

# Corto 1 - ModSim

- Diego Cordova 20212
- Gabriel Vicente 20498

```
In [ ]: import math
import random
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import poisson
```

## Task 1

Suponga que usted está trabajando en la industria relacionada con meteorología, por lo cual le interesa saber la probabilidad de que haya  $N$  huracanes este año. Se sabe que la frecuencia histórica de huracanes es 7 por año, en otras palabras, el número promedio de huracanes por año es de 7.

1. ¿Es este un escenario que se pueda modelar como una variable aleatoria de Poisson?

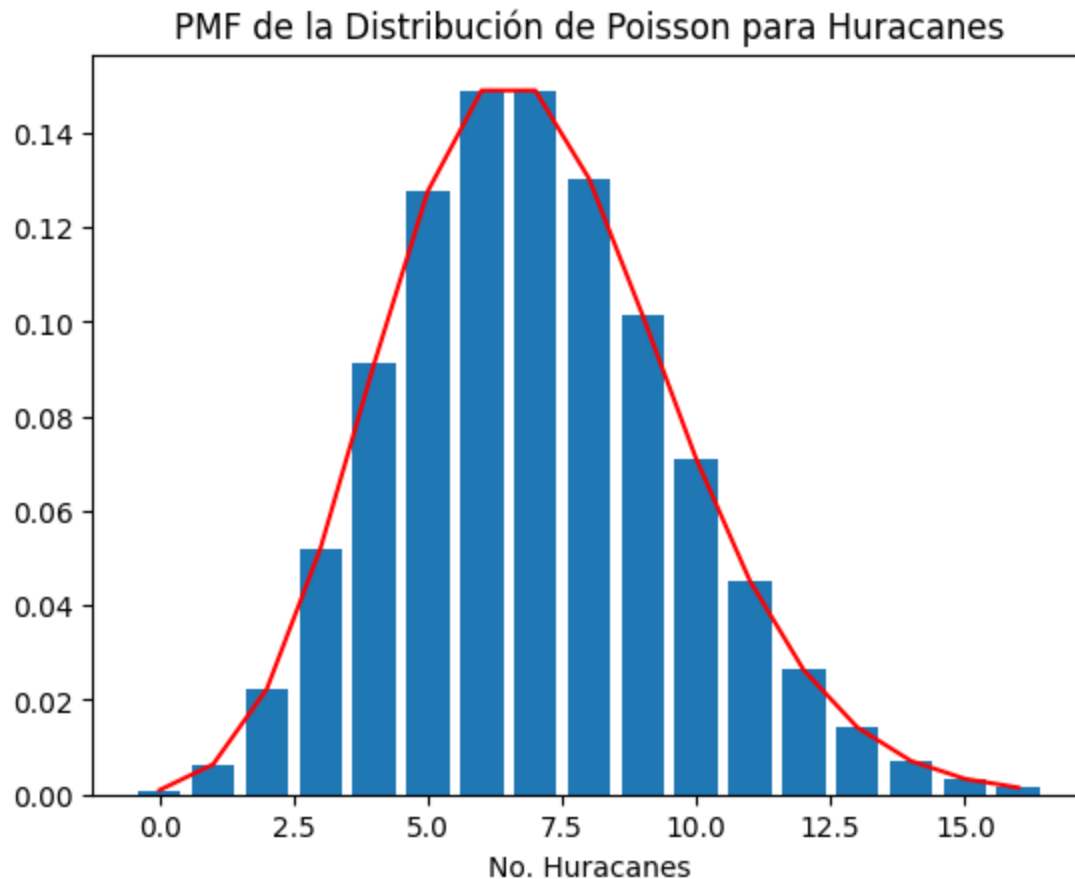
¿Por qué?

R// Si se puede modelar como una variable aleatoria de poisson en dónde la probabilidad del que sucede un evento determinado, en este caso el huracán, está dada por el intervalo de tiempo que se analiza. En este caso la variable  $\lambda$  es 7.

2. Considere que usted analizará hasta un máximo de 16 huracanes este año. Grafique PMF (probability mass function) de estos eventos

```
In [ ]: num_huracanes = range(0, 17)
pmf = poisson.pmf(num_huracanes, mu=7)

plt.bar(num_huracanes, pmf)
plt.plot(pmf, color='red')
plt.xlabel('No. Huracanes')
plt.title('PMF de la Distribución de Poisson para Huracanes')
plt.show()
```



3. Considere que usted analizará hasta un máximo de 16 huracanes este año. Grafique CDF (cumulative distribution function) de estos eventos

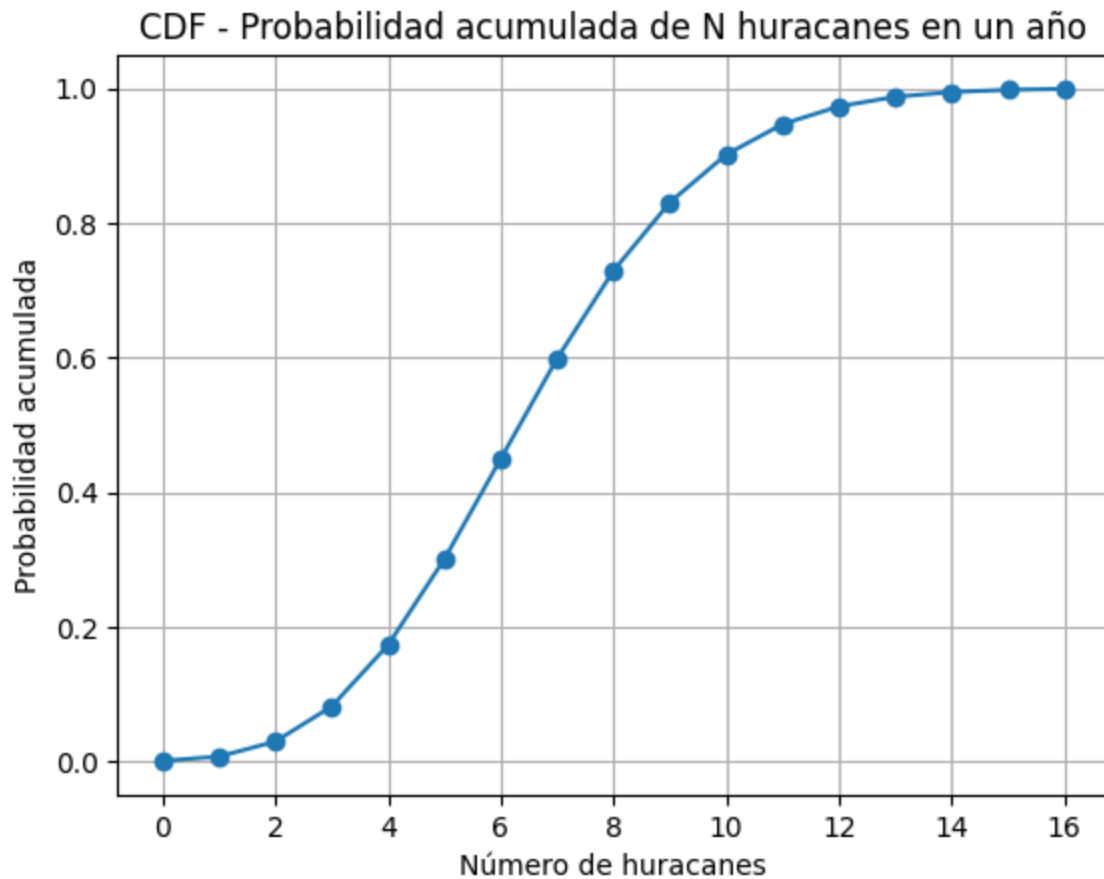
```
In [ ]: def poisson_prob(N, lam):
    return (math.exp(-lam) * lam**N) / math.factorial(N)

def poisson_cdf(N, lam):
    cdf_vals = [poisson_prob(n, lam) for n in range(N + 1)]
    cdf = [sum(cdf_vals[:i+1]) for i in range(N + 1)]
    return cdf

lambda_huracanes = 7
max_huracanes = 16

N_values = list(range(max_huracanes + 1))
cdf = poisson_cdf(max_huracanes, lambda_huracanes)

plt.plot(N_values, cdf, marker='o')
plt.xlabel('Número de huracanes')
plt.ylabel('Probabilidad acumulada')
plt.title('CDF - Probabilidad acumulada de N huracanes en un año')
plt.grid()
plt.show()
```



4. ¿Qué conclusiones puede sacar al observar las gráficas de los ejercicios anteriores?

R// De la gráfica PMF se puede observar que lo más probable es que se produzcan entre 6 a 7.5 huracanes por año, esto dado a su función de masa de probabilidad, lo cual tiene sentido dado las declaraciones iniciales. Respecto a CDF podemos observar que la probabilidad de que ocurra un huracán más a comparación del año anterior aumenta logarítmicamente, dando como resultado que pasen más de 16 huracanes en ese periodo de tiempo.

## Task 2

Usted es un analista de simulación encargado de modelar la llegada de clientes a una tienda minorista. Desea simular la cantidad de clientes que llegan por hora utilizando dos métodos diferentes: el método de transformación inversa y el método de rechazo.

### Task 2.1

Defina la distribución de probabilidad objetivo para las llegadas de clientes en función de los datos históricos. Supongamos que ha recopilado datos y descubrió que la cantidad de clientes que llegan por hora sigue una distribución de Poisson con un promedio de 10

clientes por hora ( $\lambda = 10$ ).

1. Implemente el método de transformación inversa para generar muestras aleatorias a partir de la distribución de Poisson.
2. Genere una muestra aleatoria de tamaño 1000 que represente el número de clientes que llegan en una hora.
3. Trace un histograma de la muestra generada y compárelo con el PMF teórico de la distribución de Poisson.
4. Calcule la media y la varianza de la muestra generada y compárelas con los valores teóricos.

```
In [ ]: from math import e
horas = range(0, 24)

# Datos de muestreo
def muestreo_inversa(lamb, size):
    muestra = []

    for i in range(size):
        U = np.random.uniform(0, 1)
        i = 0
        p = e**(-lamb)
        F = p

        while U >= F:
            p = (lamb * p)/(i + 1)
            F = F + p
            i = i + 1

        muestra.append(i)

    return muestra

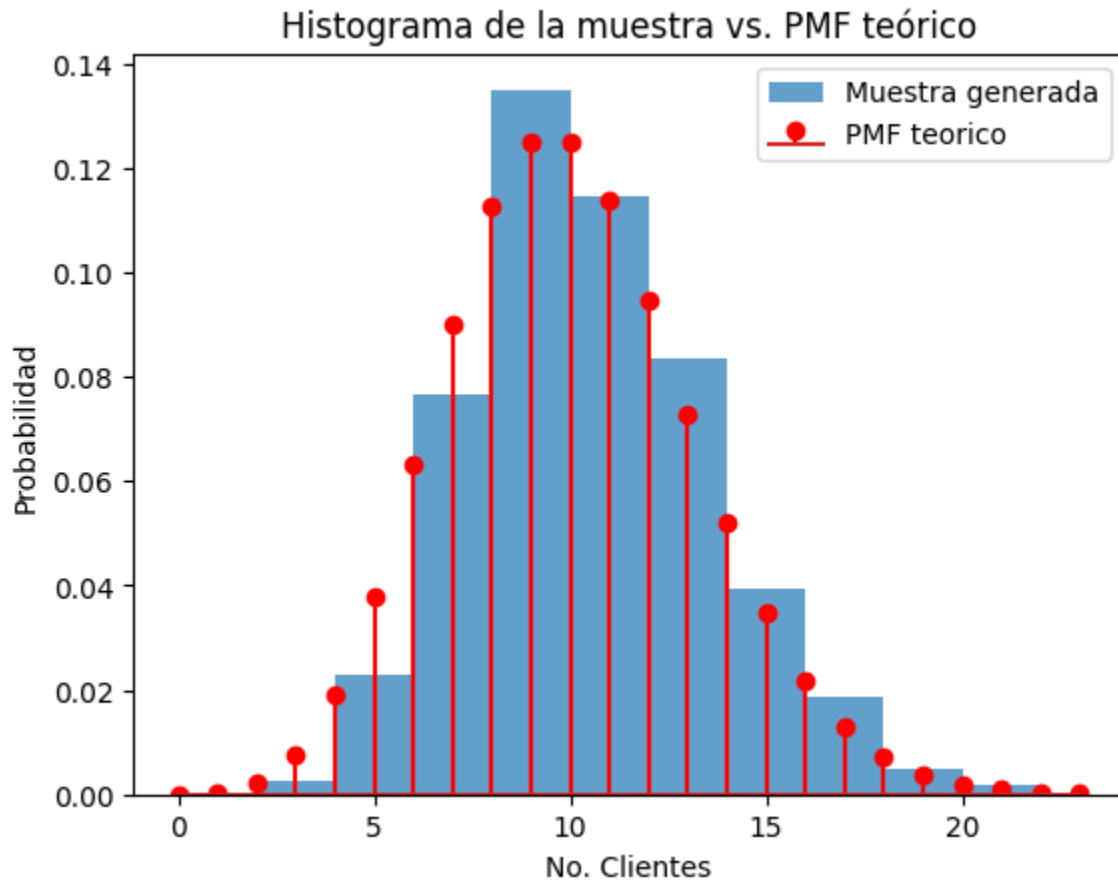
muestra_inversa = muestreo_inversa(lamb=10, size=1000)
media_inversa = np.mean(muestra_inversa)
varianza_inversa = np.var(muestra_inversa)

# Datos teoricos
pmf = poisson.pmf(horas, mu=10)
media_teorica = poisson.mean(mu=10)
varianza_teorica = poisson.var(mu=10)

# Graficos
plt.hist(muestra_inversa, density=True, alpha=0.7, label='Muestra generada')
plt.stem(pmf, 'r', label='PMF teorico')
plt.xlabel('No. Clientes')
plt.ylabel('Probabilidad')
plt.title('Histograma de la muestra vs. PMF teórico')
plt.legend()
plt.show()

# Media y Varianza
data = {
```

```
"Description": ["Media de la muestra", "Varianza de la muestra", "Media de teor",  
"Value": [media_inversa, varianza_inversa, media_teorica, varianza_teorica]  
}  
  
display(pd.DataFrame(data))
```



	Description	Value
0	Media de la muestra	10.012000
1	Varianza de la muestra	9.393856
2	Media de teórica	10.000000
3	Varianza teórica	10.000000

## Task 2.2

Defina una distribución de propuesta que sea más fácil de muestrear y que cubra el soporte de la distribución de Poisson de destino. Por ejemplo, puede elegir una distribución uniforme o geométrica.

1. Calcule la constante  $C$  para acotar la relación entre el PMF objetivo y el PMF propuesto en todo el soporte de la distribución de Poisson.

```
In [ ]: """Se eligio la distribución Geométrica"""

def C_constant(lmbd):
    p = 1 / (lmbd + 1)
    max_ratio = max((math.exp(-lmbd) * (lmbd ** k) / math.factorial(k)) / (((1 - p)
    return max_ratio

lambdaV = 10
C = C_constant(lambdaV)
print("Constante C_constant:", C)
```

Constante C\_constant: 3.2450321183453124

2. Implemente el método de rechazo para generar muestras aleatorias a partir de la distribución de Poisson.

```
In [ ]: def metodo_rechazo(n_samples, lmbd):
    p = 1 / (lmbd + 1)
    muestras_obtenidas = []

    while len(muestras_obtenidas) < n_samples:
        Y = random.randint(1, 100)
        U = random.uniform(0, 1)
        if U <= (math.exp(-lmbd) * (lmbd ** Y) / (C * ((1 - p) ** (Y - 1)) * p)):
            muestras_obtenidas.append(Y)

    return muestras_obtenidas
```

3. Genere una muestra aleatoria de tamaño 1000 que represente el número de clientes que llegan en una hora utilizando el método de rechazo.

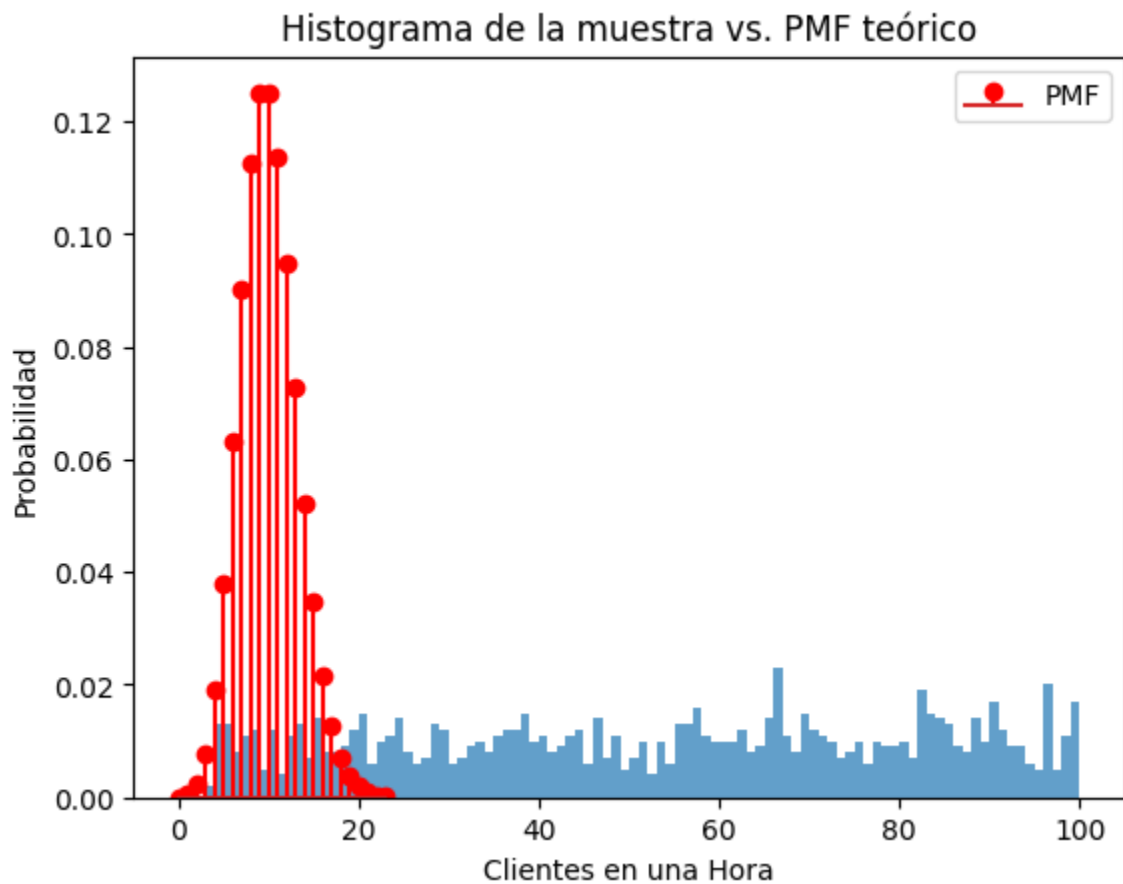
```
In [ ]: lambdaV = 10
cantidad_muestrasA = 1000
samples = metodo_rechazo(cantidad_muestrasA, lambdaV)
print("Muestra aleatoria:", samples)
```

Muestra aleatoria: [34, 32, 26, 5, 76, 84, 18, 50, 19, 37, 58, 28, 86, 74, 18, 75, 3, 3, 15, 34, 86, 85, 37, 39, 86, 20, 89, 88, 57, 67, 48, 46, 23, 10, 12, 12, 78, 57, 7, 4, 48, 15, 93, 9, 10, 59, 27, 89, 56, 54, 19, 38, 93, 85, 70, 43, 27, 67, 42, 7, 90, 39, 82, 40, 46, 10, 82, 13, 86, 63, 66, 92, 3, 28, 15, 38, 23, 90, 66, 25, 62, 5, 5, 50, 11, 66, 82, 17, 74, 85, 83, 38, 65, 96, 57, 27, 44, 97, 55, 4, 40, 68, 47, 90, 3, 0, 53, 61, 65, 75, 58, 17, 58, 58, 51, 57, 44, 20, 44, 21, 23, 67, 82, 74, 11, 13, 9, 1, 20, 83, 96, 73, 11, 38, 75, 100, 83, 41, 71, 17, 48, 49, 62, 39, 47, 55, 45, 81, 29, 84, 61, 94, 70, 41, 18, 39, 6, 72, 63, 93, 53, 29, 82, 28, 23, 100, 93, 77, 96, 36, 10, 56, 62, 70, 36, 44, 96, 76, 16, 36, 66, 63, 95, 14, 12, 19, 32, 20, 16, 60, 6, 84, 15, 41, 7, 90, 4, 85, 50, 84, 33, 87, 91, 82, 56, 96, 60, 82, 82, 85, 85, 37, 64, 32, 22, 22, 35, 84, 76, 79, 78, 83, 77, 45, 69, 76, 50, 89, 51, 69, 60, 90, 15, 70, 87, 13, 100, 4, 91, 68, 75, 81, 99, 98, 88, 33, 93, 65, 77, 90, 67, 37, 4, 8, 6, 6, 94, 57, 39, 29, 35, 20, 40, 96, 55, 18, 13, 56, 31, 78, 22, 66, 54, 61, 55, 5, 4, 2, 72, 88, 66, 33, 37, 31, 65, 61, 24, 85, 66, 70, 44, 71, 29, 63, 5, 30, 99, 73, 8, 6, 73, 78, 85, 100, 37, 40, 53, 87, 49, 66, 61, 74, 46, 72, 67, 43, 99, 18, 56, 10, 20, 5, 80, 69, 21, 96, 66, 46, 13, 91, 53, 41, 20, 90, 33, 55, 100, 89, 8, 13, 75, 6, 9, 57, 96, 56, 4, 51, 50, 46, 68, 82, 15, 53, 59, 82, 50, 77, 22, 92, 56, 16, 79, 2, 9, 6, 45, 35, 8, 94, 67, 92, 8, 28, 69, 83, 98, 57, 79, 61, 52, 10, 68, 88, 58, 72, 81, 4, 98, 43, 24, 4, 70, 92, 24, 57, 36, 90, 96, 90, 4, 10, 76, 46, 88, 4, 69, 10, 83, 99, 93, 88, 99, 72, 51, 68, 26, 24, 85, 71, 36, 19, 92, 7, 46, 29, 48, 57, 35, 4, 6, 47, 32, 54, 80, 51, 84, 83, 53, 82, 73, 74, 98, 67, 37, 39, 83, 44, 8, 59, 36, 4, 9, 59, 96, 69, 36, 13, 78, 71, 65, 70, 90, 73, 45, 22, 59, 26, 43, 24, 40, 53, 13, 1, 9, 66, 48, 78, 7, 50, 6, 53, 55, 24, 80, 56, 19, 66, 51, 21, 38, 40, 15, 77, 19, 29, 31, 80, 4, 70, 89, 31, 29, 16, 94, 16, 10, 98, 81, 7, 40, 15, 55, 66, 36, 4, 67, 80, 24, 100, 75, 41, 35, 26, 60, 84, 46, 10, 90, 34, 27, 35, 79, 5, 58, 62, 69, 34, 13, 90, 17, 29, 14, 75, 4, 58, 64, 56, 65, 37, 77, 57, 23, 32, 26, 66, 19, 98, 91, 23, 6, 1, 83, 84, 90, 12, 47, 15, 5, 12, 91, 44, 66, 65, 79, 96, 59, 20, 69, 74, 9, 40, 88, 92, 44, 79, 88, 64, 27, 55, 55, 90, 65, 66, 28, 21, 87, 52, 67, 26, 41, 25, 44, 98, 47, 16, 43, 7, 83, 83, 11, 96, 67, 58, 12, 51, 79, 85, 28, 10, 72, 60, 81, 15, 95, 1, 6, 95, 83, 33, 56, 70, 55, 71, 38, 27, 79, 69, 66, 82, 8, 40, 48, 48, 83, 7, 20, 91, 57, 22, 84, 3, 62, 46, 65, 41, 25, 98, 7, 12, 42, 81, 64, 55, 14, 72, 65, 72, 68, 5, 4, 8, 87, 22, 64, 69, 5, 64, 88, 82, 96, 79, 42, 60, 69, 32, 42, 87, 8, 82, 48, 96, 53, 62, 24, 44, 6, 80, 82, 87, 96, 65, 76, 30, 82, 19, 59, 20, 71, 68, 82, 77, 78, 5, 1, 28, 36, 86, 92, 60, 22, 51, 90, 23, 34, 8, 73, 88, 70, 35, 84, 46, 80, 9, 65, 63, 97, 28, 14, 37, 48, 20, 56, 28, 24, 20, 60, 71, 44, 96, 46, 91, 77, 23, 61, 6, 34, 7, 5, 25, 43, 7, 57, 84, 36, 96, 37, 35, 90, 89, 63, 84, 13, 96, 38, 38, 85, 15, 82, 9, 1, 60, 19, 23, 93, 33, 52, 73, 59, 38, 25, 28, 22, 89, 29, 43, 66, 17, 38, 58, 47, 8, 2, 8, 23, 86, 15, 12, 45, 53, 28, 20, 88, 98, 94, 42, 5, 24, 48, 20, 29, 62, 72, 18, 97, 69, 80, 97, 70, 62, 43, 64, 97, 38, 45, 90, 91, 30, 92, 18, 6, 15, 49, 63, 51, 4, 0, 24, 86, 32, 32, 31, 9, 92, 58, 5, 38, 88, 18, 56, 80, 57, 20, 7, 28, 69, 86, 38, 29, 33, 33, 85, 89, 55, 44, 95, 91, 24, 93, 31, 21, 25, 39, 70, 88, 39, 62, 47, 57, 8, 30, 89, 37, 71, 17, 83, 83, 55, 80, 71, 25, 46, 19, 28, 62, 13, 66, 35, 69, 64, 5, 4, 99, 61, 64, 60, 34, 17, 78, 5, 66, 9, 14, 67, 23, 39, 36, 7, 96, 14, 65, 40, 77, 33, 14, 63, 66, 39, 43, 22, 74, 41, 95, 61, 31, 35, 75, 89, 94, 88, 57, 91, 15, 66, 54, 35, 19, 87, 99, 13, 10, 81, 75, 21, 12, 4, 12, 46, 36, 59, 38, 12, 24, 100, 59, 49, 65, 42, 57, 43, 48, 99, 62, 34, 13, 56, 5, 18, 96, 42, 71, 58, 52, 6, 77, 24, 3, 2, 84, 42, 100, 16, 25, 98, 93, 37, 100, 38, 84, 98, 85, 71, 66, 27, 72, 17, 78, 62, 43, 30, 68, 8]

4. Trace un histograma de la muestra generada y compárelo con el PMF teórico de la distribución de Poisson.

```
In [ ]: plt.hist(samples, bins=range(min(samples), max(samples) + 1), density=True, alpha=0
x = range(min(samples), max(samples) + 1)
pmf_theoretical = [(math.exp(-lambdaV) * (lambdaV ** k) / math.factorial(k)) for k
```

```
plt.stem(pmf, 'r', label='PMF')
plt.xlabel('Clientes en una Hora')
plt.ylabel('Probabilidad')
plt.title('Histograma de la muestra vs. PMF teórico')
plt.legend()
plt.show()
```



5. Calcule la media y la varianza de la muestra generada y compárelas con los valores teóricos.

```
In [ ]: data = {
    "Description": ["Media de la muestra", "Varianza de la muestra", "Media de teor
    "Value": [np.mean(samples), np.var(samples), lambdaV, lambdaV]
}

display(pd.DataFrame(data))
```

	Description	Value
0	Media de la muestra	51.5500
1	Varianza de la muestra	776.2155
2	Media de teórica	10.0000
3	Varianza teórica	10.0000



**Responda:**

1. Compare los resultados de los dos métodos. ¿Qué método proporciona un mejor ajuste a la distribución de Poisson objetivo?

R// Se puede argumentar que la que tiene un mejor ajuste corresponde a la distribución con el método de transformación inversa, ya que cómo se puede apreciar en su gráfica PMF esta es fiel a los valores proporcionados

2. Discuta las ventajas y desventajas de cada método en términos de eficiencia y precisión.

R// Se puede argumentar que el método del rechazo es simple y fácil de implementar, además de que es flexible entre samples, el problema es su precisión, cómo se puede ver en el histograma del inciso anterior comparado con el del 2.1. Mientras que el método de transformación inversa presentó una mejor adaptabilidad a la muestra, dejando cómo evidencia las gráficas por lo que su precisión no está a discusión, pero es limitado respecto a que distribuciones aplicarlo.

3. Considere diferentes escenarios, como cambiar la tasa de llegada promedio ( $\lambda$ ) o usar diferentes distribuciones de propuestas. ¿Cómo funcionan los métodos en estos escenarios?

R// En estos casos, el muestreo depende también de la distribución propuesta. En este caso se puede mejorar la precisión realizando una buena elección de distribución propuesta. Sin embargo, la transformación a la inversa sigue siendo mejor en términos de desempeño que la transformación inversa debido a que aprovecha la inversa de la cdf de la distribución original para lograr una probabilidad más precisa.