ESMA 5015: Asignacion #1a

Due on March 25, 2025

Damaris Santana

Alejandro Ouslan

Problem 1

Demueste como generar una variable aleatoria $Cauchy(a, b)$ a partir de una variable aleatoria $u \sim unif(0, 1)$.

1. Proceso matematico

$$
\begin{aligned}
f(x) &= \frac{1}{\pi}\frac{b}{(x-a)^2+b^2} \\
F_X(x) &= \int_{-\infty}^{x} f(x)dx \\
F_X(x) &= \int_{-\infty}^{x} \frac{2}{\pi}\frac{b}{(x-a)^2+b^2} \\
F_X(x) &= \frac{1}{\pi}\arctan(\frac{x-a}{b}) + \frac{1}{2} \\
u &= \frac{1}{\pi}\arctan(\frac{x-a}{b}) + \frac{1}{2} \\
u - \frac{1}{2} &= \frac{1}{\pi}\arctan(\frac{x-a}{b}) \\
\pi(u - \frac{1}{2}) &= \arctan(\frac{x-a}{b}) \\
\tan(\pi(u - \frac{1}{2})) &= \frac{x-a}{b} \\
b\tan(\pi(u - \frac{1}{2})) + a &= x \\
x &= a + b\tan(\pi(u - \frac{1}{2}))
\end{aligned}
$$

2. algorithms
   (a) Generar $u \sim U(0, 1)$
   (b) Definir $x = a + b\tan(\pi(u - \frac{1}{2}))$
3. Graphicas y implementacion

```python
import numpy as np
import matplotlib.pyplot as plt

rng = np.random.default_rng(seed=787)
x = rng.uniform(0,1,1000000)
a,b = 0,1

data = b * np.tan(np.pi*(x-0.5)) + a
data = data[(data>-25) & (data<25)]
```

Listing 1: Generacion de variable aleatoria Cauchy

```python
s = rng.standard_cauchy(1000000)

s = s[(s>-25) & (s<25)]

plt.hist([data, s], bins=100, label=['Simulated Caucy', 'Actual Caucy'])
plt.legend(loc='upper right')
plt.show()
```

Listing 2: Grafica de la variable aleatoria Cauchy

Problem 1 continued on next page...

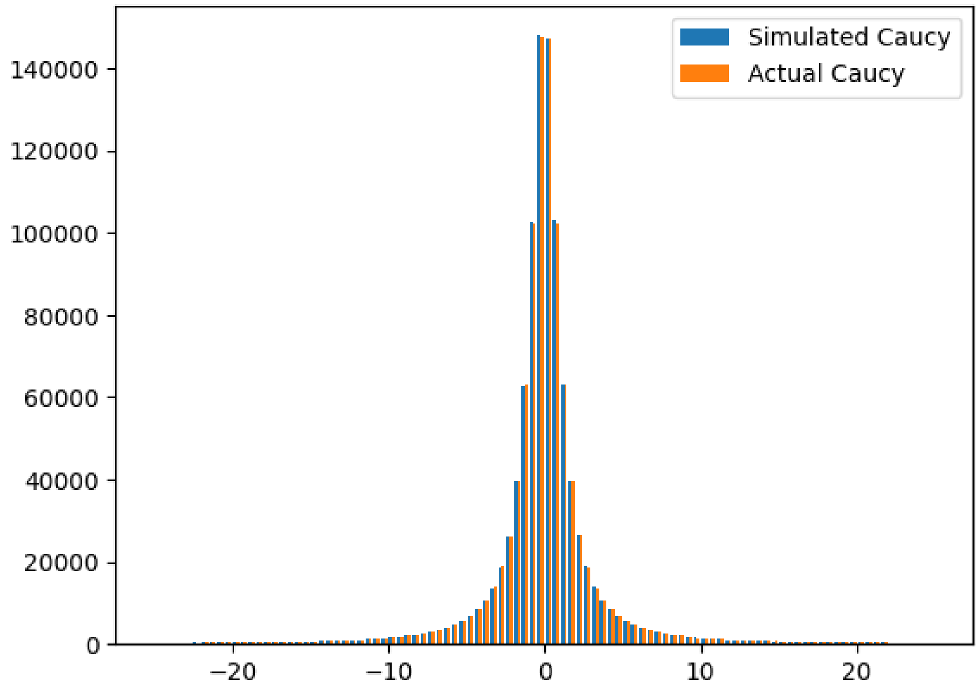

Figure 1: Histograma de la variable aleatoria Cauchy

Problem 2

Demueste como generar una variable aleatoria $X \sim DoubleExp(\mu, \beta)$ a partir de una variable aleatoria $u \sim unif(0, 1)$.

1. Proceso Matematico

$$
\begin{aligned}
f(x) &= \frac{1}{\beta} e^{-\frac{x-\mu}{\beta} + e^{-\frac{x-\mu}{\beta}}} \\
F_X(x) &= \int_{-\infty}^{x} f(x) dx \\
F_X(x) &= \int_{-\infty}^{x} \frac{1}{\beta} e^{-\frac{x-\mu}{\beta} + e^{-\frac{x-\mu}{\beta}}} \\
F_X(x) &= e^{-e^{-\frac{x-\mu}{\beta}}} \\
u &= e^{-e^{-\frac{x-\mu}{\beta}}} \\
\log(-\log(u)) &= -\frac{x-\mu}{\beta} \\
\beta \log(-\log(u)) + \mu &= x \\
x &= \beta \log(-\log(u)) + \mu
\end{aligned}
$$

2. algorithms

   (a) Generar $u \sim U(0, 1)$

   (b) Definir $x = \beta \log(-\log(u)) + \mu$

Problem 2 continued on next page. . .

3. Graphicas

```python
import numpy as np
import matplotlib.pyplot as plt

rng = np.random.default_rng(seed=787)
x = rng.uniform(0,1,1000000)
mu, beta = 0, 0.1

data = beta*(-np.log(-np.log(x))+mu)
data = data[(data>-1) & (data<1)]
```

Listing 3: Generacion de variable aleatoria DoubleExp

```python
s = rng.gumbel(mu, beta, 1000000)
s = s[(s>-1) & (s<1)]


plt.hist([data, s], bins=100, label=['Simulated Gumbel ', 'Actual Gumbel'])
plt.legend(loc='upper right')
plt.show()
```

Listing 4: Grafica de la variable aleatoria DoubleExp

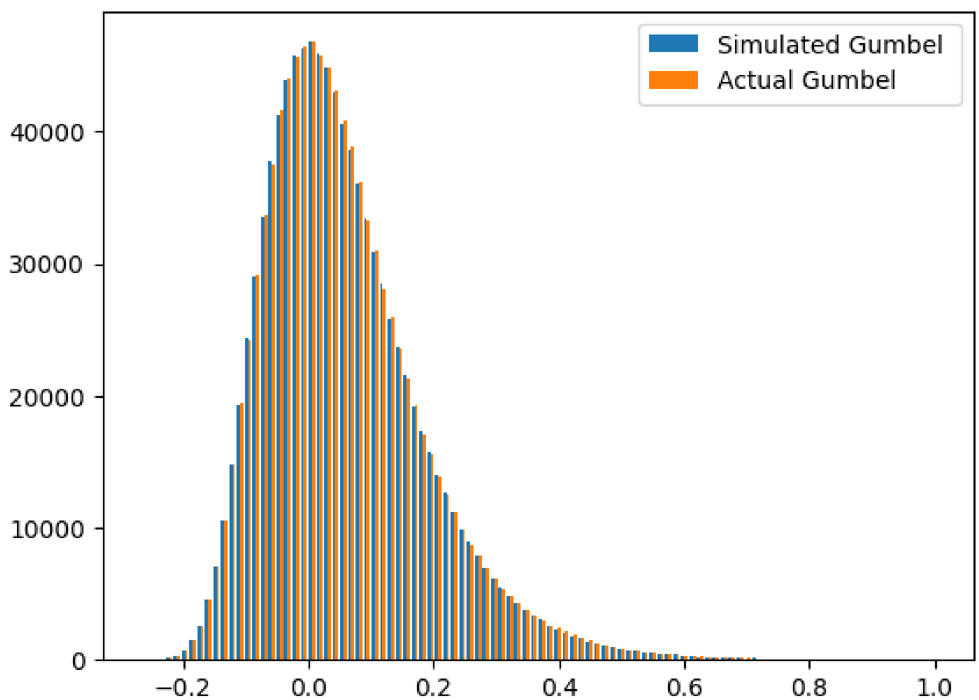

Figure 2: Histograma de la variable aleatoria DoubleExp