

Oblig 2 - STK 1100

Kevin Alexander Aalesen

Oppgave 1

$$\begin{aligned} \text{a) } \int_0^1 \int_0^{1-x} 2(x+2y) dy dx &= \int_0^1 2x \int_0^{1-x} dy dx + \int_0^1 \int_0^{1-x} 4y dy dx \\ &\quad \uparrow \\ &\quad k=2 \\ &= \int_0^1 2x(1-x) dx + \int_0^1 [2y^2]_0^{1-x} dx \\ &= \int_0^1 2x - 2x^2 dx + \int_0^1 2(1-x)^2 dx \\ &= \left[x^2 - \frac{2}{3}x^3 \right]_0^1 + \left[-\frac{2}{3}(1-x)^3 \right]_0^1 \\ &= 1 - \frac{2}{3} + \left(-0 + \frac{2}{3} \right) \\ &= 1 \end{aligned}$$

k må altså være lik 2 for at den totale sannsynligheten skal bli 1 og $f(x,y)$ skal være en gyldig sannsynlighetsfølhet.

$$\begin{aligned}
 b) \quad f_Y(y) &= \int_0^{1-y} f(x,y) dx = \int_0^{1-y} 2(x+z_y) dx \\
 &= \int_0^{1-y} 2x dx + \int_0^{1-y} 4y dx \\
 &= [x^2]_0^{1-y} + 4y [x]_0^{1-y} \\
 &= (1-y)^2 + 4y(1-y) \\
 &= 1 - 2y + y^2 + 4y - 4y^2 \\
 &= 1 + 2y - 3y^2
 \end{aligned}$$

Vi har da at:

$$F_Y(y) = \begin{cases} 1 + 2y - 3y^2, & 0 \leq y \leq 1 \\ 0, & \text{ellers} \end{cases}$$

$$\textcircled{a)} \quad f_{X|Y}(x,y) = \frac{f(x,y)}{f_Y(y)} = \frac{2(x+z_y)}{1+2y-3y^2} = \frac{2x+4y}{1+2y-3y^2}$$

d) X og Y er ikke uavhengige ettersom uttrykket i c)

ikke kan faktoriseres slik at $f_{X|Y}(x,y) = \frac{f_x(x) \cdot f_y(y)}{f_Y(y)} = f_x(x)$

Oppgave 2

a) Når $U \sim \text{uniform}(0,1)$ så har vi at den kumulative funksjonen til U er:

$$f(u) = \frac{1}{B-A}$$

$$F(u) = P(U \leq u) = \int_0^u \frac{1}{1-0} du = [u]_0^u = u$$

Den kumulative funksjonen til U er lik U selv.

Vi har da at:

$$P(F^{-1}(U) \leq x) = P(U \leq F(x))$$

$$= F(x)$$

$$= P(X \leq x)$$

Altså har $X = F^{-1}(U)$ samme kumulativ funksjon som $F(x)$.

$$\begin{aligned}
 \hookrightarrow F_x(x) &= \int_0^x f_x(x) dx = \int_0^x \frac{\alpha}{\lambda} \left(1 + \frac{x}{\lambda}\right)^{-\alpha-1} dx \\
 &= \frac{\alpha}{\lambda} \left[-\frac{\lambda}{\alpha} \left(1 + \frac{x}{\lambda}\right)^{-\alpha} \right]_0^x \\
 &= \left[-\left(1 + \frac{x}{\lambda}\right)^{-\alpha} \right]_0^x \\
 &= -\left(1 + \frac{x}{\lambda}\right)^{-\alpha} + \left(1 + 0\right)^{-\alpha} \\
 &= 1 - \left(1 + \frac{x}{\lambda}\right)^{-\alpha}
 \end{aligned}$$

ellers er $F_x(x) = 0$. Derned har vi at:

$$F_x(x) = \begin{cases} 1 - \left(1 + \frac{x}{\lambda}\right)^{-\alpha}, & x > 0 \\ 0, & \text{ellers} \end{cases}$$

Finner medianen ved å la den kumulative fordelingen være like 0.5 og løse for tiden x :

$$F_x(\tilde{x}) = 0.5 \Rightarrow 1 - \left(1 + \frac{\tilde{x}}{\lambda}\right)^{-\alpha} = 0.5$$

$$\left(1 + \frac{\tilde{x}}{\lambda}\right)^{-\alpha} = 0.5$$

$$\left(1 + \frac{\tilde{x}}{\lambda}\right)^{\alpha} = 2$$

$$1 + \frac{\tilde{x}}{\lambda} = 2^{1/\alpha}$$

$$1 + \frac{\tilde{x}}{x} = 2^{\frac{1}{\alpha}} \Rightarrow \underline{\underline{\tilde{x} = \lambda(2^{\frac{1}{\alpha}} - 1)}}$$

c) Hvis vi setter en variabel U til å ta en tilfeldig verdi mellom 0 og 1 (altså $U \sim \text{uniform}(0,1)$), så vil variablen $X = 1/F(u)$ tilsvare en tilfeldig observasjon fra Lomax-fordelingen hvis funksjonen F tilsvarer fordelingen fra b).

d) Kode:

```

1 import numpy as np
2
3 a = 3          # alpha
4 l = 48         # lambda
5 n = 10000      # antall observasjoner
6
7 def F(x):
8     # Kumulativ Lomax
9     return 1 - (1 + x/l)**(-a)
10
11 u = np.random.uniform(0, 1, n)
12 x = 1 / F(u)
13
14 median_sim = np.sort(x)[5000]
15 median_mod = l * (2**((1/a) - 1))
16
17 print(f"Simulering median: {median_sim:.3f}")
18 print(f"Modell median: {median_mod:.3f}")

```

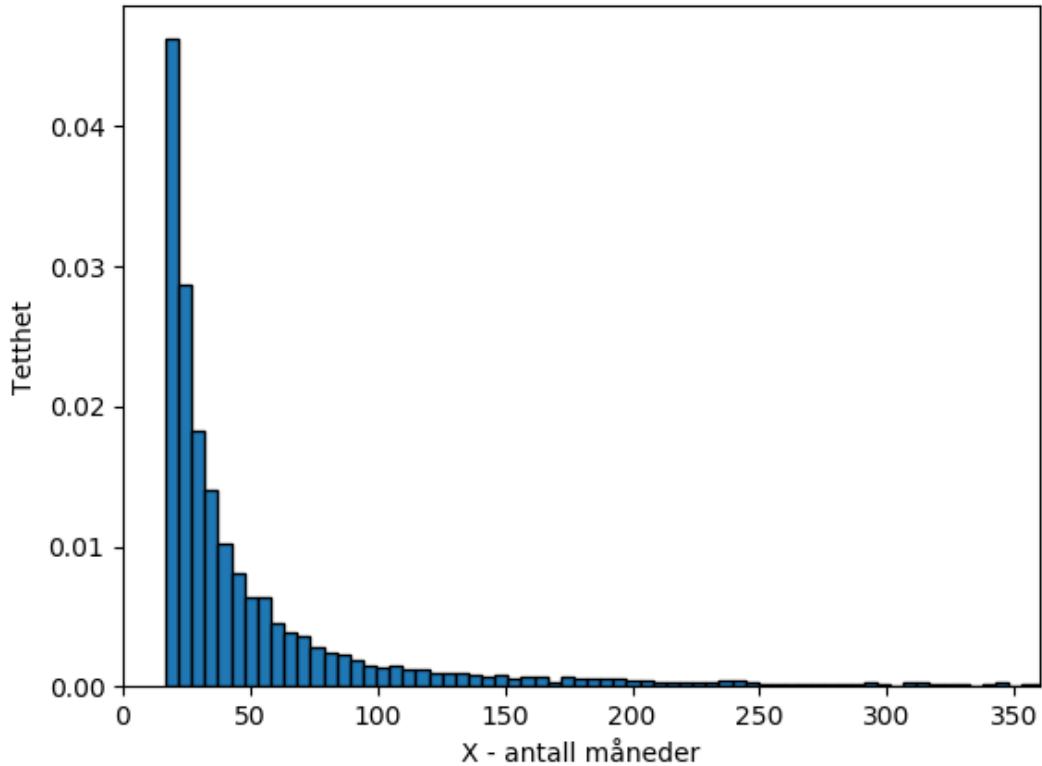
Resultat:

Simulering median: 32.795

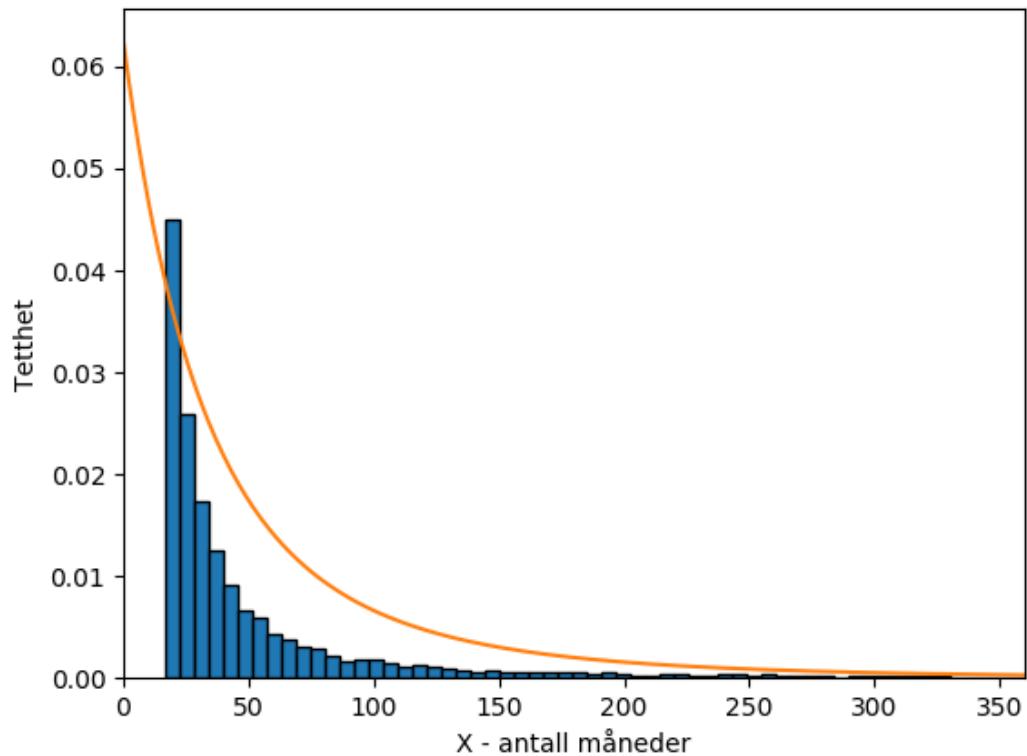
Modell median: 12.476

Ser at medianen fra de genererte observasjonene
er markant høyere enn medianen fra formellen
funnet i b). Klarer ikke å se hvorfor det
er så stor forskjell...

c) Normalt histogram av observasjonene:



→ Tettheten til Lomax-fordelingen med histogram:



heggar meir til at Lomax-fordelingen følger
samme form som toppene av histogrammet, som
tyder på at de genererte observasjonene faktisk
følgen Lomax-fordelingen som vi ønsket.

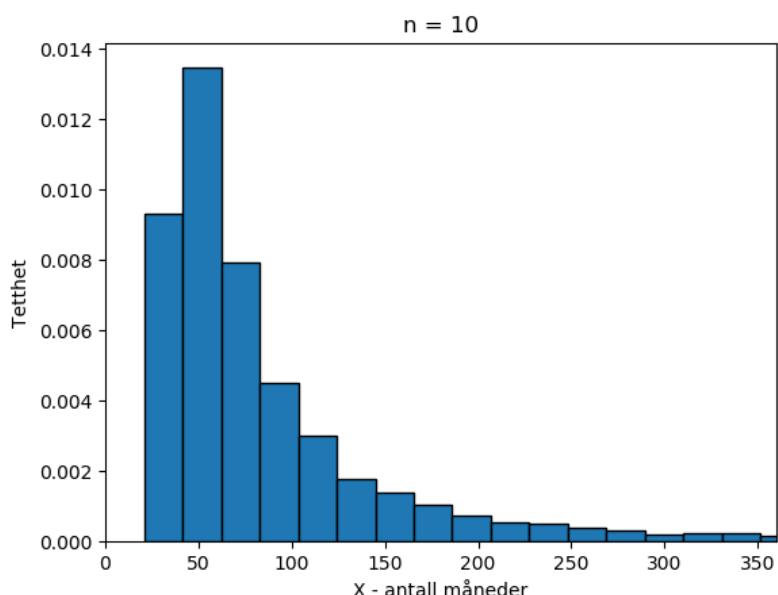
Koden berettet i ø) og f):

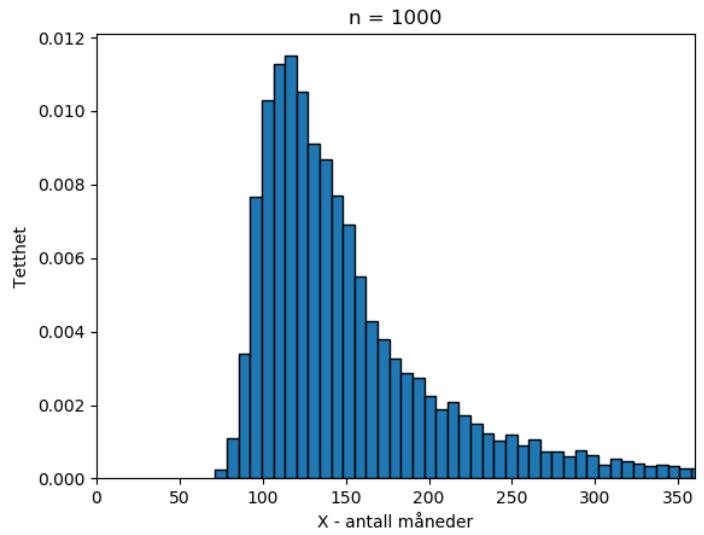
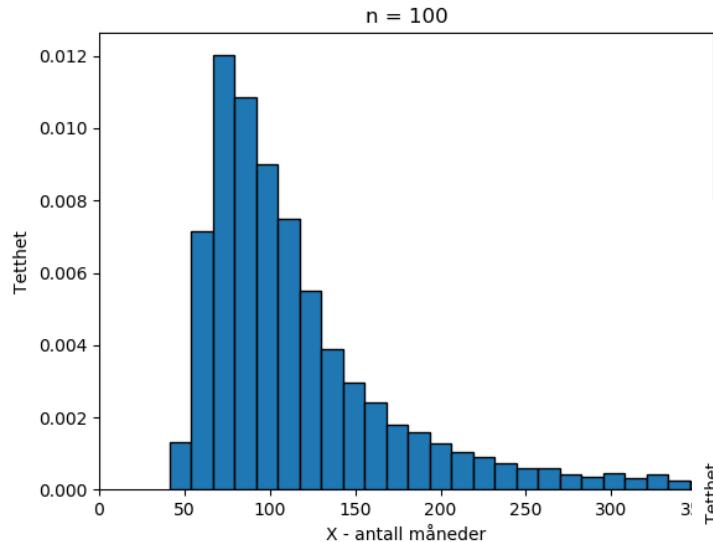
```
def f(x):
    # Lomax
    return a/l * (1 + x/l)**(-(a+1))

x_L = np.linspace(0, 360, 1000)

import matplotlib.pyplot as plt
plt.xlim(0,360)
plt.hist(x, density=True, edgecolor="black", bins=10000)
plt.plot(f(x_L))
plt.xlabel("X - antall måneder")
plt.ylabel("Tetthet")
plt.show()
```

g) Plotter histogrammer av gjennomsnittet for n=10, 100 og 1000 for Lomax fordelingen:





Det ser ut som snittet sentrer seg rundt $x = 120$ og at fordelingen har en hale som går mot høyre.

Koden brukt for å få disse plottene:

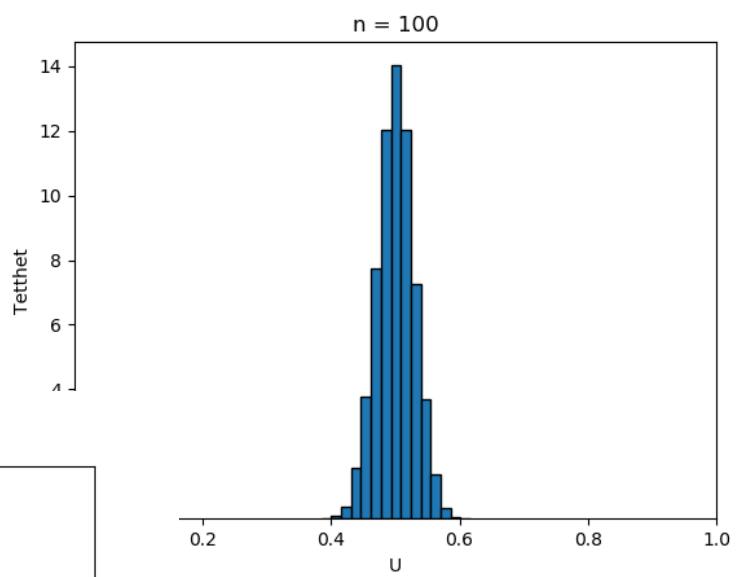
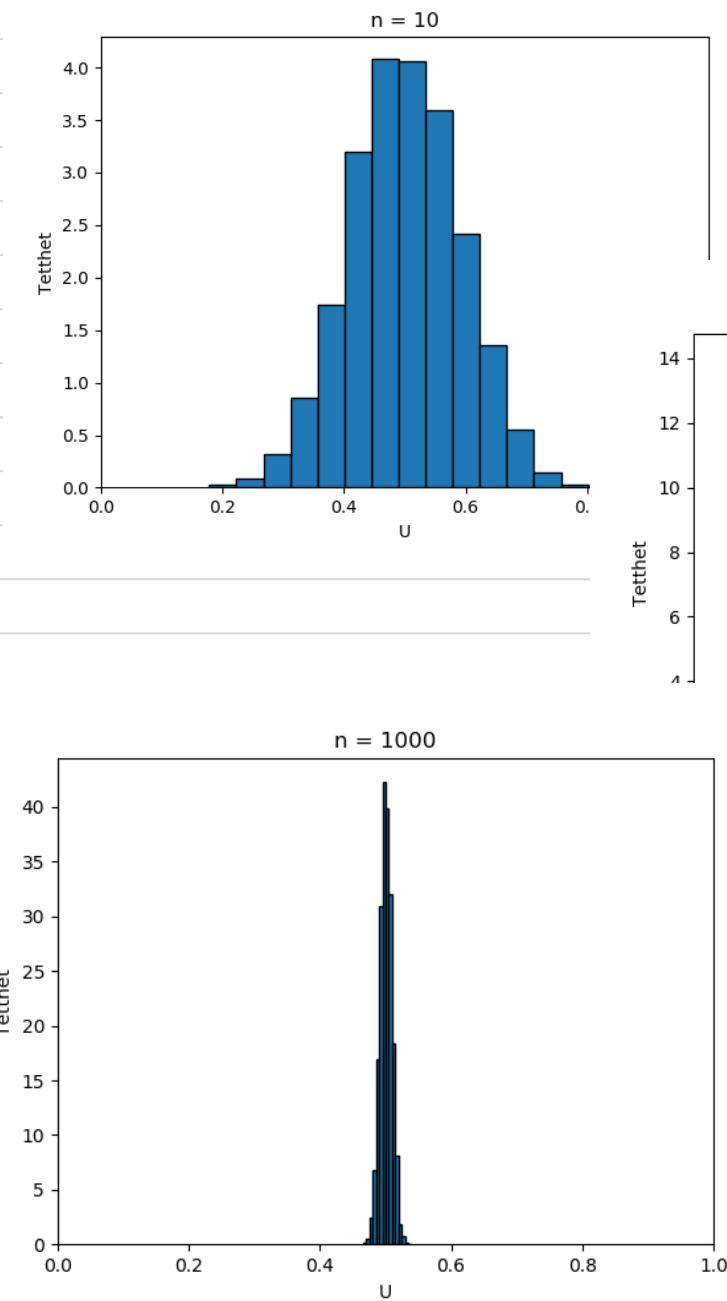
```

for n in [10, 100, 1000]:
    means = []
    for i in range(10000):
        u = np.random.uniform(0,1,n)
        x = 1/ F(u)
        mean = np.mean(x)
        means.append(mean)

    plt.xlim(0,360)
    plt.hist(means, density=True, edgecolor="black", bins=10000)
    plt.xlabel("X - antall måneder")
    plt.ylabel("Tetthet")
    plt.title(f"n = {n}")
    plt.show()

```

h) Gjentau g), men for den uniforme fordelingen på $(0, 1)$:



Dette var unntekst fra Lamex-fordelingen i g). Snittene

for den uniforme fordelingen konvergerer mye forttere

enn det for Lamex og har heller ingen hate som

i Lamex. Dette må skyldes av at den uniforme-fordelingen ikke har noen høyre og er fordelt høyt overalt slik at det ildre vil være noen verdier som ligg snittet langt ut til siden.

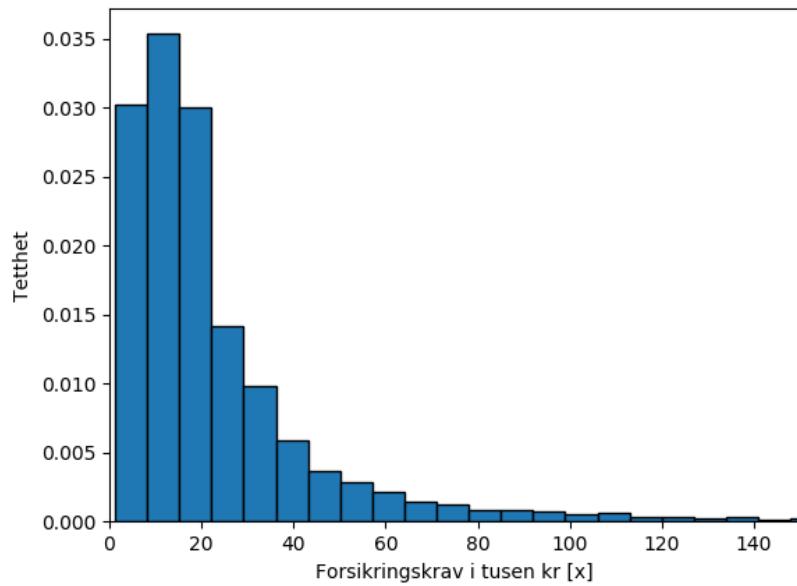
Koden benyttet for uniform-plottene:

```
for n in [10, 100, 1000]:
    means = []
    for i in range(10000):
        u = np.random.uniform(0,1,n)
        x = 1/ F(u)
        mean = np.mean(u) ← Benytter u istedenfor x
        means.append(mean)

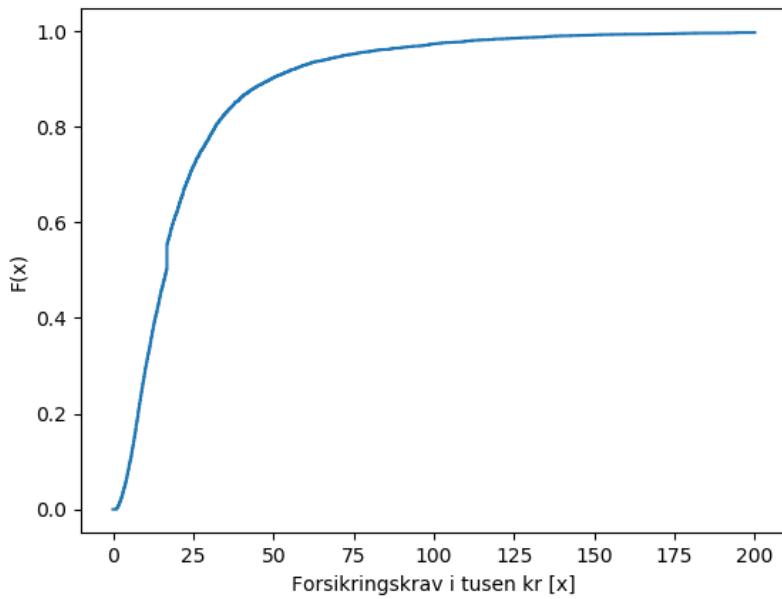
    plt.xlim(0,1)
    plt.hist(means, density=True, edgecolor="black", bins=15)
    plt.xlabel("U")
    plt.ylabel("Tetthet")
    plt.title(f"n = {n}")
    plt.show()
```

Oppgave 3

a) Histogram over forsikringskravene:



Plott av empirisk kumulativ fordeling:



Vi ser fra histogrammet og den kumulative fordelingen at de aller fleste forsikringskrav ligger under 50 000 kr. Den kumulative fordelingen tyder på at median knaret er rundt 20 000 kr, som passer godt med hvor histogrammet er sentvert.

$$b) \bar{x} = \alpha \beta \Rightarrow \alpha = \bar{x}/\beta$$

Setter inn i den andre ligningen:

$$s^2 = \bar{x}/\beta \cdot \beta^2 = \bar{x}\beta \Rightarrow \beta = s^2/\bar{x}$$

Setter tilbake inn i α :

$$\alpha = \bar{x}/s^2/\bar{x} = \bar{x}^2/s^2$$

Estimatene for α og β er altså:

$$\hat{\alpha} = \bar{x}^2/s^2 \text{ og } \hat{\beta} = s^2/\bar{x}$$

Beregner estimatorene numerisk og får:

$$\hat{\alpha} = 0.697 \text{ og } \hat{\beta} = 34.649$$

c) Hvis Y er normalfordelt så har den ptf lik:

$$f_Y(y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(y-\mu)^2/2\sigma^2}$$

V: når at $X = g(Y) = e^Y$ og $Y = g^{-1}(X) = \ln(x) = \ln X$.

Da er transformasjonen og tettheten til X lik:

$$\begin{aligned} f_X(x) &= f_Y(\ln(x)) \cdot |\ln(x)| \\ &= f_Y(\ln x) \cdot |(\ln x)'| \\ &= \frac{1}{\sqrt{2\pi}\sigma} e^{-(\ln x - \mu)^2/2\sigma^2} \cdot \left| \frac{1}{x} \right| \\ &= \frac{1}{\sqrt{2\pi}\sigma x} e^{-(\ln x - \mu)^2/2\sigma^2} \end{aligned}$$

d) Starter med å løse første ligning for μ :

$$\bar{x} = e^{\mu + \sigma^2/2} \Rightarrow \ln \bar{x} = \mu + \sigma^2/2 \Rightarrow \mu = \ln \bar{x} - \sigma^2/2$$

Setter μ inn i andre ligning:

$$S^2 = (e^{\sigma^2} - 1) e^{2\mu + \sigma^2} \Rightarrow \ln S^2 = \ln(e^{\sigma^2} - 1) + 2\mu + \sigma^2$$

(forts. neste side)

$$2 \ln S = \ln(e^{\sigma^2} - 1) + 2(\ln \bar{x} - \sigma^2 / \bar{x}) + \sigma^2$$

$$= \ln(e^{\sigma^2} - 1) + 2 \ln \bar{x}$$

$$\downarrow \\ \ln(e^{\sigma^2} - 1) = 2(\ln S - \ln \bar{x})$$

$$\ln(e^{\sigma^2} - 1) = \ln\left(\frac{s^2}{\bar{x}^2}\right)$$

$$e^{\sigma^2} - 1 = s^2 / \bar{x}^2$$

$$e^{\sigma^2} = s^2 / \bar{x}^2 + 1$$

$$\sigma^2 = \ln\left(\frac{s^2}{\bar{x}^2} + 1\right)$$

$$\sigma = \sqrt{\ln\left(\frac{s^2}{\bar{x}^2} + 1\right)}$$

Setter dette tilbake inn i μ :

$$\begin{aligned} \mu &= \ln \bar{x} - \frac{\sqrt{\ln\left(\frac{s^2}{\bar{x}^2} + 1\right)}}{2}^2 = \ln \bar{x} - \frac{1}{2} \ln\left(\frac{s^2}{\bar{x}^2} + 1\right) \\ &= \ln\left(\frac{\bar{x}}{\sqrt{\frac{s^2}{\bar{x}^2} + 1}}\right) \end{aligned}$$

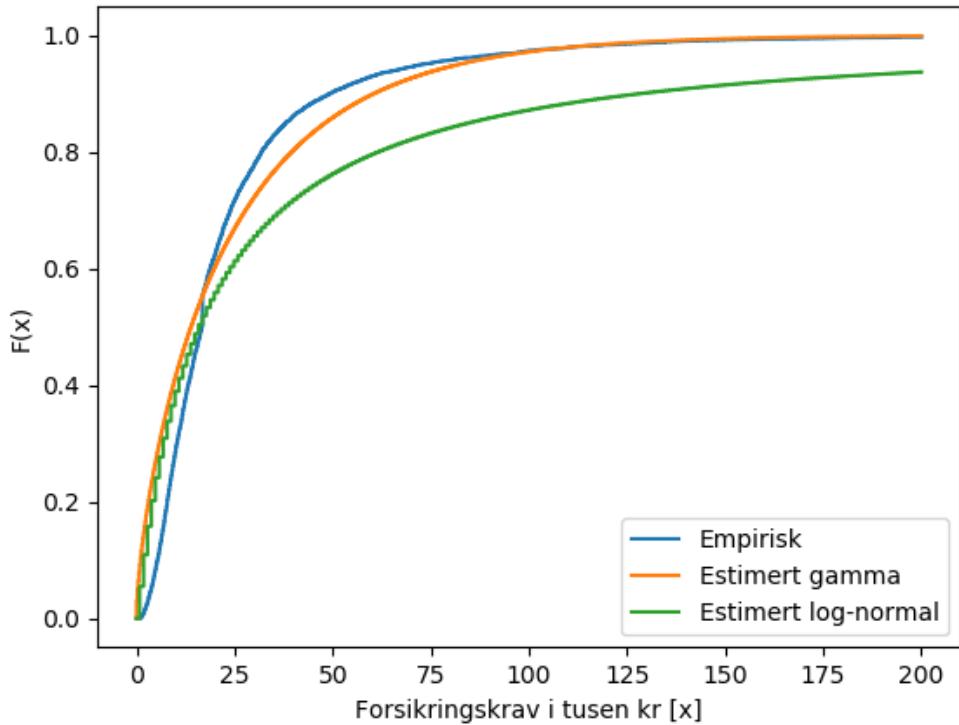
Momentestimatorene for μ og σ er altså:

$$\hat{\mu} = \ln\left(\frac{\bar{x}}{\sqrt{\frac{s^2}{\bar{x}^2} + 1}}\right) \text{ og } \hat{\sigma} = \sqrt{\frac{s^2}{\bar{x}^2} + 1}$$

Beregn estimatene numerisk og få:

$$\hat{\mu} = 2.739 \quad \text{og} \quad \hat{\sigma} = 1.561$$

c) Plotter de forskjellige kumulative fordelingene sammen:



Av disse tre modellene syns jeg ~~at~~ gamma fordelingen passer best ettersom den forklarer godt de store verdiene rett etter $x=0$ samt de lavere verdiene når $x > 100$, sammenlignet med log-normal fordelingen.

Koden benytter for oppg. 3 :

```
#a)
import pandas as pd
url = "https://www.uio.no/studier/emner/matnat/math/STK1100/data/forsikringskrav.txt"
forsikringskrav=pd.read_csv(url, header=None)[0]

import matplotlib.pyplot as plt
import numpy as np

plt.xlim(0, 150)
plt.hist(forsikringskrav, density=True, edgecolor="black", bins=100)
plt.xlabel("Forsikringskrav i tusen kr [x]")
plt.ylabel("Tetthet")
plt.show()

from statsmodels.distributions.empirical_distribution import ECDF
ecdf2 = ECDF(forsikringskrav)
z = np.linspace(0, 200, 1000)
plt.step(z, ecdf2(z))
plt.xlabel("Forsikringskrav i tusen kr [x]")
plt.ylabel("F(x)")
plt.show()
```

```
#b)
mean = np.mean(forsikringskrav)
sd = np.std(forsikringskrav)

alpha = mean**2 / sd**2
beta = sd**2 / mean

print(f"Alpha: {alpha:.3f}")
print(f"Beta: {beta:.3f}")
```

```
#d)
my = np.log(mean / (np.sqrt(sd**2/mean**2 + 1)))
sigma = np.sqrt(sd**2/mean**2 + 1)

print(f"My: {my:.3f}")
print(f"Sigma: {sigma:.3f}")
```

```
#e)
from scipy import stats

def lognormal(x):
    return (1/(np.sqrt(2*np.pi)*sigma*x)) * np.exp(-(np.log(x)-my)**2 / (2*sigma**2))

def lcdf(x):
    cdf = []
    for i in x:
        sum = 0
        for j in range(1, int(i)+1):
            sum += lognormal(j)
        cdf.append(sum)
    return cdf

plt.step(z, ecdf2(z), label="Empirisk")
plt.step(z, stats.gamma.cdf(z, a=alpha, scale=beta), label="Estimert gamma")
plt.step(z, lcdf(z), label="Estimert log-normal")
plt.xlabel("Forsikringskrav i tusen kr [x]")
plt.ylabel("F(x)")
plt.legend()
plt.show()
```