

# Corto 1

## Integrantes:

- Juan Diego Avila 20090
- Diego Franco 20240

## Parte 1 - Ejercicios Prácticos

### Tasks 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?**

Si ya que se tiene un intervalo de tiempo continuo y se tiene la frecuencia histórica en la que ocurre un huracán. Asimismo se tiene el promedio de huracanes por año y son eventos independientes. La distribución de Poisson es adecuada para modelar la ocurrencia de eventos raros en un rango de tiempo continuo.

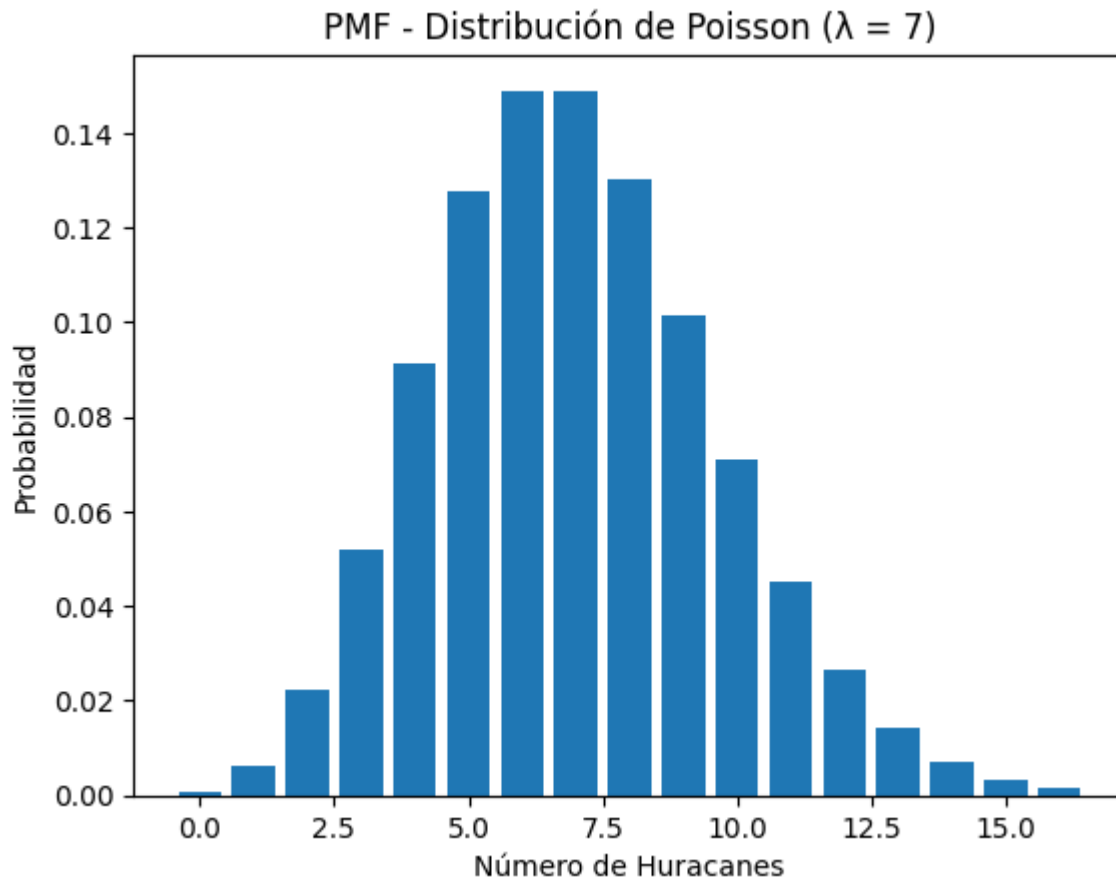
**2. Considere que usted analizará hasta un máximo de 16 huracanes este año. Grafique PMF (probability mass function) de estos eventos**

```
import matplotlib.pyplot as plt
from scipy.stats import poisson

media = 7
maximo = 16

x = range(maximo + 1)
pmf_values = poisson.pmf(x, media)

plt.bar(x, pmf_values)
plt.xlabel('Número de Huracanes')
plt.ylabel('Probabilidad')
plt.title(f'PMF - Distribución de Poisson ( $\lambda = \{media\}$ )')
plt.show()
```



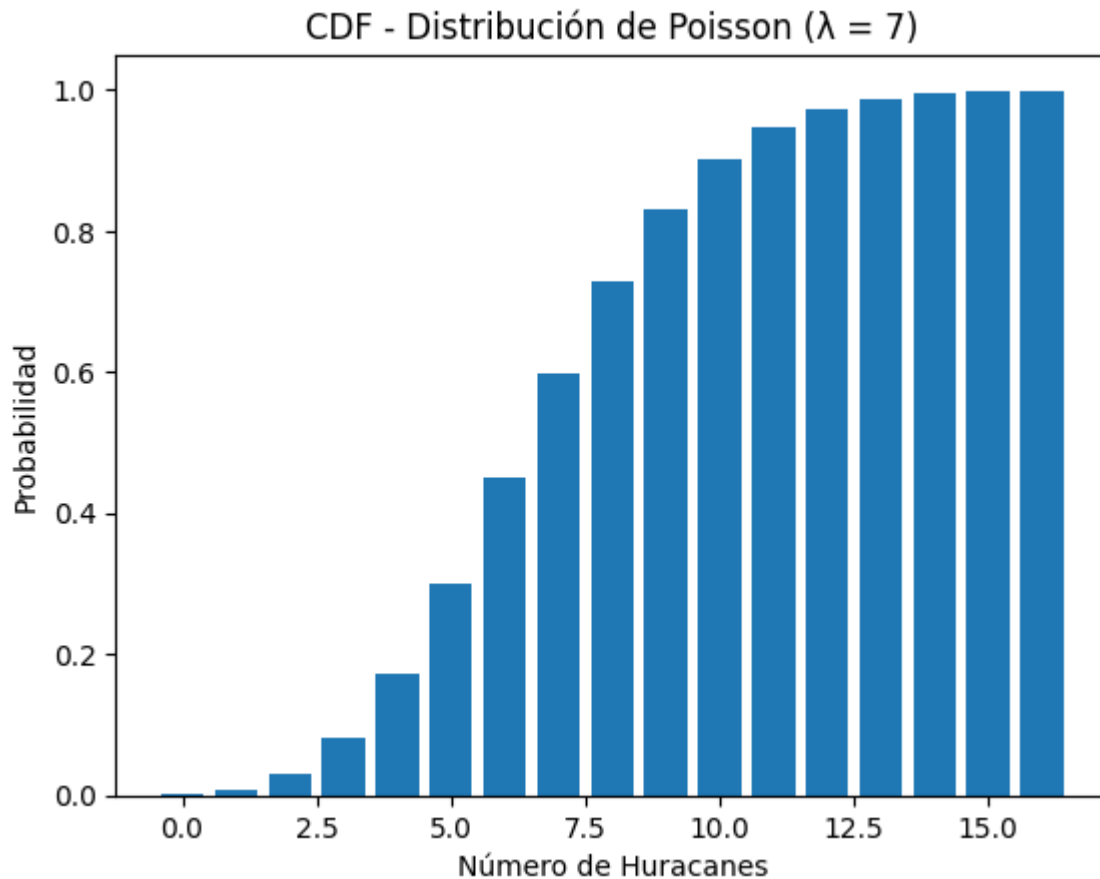
3. Considere que usted analizará hasta un máximo de 16 huracanes este año. Grafique CDF (cumulative distribution function) de estos eventos

```
import matplotlib.pyplot as plt
from scipy.stats import poisson

media = 7
maximo = 16

x = range(maximo + 1)
pmf_values = poisson.cdf(x, media)

plt.bar(x, pmf_values)
plt.xlabel('Número de Huracanes')
plt.ylabel('Probabilidad')
plt.title(f'CDF - Distribución de Poisson ( $\lambda = \{media\}$ )')
plt.show()
```



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

- De la PMF se puede concluir que la probabilidad mas alta de ocurrencia de un huracán tiende siempre a la media, siendo desta de 7 huracanes al año. Disminuyendo la probabilidad de ocurrencia a medida que se aleja de la media.
- De la CMF indica que la probabilidad de que ocurran hasta cierta cantidad de huracanes al año. Cuando la cantidad de huracanes por año es de 7 o menor la probabilidad es de 0.5, y a medida de que se acerca a el limite de huracanes, la probabilidad tiene a 1.
- En general se puede observar la tendencia de cantidad de huracanes por año, lo que ayuda a entender la probabilidad de ocurrencia de huracanes, y permitir tomar decisiones en base a esto.

## Tasks 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

```
import numpy as np
import math
import matplotlib.pyplot as plt
from scipy.stats import poisson

def poisson_cdf(lmbda, k):
    cdf = 0
    for i in range(k + 1):
        cdf += (lmbda ** i) * np.exp(-lmbda) / math.factorial(i)
    return cdf

def poisson_inversa_cdf(lmbda, y):
    k = 0
    cdf = poisson_cdf(lmbda, k)
    while cdf < y:
        k += 1
        cdf = poisson_cdf(lmbda, k)
    return k

def generar_muestra_poisson(num_samples, lmbda):
    samples = []
    for _ in range(num_samples):
        U = np.random.uniform()
        x = poisson_inversa_cdf(lmbda, U)
        samples.append(x)
    return samples
```

## 2. Genere una muestra aleatoria de tamaño 1000 que represente el número de clientes que llegan en una hora.

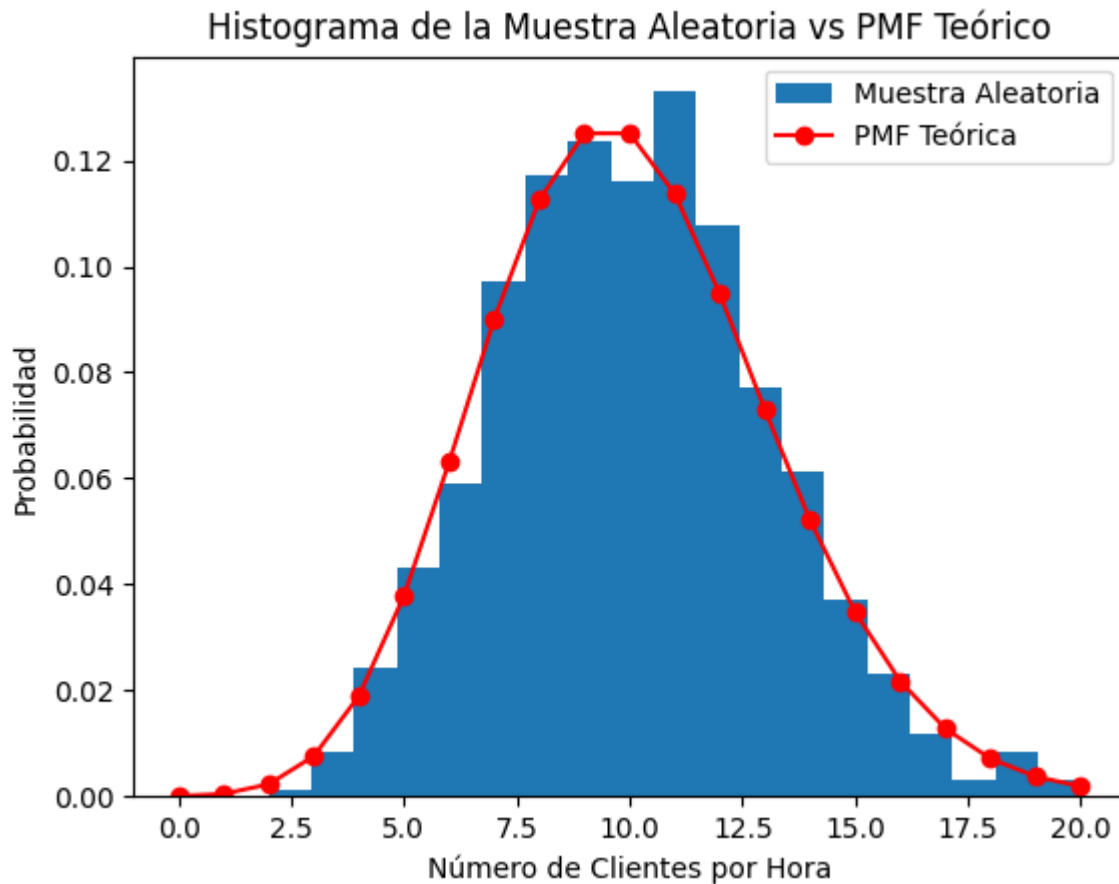
```
promedio_por_hora = 10
tamano_muestra = 1000

muestra_aleatoria = generar_muestra_poisson(tamano_muestra, promedio_por_hora)
```

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

```
histograma, bins, _ = plt.hist(muestra_aleatoria, bins=max(muestra_aleatoria)-min(muestra_aleatoria)+1)
x = np.arange(0, max(muestra_aleatoria)+1)
pmf_teorica = [poisson.pmf(k, promedio_por_hora) for k in x]
plt.plot(x, pmf_teorica, 'ro-', label='PMF Teórica')
```

```
plt.title('Histograma de la Muestra Aleatoria vs PMF Teórico')
plt.xlabel('Número de Clientes por Hora')
plt.ylabel('Probabilidad')
plt.legend()
plt.show()
```



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

```
media_muestra = np.mean(muestra_aleatoria)
varianza_muestra = np.var(muestra_aleatoria)

media_teórica = promedio_por_hora
varianza_teórica = promedio_por_hora

print("Media de la muestra generada:", media_muestra)
print("Media teórica:", media_teórica)
print("Varianza de la muestra generada:", varianza_muestra)
print("Varianza teórica:", varianza_teórica)
```

Media de la muestra generada: 10.045

Media teórica: 10

Varianza de la muestra generada: 10.056975

Varianza teórica: 10

Como se puede observar el valor de la media de la muestra generada es bastante cercano al valor teórico, por lo que se puede concluir que la muestra generada es representativa de la distribución teórica.

Por otro lado sobre la comparación de las varianzas se puede observar que el valor de la varianza de la muestra generada es bastante cercano al valor teórico, por lo que se puede concluir que la dispersión de los datos de la muestra generada es consistente con la de la distribución teórica.

## 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

*# Se va a utilizar una distribucion uniforme*

```
import numpy as np
from scipy.stats import poisson

lmbda = 10

def calculr_C(lmbda, k):
    p = 1 / (lmbda + 1)
    max_poisson_pmf = poisson.pmf(lmbda, k)
    max_geom_pmf = p * (1 - p) ** k
    C = max_poisson_pmf / max_geom_pmf
    return C
```

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

```
def pmf_geometrica(x, p):
    return p * (1 - p) ** (x - 1)

def metodo_de_rechazo(size):
    C = calculr_C(lmbda, size)
    p = 1 / lmbda
    samples = []
    while len(samples) < size:
        x_geom = np.random.geometric(p)
        u = np.random.uniform()
        if u <= poisson.pmf(x_geom, lmbda) / (C * pmf_geometrica(x_geom, p)):
            samples.append(x_geom)
    return samples
```

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

```
muestra_aleatoria = metodo_de_rechazo(1000)

print(muestra_aleatoria)
```

```
[19, 11, 3, 12, 2, 23, 3, 26, 10, 2, 18, 5, 1, 29, 11, 4, 8, 12, 3, 2, 13, 20, 1, 18, 1, 14, 9,
2, 8, 14, 3, 5, 12, 25, 12, 16, 19, 5, 4, 8, 35, 5, 1, 29, 1, 18, 9, 9, 2, 5, 7, 11, 20, 20, 2,
6, 12, 25, 1, 16, 3, 11, 4, 23, 14, 13, 23, 26, 13, 12, 15, 5, 45, 16, 8, 1, 1, 5, 12, 1, 12, 15,
6, 7, 9, 14, 15, 3, 11, 19, 34, 9, 1, 34, 2, 7, 15, 1, 5, 17, 1, 1, 19, 19, 6, 8, 5, 5, 9, 14, 2,
2, 9, 9, 8, 3, 8, 3, 43, 23, 2, 1, 25, 10, 13, 38, 4, 21, 9, 30, 18, 5, 13, 2, 28, 3, 8, 1, 4, 1,
10, 7, 3, 6, 4, 2, 8, 1, 2, 2, 7, 14, 14, 12, 7, 6, 1, 10, 2, 17, 8, 27, 15, 4, 2, 11, 3, 22, 13,
1, 1, 47, 11, 8, 9, 4, 4, 33, 4, 6, 1, 23, 13, 12, 12, 14, 16, 5, 3, 4, 4, 5, 14, 19, 4, 8, 2, 2,
7, 2, 5, 5, 5, 6, 3, 5, 22, 6, 3, 1, 23, 25, 33, 4, 57, 7, 8, 14, 30, 1, 11, 4, 3, 13, 12, 1, 9,
6, 6, 4, 11, 15, 3, 9, 1, 7, 6, 4, 6, 15, 3, 21, 49, 18, 31, 14, 3, 8, 16, 8, 3, 21, 15, 5, 19,
10, 4, 4, 12, 10, 28, 6, 17, 14, 12, 2, 23, 36, 3, 7, 7, 26, 25, 1, 16, 15, 24, 9, 5, 1, 1, 17,
9, 13, 7, 1, 32, 28, 15, 2, 5, 12, 4, 2, 7, 1, 5, 2, 2, 6, 3, 2, 7, 3, 2, 11, 17, 2, 24, 21, 17,
1, 19, 8, 2, 7, 33, 9, 4, 3, 4, 22, 14, 25, 10, 7, 4, 4, 3, 9, 15, 18, 1, 12, 34, 3, 13, 1, 9, 9,
10, 9, 1, 6, 1, 12, 8, 2, 10, 5, 10, 14, 6, 9, 15, 10, 8, 16, 32, 5, 22, 1, 15, 2, 30, 6, 12, 4,
9, 1, 10, 1, 31, 1, 17, 18, 1, 3, 7, 9, 11, 4, 4, 2, 9, 16, 8, 13, 3, 6, 15, 1, 2, 1, 6, 2, 1, 7,
23, 16, 19, 12, 7, 16, 9, 3, 2, 3, 23, 12, 10, 16, 1, 5, 10, 7, 4, 3, 7, 3, 12, 3, 29, 2, 8, 9,
6, 4, 7, 17, 8, 4, 12, 22, 8, 3, 38, 4, 1, 12, 4, 10, 3, 10, 1, 7, 8, 1, 4, 2, 4, 1, 16, 6, 3, 3,
2, 13, 1, 4, 33, 9, 12, 23, 10, 15, 3, 3, 3, 2, 11, 6, 32, 3, 1, 13, 18, 4, 9, 27, 17, 2, 4, 2,
4, 3, 2, 19, 3, 7, 14, 2, 2, 9, 6, 18, 28, 8, 20, 4, 7, 3, 5, 8, 8, 9, 6, 8, 2, 6, 18, 22, 4, 3,
3, 12, 8, 18, 12, 10, 7, 8, 5, 25, 19, 11, 29, 1, 14, 7, 3, 1, 11, 15, 12, 13, 11, 24, 4, 1, 77,
2, 5, 3, 7, 1, 6, 2, 7, 5, 4, 2, 38, 3, 5, 13, 4, 2, 3, 9, 13, 2, 21, 4, 1, 1, 10, 6, 4, 9, 2,
23, 22, 3, 18, 8, 4, 42, 1, 3, 4, 1, 1, 7, 17, 5, 12, 7, 15, 15, 1, 8, 6, 9, 2, 13, 10, 7, 2, 6,
4, 4, 1, 3, 10, 15, 5, 1, 2, 3, 17, 3, 6, 3, 11, 9, 4, 14, 1, 24, 1, 3, 7, 10, 1, 1, 4, 1, 20, 9,
8, 11, 1, 11, 12, 7, 7, 24, 3, 7, 3, 4, 6, 10, 28, 9, 18, 14, 6, 2, 27, 13, 40, 4, 5, 21, 5, 2,
11, 1, 2, 2, 41, 13, 2, 9, 7, 14, 72, 17, 13, 4, 4, 3, 5, 11, 14, 21, 2, 36, 26, 9, 8, 12, 5, 4,
11, 3, 15, 14, 3, 24, 12, 1, 8, 1, 22, 28, 7, 12, 1, 22, 7, 4, 9, 17, 2, 16, 1, 2, 24, 31, 11, 1,
4, 5, 7, 12, 3, 15, 19, 19, 3, 7, 12, 3, 6, 42, 6, 3, 3, 17, 3, 8, 2, 3, 22, 3, 20, 1, 2, 14, 21,
6, 4, 12, 9, 13, 8, 3, 8, 10, 13, 17, 10, 4, 4, 11, 42, 1, 30, 9, 17, 18, 12, 11, 38, 11, 6, 12,
1, 1, 10, 3, 4, 3, 6, 5, 1, 3, 20, 3, 16, 1, 14, 4, 8, 2, 21, 1, 1, 4, 3, 5, 1, 8, 13, 5, 8, 1,
25, 6, 15, 4, 18, 4, 1, 5, 11, 4, 12, 6, 11, 12, 8, 3, 19, 11, 3, 19, 17, 4, 9, 4, 2, 8, 3, 6, 1,
34, 8, 6, 1, 7, 4, 28, 6, 8, 14, 28, 1, 3, 1, 6, 1, 1, 1, 4, 6, 34, 29, 3, 11, 2, 2, 3, 4, 24,
18, 3, 18, 46, 15, 12, 6, 8, 9, 2, 1, 1, 1, 11, 8, 2, 39, 1, 1, 6, 5, 6, 7, 28, 9, 2, 3, 4, 10,
2, 1, 2, 20, 3, 39, 3, 6, 3, 11, 6, 16, 11, 7, 1, 11, 10, 11, 5, 5, 3, 5, 5, 2, 5, 1, 10, 24, 11,
25, 10, 4, 4, 5, 3, 1, 18, 3, 12, 6, 35, 39, 2, 3, 5, 2, 4, 2, 1, 3, 2, 17, 6, 9, 2, 8, 1, 4, 13,
13, 26, 23, 3, 2, 14, 2, 4, 6, 31, 3, 9, 5, 7, 1, 6, 1, 21, 7, 3, 22, 16, 7, 3, 15, 14, 1, 14,
20, 6, 8, 2, 12, 12, 18, 10, 16, 5, 2, 2, 4, 5, 5, 2, 12, 4, 13, 25, 2, 14, 5, 7, 1, 1]
```

C:\Users\diego\AppData\Local\Temp\ipykernel\_17496\4120136128.py:11: RuntimeWarning: divide by zero encountered in double\_scalars

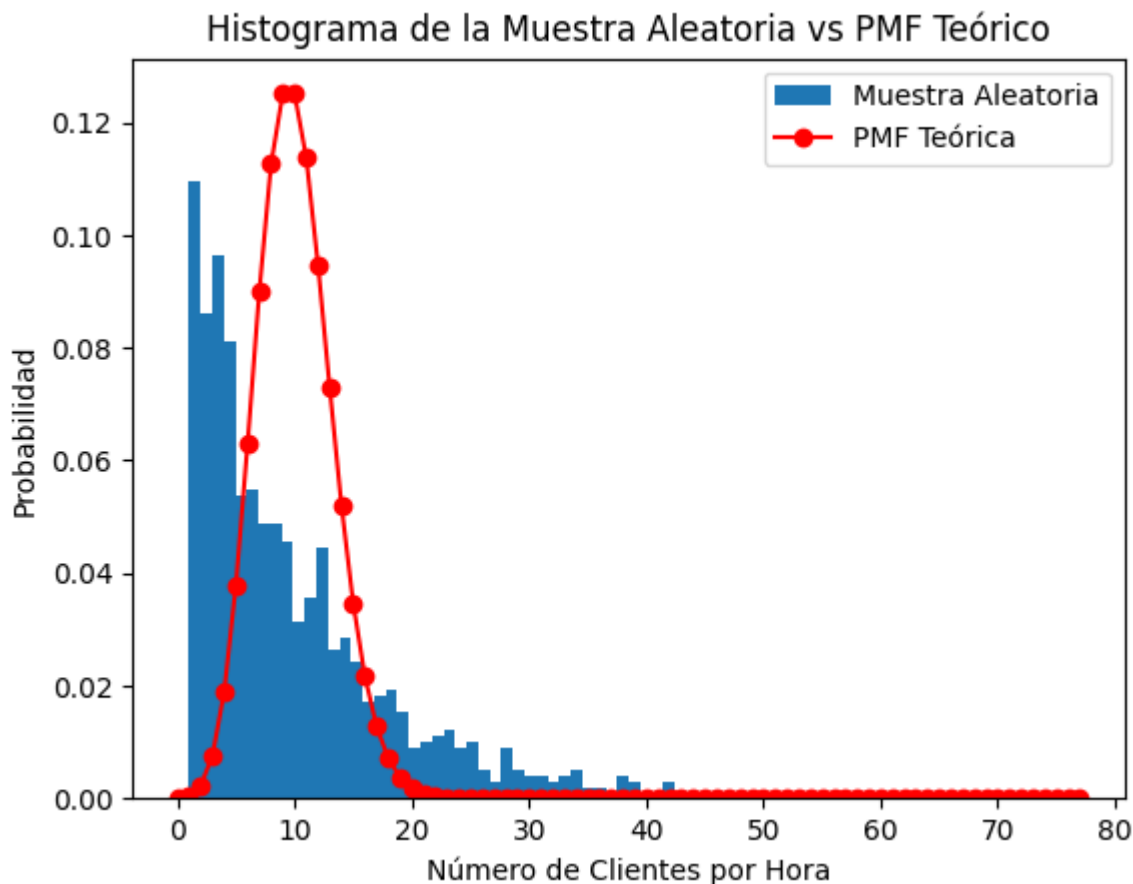
```
if u <= poisson.pmf(x_geom, lmbda) / (C * pmf_geometrica(x_geom, p)):
```

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

```

histograma, bins, _ = plt.hist(muestra_aleatoria, bins=max(muestra_aleatoria)-min(muestra_aleatoria)+1)
x = np.arange(0, max(muestra_aleatoria)+1)
pmf_teorica = [poisson.pmf(k, lambda) for k in x]
plt.plot(x, pmf_teorica, 'ro-', label='PMF Teórica')
plt.title('Histograma de la Muestra Aleatoria vs PMF Teórico')
plt.xlabel('Número de Clientes por Hora')
plt.ylabel('Probabilidad')
plt.legend()
plt.show()

```



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

```

media_muestra = np.mean(muestra_aleatoria)
varianza_muestra = np.var(muestra_aleatoria)

media_teórica = lambda
varianza_teórica = lambda

print("Media de la muestra generada:", media_muestra)
print("Media teórica:", media_teórica)
print("Varianza de la muestra generada:", varianza_muestra)
print("Varianza teórica:", varianza_teórica)

```



Media de la muestra generada: 9.793

Media teórica: 10

Varianza de la muestra generada: 86.60815099999998

Varianza teórica: 10

Al comparar el valor de la media de la muestra generada y la muestra teórica se puede observar que son valores bastante cercanos, por lo que se concluye que la muestra que se generó se asemeja bastante a la distribución teórica.

Por otro lado sobre la comparación de las varianzas se puede observar que el valor de la varianza de la muestra generada es bastante alta en comparación con el valor teórico, por lo que se puede concluir que la dispersión de los datos de la muestra generada es mayor a la de la distribución teórica.

## Responda:

1. Compare los resultados de los dos métodos. ¿Qué método proporciona un mejor ajuste a la distribución de Poisson objetivo?
  - Como se pudo observar, los dos métodos dieron buenos resultados, pero el que tiene mejor ajuste a la distribución de Poisson objetivo es el método de transformación inversa, ya que los puntos en la gráfica se encuentran más cerca de la línea de la distribución de Poisson objetivo y los valores tanto de media y varianza se encuentran más cerca de los valores de la distribución de Poisson objetivo. Cabe mencionar que en el método de rechazo los valores generados muestran una dispersión bastante mayor a la de la inversa lo que hace que el modelo no sea tan bueno.
2. Discuta las ventajas y desventajas de cada método en términos de eficiencia y precisión.
  - Ventajas:
    - Método de transformación inversa:
      - El método de transformación inversa genera valores que si se ajustan a la distribución de Poisson objetivo.
      - En general, este método es más eficiente que el método de rechazo y más acertado, lo que se pudo demostrar con la comparación de los valores generados y teóricos.
    - Método de rechazo:
      - El método de rechazo es más flexible y puede aplicarse a una amplia gama de distribuciones de Poisson.
      - La implementación del método de rechazo es más simple.
  - Desventajas:
    - Método de transformación inversa:
      - En algunos casos, encontrar la función inversa puede ser complicado o incluso imposible.
      - El método de transformación inversa puede no ser adecuado para algunas distribuciones de Poisson complejas.
    - Método de rechazo:
      - El método de rechazo puede ser menos eficiente, especialmente cuando la tasa de rechazo de valores generados es alta.

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?
- El metodo que mejor funcionaria para diferentes distribuciones propuestas seria el de rechazo, por su flexibilidad y facilidad de implementación.
  - Si cambia la tasa de llegada promedio ( $\lambda$ ) el metodo que mejor funcionaria seria el de transformación inversa, ya que este metodo genera valores que se ajustan a la distribución de Poisson objetivo.