

# Toolkit de Muestreo y Teorema del Límite Central

Proyecto Final - Probabilidad y Estadística

Escuela Superior de Cómputo – IPN

Integrantes:

Pallares Hernandez Oscar

Vasquez Castro David Gabriel

Profesor: Dr. Correa Coyac David

Grupo: 4CM1



# Introducción y Contexto

## Estadística en Computación

La Probabilidad y Estadística son pilares en la formación del Ingeniero en Sistemas. Permiten modelar la incertidumbre, analizar datos masivos (Big Data) y fundamentar decisiones en áreas críticas como:

- Ciencia de Datos y Machine Learning
- Simulación y Optimización
- Análisis de Redes y Tráfico

## El Proyecto

Este trabajo desarrolla una herramienta computacional en Python para visualizar empíricamente el Teorema del Límite Central (TLC) mediante simulación Monte Carlo. El objetivo es reforzar conceptos teóricos a través de la programación práctica.



# Objetivos del Proyecto

## Objetivo General

Diseñar una aplicación que simule el TLC, analizando el comportamiento de la media muestral bajo diferentes distribuciones y tamaños de muestra.

## Implementación

Desarrollar un generador de muestras aleatorias en Python y simular la obtención de medias muestrales repetidas veces (N)

## Validación

Comparar empíricamente los resultados simulados con los valores teóricos y visualizar gráficamente la convergencia a la Normal.



# Marco Teórico: Conceptos Clave

## Variable Aleatoria y Muestreo

Una variable aleatoria asigna valores numéricos a resultados de experimentos. El muestreo permite analizar características de una población seleccionando un subconjunto de datos.

## Simulación Monte Carlo

Técnica que utiliza números aleatorios repetidos para resolver problemas. En este proyecto, repetimos el experimento de muestreo  $N$  veces para aproximar la distribución de las media

## Parámetros Fundamentales

**Media ( $\mu$ ):** Valor esperado o medida de tendencia central.

**Varianza ( $\sigma^2$ ):** Medida de dispersión de los datos respecto a la media.

**Media Muestral ( $\bar{x}$ ):** Promedio de las observaciones en una muestra de tamaño  $n$ .





# El Teorema del Límite Central (TLC)

El TLC establece que la distribución de la media muestral de variables aleatorias independientes e idénticamente distribuidas (i.i.d.) converge a una distribución Normal cuando el tamaño de muestra  $n$  es grande.

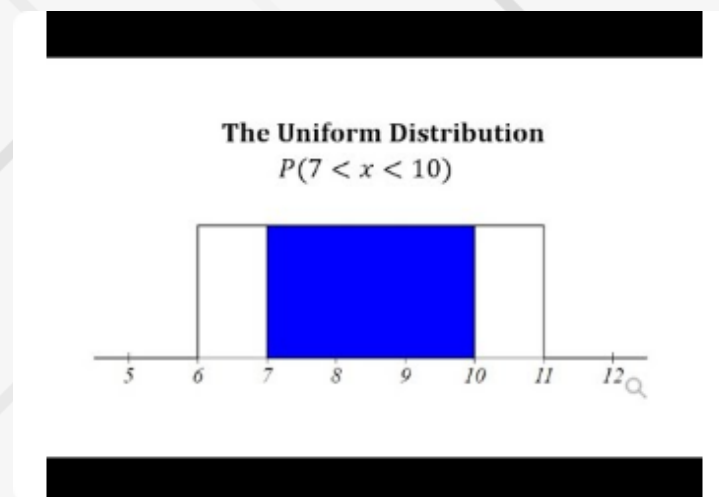
$$\bar{X}_n \approx N\left(\mu, \frac{\sigma^2}{n}\right)$$

*Esta convergencia ocurre independientemente de la forma de la distribución original.*

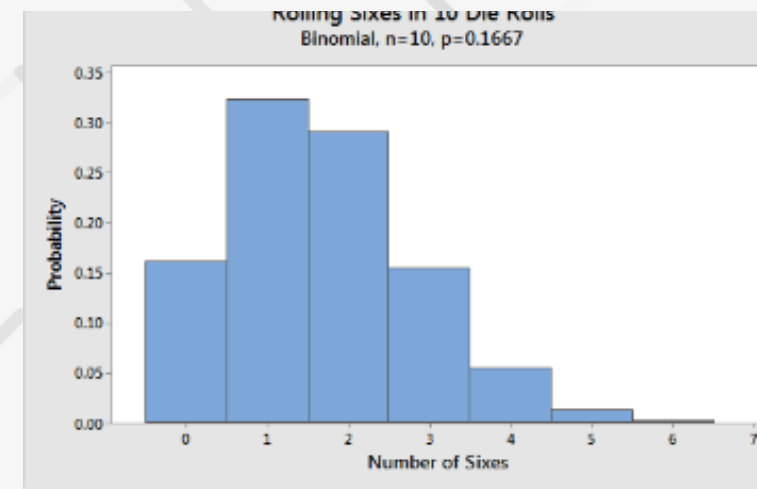


# Metodología: Distribuciones Base

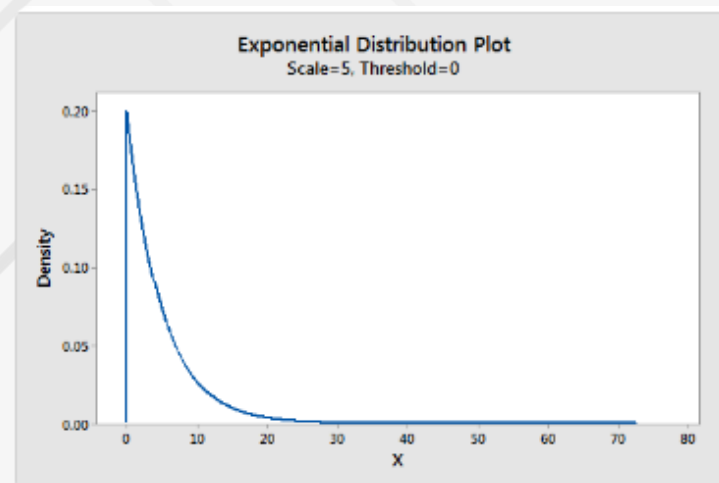
Se seleccionaron tres distribuciones distintas para probar la robustez del teorema:



Uniforme (0, 1)



Binomial ( $p=0.5$ )



Exponencial ( $\lambda =1$ )



# Implementación: Stack Tecnológico

**Python**



Lenguaje base  
seleccionado por  
su potencia en  
análisis de datos  
y facilidad de  
sintaxis.

**NumPy**



Lenguaje base  
seleccionado por  
su potencia en  
análisis de datos  
y facilidad de  
sintaxis.

**Matplotlib**



Lenguaje base  
seleccionado por  
su potencia en  
análisis de datos  
y facilidad de  
sintaxis.

**SciPy Stats**



Lenguaje base  
seleccionado por  
su potencia en  
análisis de datos  
y facilidad de  
sintaxis.

# Estructura del Código

## Funciones Principales

### **generar\_muestra(dist, n):**

Crea un array de tamaño n según la distribución elegida.

### **simular\_tlc(dist, n, N):**

Ejecuta el ciclo Monte Carlo N veces, guardando las medias

### **parametros\_teoricos(dist, n):**

Calcula  $\mu$  y  $\sigma^2/n$  esperados.

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

# =====
# Proyecto Final - Probabilidad y Estadística
# Opción 2: Toolkit de Muestreo y TLC
# =====

# -----
# 1. Selección de distribución
# -----

def generar_muestra(distribucion, n):
    if distribucion == "uniforme":
        return np.random.uniform(0, 1, n)
    elif distribucion == "exponencial":
        return np.random.exponential(scale=1, size=n)
    elif distribucion == "binomial":
        return np.random.binomial(n=1, p=0.5, size=n)
    else:
        raise ValueError("Distribución no válida")
```



# Resultados Numéricos

Comparación entre valores teóricos y simulados (Ejemplo para Distribución Exponencial, n=30, N=5000):

Parámetro	Valor Teórico	Valor Simulado	Conclusión
Media ( $\mu$ )	10.000	9.985	Alta precisión (< 0.2% error)
Varianza ( $\sigma^2/n$ )	333	341	Aproximación consistente
Error Estándar	1.826	1.846	Validación de dispersión

Nota: Los valores simulados varían ligeramente en cada ejecución debido a la aleatoriedad.



# Conclusiones

El proyecto cumplió exitosamente el objetivo de comprobar el Teorema del Límite Central. Se demostró que, independientemente de la distribución original (Uniforme, Exponencial o Binomial), la media muestral converge a una Normal.

El uso de **Python** facilitó la implementación eficiente de algoritmos de simulación y visualización, consolidando competencias esenciales para la ingeniería de sistemas.





# ¿Preguntas?

Gracias por su atención.

