

Método de la Transformada Inversa

Curso: Temas Selectos I: O25 LAT4032 1

Profesor: Rubén Blancas Rivera

Equipo: Ana Ximena Bravo, Heriberto Espino, Celeste Núñez

Universidad de las Américas Puebla

2 de Septiembre de 2025

Índice

```
[1]: import math
import random
from collections import Counter

import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
[2]: color = sns.color_palette("muted")
np.random.shuffle(color)
sns.set(style="whitegrid", context="paper", palette=color)

import matplotlib as mpl
mpl.rcParams.update({
    "text.usetex": True,           # route all text through LaTeX
    "pgf.texsystem": "xelatex",
    "pgf.rcfonts": False,         # do not override with mpl fonts
    "font.family": "serif",
    "text.latex.preamble": r"""
\usepackage{libertinust1math}
""",
})
```

1. Distribución Uniforme (a, b)

Sea $U \sim \text{Unif}(0, 1)$. Si $X \sim \text{Unif}(a, b)$, entonces su función de distribución acumulada es:

$$F_X(x) = \frac{x-a}{b-a} \mathbf{1}_{[a,b]}(x) + \mathbf{1}_{(b,\infty)}(x)$$

Encontrando la inversa:

$$\begin{aligned} F_X(x) = u &\iff \frac{x-a}{b-a} = u, \\ &\iff x-a = (b-a)u, \\ &\iff x = a + (b-a)u. \end{aligned}$$

Entonces:

$$F_X^{-1}(u) = a + (b-a)u.$$

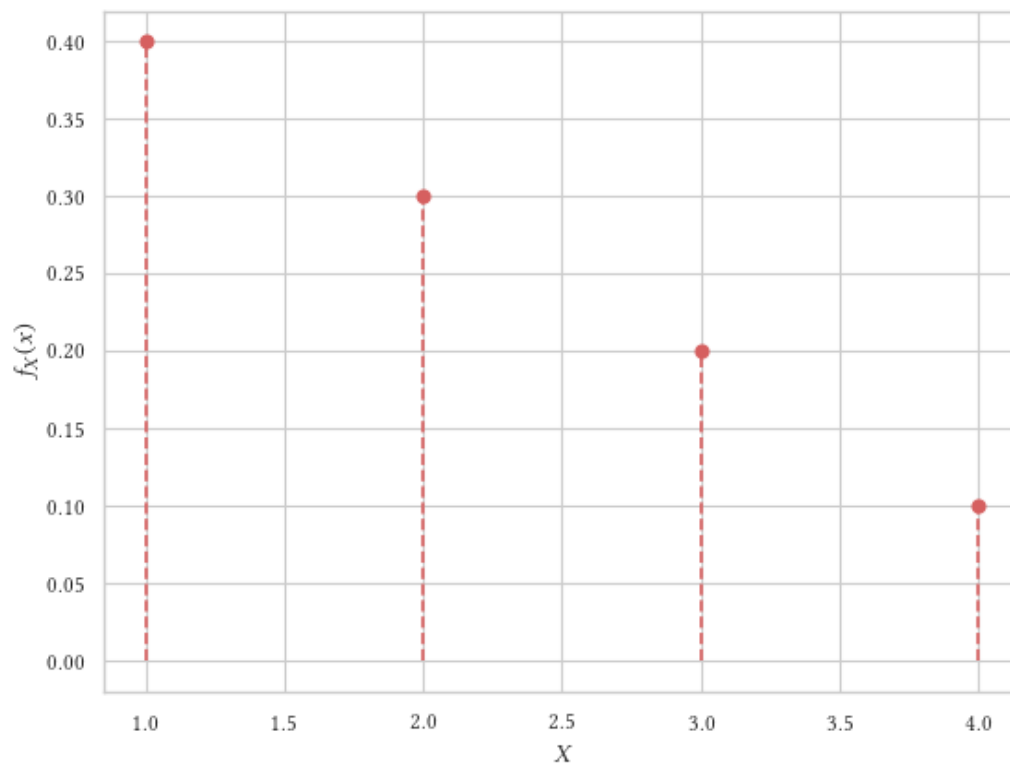
2. Distribución Discreta

a) Gráfica de $f_X(x)$

$$f_X(x) = \mathbb{P}(X = x) = \begin{cases} 0.4, & x = 1, \\ 0.3, & x = 2, \\ 0.2, & x = 3, \\ 0.1, & x = 4, \\ 0, & \text{en otro caso} \end{cases}$$

```
[3]: x_vals = [1, 2, 3, 4]
      pmf = [0.4, 0.3, 0.2, 0.1]

      plt.vlines(x_vals, 0, pmf, linestyle='--')
      plt.plot(x_vals, pmf, 'o')
      plt.xlabel('$X$')
      plt.ylabel('$f_X(x)$')
      plt.show()
```

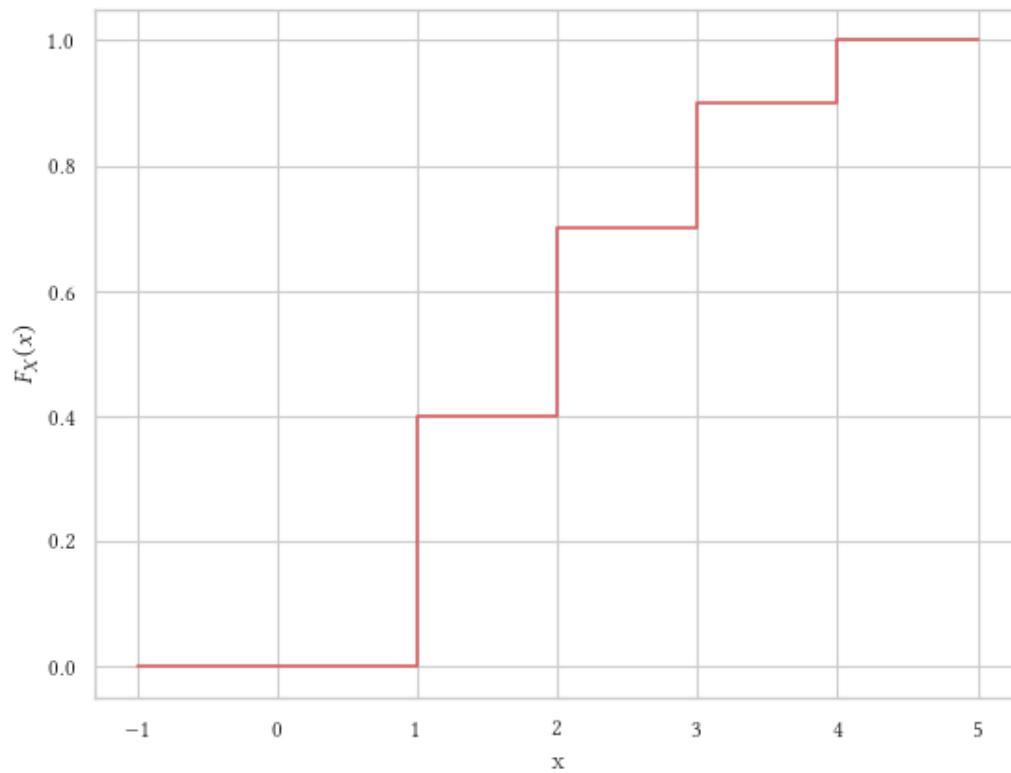


b) Gráfica de $F_X(x)$

$$F_X(x) = \mathbb{P}(X \leq x) = \begin{cases} 0, & x < 1, \\ 0.4, & 1 \leq x < 2, \\ 0.7, & 2 \leq x < 3, \\ 0.9, & 3 \leq x < 4, \\ 1, & x \geq 4. \end{cases}$$

```
[4]: x_cdf = [-1, 1, 2, 3, 4, 5]
      F_cdf = [0, 0.4, 0.7, 0.9, 1, 1]

      plt.step(x_cdf, F_cdf, where='post')
      plt.xlabel('x')
      plt.ylabel('$F_X(x)$')
      plt.show()
```



c) Gráfica de $F_X^{-1}(u)$

Sea $U \sim \text{Unif}(0, 1)$.

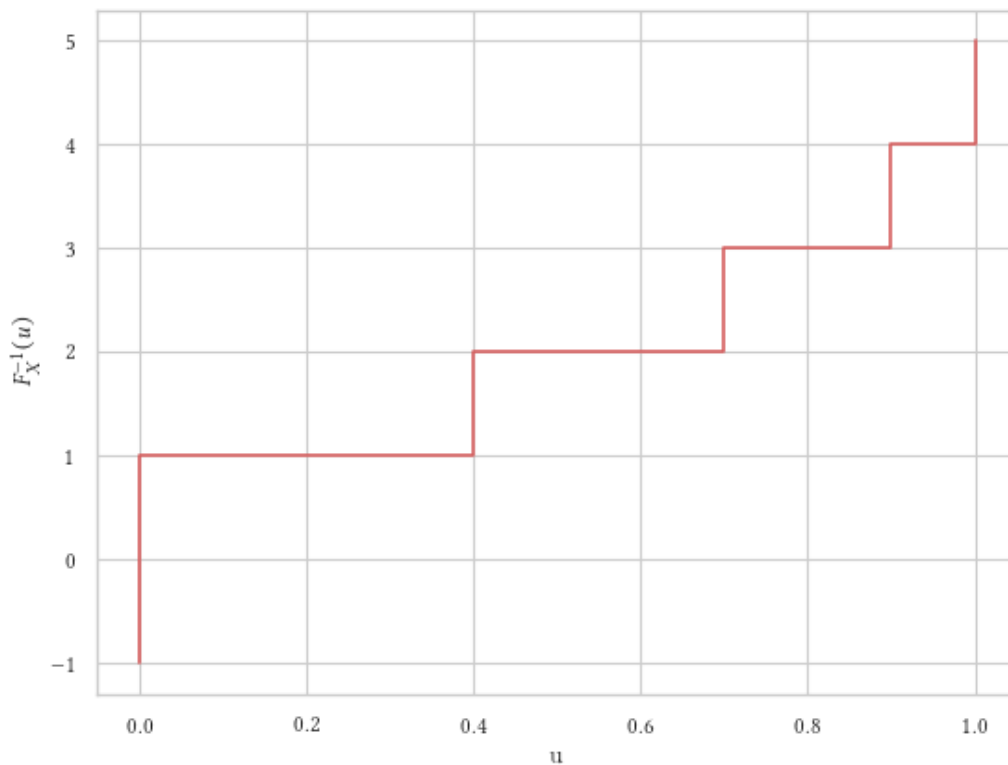
$$\begin{aligned} F_X(x) = u &\iff u \in (0, 0.4] \Rightarrow x = 1, \\ &\iff u \in (0.4, 0.7] \Rightarrow x = 2, \\ &\iff u \in (0.7, 0.9] \Rightarrow x = 3, \\ &\iff u \in (0.9, 1] \Rightarrow x = 4. \end{aligned}$$

Entonces:

$$F_X^{-1}(u) = \begin{cases} 1, & 0 < u \leq 0.4, \\ 2, & 0.4 < u \leq 0.7, \\ 3, & 0.7 < u \leq 0.9, \\ 4, & 0.9 < u \leq 1. \end{cases}$$

```
[5]: u_vals = [0, 0, 0.4, 0.7, 0.9, 1]
     F_inv_vals = [-1, 1, 2, 3, 4, 5]

     plt.step(u_vals, F_inv_vals, where="post")
     plt.xlabel("u")
     plt.ylabel("$F_X^{-1}(u)$")
     plt.show()
```

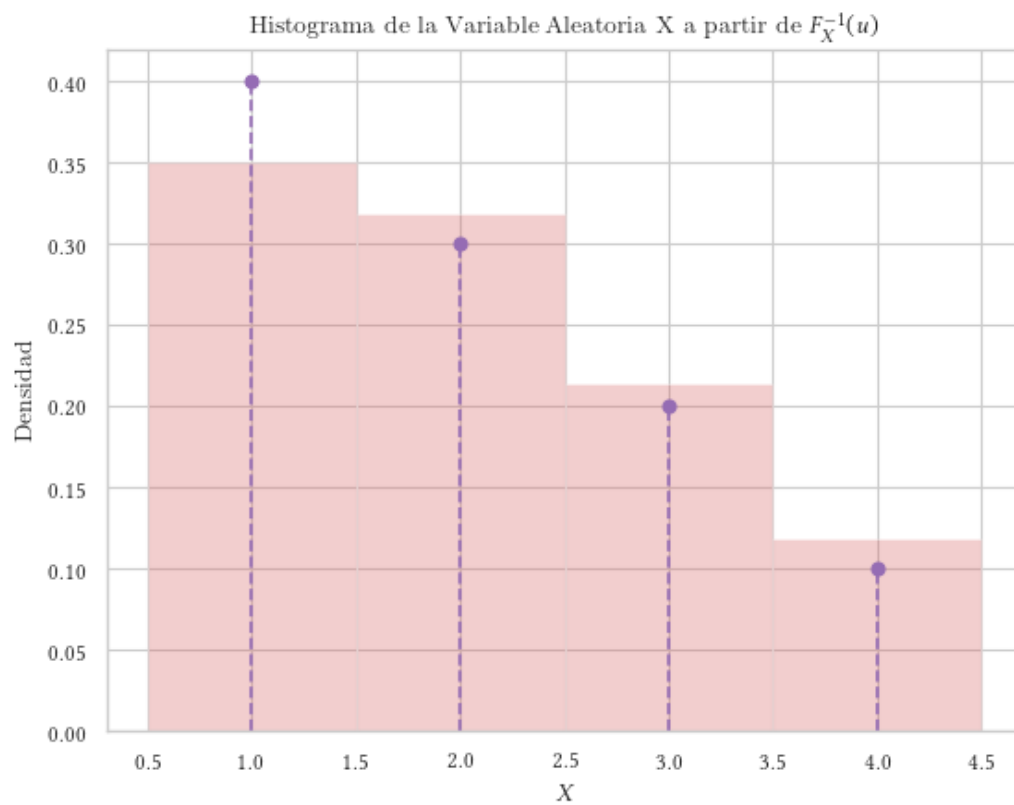


d) Programa

```
[6]: def F_inv(u):  
    if u <= 0.4:  
        return 1  
    elif u <= 0.7:  
        return 2  
    elif u <= 0.9:  
        return 3  
    else:  
        return 4  
  
u_samples = np.random.uniform(0, 1, 500)  
x_samples = [F_inv(u) for u in u_samples]
```

e) Histograma

```
[7]: plt.hist(x_samples, bins=np.arange(0.5, 5.5, 1), density=True, alpha=0.3)
plt.vlines(x_vals, 0, pmf, linestyle='--', color=color[1])
plt.plot(x_vals, pmf, 'o', color=color[1])
plt.xlabel("$X$")
plt.ylabel("Densidad")
plt.title("Histograma de la Variable Aleatoria X a partir de $F_X^{-1}(u)$")
plt.show()
```



3. Exponencial $\text{Exp}(\lambda)$

Sea $U \sim \text{Unif}(0, 1)$. Si $X \sim \text{Exp}(\lambda)$, entonces su función de distribución acumulada es:

$$F_X(x) = (1 - e^{-\lambda x}) \mathbf{1}_{[0, \infty)}(x)$$

Encontrando la inversa:

$$\begin{aligned} F_X(x) = u &\iff 1 - e^{-\lambda x} = u, \\ &\iff e^{-\lambda x} = 1 - u, \\ &\iff -\lambda x = \ln(1 - u), \\ &\iff x = -\frac{1}{\lambda} \ln(1 - u). \end{aligned}$$

Entonces:

$$F_X^{-1}(u) = -\frac{1}{\lambda} \ln(1 - u).$$

Por lo tanto, con $U \sim \text{Unif}(0, 1)$,

$$X = F_X^{-1}(U) = -\frac{1}{\lambda} \ln(1 - U) \sim \text{Exp}(\lambda).$$

4. Weibull (r, λ)

Sea $U \sim \text{Unif}(0, 1)$, con $r > 0$ y $\lambda > 0$. Si $X \sim \text{Weibull}(r, \lambda)$, entonces su función de distribución acumulada es:

$$F_X(x) = (1 - e^{-(\lambda x)^r}) \mathbf{1}_{[0, \infty)}(x)$$

Encontrando la inversa:

$$\begin{aligned} F_X(x) = u &\iff 1 - e^{-(\lambda x)^r} = u, \\ &\iff e^{-(\lambda x)^r} = 1 - u, \\ &\iff -(\lambda x)^r = \ln(1 - u), \\ &\iff (\lambda x)^r = -\ln(1 - u), \\ &\iff x = \frac{1}{\lambda} [-\ln(1 - u)]^{1/r}. \end{aligned}$$

Entonces:

$$F_X^{-1}(u) = \frac{1}{\lambda} [-\ln(1 - u)]^{1/r}.$$

Por lo tanto, con $U \sim \text{Unif}(0, 1)$:

$$X = F_X^{-1}(U) = \frac{1}{\lambda} [-\ln(1 - U)]^{1/r} \sim \text{Weibull}(r, \lambda).$$

5. Cauchy (a, b)

Sea $U \sim \text{Unif}(0, 1)$. Si $X \sim \text{Cauchy}(a, b)$, entonces su funcion de distribucion acumulada es:

$$F_X(x) = \frac{1}{\pi} \arctan\left(\frac{x-a}{b}\right) + \frac{1}{2}, \quad x \in \mathbb{R}, \ b > 0.$$

Encontrando la inversa:

$$\begin{aligned} F_X(x) = u &\iff \frac{1}{\pi} \arctan\left(\frac{x-a}{b}\right) + \frac{1}{2} = u, \\ &\iff \arctan\left(\frac{x-a}{b}\right) = \pi\left(u - \frac{1}{2}\right), \\ &\iff \frac{x-a}{b} = \tan\left(\pi\left(u - \frac{1}{2}\right)\right), \\ &\iff x = a + b \tan\left(\pi\left(u - \frac{1}{2}\right)\right). \end{aligned}$$

Entonces:

$$F_X^{-1}(u) = a + b \tan\left(\pi\left(u - \frac{1}{2}\right)\right).$$

Por lo tanto, $X = F_X^{-1}(U) = a + b \tan\left(\pi\left(U - \frac{1}{2}\right)\right) \sim \text{Cauchy}(a, b)$.

6. Pareto I (a, b)

Sea $U \sim \text{Unif}(0, 1)$, con $a > 0$ y $b > 0$. Si $X \sim \text{Pareto I}(a, b)$, entonces su función de distribución acumulada es:

$$F_X(x) = (1 - (b/x)^a) \mathbf{1}_{[b, \infty)}(x)$$

Encontrando la inversa:

$$\begin{aligned} F_X(x) = u &\iff 1 - \left(\frac{b}{x}\right)^a = u, \\ &\iff \left(\frac{b}{x}\right)^a = 1 - u, \\ &\iff \frac{b}{x} = (1 - u)^{1/a}, \\ &\iff x = b(1 - u)^{-1/a}. \end{aligned}$$

Entonces:

$$F_X^{-1}(u) = b(1 - u)^{-1/a}.$$

Por lo tanto,

$$X = F_X^{-1}(U) = b(1 - U)^{-1/a} \sim \text{Pareto I}(a, b).$$

7. Mínimo $X_{(1)} = \min\{X_1, \dots, X_n\}$

Sea $U \sim \text{Unif}(0, 1)$ y $X_{(1)} := \min\{X_1, \dots, X_n\}$ con X_i i.i.d. de CDF F .

$$F_{X_{(1)}}(x) = \mathbb{P}(X_{(1)} \leq x) = 1 - \mathbb{P}(X_1 > x, \dots, X_n > x) = 1 - (1 - F(x))^n.$$

Encontrando la inversa:

$$\begin{aligned} F_{X_{(1)}}(x) = u &\iff 1 - (1 - F(x))^n = u, \\ &\iff (1 - F(x))^n = 1 - u, \\ &\iff 1 - F(x) = (1 - u)^{1/n}, \\ &\iff F(x) = 1 - (1 - u)^{1/n}, \\ &\iff x = F^{-1}(1 - (1 - u)^{1/n}). \end{aligned}$$

Entonces:

$$F_{X_{(1)}}^{-1}(u) = F^{-1}(1 - (1 - u)^{1/n}), \quad 0 < u < 1.$$

8. Mixta $X = \min\{Y, M\}$ con $Y \sim \text{Exp}(\lambda)$

Sea con $U \sim \text{Unif}(0, 1)$,

$$X = F_X^{-1}(U) = -\frac{1}{\lambda} \ln(1 - U) \sim \text{Exp}(\lambda).$$

```
[8]: def cdf_exp(lam, x):
      return 1.0 - np.exp(-lam * x)

      def inv_exp(lam, u):
          return - np.log(u) / lam

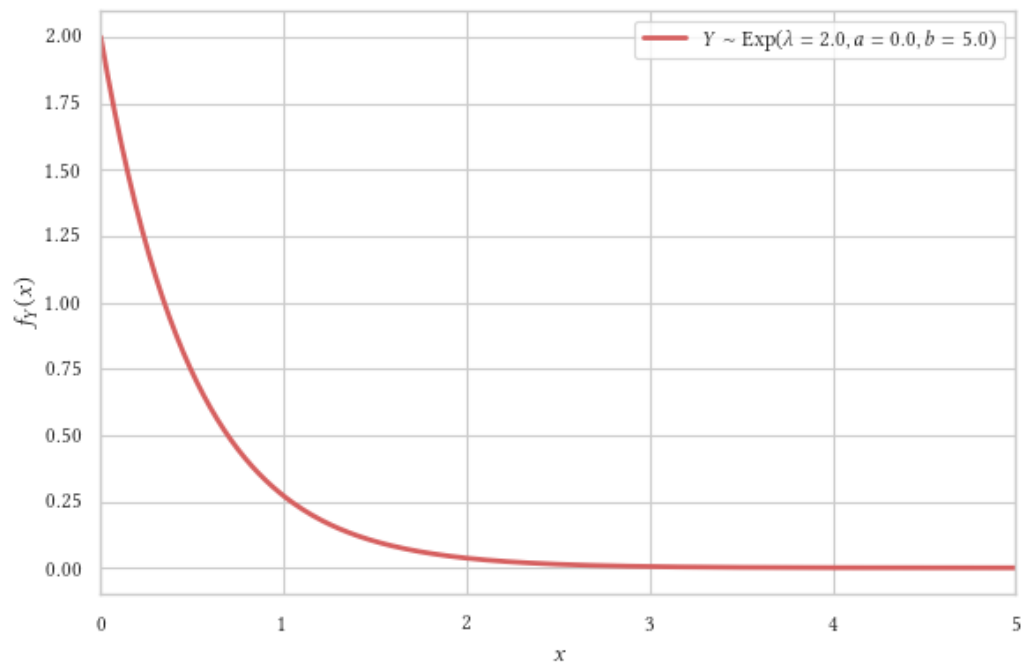
      def exp_ab(lam, a, b, n):
          Fa, Fb = cdf_exp(lam, a), cdf_exp(lam, b)
          U01 = np.random.rand(n)          # ~ Unif(0,1)
          Uab = Fa + U01 * (Fb - Fa)        # ~ Unif(Fa,Fb)
          return inv_exp(lam, Uab)         # Y = F^{-1}(U) ∈ [a,b]

      def pmf_exp(lam, x):
          return lam * np.exp(-lam * x)
```

```
[9]: lam, a, b, n = 2.0, 0.0, 5.0, 50_000
      Y = exp_ab(lam, a, b, n)

      x = np.linspace(0, 5, 400)
      Fa, Fb = cdf_exp(lam, a), cdf_exp(lam, b)
      fY = np.where((x>=a)&(x<=b), pmf_exp(lam, x)/(Fb-Fa), 0.0)

      plt.figure(figsize=(6,4))
      plt.plot(x, fY, linewidth=2, label=f"$Y \sim \mathrm{{Exp}}(\lambda={lam}, a={a}, b={b})$")
      plt.xlim(0,5)
      plt.legend()
      plt.xlabel("$x$")
      plt.ylabel("$f_Y(x)$")
      plt.tight_layout()
      plt.show()
```

a) Grafica de $F_X(x)$

Por definición,

$$F_X(x) = \mathbb{P}(X \leq x) = \begin{cases} 0, & x < 0, \\ \mathbb{P}(\min\{Y, M\} \leq x), & 0 \leq x < M, \\ 1, & x \geq M. \end{cases}$$

Para $0 \leq x < M$: $\min\{Y, M\} \leq x$ equivale a $Y \leq x$. Como $Y \sim \text{Exp}(\lambda)$,

$$\mathbb{P}(Y \leq x) = 1 - e^{-\lambda x}.$$

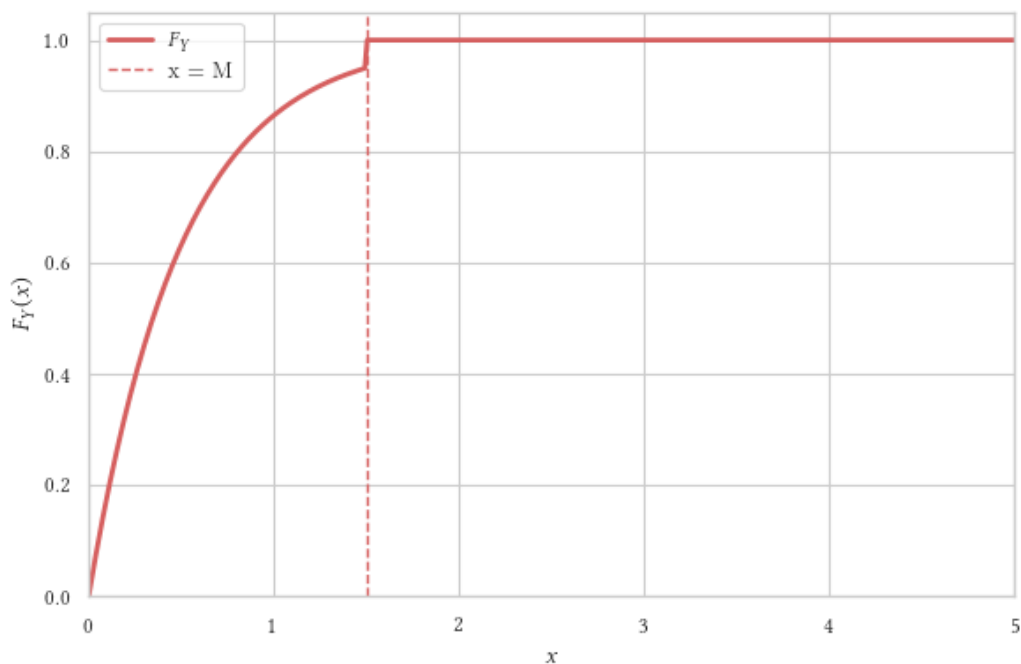
Entonces,

$$F_X(x) = \begin{cases} 0, & x < 0, \\ 1 - e^{-\lambda x}, & 0 \leq x < M, \\ 1, & x \geq M. \end{cases}$$

[10]:

```
M = 1.5
FY = np.where(x<0, 0.0, np.where(x<M, cdf_exp(lam, x), 1.0))

plt.figure(figsize=(6,4))
plt.plot(x, FY, lw=2, label=r"$F_Y$")
plt.axvline(M, ls="--", lw=1, label="x = M")
plt.ylim(0,1.05)
plt.xlim(0,5)
plt.legend()
plt.xlabel("$x$")
plt.ylabel("$F_Y(x)$")
plt.tight_layout()
```



b) Grafica de $F_X^{-1}(u)$

Sea $Y \sim \text{Exp}(\lambda)$, $M > 0$, $X = \min\{Y, M\}$ y $U \sim \text{Unif}(0, 1)$. Usamos la pseudoinversa izquierda $F_X^{-1}(u) = \inf\{x : F_X(x) \geq u\}$.

Encontrando la inversa:

- Para $x < 0, u \in \{0\}$:

$$F_X^{-1}(0) = 0.$$

- Para $0 \leq x < M, u \in (0, 1 - e^{-\lambda M})$:

$$\begin{aligned} F_X(x) = u &\iff 1 - e^{-\lambda x} = u, \\ &\iff e^{-\lambda x} = 1 - u, \\ &\iff x = -\frac{1}{\lambda} \ln(1 - u). \end{aligned}$$

- En el salto $x = M, u \in [1 - e^{-\lambda M}, 1]$:

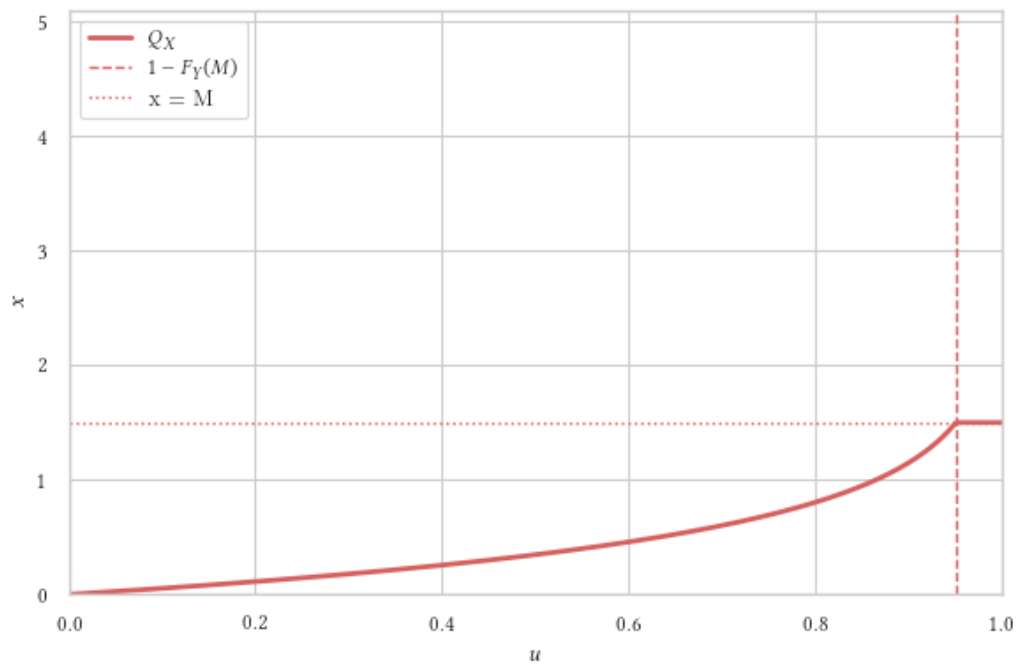
$$F_X^{-1}(u) = M.$$

Entonces:

$$F_X^{-1}(u) = \begin{cases} 0, & u = 0, \\ -\frac{1}{\lambda} \ln(1 - u), & 0 < u < 1 - e^{-\lambda M}, \\ M, & 1 - e^{-\lambda M} \leq u \leq 1. \end{cases}$$

```
[11]: def q_X(u, lam=1.0, M=3):
    u = np.asarray(u)
    t = 1 - np.exp(-lam*M)
    x = np.empty_like(u, dtype=float)
    x[u==0] = 0.0
    mask = (u>0) & (u<t)
    x[mask] = -np.log1p(-u[mask]) / lam
    x[u>=t] = M
    return x

u = np.linspace(0, 1, 1000, endpoint=True)
xq = q_X(u, lam, M)
plt.figure(figsize=(6,4))
plt.plot(u, xq, lw=2, label=r"$Q_X$")
plt.axvline(1-np.exp(-lam*M), ls="--", lw=1, label=r"$1 - F_Y(M)$")
plt.axhline(M, ls=":", lw=1, label="x = M")
plt.xlim(0,1)
plt.ylim(0, max(5,M)+0.1)
plt.legend()
plt.xlabel("$u$")
plt.ylabel("$x$")
plt.tight_layout()
```



c) $F_X(F_X^{-1}(u)) \geq u$, **para** $u \in [0, 1]$.

- Si $0 < u < 1 - e^{-\lambda M}$: Como $x = -(1/\lambda) \ln(1 - u) \in (0, M)$, entonces:

$$F_X(F_X^{-1}(u)) = F_X(x) = 1 - e^{-\lambda x} = u.$$

- Si $1 - e^{-\lambda M} \leq u \leq 1$: $F_X^{-1}(u) = M$ y

$$F_X(F_X^{-1}(u)) = F_X(M) = 1 \geq u.$$

- En $u = 0$: $F_X^{-1}(0) = 0$ y $F_X(0) = 0 \geq 0$.

d) $F_X^{-1}(F_X(x)) \leq x$, para x tal que $F_X(x) \in [0, 1]$.

- Si $x < 0$: $F_X(x) = 0$ y $F_X^{-1}(0) = 0 \leq x$
- Si $x \geq M$: $F_X(x) = 1$ y $F_X^{-1}(1) = M \leq x$.
- Si $0 \leq x < M$:

$$F_X^{-1}(F_X(x)) = F_X^{-1}(1 - e^{-\lambda x}) = -\frac{1}{\lambda} \ln(1 - (1 - e^{-\lambda x})) = -\frac{1}{\lambda} \ln(e^{-\lambda x}) = x.$$

e) Cómo generar valores con el método de la función inversa

1. $U \sim \text{Unif}(0, 1)$.
2. $X = F^{-1}(U)$. Aquí:

$$X = \begin{cases} -\frac{1}{\lambda} \ln(1 - U), & U < 1 - e^{-\lambda M}, \\ M, & U \geq 1 - e^{-\lambda M}. \end{cases}$$

9. Mixta $X = \max\{Y, M\}$ **con**
 $Y \sim \text{Exp}(\lambda)$

a) Grafica de $F_X(x)$

Para $X = \max\{Y, M\}$ con $Y \sim \text{Exp}(\lambda)$ y $M > 0$:

$$F_X(x) = \mathbb{P}(X \leq x) = \begin{cases} 0, & x < M, \\ 1 - e^{-\lambda x}, & x \geq M. \end{cases}$$

Masa en M : $\mathbb{P}(X = M) = 1 - e^{-\lambda M}$. Densidad en (M, ∞) :

$$f_X(x) = \lambda e^{-\lambda x} \mathbf{1}_{\{x > M\}}.$$

```
[12]: lam = 1.2 # > 0
      M = 2.0 # M > 0

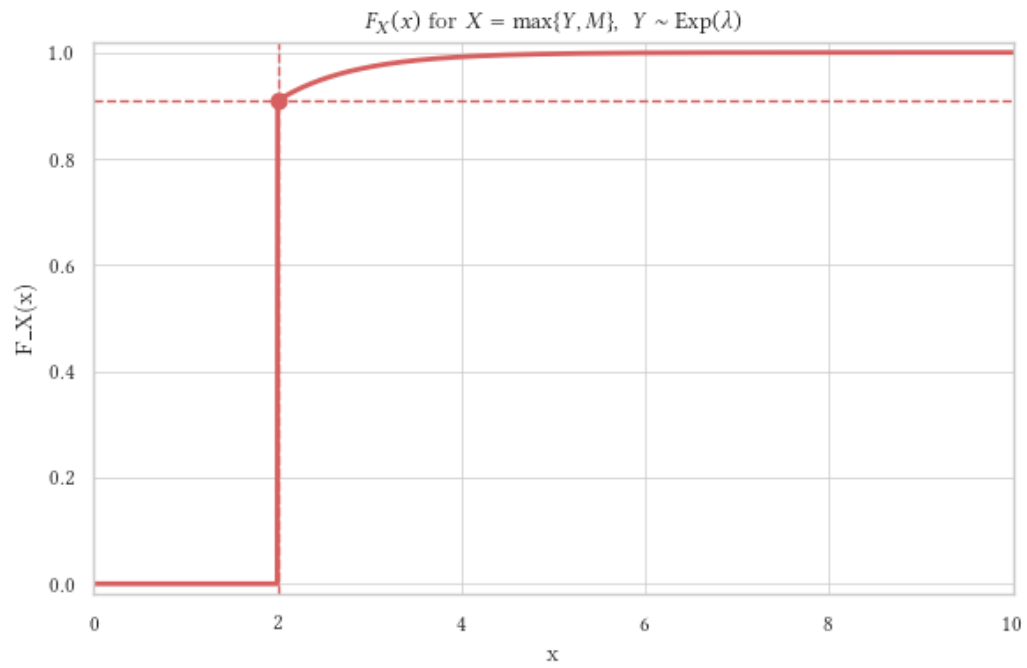
def F_X(x, lam, M):
    x = np.asarray(x, dtype=float)
    out = np.zeros_like(x)
    mask = x >= M
    out[mask] = 1.0 - np.exp(-lam * x[mask])
    return out

def F_inv(u, lam, M):
    u = np.asarray(u, dtype=float)
    p0 = 1.0 - np.exp(-lam * M)
    out = np.empty_like(u)
    left = u <= p0
    out[left] = M
    right = ~left
    out[right] = -np.log(1.0 - u[right]) / lam
    return out

p0 = 1.0 - np.exp(-lam * M)

x_min = max(0.0, M - 3.0)
x_max = M + 8.0
x = np.linspace(x_min, x_max, 2000)
Fx = F_X(x, lam, M)
```

```
[13]: plt.figure(figsize=(6, 4))
      plt.plot(x, Fx, linewidth=2)
      plt.axvline(M, linestyle="--", linewidth=1)
      plt.axhline(p0, linestyle="--", linewidth=1)
      plt.scatter([M], [p0], s=40) # value at x=M
      plt.scatter([M], [0.0], s=40, facecolors='none') # left limit at M-
      plt.title(r"$F_X(x)$ for $X=\max\{Y,M\}$, $Y\sim \mathrm{Exp}(\lambda)$")
      plt.xlabel("x")
      plt.ylabel("$F_X(x)$")
      plt.ylim(-0.02, 1.02)
      plt.xlim(x_min, x_max)
      plt.grid(True, linewidth=0.5)
      plt.tight_layout()
      plt.show()
```



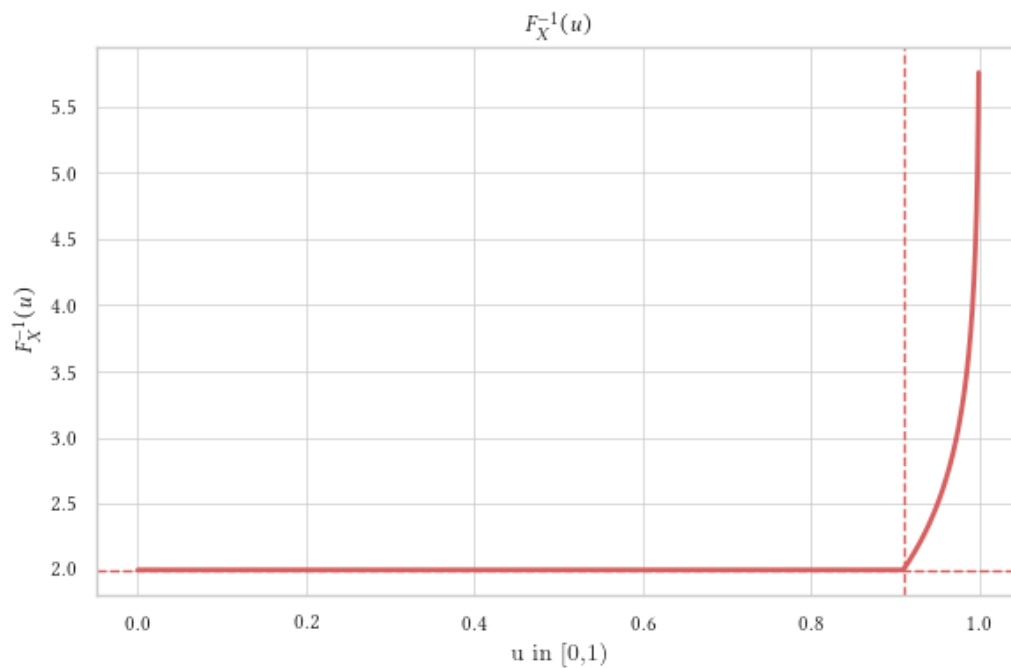
b) Grafica de $F_X^{-1}(u)$

Sea $p_0 := 1 - e^{-\lambda M}$.

$$F_X^{-1}(u) = \begin{cases} M, & 0 \leq u \leq p_0, \\ -\frac{1}{\lambda} \ln(1 - u), & p_0 < u < 1. \end{cases}$$

```
[14]: u = np.linspace(0.0, 0.999, 2000)
      Finv = F_inv(u, lam, M)

      plt.figure(figsize=(6, 4))
      plt.plot(u, Finv, linewidth=2)
      plt.axvline(p0, linestyle="--", linewidth=1)
      plt.axhline(M, linestyle="--", linewidth=1)
      plt.title(r"$F_X^{-1}(u)$")
      plt.xlabel("u in [0,1)")
      plt.ylabel(r"$F_X^{-1}(u)$")
      plt.grid(True, linewidth=0.5)
      plt.tight_layout()
      plt.show()
```



$$\mathbf{c)} \ F_X(F_X^{-1}(u)) \geq u, \text{ para } u \in [0, 1].$$

Si $u \leq p_0$, $F_X^{-1}(u) = M$ y $F_X(M) = p_0 \geq u$.

Si $u > p_0$, $F_X^{-1}(u) = -(1/\lambda) \ln(1 - u)$ y $F_X(F_X^{-1}(u)) = u$.

d) $F_X^{-1}(F_X(x)) \geq x$, para x tal que $F_X(x) \in [0, 1]$.

Si $x < M$, $F_X(x) = 0$ y $F_X^{-1}(0) = M \geq x$.

Si $x \geq M$, $F_X(x) = 1 - e^{-\lambda x} \geq p_0$ y $F_X^{-1}(F_X(x)) = x$.

e) Cómo generar valores con el método de la función inversa

Muestrea $U \sim \text{Unif}(0, 1)$ y define

$$X = \begin{cases} M, & U \leq 1 - e^{-\lambda M}, \\ -\frac{1}{\lambda} \ln(1 - U), & U > 1 - e^{-\lambda M}. \end{cases}$$

Cambios mínimos respecto al caso mín: sustituye los tramos $x < M$ por cero, coloca la masa en M con tamaño $1 - e^{-\lambda M}$, y mantiene la cola exponencial intacta para $x > M$.

10. Variable con CDF por tramos

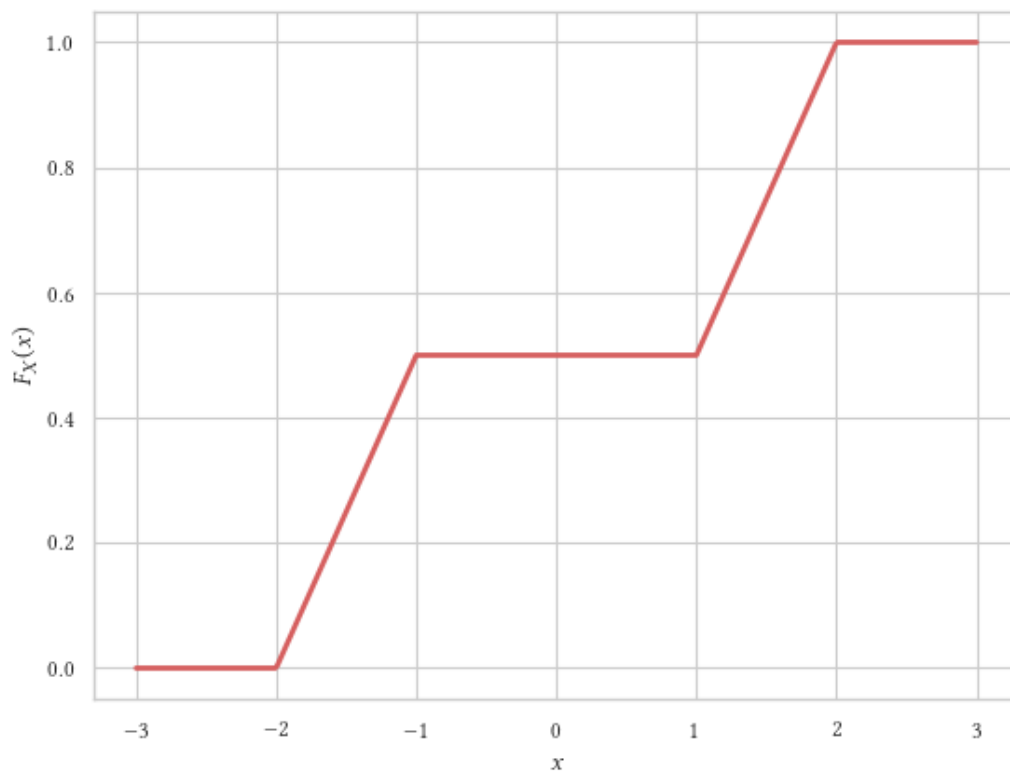
a) Grafica de $F_X(x)$

$$F_X(x) = \begin{cases} 0, & x \leq -2, \\ \frac{x+2}{2}, & -2 < x < -1, \\ \frac{1}{2}, & -1 \leq x < 1, \\ \frac{x}{2}, & 1 \leq x < 2, \\ 1, & x \geq 2. \end{cases}$$

```
[15]: def F(x):
    if x <= -2:
        return 0
    elif -2 < x < -1:
        return (x + 2) / 2
    elif -1 <= x < 1:
        return 0.5
    elif 1 <= x < 2:
        return x / 2
    elif x >= 2:
        return 1

x = np.linspace(-3, 3, 400)
Fx = [F(xi) for xi in x]

plt.plot(x, Fx, lw=2, color='red')
plt.xlabel('$x$')
plt.ylabel('$F_X(x)$')
plt.show()
```



b) Grafica de $F_X^{-1}(u)$

Sea $U \sim \text{Unif}(0, 1)$.

Encontrando la inversa:

- Para $x \leq -2, u \in \{0\}$

$$F_X^{-1}(0) = -2$$

.

- Para $-2 < x < -1, u \in (0, \frac{1}{2})$:

$$\begin{aligned} F_X(x) = u &\iff \frac{x+2}{2} = u, \\ &\iff x = 2u - 2. \end{aligned}$$

- Para $-1 \leq x < 1, u \in \{\frac{1}{2}\}$:

$$F_X^{-1}(\frac{1}{2}) = -1$$

.

- Para $1 \leq x < 2, u \in (\frac{1}{2}, 1)$:

$$\begin{aligned} F_X(x) = u &\iff \frac{x}{2} = u, \\ &\iff x = 2u. \end{aligned}$$

- Para $x \geq 2, u \in \{1\}$:

$$F_X^{-1}(1) = 2$$

.

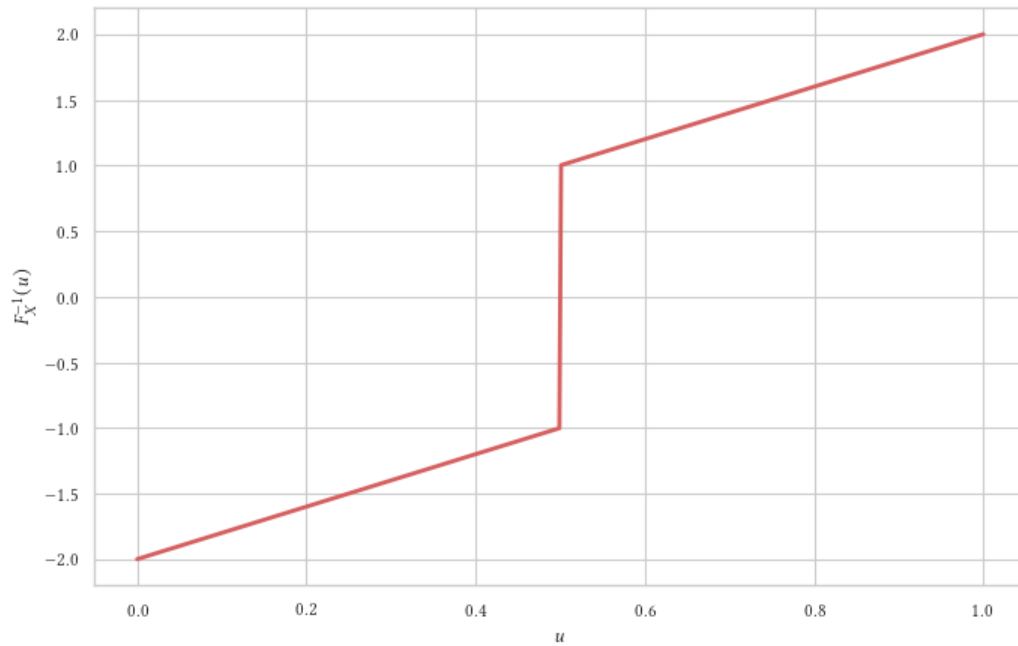
Entonces:

$$F_X^{-1}(u) = \begin{cases} -2, & u = 0, \\ 2u - 2, & 0 < u < \frac{1}{2}, \\ -1, & u = \frac{1}{2}, \\ 2u, & \frac{1}{2} < u < 1, \\ 2, & u = 1. \end{cases}$$

```
[16]: def F_inv_func(u):
    if u == 0:
        return -2
    elif 0 < u < 0.5:
        return 2 * u - 2
    elif u == 0.5:
        return -1
    elif 0.5 < u < 1:
        return 2 * u
    elif u == 1:
        return 2
```

```
u = np.linspace(0, 1, 500)
F_inv = [F_inv_func(val) for val in u]

plt.figure(figsize=(8, 5))
plt.plot(u, F_inv, color=color[0], lw=2)
plt.xlabel('$u$')
plt.ylabel('$F_X^{-1}(u)$')
plt.show()
```



c) $F_X(F_X^{-1}(u)) \geq u$, **para** $u \in [0, 1]$

Sea $x(u) = F_X^{-1}(u)$.

▪ $u = 0$:

$$F_X^{-1}(0) = -2.$$

$$F_X(-2) = 0 = u.$$

▪ $0 < u < \frac{1}{2}$:

$$F_X^{-1}(u) = 2u - 2 \in (-2, -1).$$

$$F_X(2u - 2) = \frac{(2u - 2) + 2}{2} = u.$$

▪ $u = \frac{1}{2}$:

$$F_X^{-1}\left(\frac{1}{2}\right) = -1.$$

$$F_X(-1) = \frac{1}{2} = u.$$

▪ $\frac{1}{2} < u < 1$:

$$F_X^{-1}(u) = 2u \in (1, 2).$$

$$F_X(2u) = \frac{2u}{2} = u.$$

▪ $u = 1$:

$$F_X^{-1}(1) = 2.$$

$$F_X(2) = 1 = u.$$

d) $F_X^{-1}(F_X(x)) \leq x$, para x tal que $F_X(x) \in (0, 1)$.

- $x < -2$:

$$F_X(x) = 0.$$

$$F_X^{-1}(F_X(x)) = F_X^{-1}(0) = -2 \leq x.$$

- $x \in (-2, -1)$:

$$F_X(x) = \frac{x+2}{2} \in (0, \frac{1}{2}).$$

$$F_X^{-1}(F_X(x)) = 2 \cdot \frac{x+2}{2} - 2 = x. \quad \text{Igualdad.}$$

- $x \in [-1, 1)$:

$$F_X(x) = \frac{1}{2}.$$

$$F_X^{-1}(F_X(x)) = F_X^{-1}(\frac{1}{2}) = -1 \leq x, \quad \text{igualdad sólo en } x = -1.$$

- $x \in [1, 2)$:

$$F_X(x) = \frac{x}{2} \in [\frac{1}{2}, 1).$$

$$F_X^{-1}(F_X(x)) = 2 \cdot \frac{x}{2} = x. \quad \text{Igualdad.}$$

- $x \geq 2$:

$$F_X(x) = 1.$$

$$F_X^{-1}(F_X(x)) = F_X^{-1}(1) = 2 \leq x, \quad \text{igualdad sólo en } x = 2.$$

e) Cómo generar valores con el método de la función inversa

1. Generar $U \sim \text{Unif}(0, 1)$.
2. Definir la inversa

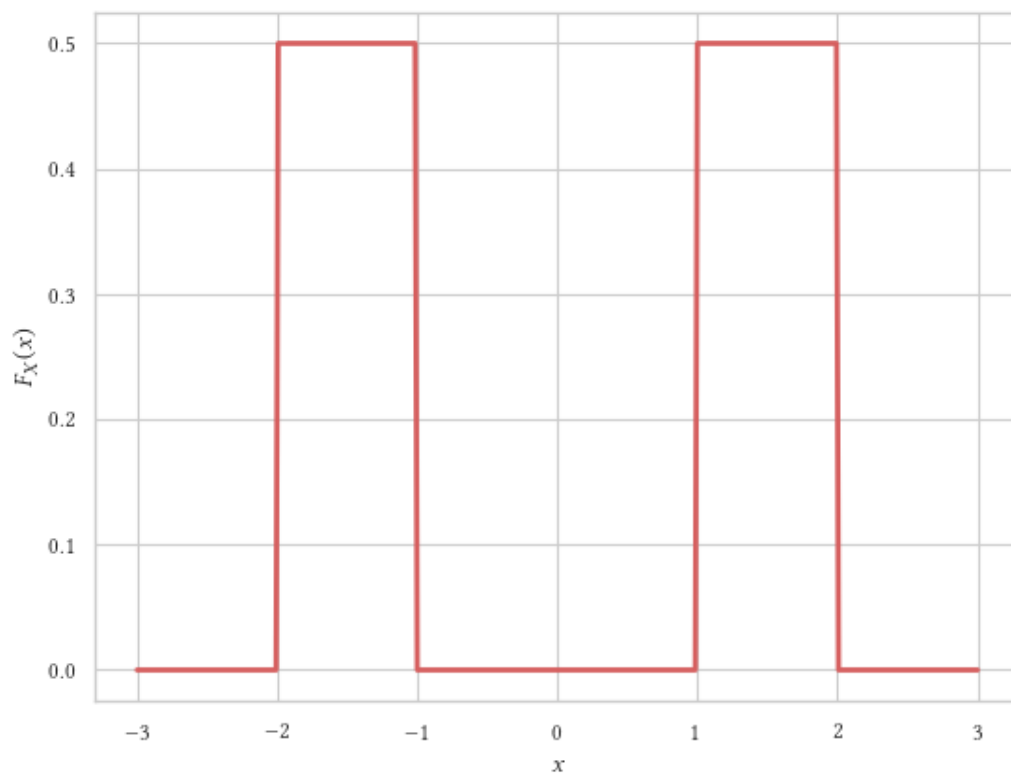
$$F^{-1}(u) = \inf\{x : F(x) \geq u\}.$$

3. Asignar $X = F^{-1}(U)$.
4. Entonces:

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

f) $f_X(x)$.

```
[17]: def f(x):  
    if x < -2:  
        return 0  
    elif -2 < x < -1:  
        return 1/2  
    elif -1 <= x < 1:  
        return 0  
    elif 1 <= x < 2:  
        return 1/2  
    else:  
        return 0  
  
x = np.linspace(-3, 3, 400)  
fx = [f(xi) for xi in x]  
  
plt.plot(x, fx, lw=2, color='red')  
plt.xlabel('$x$')  
plt.ylabel('$f_X(x)$')  
plt.show()
```



g) Programa

Está en el inciso anterior.

h) $E[X] = 0$.

La densidad es

$$f(x) = \frac{1}{2} \mathbf{1}_{(-2,-1)}(x) + \frac{1}{2} \mathbf{1}_{(1,2)}(x),$$

Simetría.

Es simétrica, $f(x) = f(-x)$. Entonces

$$E[X] = \int_{\mathbb{R}} x f(x) dx = \int_{-\infty}^{\infty} x f(x) dx = - \int_{-\infty}^{\infty} x f(x) dx$$

por el cambio $x \mapsto -x$ y la simetría, así $E[X] = 0$.

Cálculo.

$$E[X] = \frac{1}{2} \int_{-2}^{-1} x dx + \frac{1}{2} \int_1^2 x dx = \frac{1}{2} \left[\frac{x^2}{2} \right]_{-2}^{-1} + \frac{1}{2} \left[\frac{x^2}{2} \right]_1^2 = \frac{1}{2} \left(\frac{1}{2} - 2 \right) + \frac{1}{2} \left(2 - \frac{1}{2} \right) = 0.$$

Mezcla.

$$X \sim \frac{1}{2} \text{Unif}(-2, -1) + \frac{1}{2} \text{Unif}(1, 2).$$

$$E[\text{Unif}(-2, -1)] = -1.5, E[\text{Unif}(1, 2)] = 1.5.$$

$$E[X] = \frac{1}{2}(-1.5) + \frac{1}{2}(1.5) = 0.$$

i) $\bar{x} \approx 0$.

```
[18]: x = np.random.uniform(-10000, 10000, 1000)
      fx = [f(xi) for xi in x]
      np.mean(fx)
```

```
[18]: np.float64(0.0005)
```

11. Bernoulli (p) desde $U(0, 1)$

Sea $U \sim \text{Unif}(0, 1)$ y $0 < p < 1$. Defina

$$X = \mathbf{1}_{(0,p]}(U) = \begin{cases} 1, & U \leq p, \\ 0, & U > p. \end{cases}$$

$$\mathbb{P}(X = 1) = \mathbb{P}(U \leq p) = p, \quad \mathbb{P}(X = 0) = \mathbb{P}(U > p) = 1 - p,$$

usando que $\mathbb{P}(U = p) = 0$. Por tanto $X \sim \text{Bernoulli}(p)$.

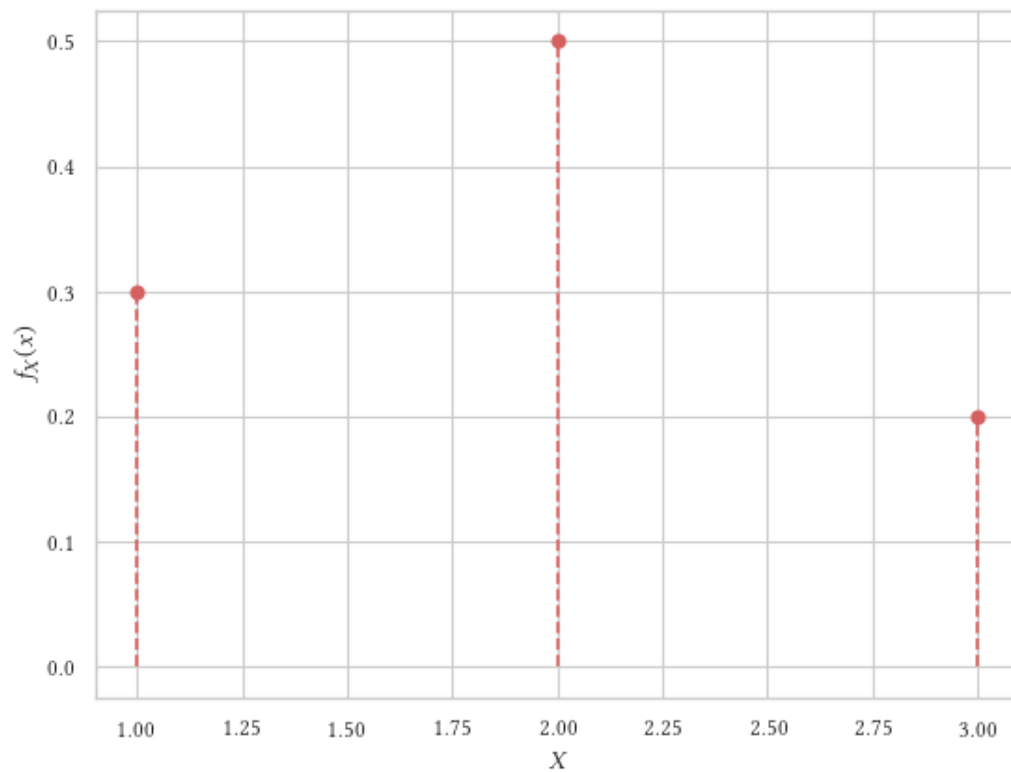
12. Variable aleatoria discreta

a) Grafica de $f_X(x)$

$$f_X(x) = \begin{cases} 0.3 & x = 1, \\ 0.5, & x = 2, \\ 0.2, & x = 3, \\ 0, & \text{en otro caso.} \end{cases}$$

```
[19]: x_vals = [1, 2, 3]
      pmf = [0.3, 0.5, 0.2]

      plt.vlines(x_vals, 0, pmf, linestyle='--')
      plt.plot(x_vals, pmf, 'o')
      plt.xlabel('$X$')
      plt.ylabel('$f_X(x)$')
      plt.show()
```

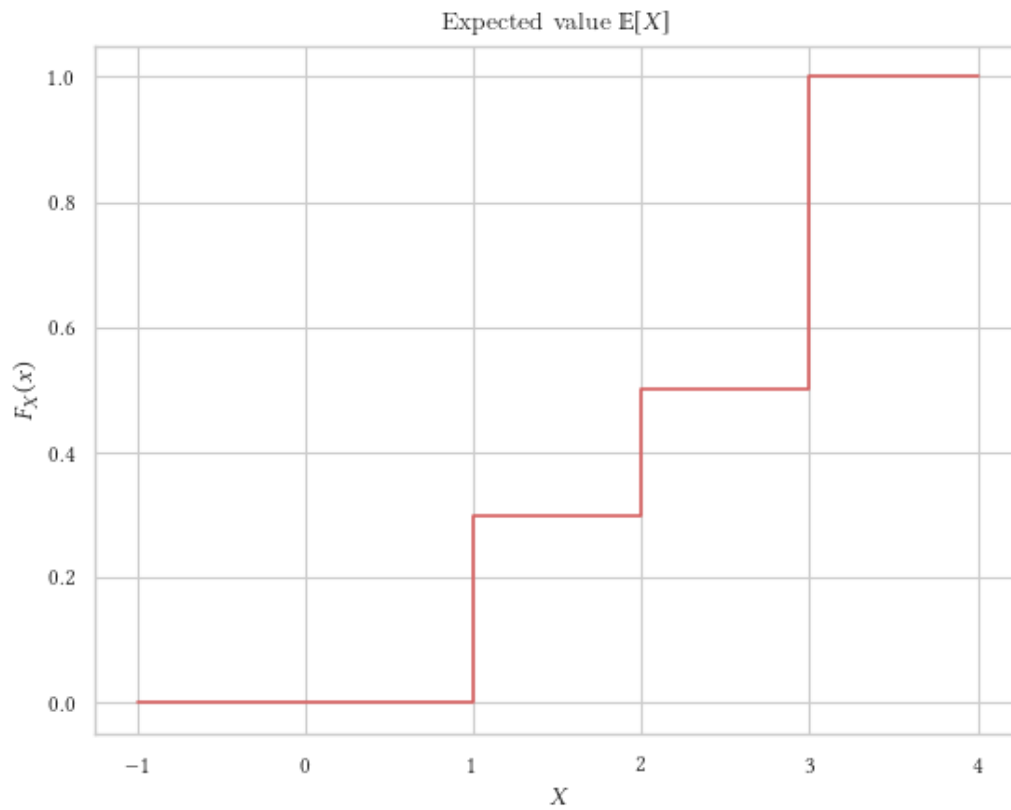


b) Grafica de $F_X(x)$

$$F_X(x) = \mathbb{P}(X \leq x) = \begin{cases} 0, & x < 1, \\ 0.3, & 1 \leq x < 2, \\ 0.8, & 2 \leq x < 3, \\ 1, & x \geq 3. \end{cases}$$

```
[20]: x_cdf = [-1, 1, 2, 3, 4]
      F_cdf = [0, 0.3, 0.5, 1, 1]

      plt.step(x_cdf, F_cdf, where='post', linestyle='-')
      plt.xlabel('$X$')
      plt.ylabel('$F_X(x)$')
      plt.title('Expected value  $\mathbb{E}[X]$ ')
      plt.show()
```



c) Gráfica de $F_X^{-1}(u)$

Sea $U \sim \text{Unif}(0, 1)$.

$$\begin{aligned} F_X(x) = u &\iff u \in (0, 0.3] \Rightarrow x = 1, \\ &\iff u \in (0.3, 0.8] \Rightarrow x = 2, \\ &\iff u \in (0.8, 1] \Rightarrow x = 3. \end{aligned}$$

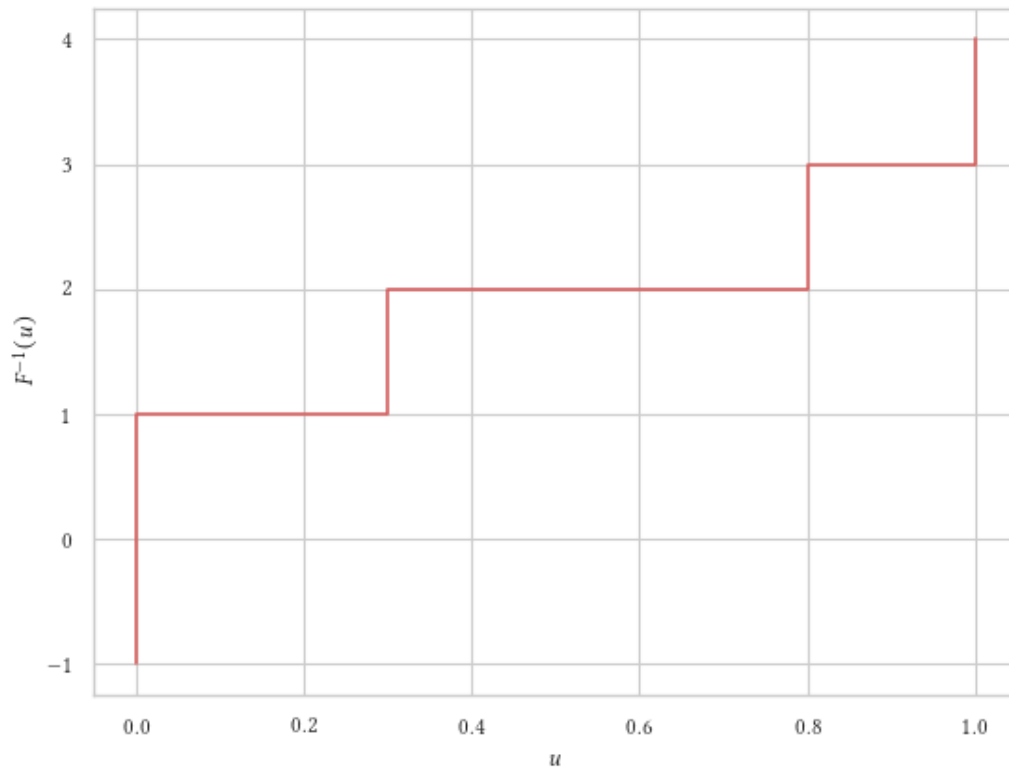
Entonces:

$$F_X^{-1}(u) = \begin{cases} 1, & 0 < u \leq 0.3, \\ 2, & 0.3 < u \leq 0.8, \\ 3, & 0.8 < u \leq 1. \end{cases}$$

[21]:

```
u_vals = [0, 0, 0.3, 0.8, 1.0]
F_inv_values = [-1, 1, 2, 3, 4]

plt.step(u_vals, F_inv_values, where='post')
plt.xlabel('$u$')
plt.ylabel('$F^{-1}(u)$')
plt.show()
```



d) Valor esperado $\mathbb{E}[X]$

$$\mathbb{E}[X] = \sum_{x=1}^3 x \cdot f_X(x) = 1 \cdot 0.3 + 2 \cdot 0.5 + 3 \cdot 0.2 = 1.9.$$

e) Varianza $V[X]$

$$V[X] = \sum_{x=1}^3 (x - E[X])^2 \cdot f_X(x) = (1 - 1.9)^2 \cdot 0.3 + (2 - 1.9)^2 \cdot 0.5 + (3 - 1.9)^2 \cdot 0.2 = 0.49.$$

$$V[X] = E[X^2] - (E[X])^2 = 0.3 \cdot 1^2 + 0.5 \cdot 2^2 + 0.2 \cdot 3^2 - (1.9)^2 = 2.79 - (1.9)^2 = 0.49.$$

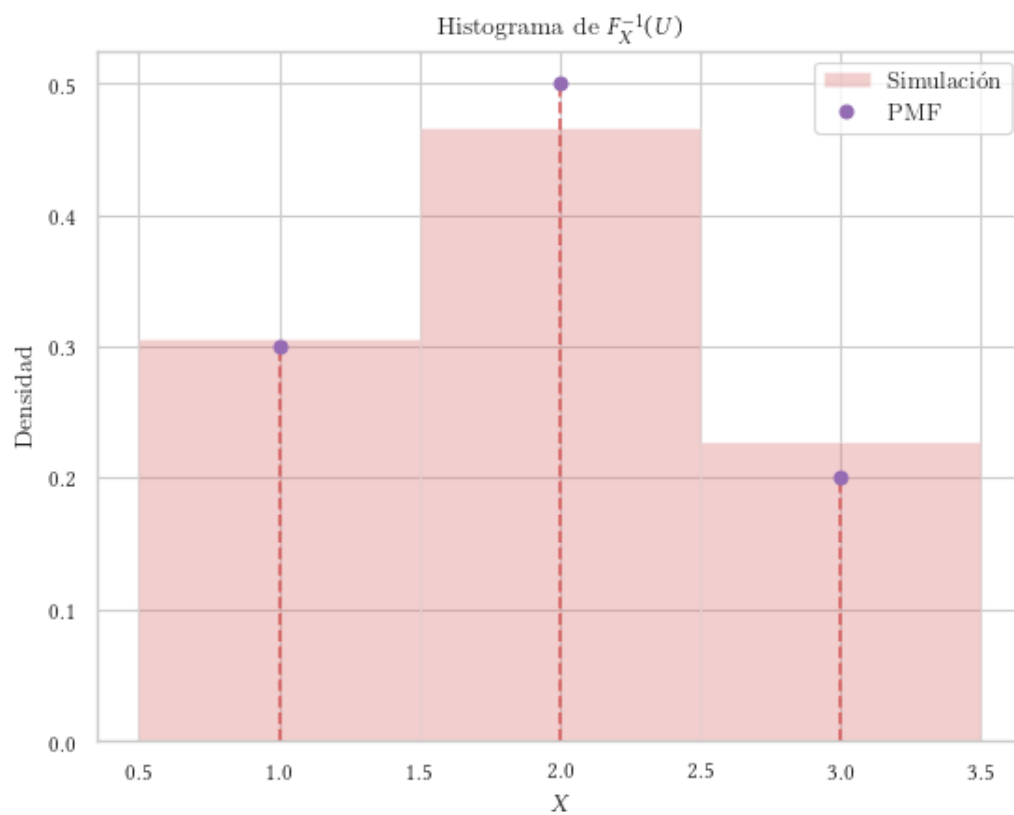
f) Programa

```
[22]: def F_inv(u):  
      if 0.3 < u <= 0.8:  
          return 2  
      elif u <= 0.3:  
          return 1  
      else:  
          return 3
```

g) Histograma

```
[23]: u_samples = np.random.uniform(0, 1, 500)
      x_samples = [F_inv(u) for u in u_samples]
```

```
[24]: plt.hist(x_samples, bins=np.arange(0.5, 4.5, 1), density=True, alpha=0.3, label='Simulación')
      plt.vlines(x_vals, 0, pmf, linestyle='--')
      plt.plot(x_vals, pmf, 'o', label='PMF')
      plt.xlabel("$X$")
      plt.ylabel("Densidad")
      plt.title("Histograma de $F_X^{-1}(U)$")
      plt.legend()
      plt.show()
```



$$\mathbf{h)} E[X] \approx \bar{x}$$

```
[25]: x_mean = np.mean(x_samples)
      print(f"Media muestral de X: {x_mean}")
      print(f"Media teórica de X: 1.9")
      print(f"Diferencia: {x_mean - 1.9:.4f}")
```

```
Media muestral de X: 1.922
Media teórica de X: 1.9
Diferencia: 0.0220
```

$$\mathbf{i) } V[X] \approx s^2$$

```
[26]: x_var = np.var(x_samples)
      print(f"Varianza muestral de X: {x_var}")
      print(f"Varianza teórica de X: 0.49")
      print(f"Diferencia: {x_var - 0.49:.4f}")
```

```
Varianza muestral de X: 0.5279159999999999
Varianza teórica de X: 0.49
Diferencia: 0.0379
```

13. Binomial, Geométrica y Poisson

Implementa generadores por inversión para: - $X \sim \text{Bin}(m, p)$ con $m = 10$ y $p = 1/3$ - $X \sim \text{Geo}(p)$ con $\mathbb{P}(X = k) = p(1 - p)^{k-1}, k \geq 1$ con $p = 3/4$ - $X \sim \text{Poisson}(\lambda)$ con $\lambda = 2$

a) Binomial

```
[27]: def binomial(n: int, p: float, size: int):  
  
    c = p/(1.0 - p)          # paso 2: c  
    out = []  
    for _ in range(size):  
        U = random.random()  # paso 1  
        i = 0                 # paso 2  
        pr = (1.0 - p)**n     # pr = P(X=0)  
        F = pr  
        if U < F:             # paso 3-5  
            out.append(i); continue  
        while True:          # paso 6  
            pr *= c * (n - i) / (i + 1) # paso 7  
            F += pr  
            i += 1  
            if U < F or i == n:    # paso 8-11 (con tope)  
                out.append(i)  
                break  
    return out  
  
m = binomial(n=10, p=1/3, size=100)  
print(m[:20])
```

```
[2, 5, 3, 2, 5, 3, 4, 4, 5, 4, 3, 6, 3, 5, 4, 2, 3, 3, 3, 5]
```


b) Geometrica

```
[28]: def geometrica(p: float, n: int):  
  
    out = []  
    for _ in range(n):  
        u = random.random()  
        x = math.ceil(math.log(u) / math.log(1.0 - p))  
        out.append(x)  
    return out
```

```
muestras = geometrica(p=3/4, n=100)  
print(muestras[:20])
```

```
[1, 1, 1, 1, 2, 2, 1, 2, 1, 1, 1, 2, 3, 1, 2, 2, 1, 1, 1]
```

c) Poisson

```
[29]: def poisson(lam: float, n: int):  
  
    out = []  
    for _ in range(n):  
        # 1)  $U \sim U(0,1)$   
        U = random.random()  
        # 2)  $i=0, p=e^{-\lambda}, F=p$   
        i = 0  
        p = math.exp(-lam)  
        F = p  
        # 3) if  $U < F: X=i$   
        if U < F:  
            out.append(i)  
            continue  
        # 6-11) loop:  $p = (\lambda/(i+1))*p; F = F + p; i = i + 1; \text{if } U < F: X=i$   
        while True:  
            i += 1  
            p = p * lam / i  
            F = F + p  
            if U < F:  
                out.append(i)  
                break  
    return out  
  
muestras = poisson(lam=2.0, n=100)  
print(muestras[:20])
```

```
[1, 2, 0, 3, 0, 4, 1, 0, 3, 0, 0, 1, 3, 1, 2, 3, 2, 2, 3, 1]
```