15. april 2024

STK1100

Obligatorisk oppgavesett 2 av 2.

Innleveringsfrist

Torsdag 2. mai 2024, klokken 14:30 i Canvas (canvas.uio.no).

Instruksjoner

Du velger selv om du skriver besvarelsen for hånd og scanner besvarelsen eller om du skriver løsningen direkte inn på datamaskin (for eksempel ved bruk av LaTeX). NB! Besvarelsen skal leveres som én PDF-fil. Scannede ark må være godt lesbare. Besvarelsen skal inneholde navn, emne og oblignummer.

Det forventes at man har en klar og ryddig besvarelse med tydelige begrunnelser. Husk å inkludere alle relevante plott og figurer, med forklaringer. Besvarelser som kun består av kode, godkjennes ikke. **Besvarelser som ikke har forsøkt på oppgavene som krever programmering, blir heller ikke godkjent.** Samarbeid og alle slags hjelpemidler er tillatt, men den innleverte besvarelsen skal være skrevet av deg og reflektere din forståelse av stoffet. Er vi i tvil om du virkelig har forstått det du har levert inn, kan vi be deg om en muntlig redegjørelse.

I oppgaver der du blir bedt om å programmere må du legge ved programkoden og levere den sammen med resten av besvarelsen.

Søknad om utsettelse av innleveringsfrist

Hvis du blir syk eller av andre grunner trenger å søke om utsettelse av innleveringsfristen, må du ta kontakt med studieadministrasjonen ved Matematisk institutt (e-post: studieinfo@math.uio.no) i god tid før innleveringsfristen.

For å få adgang til avsluttende eksamen i STK1100, må man bestå begge de obligatoriske oppgavesettene i ett og samme semester.

For fullstendige retningslinjer for innlevering av obligatoriske oppgaver, se her:

www.uio.no/studier/admin/obligatoriske-aktiviteter/mn-math-oblig.html

Spesielt om det obligatoriske oppgavesettet i STK1100

Det anbefales på det sterkeste at du bruker Python til å gjøre beregningene i oppgave 2 og 3. Hvis du bruker et annet programmeringsspråk, kan vi ikke hjelpe deg hvis du får problemer. Uansett hvilket programmeringsspråk du bruker, må du angi hvilke kommandoer du har brukt for å komme fram til svarene dine. Hvis du trenger hjelp til å løse oppgavene, kan du få det på en av de åpne gruppene i STK1100.

LYKKE TIL!

Oppgave 1. De stokastiske variablene X og Y har simultan sannsynlighetstetthet

$$f(x,y) = \begin{cases} k(x+2y) & \text{for } 0 \le x \le 1,\ 0 \le y \le 1,\ x+y \le 1 \\ 0 & \text{ellers,} \end{cases}$$

der k er en konstant.

a) Vis at $k = 2$.

b) Vis at den marginale sannsynlighetstettheten til Y er

$$f_Y(y) = \begin{cases} 1 + 2y - 3y^2 & \text{for } 0 \le y \le 1 \\ 0 & \text{ellers} \end{cases}$$

c) Bestem den betingede sannsynlighetstettheten til $X|Y = y$.

d) Er X og Y uavhengige? Svaret skal begrunnes!

Oppgave 2. I denne oppgaven skal vi se hvordan du kan bruke en datamaskin til å generere observasjoner fra Lomax-fordelingen, som er oppgitt under. Vi tar som utgangspunkt at datamaskiner kan generere tilfeldige tall i intervallet $[0,1]$, dvs. observasjoner av en stokastisk variabel U som er uniformt fordelt på $[0,1]$.

a) La $F(x)$ være en kumulativ fordeling der den inverse funksjonen $F^{-1}(u)$ eksisterer. Anta at $U \sim \text{uniform}(0,1)$. Vis at da har $X = F^{-1}(U)$ kumulativ fordeling $F(x)$.

La X være tida fram til en bedrift går konkurs. Da er det er vanlig å anta at X er Lomax-fordelt, det vil si at X har sannsynlighetstettheten

$$f_X(x) = \begin{cases} \frac{\alpha}{\beta}\left(1 + \frac{x}{\beta}\right)^{-(\alpha+1)} & \text{for} \quad x > 0, \\ 0 & \text{ellers.} \end{cases}$$

Her er $\alpha, \beta > 0$ parametere i fordelingen.

b) Den kumulative sannsynlighetsfordelingen til X er gitt ved

$$F_X(x) = \begin{cases} 1 - \left(1 + \frac{x}{\beta}\right)^{-\alpha} & \text{for} \quad x > 0, \\ 0 & \text{ellers.} \end{cases}$$

Bestem median tid fram til en tilfeldig bedrift går konkurs.

c) Bruk resultatene i punkt a) og b) til å angi en framgangsmåte for å generere observasjoner fra Lomax-fordelingen.

d) Bruk framgangsmåten i forrige punkt til å generere 10 000 observasjoner fra Lomax-fordelingen med $\beta = 48$ måneder og $\alpha = 3$. Beregn medianen av de genererte observasjonene, og sammenlign med resultatene i punkt b).

PYTHON hjelp: Du kan generere n observasjoner av $U \sim \text{uniform}(0,1)$ ved kommandoene

```
from numpy import *
u=np.random.uniform(0,1,n)
```

e) Lag et normert histogram av de Lomax-fordelte observasjonene fra punkt d) (dvs. et histogram der arealet av alle stolpene til sammen er lik én.)

PYTHON hjelp: Du kan tegne et normert histogram av observasjoner i en vektor `x` ved kommandoene

```
import matplotlib.pyplot as plt
plt.hist(x,density=True,edgecolor="black")
```

Siden noen av observasjonene fra Lomax-fordelingen vil være veldig store, må du avgrense de `x`-verdiene du plotter histogrammet for. For eksempel får du tegnet histogrammet over intervallet fra 0 til 360 måneder ved kommandoene

```
plt.xlim(0,360)
plt.hist(x,density=True,edgecolor="black",bins=50).
```

Noen ganger kan du få så mange store verdier at du må jobbe litt med antall bins i histogrammet for å få det pent.

f) Tegn tettheten til Lomax-fordelingen med $\beta = 48$ måneder og $\alpha = 3$ i samme figur som histogrammet og kommentér resultatet.

g) Lomax-fordelingen er en fordeling med tung hale. Du skal se på hvordan dette slår ut for sentralgrenseteoremet i praksis. Illustrér vha. simuleringer hvordan den empiriske fordelingen til gjennomsnittet av n Lomax-fordelte observasjoner nærmer seg normalfordelingen for $n = 10, 100$ og 1000. Dette gjør du ved å simulere 10 000 utvalg av størrelse n[1], ta gjennomsnittet i hvert utvalg, og se på hvordan de 10 000 gjennomsnittene fordeler seg, enten ved å lage histogrammer, eller ved å se på den empiriske kumulative fordelingen, og sammenligne med standard normalfordeling. La deg inspirere av forelesningen fra kapittel 6. Du kan få bruk for å vite at forventning og standardavvik i Lomax-fordelingen med parametre β og α er gitt ved

$$\mathbb{E}(X) = \frac{\beta}{\alpha - 1} \quad \text{og} \quad \mathrm{Sd}(X) = \frac{\beta}{\alpha - 1}\sqrt{\frac{\alpha}{\alpha - 2}}$$

h) Gjenta punkt g), men denne gangen med den uniforme fordelingen på $(0, 1)$. Du skal med andre ord gjøre akkurat det samme som i forrige punkt, bortsett fra at du genererer og tar gjennomsnittet av uniformt fordelte observasjoner i stedet for Lomax-fordelte. Sammenligne den empiriske fordelingen for gjennomsnittet av de uniformt fordelte observasjonene med den for de Lomax-fordelte i punkt g). Kommentér og forklar den forskjellen du ser i resultatene for Lomax-fordelingen og den uniforme fordelingen.

Oppgave 3. I denne oppgaven skal du studere fordelingen til størrelsen på forsikringskrav til et gitt forsikringsselskap. Dette er altså størrelsen (i tusen kroner) på alle kravene kundene har sendt inn til selskapet i løpet av en gitt tidsperiode. Det er 6377 registrerte krav i datasettet. Datasettet finner du i filen `forsikringskrav.txt` under fanen Data på hjemmesiden til kurset.

PYTHON hjelp: Du kan lese inn datasettet ved kommandoene:

```
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]
```

[1] Bruk kommandoen `np.random.uniform(0,1,size=(n,10000))` for å generere U-ene.

For et utvalg $x_1, x_2, \ldots, x_n$ er den *empiriske kumulative fordelingsfunksjonen* gitt ved

$$\widehat{F}(x) = \frac{1}{n}\sum_{i=1}^{n} I(x_i \leq x),$$

som altså betyr at $\widehat{F}(x)$ er gitt ved *andelen* av alle observasjonene i utvalget som er mindre enn eller lik x.

a) Lag et histogram over forsikringskravene. Lag også et plott av den empiriske kumulative fordelingsfunksjonen for kravene. Kommenter kort om formen på den empiriske fordelingen.

PYTHON hjelp: Du kan enten bruke ECDF fra statsmodels, som vi brukte i forelesningene, til å plotte den empiriske kumulative fordelingen, på denne måten:

```
from statsmodels.distributions.empirical_distribution import ECDF
ecdf2 = ECDF(forsikringskrav)
z = np.linspace(0, 200, 1000)
plt.step(z, ecdf2(z))
```

Eller du kan programmere funksjonen selv:

```
import numpy as np
from scipy import interpolate as inter

def lag_empirisk_fordelingsfunksjon(x):
    empirisk_fordeling = inter.interp1d(np.sort(x),
                          np.arange(len(x))/float(len(x)),
                          kind = "zero",
                          fill_value = "extrapolate")
    return empirisk_fordeling
ecdf = lag_empirisk_fordelingsfunksjon(forsikringskrav)
z = np.linspace(0, 200, 1000)
plt.step(z, ecdf(z))
```

En mulig parametrisk modell for forsikringskrav, er gammafordelingen, altså en modell med tetthet

$$f(x) = \frac{x^{\alpha-1}e^{-\frac{x}{\beta}}}{\Gamma(\alpha)\beta^{\alpha}}, \qquad x > 0.$$

I gammafordelingen er $\mathbb{E}(X) = \alpha\beta$, og $\mathrm{Var}(X) = \alpha\beta^2$.

b) Finn estimatorer for α og β ved å løse ligningssettet under for α og β

$$\overline{x} = \alpha\beta,$$

$$S^2 = \alpha\beta^2,$$

der $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$, og $S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2$. Disse kalles for momentestimatorene for α og β. Finn de tilhørende estimatene for α og β ved å sette inn de observerte forsikringskravene.

En annen mulig parametrisk modell for forsikringskrav er en såkalt *log-normal* fordeling. Denne fordelingen kan man konstruere som $X = e^Y$, der Y er normalfordelt med forventning μ og varians σ^2. For den log-normale fordelingen har man at

$$\mathbb{E}(X) = \exp\left(\mu + \frac{\sigma^2}{2}\right),$$

og at

$$\mathrm{Var}(X) = \left[\exp(\sigma^2) - 1\right] \exp\left(2\mu + \sigma^2\right).$$

c) Finn momentestimatorene for μ og σ ved å løse ligningssettet under for μ og σ

$$\bar{x} = \exp\left(\mu + \frac{\sigma^2}{2}\right),$$

og

$$S^2 = \left[\exp(\sigma^2) - 1\right] \exp\left(2\mu + \sigma^2\right).$$

Finn de tilhørende estimatene for μ og σ ved å sette inn de observerte forsikringskravene slik som i punkt b).

d) Lag en figur, der du plotter den empiriske kumulative fordelingen i a) sammen med de estimerte kumulative modellene basert på log normal- og gamma-fordelingene fra b) og c). Hvilken modell syns du passer best? Begrunn svaret ditt.

PYTHON hjelp: For å plotte den kumulative fordeling til gamma(α, β) fordelingen, så kan du bruke følgende kommandoer:

```
from scipy.stats import gamma
z = np.linspace(0,200,1000)
plt.step(z,gamma.cdf(z/beta,alpha))
```

Tilsvarende for å plotte den kumulative til lognormal fordelingen med parametre μ, σ, så kan du bruke kommandoenen

```
from scipy.stats import lognorm
z = np.linspace(0,200,1000)
plt.step(z,lognorm.cdf(z,s=sigma,scale=exp(mu)))
```