Tarea 3

Probabilidad y Estadística Aplicada

Prof: Maglis Mujica

Autores: Juan Nocetti - 5.390.966-5

Franco Barlocco - 5.308.038-6

Ionas Josponis - 5.242.903-0

10 de Mayo de 2024

Índice

Introducción

En este proyecto aplicamos los conceptos estudiados en clase para simular variables aleatorias discretas utilizando Python. El objetivo es generar muestras a partir de distribuciones binomiales, geométricas y de Poisson según los valores proporcionados, para luego realizar una estadística descriptiva.

Objetivos

1- Simular variables aleatorias discretas en Python.
2- Obtener muestras para analizar estadísticas descriptivas para cada distribución.
3- Hallar la media, mediana y la moda.
4- Comparar las medidas de tendencia central con la esperanza y varianza.
5- Representar gráficamente dichas muestras.
6- Verificar empíricamente la ley de los grandes números

Marco Teórico

Estadística descriptiva: La estadística descriptiva es una rama de la estadística que se usa para resumir información y poder sacar conclusiones sobre datos. Nos permite tener una mejor comprensión de la información cuantitativa. Además, capacita a la persona para evaluar objetivamente y efectivamente la información que recibe (vía tablas, gráficos, porcentajes, tasas, etc.).

Variable aleatorio: Definir una variable aleatoria en un experimento consiste en asociar un valor numérico a cada suceso elemental del experimento. Interesa fundamentalmente asignar probabilidades a dichos valores numérico

Variables aleatorias discretas: Una variable aleatoria es discreta cuando sólo puede tomar unos ciertos valores enteros en un número finito de valores.

Variable aleatoria continua: es un tipo de variable aleatoria que puede tomar cualquier valor dentro de un intervalo de resultados.

Mediana: La mediana es un número que divide una lista de valores en dos partes iguales. Esto significa que la mitad de los valores son menores que la mediana y la otra mitad son mayores. Se obtiene ordenando primero los datos desde el más pequeño al más grande (con cualquier valor repetido incluido de modo que cada observación muestral aparezca en la lista ordenada).
Si tenemos n datos, se puede usar las siguientes fórmulas:

- Si n es impar, la mediana es el dato que ocupa la posición central

$$X = x\left(\frac{n+1}{2}\right)$$

- Si n es par, la mediana es el promedio de los dos datos centrales.

$$X = \frac{x\left(\frac{n}{2}\right) + x\left(\frac{n+1}{2}\right)}{2}$$

Media: La media es el valor promedio de un conjunto de datos numéricos. Se calcula como la suma del conjunto de valores dividida entre el número total de valores.

$$X = \frac{x_1 + x_2 + \ldots + x_n}{n}$$

Moda: La moda es el dato que ocurre con mayor frecuencia. No tiene una fórmula en sí mismo. Lo que habría que realizar es la suma de las repeticiones de cada valor.

Ley de los grandes números: La ley de los grandes números es un teorema fundamental de la teoría de la probabilidad que indica que si repetimos muchas veces (tendiendo al infinito) un mismo experimento, la frecuencia de que suceda un cierto evento tiende a ser una constante.

Distribución Binomial: Esta distribución consiste en calcular la probabilidad de tener cierta cantidad de éxitos en las repeticiones de un evento x veces. Dependiendo claramente de la probabilidad general en la que ocurra dicho evento.

$$P[X \leq x] = \sum_{k=0}^{x} \binom{n}{k} p^k (1-p)^{n-k}$$

x = cantidad de repeticiones deseadas con éxito.
n = cantidad de repeticiones.
p = probabilidad de ocurrencia del evento.

Distribución Geométrica: Es una distribución que tiene como resultado la probabilidad que se de un evento por primera vez en un número de intentos dados. Esta función recibe como parámetro el número de ensayos, y se debe saber calcular la probabilidad de que ocurra dicho evento (p).

Su fórmula es la siguiente:

$$P[X \leq x] = 1 - (1 - p)^x$$

x = número de intentos.
p = probabilidad de ocurrencia del evento.

Ensayo de Bernoulli: Ensayo que tiene como resultado dos posibles eventos, éxito y fracaso. Tomándose éxito como “p” y fracaso como “1 - p”.

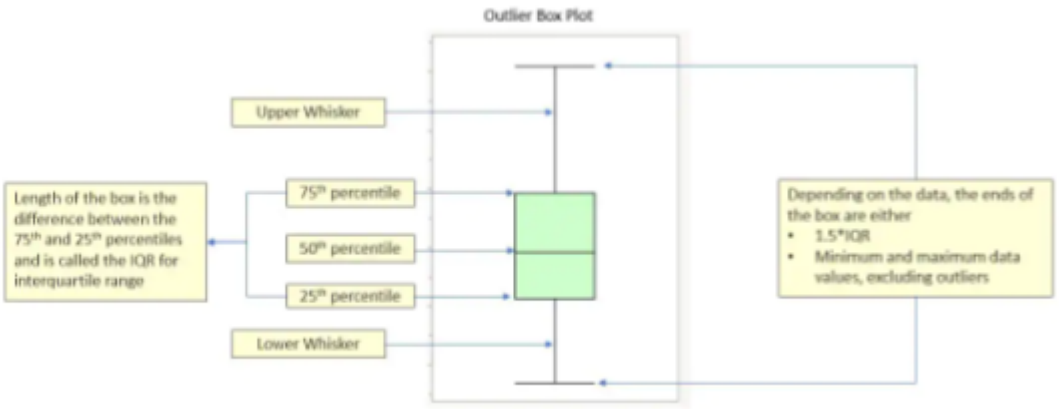
Distribución de Poisson: La distribución de Poisson indica la probabilidad de que ocurra un cierto número de eventos en un período de tiempo o en un espacio en específico, bajo la condición de que los eventos sucedan a una tasa promedio constante.

$$P(X = k) = e^{-\lambda}(\frac{\lambda^k}{k!})$$

k = número de éxitos

λ = probabilidad de suceso en un tiempo dado

Diagrama de caja: Es un diagrama de cajas que muestran las distribución de datos para variables continuas. Ayudan a ver el centro y la extensión de los datos.



Partes del diagrama:

Línea central: La línea central de la caja indica la mediana de los datos. Una mitad de los datos está por debajo de este valor, y la otra por encima. Si los datos son simétricos, la mediana estará en el centro de la caja. Si los datos están sesgados, la mediana estará más cerca de la parte superior o inferior de la caja.

Extremos superiores e inferiores: Los extremos de arriba y abajo de la caja indican los cuartiles, o percentiles, 25 y 75.
Se conocen como cuartiles porque separan cuartos (25 %) de los datos. La longitud de la caja es la diferencia entre estos dos percentiles y se conoce como rango intercuartílico (IQR).

Rango intercuartílico: Representa la distancia entre el primer y el tercer cuartil.

Bigotes: Las líneas que se extienden desde la caja se llaman bigotes. Los bigotes representan la varianza esperada de los datos.

Histogramas: Es un gráfico que permite mostrar cómo se distribuyen los datos de una muestra estadística. Esto, respecto a alguna variable numérica.
En el histograma se suelen usar barras, cuya altura dependerá de la frecuencia de los datos, que corresponde al eje Y. En tanto, en el eje X podemos observar la variable de estudio.

Esperanza: La esperanza en estadística es igual al sumatorio de las probabilidades de que exista un suceso aleatorio, multiplicado por el valor del suceso aleatorio. Dicho de otra forma, es el valor medio de un conjunto de datos.

Varianza: La varianza es una forma de entender cuanto se dispersan o varían los datos respecto a su media. Cuanto más se desvíe un valor de su media, más alta será la varianza, y viceversa.

Herramientas

- IDE: Es un entorno de desarrollo para poder programar, en nuestra aplicación usamos PyCharm de JetBrains y Visual Studio Code.

- **Python:** Lenguaje de desarrollo.

-**GitHub:** Es una herramienta para desarrollar software colectivamente.

-Numpy: Librería que brinda funcionalidades para análisis de datos en gran volumen.

-Matplotlib: Librería que brinda métodos para la creación de gráficos en 2D.

-Seaborn: Librería que permite la personalización de distintos gráficos.

-Scipy: Librería que tiene disponible funciones de cálculo científico.

Desarrollo

Para nuestro desarrollo del programa, generamos 4 archivos distintos “binomial.py”, “geometrica.py”, “poisson.py” y “main.py”.

Con estos archivos podemos ejecutar el programa(ejecutando el archivo main.py) para que nos muestre los resultados esperados.

binomial.py

generate_binomial_samples: Esta función toma tres argumentos: n, p y sample_sizes. En este caso n y p son los parámetros de la distribución binomial, mientras que sample_sizes es una lista de los tamaños de muestra deseados.

La función genera muestras aleatorias de la distribución binomial para cada tamaño de muestra especificado en sample_sizes.

```python
from scipy.stats import binom, mode
import numpy as np

def generate_binomial_samples(n, p, sample_sizes):
    samples = {size: binom.rvs(n, p, size=size) for size in sample_sizes}
    stats = {}
    for size, sample in samples.items():
        sample_mode = mode(sample)
        moda = sample_mode.mode[0] if isinstance(sample_mode.mode, np.ndarray) and len(sample_mode.mode) > 0 else sample_mode.mode
        stats[size] = {
            'mediana': np.median(sample),
            'moda': moda,
            'media': np.mean(sample),
            'varianza': np.var(sample)
        }
    return samples, stats
```

Luego, calcula varias estadísticas descriptivas para cada muestra generada:

- Calcula la moda utilizando la función mode(sample). Si hay múltiples modas, selecciona la primera. Si no hay ninguna moda, la moda se establece como None.
- Calcula la mediana utilizando np.median(sample).
- Calcula la media utilizando np.mean(sample).
- Calcula la varianza utilizando np.var(sample).

Por ultimo, retornamos un diccionario de muestras y un diccionario de estadísticas para cada tamaño de muestra.

geometrica.py

Este archivo hace lo mismo que “binomial.py” pero la diferencia que tienen es que en la función generate_geometric_samples, las muestras se generan utilizando geom.rvs(p, size=size), donde p es el parámetro de la distribución geométrica.
Al igual que Binomial, retornamos un diccionario de muestras y un diccionario de estadísticas para cada tamaño de muestra.

```python
from scipy.stats import geom, mode
import numpy as np

def generate_geometric_samples(p, sample_sizes):
    samples = {size: geom.rvs(p, size=size) for size in sample_sizes}
    stats = {}
    for size, sample in samples.items():
        sample_mode = mode(sample)
        moda = sample_mode.mode[0] if isinstance(sample_mode.mode, np.ndarray) and len(sample_mode.mode) > 0 else sample_mode.mode
        stats[size] = {
            'mediana': np.median(sample),
            'moda': moda,
            'media': np.mean(sample),
            'varianza': np.var(sample)
        }
    return samples, stats
```

poisson.py

Similar a las funciones anteriores, esta función utiliza poisson.rvs(lambda_, size=size) para generar muestras aleatorias de la distribución de Poisson. En esta distribución, lambda_ es el parámetro de la distribución de Poisson, que representa el número promedio de eventos que ocurren en un intervalo fijo de tiempo o espacio.

Repetimos el paso como en las demás distribuciones retornando un diccionario de muestras y un diccionario de estadísticas para cada tamaño de muestra.

```python
from scipy.stats import poisson, mode
import numpy as np

def generate_poisson_samples(lambda_, sample_sizes):
    samples = {size: poisson.rvs(lambda_, size=size) for size in sample_sizes}
    stats = {}
    for size, sample in samples.items():
        sample_mode = mode(sample)
        moda = sample_mode.mode[0] if isinstance(sample_mode.mode, np.ndarray) and len(sample_mode.mode) > 0 else sample_mode.mode
        stats[size] = {
            'mediana': np.median(sample),
            'moda': moda,
            'media': np.mean(sample),
            'varianza': np.var(sample)
        }
    return samples, stats
```

main.py

Desglosamos por partes el código con el fin de que se entienda las funciones del main.

Primero tenemos las importaciones que necesitamos para desarrollar el programa.

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy import stats
from binomial import generate_binomial_samples
from geometrica import generate_geometric_samples
from poisson import generate_poisson_samples
```

Siguiente a esto, seteamos los parámetros que necesitamos en base a la letra del entregable.

```
n = 100
p_binomial = 0.35
p_geometric = 0.08
lambda_poisson = 30
sample_sizes = [100,1000,10000,100000]

def plot_boxplots(samples, title):
    plt.figure(figsize=(10, 6))
    sns.boxplot(data=list(samples.values()))
    plt.title(title)
    plt.xlabel("Tamaño de Muestra")
    plt.ylabel("Valor")
    plt.show()

def plot_histograms(samples, title):
    plt.figure(figsize=(10, 6))
    for size, sample in samples.items():
        sns.histplot(sample, kde=False, bins=30, label=f'Tamaño={size}', element='step')
    plt.legend()
    plt.title(title)
    plt.xlabel("Valor")
    plt.ylabel("Frecuencia")
    plt.show()
```

La función **plot_boxplots**, crea un diagrama de caja para las muestras proporcionadas. Un diagrama de caja muestra la distribución de los datos y resalta la mediana, los cuartiles y los valores atípicos de haberlos.

La función **plot_histograms**, crea un histograma para las muestras proporcionadas. Un histograma muestra la frecuencia de ocurrencia de diferentes valores en los datos.

Con estas funciones, vamos a lograr visualizar los datos que necesitemos en base a diagramas de cajas e histogramas.

Luego tenemos el siguiente fragmento del código:

```python
def compare_stats_with_theoretical(stats, theoretical_values):
    for size, stat in stats.items():
        print(f"Muestra de tamaño {size}:")
        for key, value in stat.items():
            print(f"{key.capitalize()}: {value} (Teórico: {theoretical_values[size][key]})")

samples_binomial, stats_binomial = generate_binomial_samples(n, p_binomial, sample_sizes)

theoretical_values_binomial = {
    size: {
        'media': n * p_binomial,
        'varianza': n * p_binomial * (1 - p_binomial),
        'mediana': np.median(np.random.binomial(n, p_binomial, size)),
        'moda': stats.mode(np.random.binomial(n, p_binomial, size)).mode.item() if stats.mode(np.random.binomial(n, p_binomial, size))
        .count > 0 else np.nan
    } for size in sample_sizes
}

print("BINOMIAL")
plot_boxplots(samples_binomial, "Diagramas de Caja para Distribución Binomial")
plot_histograms(samples_binomial, "Histogramas para Distribución Binomial")
compare_stats_with_theoretical(stats_binomial, theoretical_values_binomial)
print("-----------------")
```

La media y varianza empírica se calculan en generate_binomial_samples y se comparan con la esperanza y varianza teórica en compare_stats_with_theoretical. Decidimos agregar la mediana teórica y la moda teórica para que el análisis quede más completo.

Esta función, toma dos argumentos: stats y theoretical_values.
El parámetro stats contiene las estadísticas calculadas a partir de las muestras generadas, mientras que theoretical_values contiene los valores teóricos esperados para estas estadísticas. La función itera sobre cada muestra y compara las estadísticas calculadas con los valores teóricos correspondientes.

La función **generate_binomial_samples:** Genera muestras de una distribución binomial y calcula las estadísticas para cada muestra.

Luego se generan las **Funciones de Visualización:**

- plot_boxplots: Muestra diagramas de caja para las muestras generadas.
- plot_histograms: Muestra histogramas para las muestras generadas.

De esta manera logramos visualizar el diagrama de caja e histograma para la distribución binomial. Por último, repetimos los últimos pasos para mostrar los diagramas para la geométrica y la de poisson.

Resultado:

Diagrama de caja distribución binomial

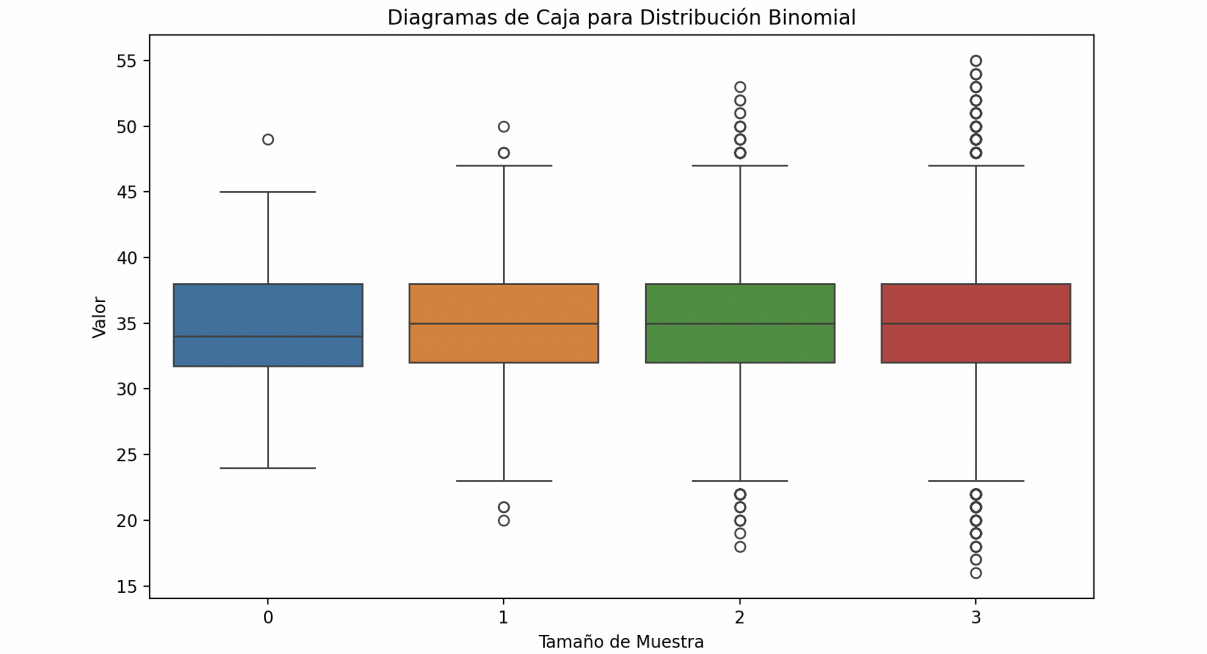


Diagrama de caja distribución geométrica

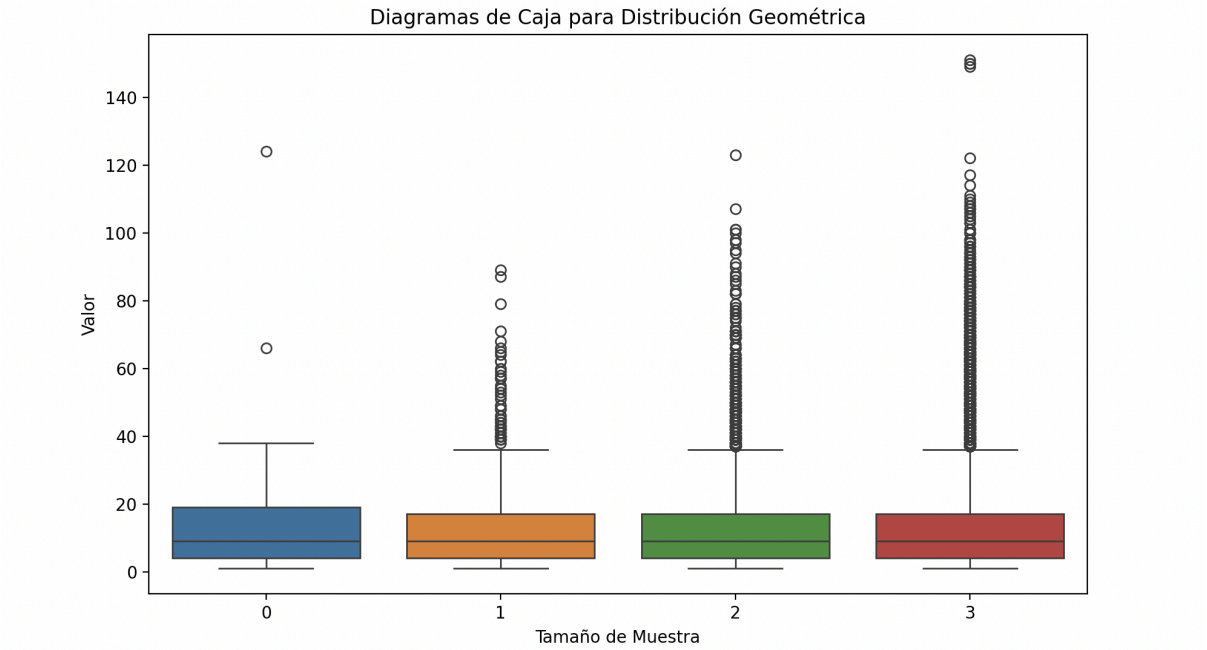
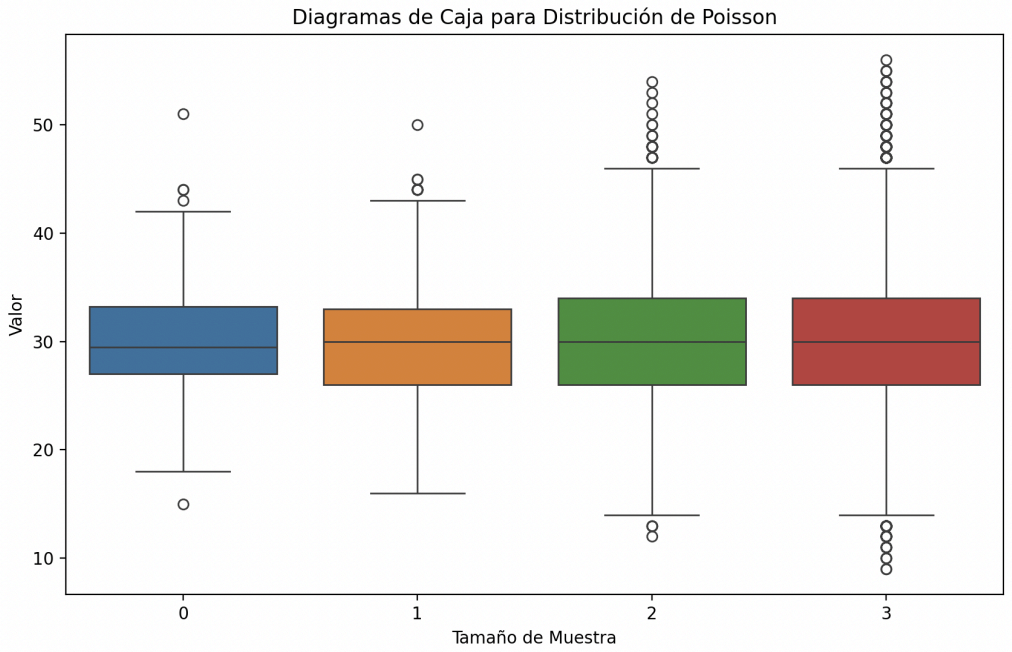


Diagrama de distribución de poisson



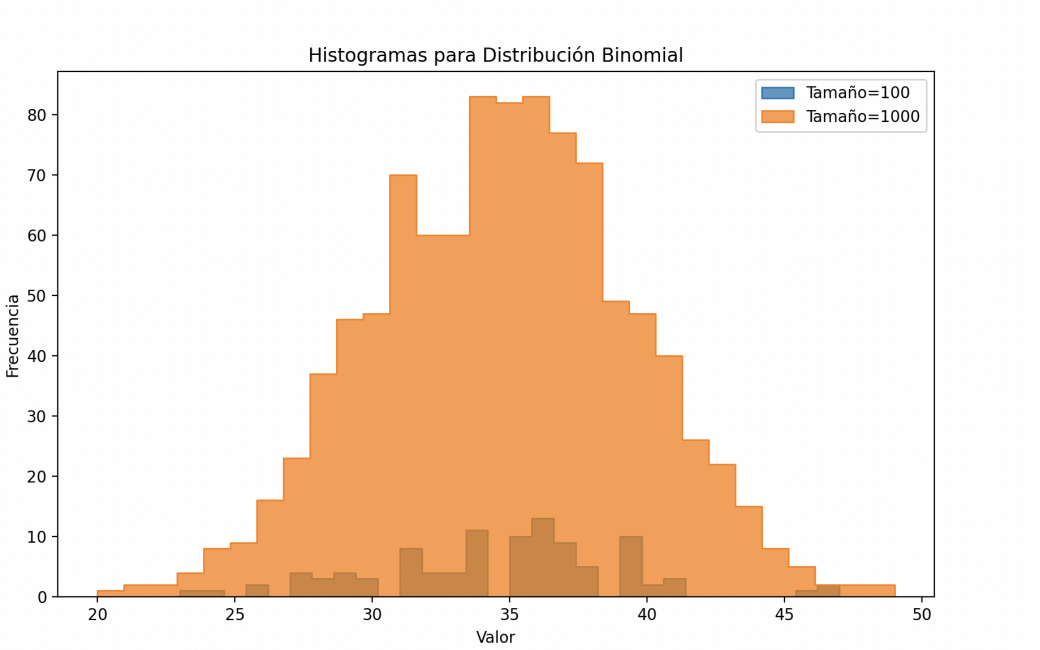
Estas imágenes nos muestran los valores atípicos en cada diagrama

Para los diagramas del histograma tuvimos que hacer algunos ajustes, ya que la muestra de 100.000 no dejaba visualizar todos los resultados correctamente ya que es una muestra muy grande comprada a los otros valores.

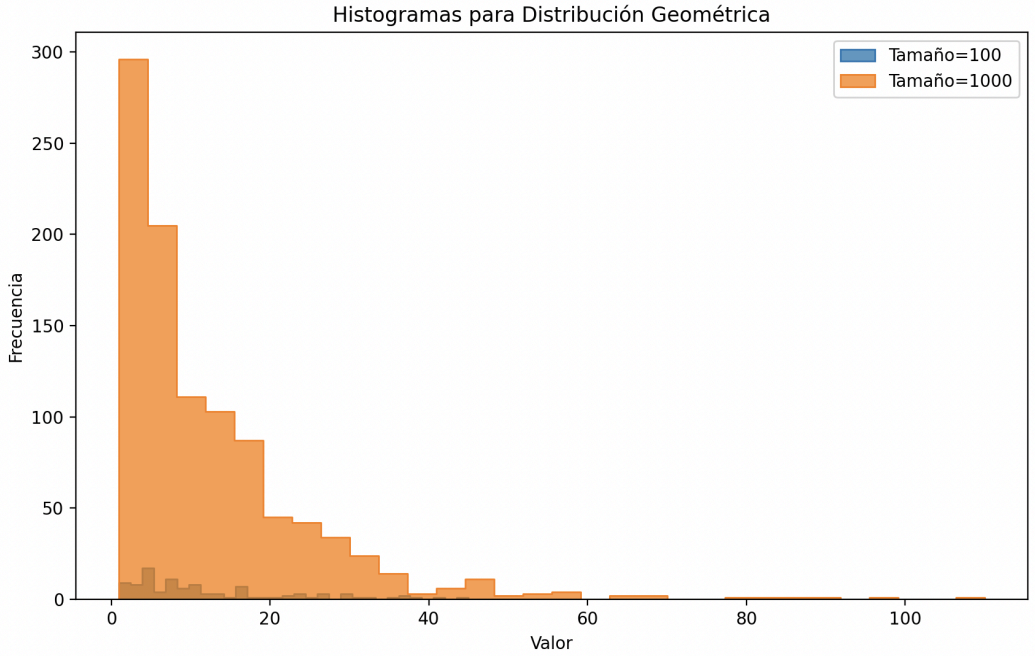
Para mostrar los histogramas, dividimos las muestras en dos partes:

Parte 1 - La muestras son 100 y 1000

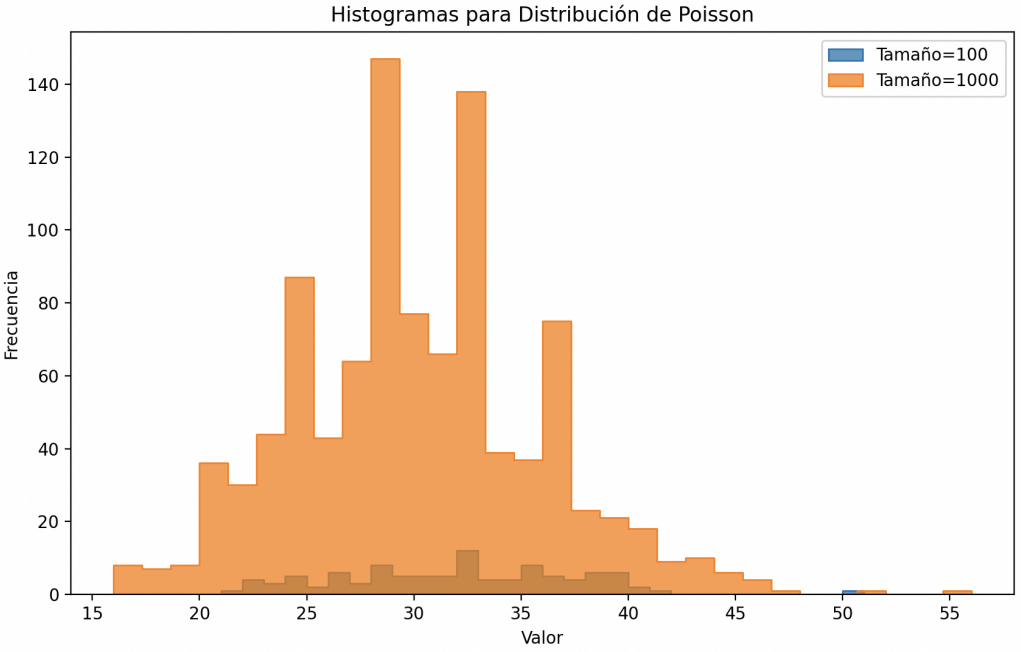
Histograma para la distribución binomial



Histograma para distribución geométrica

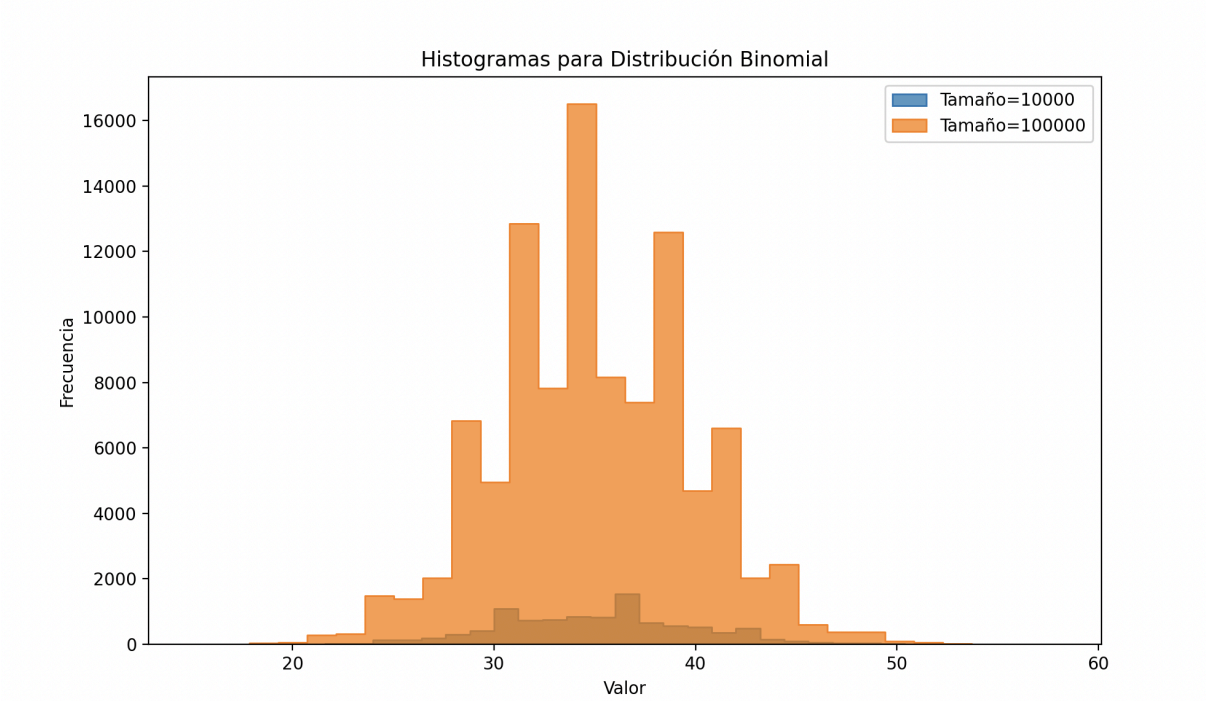


Histograma para distribución de poisson

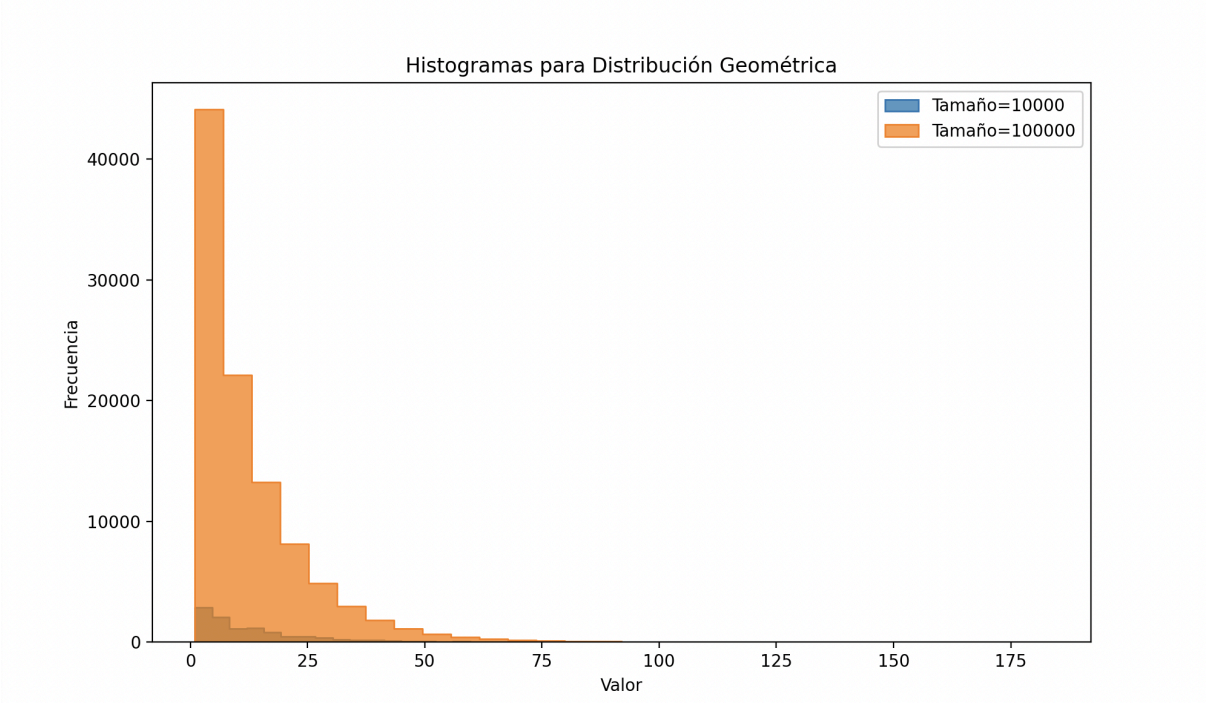


Parte 2 - Muestras de 10.000 y 100.000

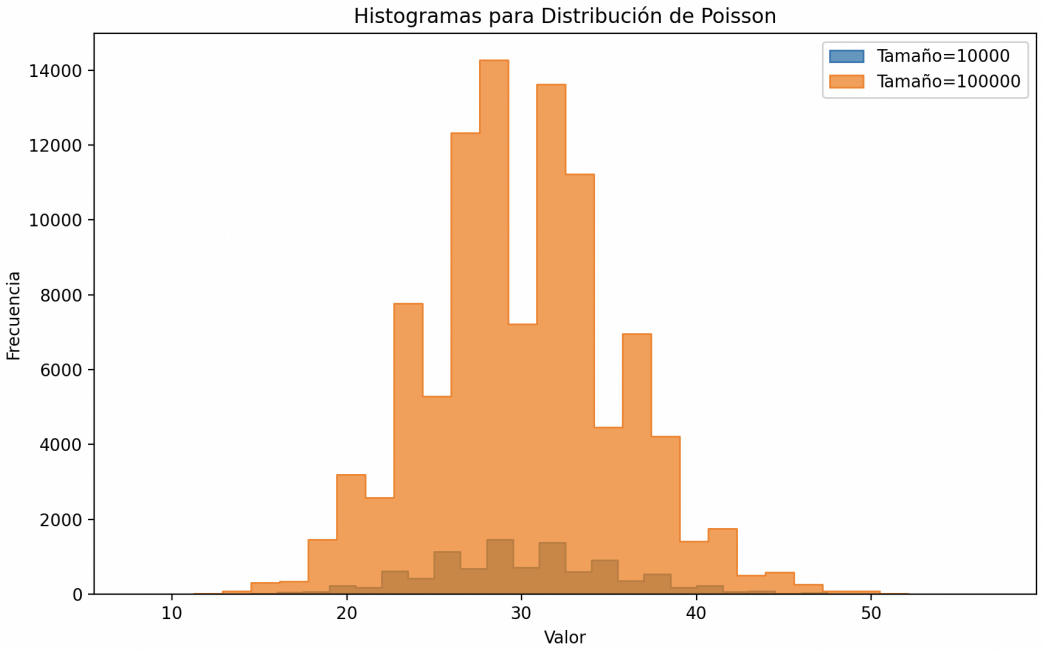
Histograma para la distribución binomial



Histograma para la distribución geométrica



Histograma para la distribución de poisson



Calculo de media, moda, mediana y varianza con sus respectivas varianzas teóricas para la distribución binomial:

```
BINOMIAL
Muestra de tamaño 100:
Mediana: 36.0 (Teórico: 35.0)
Moda: 36 (Teórico: 35)
Media: 35.78 (Teórico: 35.0)
Varianza: 20.2116 (Teórico: 22.75)
Muestra de tamaño 1000:
Mediana: 35.0 (Teórico: 35.0)
Moda: 34 (Teórico: 35)
Media: 34.909 (Teórico: 35.0)
Varianza: 22.618719000000002 (Teórico: 22.75)
Muestra de tamaño 10000:
Mediana: 35.0 (Teórico: 35.0)
Moda: 33 (Teórico: 35)
Media: 34.9975 (Teórico: 35.0)
Varianza: 23.139893750000006 (Teórico: 22.75)
Muestra de tamaño 100000:
Mediana: 35.0 (Teórico: 35.0)
Moda: 35 (Teórico: 35)
Media: 34.9804 (Teórico: 35.0)
Varianza: 22.947395840000002 (Teórico: 22.75)
```

Cálculos para la distribución geométrica:

```
GEOMETRICA
Muestra de tamaño 100:
Mediana: 8.0 (Teórico: 12.0)
Moda: 1 (Teórico: 3)
Media: 11.03 (Teórico: 12.5)
Varianza: 113.76909999999998 (Teórico: 143.75)
Muestra de tamaño 1000:
Mediana: 9.0 (Teórico: 9.0)
Moda: 1 (Teórico: 4)
Media: 12.453 (Teórico: 12.5)
Varianza: 127.76579099999998 (Teórico: 143.75)
Muestra de tamaño 10000:
Mediana: 9.0 (Teórico: 9.0)
Moda: 1 (Teórico: 1)
Media: 12.3982 (Teórico: 12.5)
Varianza: 140.48403675999998 (Teórico: 143.75)
Muestra de tamaño 100000:
Mediana: 9.0 (Teórico: 9.0)
Moda: 1 (Teórico: 1)
Media: 12.50949 (Teórico: 12.5)
Varianza: 144.7963699399 (Teórico: 143.75)
```

Cálculos para la distribución de poisson:

```
POISSON
Muestra de tamaño 100:
Mediana: 30.0 (Teórico: 31.0)
Moda: 32 (Teórico: 26)
Media: 29.86 (Teórico: 30)
Varianza: 30.120400000000004 (Teórico: 30)
Muestra de tamaño 1000:
Mediana: 30.0 (Teórico: 30.0)
Moda: 32 (Teórico: 29)
Media: 30.026 (Teórico: 30)
Varianza: 29.895324000000002 (Teórico: 30)
Muestra de tamaño 10000:
Mediana: 30.0 (Teórico: 30.0)
Moda: 29 (Teórico: 32)
Media: 30.0444 (Teórico: 30)
Varianza: 29.63482864 (Teórico: 30)
Muestra de tamaño 100000:
Mediana: 30.0 (Teórico: 30.0)
Moda: 30 (Teórico: 30)
Media: 30.00267 (Teórico: 30)
Varianza: 30.057342871099998 (Teórico: 30)
```

Conclusiones

1- Simulamos las variables aleatorias discretas en Python para distintos valores.

2- Obtuvimos correctamente las muestras con valores de 10^2, 10^3, 10^4, 10^5

3- Para cada distribución logramos hallar las medidas de tendencia central usando las diferentes librerías de Python.

4- La media y varianza empírica tiende a estabilizarse cerca de la esperanza y varianza teórica con muestras más grandes. A medida que el tamaño de las muestras aumenta, las medias y varianzas empíricas se aproximan cada vez más a los valores teóricos.

5- Representamos gráficamente las muestras, los diagramas de cajas y histogramas proporcionan una forma visual efectiva para evaluar la distribución de los datos y detectar datos atípicos. En este caso particular, se observó un aumento en el número de valores atípicos (representados como puntos negros) en los diagramas de caja a medida que aumentaba el tamaño de las muestras.

6- Al generar las muestras aleatorias para tamaños grandes para cada distribución, observamos que las medias empíricas se acercan progresivamente a las esperanzas teóricas de las respectivas distribuciones a medida que aumenta el tamaño de la muestra. Esto nos confirma que se cumple la ley de los grandes números para valores suficientemente grandes de repeticiones.

Pasos a futuro

1- Añadir la capacidad de generar muestras y calcular estadísticas para más distribuciones.

2- Incluir análisis más avanzados, como pruebas de hipótesis y análisis de varianza, para comparar las muestras generadas con las distribuciones teóricas.

3- Desarrollar una interfaz gráfica de usuario (GUI) usando bibliotecas como Tkinter o PyQt para facilitar la interacción con el programa.

4- Integrar el programa con bases de datos para almacenar y recuperar muestras y resultados de análisis.

Bibliografía

- https://riunet.upv.es/bitstream/handle/10251/7942/Distribuciones%20de%20Probabilidad%20para%20V%20A%20discretas.pdf;jsessionid=8A0570736ED675229B237109EA85A80E?sequence=3
- https://www.dm.uba.ar/materias/estadistica_Q/2010/2/C04Variables%20Aleatorias.pdf
- https://www.probabilidadyestadistica.net/distribucion-geometrica/
- https://economipedia.com/definiciones/media.html
- -https://economipedia.com/definiciones/medidas-de-tendencia-central.html#:~:text=ver%20m%C3%A1s-,Moda,las%20repeticiones%20de%20cada%20valor.
- https://economipedia.com/definiciones/variable-continua.html
- https://www.jmp.com/es_mx/statistics-knowledge-portal/exploratory-data-analysis/box-plot.html
- https://www.probabilidadyestadistica.net/distribucion-binomial/
- https://economipedia.com/definiciones/histograma.html
- https://economipedia.com/definiciones/esperanza-matematica.html
- https://economipedia.com/definiciones/varianza.html
- https://economipedia.com/definiciones/ley-los-grandes-numeros.html
- https://www.probabilidadyestadistica.net/distribucion-de-poisson/
- https://aprendeconalf.es/docencia/python/manual/numpy/
- https://aprendeconalf.es/docencia/python/manual/matplotlib/
- https://python-charts.com/es/seaborn/
- https://lovtechnology.com/que-es-scipy-como-funciona-y-para-que-sirve/
- https://openstax.org/books/introducci%C3%B3n-estad%C3%ADstica-empresarial/pages/4-4-distribucion-de-poisson
- https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.binom.html
- https://matplotlib.org/stable/tutorials/pyplot.html