $X \sim \text{Pareto}(50, 3)$ og la $\alpha = 0.2$.

a) Finn $V_\alpha(X)$ ved å benytte metoden beskrevet i Subseksjon 3.3.

b) Finn $V_\alpha(X)$ ved å lage et python-script som benytter funksjonen `pareto.isf` beskrevet i Subseksjon 3.4.

Løsning:

a) $V_{0.2}(X) = \frac{x_{\text{min}}}{\alpha^{1/\beta}} = \frac{50}{0.2^{1/3}} = 85.499$

Oppgave 3

Anta at $X \sim \text{Pareto}(50, 3)$ og la $\alpha = 0.2$.

a) Finn $V_\alpha(X)$ ved å benytte metoden beskrevet i Subseksjon 3.3.

b) Finn $V_\alpha(X)$ ved å lage et python-script som benytter funksjonen `pareto.isf` beskrevet i Subseksjon 3.4.

Løsning:

a) $V_{0.2}(X) = \frac{x_{\text{min}}}{\alpha^{1/\beta}} = \frac{50}{0.2^{1/3}} = 85.499$

Oppgave 3 (forts.)

b) Et python-script som benytter funksjonen `pareto.isf`:

```
#############################################################
#                                                           #
#    Bestem VaR for alpha = 0.2 for en risiko               #
#    som er Paretofordelt med parametre 50 og 3.            #
#                                                           #
#############################################################

from scipy.stats import pareto

alpha = 0.2

V_alpha = pareto.isf(alpha, b=3, scale=50)

print(f"V_alpha: {V_alpha:.3f}")
```

Output:

```
V_alpha: 85.499
```

Oppgave 4

Anta at $X \sim \text{Pareto}(60, 2)$ og la $\alpha = 0.1$.

a) Finn $V_\alpha(1 - e^{-X})$.

b) Finn $V_\alpha(\sqrt{X})$.

Løsning:

Vi finner først $V_{0.1}(X)$ ved å benytte formelen:

$$V_{0.1}(X) = \frac{x_{\text{min}}}{\alpha^{1/\beta}} = \frac{60}{0.1^{1/2}} = 189.737$$

a) $V_\alpha(1 - e^{-X}) = (1 - e^{-V_{0.1}(X)}) = 1.000$

b) $V_\alpha(\sqrt{X}) = \sqrt{V_{0.1}(X)} = 13.774$

Oppgave 4

Anta at $X \sim \text{Pareto}(60, 2)$ og la $\alpha = 0.1$.

a) Finn $V_\alpha(1 - e^{-X})$.

b) Finn $V_\alpha(\sqrt{X})$.

Løsning:

Vi finner først $V_{0.1}(X)$ ved å benytte formelen:

$$V_{0.1}(X) = \frac{x_{\text{min}}}{\alpha^{1/\beta}} = \frac{60}{0.1^{1/2}} = 189.737$$

a) $V_\alpha(1 - e^{-X}) = (1 - e^{-V_{0.1}(X)}) = 1.000$

b) $V_\alpha(\sqrt{X}) = \sqrt{V_{0.1}(X)} = 13.774$

Oppgave 4

Anta at $X \sim \text{Pareto}(60, 2)$ og la $\alpha = 0.1$.

a) Finn $V_\alpha(1 - e^{-X})$.

b) Finn $V_\alpha(\sqrt{X})$.

Løsning:

Vi finner først $V_{0.1}(X)$ ved å benytte formelen:

$$V_{0.1}(X) = \frac{x_{\text{min}}}{\alpha^{1/\beta}} = \frac{60}{0.1^{1/2}} = 189.737$$

a) $V_\alpha(1 - e^{-X}) = (1 - e^{-V_{0.1}(X)}) = 1.000$

b) $V_\alpha(\sqrt{X}) = \sqrt{V_{0.1}(X)} = 13.774$

Oppgave 4

Anta at $X \sim \text{Pareto}(60, 2)$ og la $\alpha = 0.1$.

a) Finn $V_\alpha(1 - e^{-X})$.

b) Finn $V_\alpha(\sqrt{X})$.

Løsning:

Vi finner først $V_{0.1}(X)$ ved å benytte formelen:

$$V_{0.1}(X) = \frac{x_{\text{min}}}{\alpha^{1/\beta}} = \frac{60}{0.1^{1/2}} = 189.737$$

a) $V_\alpha(1 - e^{-X}) = (1 - e^{-V_{0.1}(X)}) = 1.000$

b) $V_\alpha(\sqrt{X}) = \sqrt{V_{0.1}(X)} = 13.774$

Oppgave 5

Anta at $Y_i \sim \log \mathcal{N}(\mu_i, \sigma_i^2)$, $i = 0, 1, 2$, der:

$$\begin{aligned} \mu_1 &= 2, \quad \mu_2 = 3, \quad \mu_1 = 4 \\ \sigma_1 &= 1, \quad \sigma_1 = 2, \quad \sigma_1 = 1.5 \end{aligned}$$

La så $\alpha = 0.1$, og innfør $X_1 = Y_0 + Y_1 + Y_2$, og $X_2 = Y_0 \cdot Y_1 \cdot Y_2$. Benytt Monte Carlo simulering til å estimere:

a) $V_\alpha(X_1)$.

b) $V_\alpha(X_2)$.

Vi løser denne oppgaven ved hjelp av et python-script.

Oppgave 5 (forts.)

```
#############################################################
#                                                           #
#    Bestem VaR for alpha = 0.1 for en risiko               #
#    for a sum and product of three lognormal risks         #
#    using Monte Carlo simulation                           #
#                                                           #
#############################################################

import numpy as np
from random import lognormvariate

alpha = 0.1

mu = [2, 3, 4]
sigma = [1, 2, 1.5]

m = len(mu)
N = 100000
X1 = np.zeros(N)
X2 = np.zeros(N)
```

Oppgave 5 (forts.)

```
for j in range(N):
    X1[j] = 0
    X2[j] = 1
    for i in range(m):
        Y = lognormvariate(mu[i], sigma[i])
        X1[j] += Y
        X2[j] *= Y

X1.sort()
X2.sort()

k = int(N * (1-alpha))

print(f"V_alpha(X1): {X1[k]:.3f}")
print(f"V_alpha(X2): {X2[k]:.3f}")
```

Output:

```
V_alpha(X1): 674.747
V_alpha(X2): 259197.370
```

Oppgave 6

Beregn den optimale reassuransekontrakten der $\alpha = 0.01$ og $\theta = 2$ for risikoen X der:

a) $X \sim \mathcal{N}(\mu, \sigma^2)$ der $\mu = 130$ og $\sigma = 20$.

b) $X \sim \log \mathcal{N}(\mu, \sigma^2)$ der $\mu = 4$ og $\sigma = 1.5$.

c) $X \sim \text{Pareto}(90, 2)$.

Oppgave 6 (forts.)

Løsning:

```
############################################################
#                                                          #
#    Finding optimal interval-reinsurance contracts        #
#                                                          #
############################################################

from math import exp
from scipy.stats import norm, lognorm, pareto

alpha = 0.01
theta = 2.00
gamma = 1 / (1 + theta)

# Find the optimal a- and b-value for a norm(100, 20)-distribution
opt_a = norm.isf(gamma, loc=130, scale=20)
opt_b = norm.isf(alpha, loc=130, scale=20)

print(f"Optimal a, norm(130, 20): {opt_a:.3f}")
print(f"Optimal b, norm(130, 20): {opt_b:.3f}")
print("")
```

Oppgave 6 (forts.)

```
# Find the optimal a- and b-value for a lognorm(4, 1.5)-distribution
opt_a = lognorm.isf(gamma, s=1.5, scale=exp(4))
opt_b = lognorm.isf(alpha, s=1.5, scale=exp(4))

print(f"Optimal a, lognorm(4, 1.5): {opt_a:.3f}")
print(f"Optimal b, lognorm(4, 1.5): {opt_b:.3f}")
print("")

# Find the optimal a- and b-value for a pareto(90, 2)-distribution
opt_a = pareto.isf(gamma, b=2, scale=90)
opt_b = pareto.isf(alpha, b=2, scale=90)

print(f"Optimal a, pareto(90, 2): {opt_a:.3f}")
print(f"Optimal b, pareto(90, 2): {opt_b:.3f}")
print("")
```

Output:

```
Optimal a, norm(130, 20): 138.615
Optimal b, norm(130, 20): 176.527

Optimal a, lognorm(4, 1.5): 104.177
Optimal b, lognorm(4, 1.5): 1789.196

Optimal a, pareto(90, 2): 155.885
Optimal b, pareto(90, 2): 900.000
```

Oppgave 7

Anta $X \sim \log \mathcal{N}(\mu, \sigma^2)$, $\mu = 3$ og $\sigma = 2$. Beregn den optimale reassuransekontrakten der $\alpha = 0.01$ og $\theta = 2$ for risikoen X og benytt Monte Carlo simulering til å finne:

a) Prisen cedenten må betale for reassuransekontrakten.

b) VaR dersom cedenten *ikke* tegner en reassuransekontrakt.

c) VaR dersom cedenten tegner den optimale reassuransekontrakten.

d) Netto gevinst ved å tegne den optimale reassuransekontrakten.

Oppgave 7 (forts.)

Løsning:

```
############################################################
#                                                          #
#    1. Finn optimale kontraktsparametre for en            #
#    lognormalt fordelt risiko med mu = 3 og sigma = 2     #
#    alpha = 0.01, theta = 2.0                             #
#                                                          #
#    2. Estimer pris for reassuransekontrakten ved Monte   #
#    Carlo simulering, og sammenlign avsetning med og      #
#    uten reassuranse.                                     #
#                                                          #
############################################################

from math import exp
from scipy.stats import lognorm
from random import lognormvariate

alpha = 0.01
theta = 2.00

gamma = 1 / (1 + theta)

opt_a = lognorm.isf(gamma, s=2, scale=exp(3))
opt_b = lognorm.isf(alpha, s=2, scale=exp(3))

print(f"Optimal a: {opt_a:.3f}")
print(f"Optimal b: {opt_b:.3f}")
```

Oppgave 7 (forts.)

```
N = 100000
R_sum = 0

for j in range(N):
    X = lognormvariate(3, 2)
    if X <= opt_a:
        R_sum += 0
    elif opt_a <= X and X <= opt_b:
        R_sum += (X - opt_a)
    else:
        R_sum += (opt_b - opt_a)

reinsurancePrice = (1 + theta) * R_sum / N
VaR_WithoutReinsurance = opt_b
VaR_WithReinsurance = opt_a + reinsurancePrice
netGain = VaR_WithoutReinsurance - VaR_WithReinsurance

print(f"Reinsurance price: {reinsurancePrice:.3f}")
print(f"VaR without reinsurance: {VaR_WithoutReinsurance:.3f}")
print(f"VaR with reinsurance: {VaR_WithReinsurance:.3f}")
print(f"netGain: {netGain:.3f}")
```

Output:

```
Optimal a: 47.534
Optimal b: 2106.316
Reinsurance price: 270.832
VaR without reinsurance: 2106.316
VaR with reinsurance: 318.366
netGain: 1787.950
```

Oppgave 8

Anta $X \sim Pareto(x_{\min}, \beta)$, $x_{\min} = 60$ og $\beta = 2$. Beregn den optimale reassuransekontrakten der $\alpha = 0.01$ og $\theta = 2$ for risikoen X og benytt Monte Carlo simulering til å finne:

a) Prisen cedenten må betale for reassuransekontrakten.

b) VaR dersom cedenten *ikke* tegner en reassuransekontrakt.

c) VaR dersom cedenten tegner den optimale reassuransekontrakten.

d) Netto gevinst ved å tegne den optimale reassuransekontrakten.

Oppgave 8 (forts.)

Løsning:

```
############################################################
#                                                          #
#    1. Finn optimale kontraktsparametre for en            #
#    Paretofordelt risiko med xmin = 60 og b = 2           #
#    alpha = 0.01, theta = 2.0                             #
#                                                          #
#    2. Estimer pris for reassuransekontrakten ved Monte   #
#    Carlo simulering, og sammenlign avsetning med og      #
#    uten reassuranse.                                     #
#                                                          #
############################################################

from scipy.stats import pareto
from random import paretovariate

alpha = 0.01
theta = 2.00

gamma = 1 / (1 + theta)

opt_a = pareto.isf(gamma, b=2, scale=60)
opt_b = pareto.isf(alpha, b=2, scale=60)

print(f"Optimal a: {opt_a:.3f}")
print(f"Optimal b: {opt_b:.3f}")
```

Oppgave 8 (forts.)

```
N = 100000
R_sum = 0

for j in range(N):
    X = 60 * paretovariate(2)
    if X <= opt_a:
        R_sum += 0
    elif opt_a <= X and X <= opt_b:
        R_sum += (X - opt_a)
    else:
        R_sum += (opt_b - opt_a)

reinsurancePrice = (1 + theta) * R_sum / N
VaR_WithoutReinsurance = opt_b
VaR_WithReinsurance = opt_a + reinsurancePrice
netGain = VaR_WithoutReinsurance - VaR_WithReinsurance

print(f"Reinsurance price: {reinsurancePrice:.3f}")
print(f"VaR without reinsurance: {VaR_WithoutReinsurance:.3f}")
print(f"VaR with reinsurance: {VaR_WithReinsurance:.3f}")
print(f"netGain: {netGain:.3f}")
```

Output:

```
Optimal a: 103.923
Optimal b: 600.000
Reinsurance price: 85.083
VaR without reinsurance: 600.000
VaR with reinsurance: 189.006
netGain: 410.994
```