TP 1.1 - SIMULACIÓN DE UNA RULETA

Francisca Gramaglia
Leg. 51424
Universidad Tecnológica Nacional - FRRO
Zeballos 1341, S2000, Argentina
franciscagramaglia714@gmail.com

Mateo Simón Arach
Leg. 51394
Universidad Tecnológica Nacional - FRRO
Zeballos 1341, S2000, Argentina
matearach@gmail.com

Milton Rubén Borsato Gimenez
Leg. 51397
Universidad Tecnológica Nacional - FRRO
Zeballos 1341, S2000, Argentina
borsatomilton@gmail.com

José Ignacio Dayer
Leg. 51508
Universidad Tecnológica Nacional - FRRO
Zeballos 1341, S2000, Argentina
nachodayer@gmail.com

Valentin David Marchese
Leg. 51745
Universidad Tecnológica Nacional - FRRO
Zeballos 1341, S2000, Argentina
marchese52002@gmail.com

Joaquin Garcia Forestello
Leg. 51462
Universidad Tecnológica Nacional - FRRO
Zeballos 1341, S2000, Argentina
joaquingarciaforestello@gmail.com

April 23, 2025

ABSTRACT

El siguiente documento tiene por objetivo detallar el trabajo de investigación que debe realizarse como introducción a la materia simulación. El mismo consiste en desarrollar un modelo simple de ruleta, cuyo funcionamiento será verificado mediante distintos tests rudimentarios.

***K*eywords** Simulación · Trabajo Práctico · Ruleta

1 Introducción

La ruleta[1] es un juego de azar típico de los casinos, cuyo nombre viene del término francés roulette, que significa "ruedita" o "rueda pequeña". Su uso como elemento de juego de azar, aún en configuraciones distintas de la actual, no está documentado hasta bien entrada la Edad Media. Es de suponer que su referencia más antigua es la llamada Rueda de la Fortuna, de la que hay noticias a lo largo de toda la historia, prácticamente en todos los campos del saber humano.

La “magia” del movimiento de las ruedas tuvo que impactar a todas las generaciones. La aparente quietud del centro, el aumento de velocidad conforme nos alejamos de él, la posibilidad de que se detenga en un punto al azar; todo esto tuvo que influir en el desarrollo de distintos juegos que tienen la rueda como base.

Las ruedas, y por extensión las ruletas, siempre han tenido conexión con el mundo mágico y esotérico. Así, una de ellas forma parte del tarot, más precisamente de los que se conocen como arcanos mayores.

Según los indicios, la creación de una ruleta y sus normas de juego, muy similares a las que conocemos hoy en día, se debe a Blaise Pascal, matemático francés, quien ideó una ruleta con treinta y seis números (sin el cero), en la que se halla un extremado equilibrio en la posición en que está colocado cada número. La elección de 36 números da un alcance aún más vinculado a la magia (la suma de los primeros 36 números da el número mágico por excelencia: seiscientos sesenta y seis).

[1]WIKIPEDIA - https://es.wikipedia.org/wiki/Ruleta

Esta ruleta podía usarse como entretenimiento en círculos de amistades. Sin embargo, a nivel de empresa que pone los medios y el personal para el entretenimiento de sus clientes, no era rentable, ya que estadísticamente todo lo que se apostaba se repartía en premios (probabilidad de 1/36 de acertar el número y ganar 36 veces lo apostado).

En 1842, los hermanos Blanc modificaron la ruleta añadiéndole un nuevo número, el 0, y la introdujeron inicialmente en el Casino de Montecarlo. Ésta es la ruleta que se conoce hoy en día, con una probabilidad de acertar de 1/37 y ganar 36 veces lo apostado, consiguiendo un margen para la casa del 2,7% (1/37).

Más adelante, en algunas ruletas (sobre todo las que se usan en países anglosajones) se añadió un nuevo número (el doble cero), con lo cual el beneficio para el casino resultó ser doble (2/38 o 5,26%)

2 Descripción del trabajo de investigación

El trabajo a investigar consiste en construir un programa en lenguaje Python 3.x que simule el funcionamiento del plato de una ruleta. Para esto se debe tener en cuenta lo siguientes temas:

- Generación de valores aleatorios enteros.
- Uso de listas para el almacenamiento de datos.
- Uso de la estructura de control **FOR** para iterar las listas.
- Empleo de funciones estadísticas.
- Gráficas de los resultados mediante el paquete Matplotlib.
- Ingreso por consola de parámetros para la simulación (cantidad de tiradas, corridas y número elegido, Ejemplo **python -c XXX -n YYY -e ZZ**).

3 Código desarrollado en Python

```python
import random
import sys
import statistics as stats
import matplotlib.pyplot as plt
import argparse

#-------------- Definicion de variables --------------
ruleta = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36]
promedio_esperado = stats.mean(ruleta)
varianza_esperada = stats.variance(ruleta)
desvio_esperado = stats.stdev(ruleta)
frecuencia_esperada = 1/37
valores_por_corrida = []
promedio_por_corrida = []
desvio_por_corrida = []
varianza_por_corrida = []
freq_relativa_por_corrida = []
aciertos_por_corrida = 0
MAX_TIRADAS = 10000
MAX_CORRIDAS = 20

#-------------- Definicion de funciones --------------
def graficar_corrida(freq_relativa_por_corrida, promedio_por_corrida, varianza_por_corrida, desvio_por_corrida):
    fig, axs = plt.subplots(nrows=2, ncols=2)
    fig.suptitle('Datos corrida ' + str(i + 1))

    x_vals = range(1, num_tiradas + 1)

    axs[0, 0].plot(x_vals, freq_relativa_por_corrida)
```

```python
    axs[0, 0].hlines(frecuencia_esperada, 1, num_tiradas, colors='r')
    axs[0, 0].set_xlabel('Número de tirada')
    axs[0, 0].set_ylabel('Frecuencia relativa')
    axs[0, 0].legend(['Frec. relativa del número ' + str(num_elegido), 'Frecuencia esperada'])

    axs[0,1].plot(x_vals, promedio_por_corrida)
    axs[0,1].hlines(promedio_esperado, 1, num_tiradas, colors='r')
    axs[0,1].set_xlabel('Número de tirada')
    axs[0,1].set_ylabel('Promedio')
    axs[0, 1].legend(['Progreso de la media' + str(num_elegido), 'Promedio esperado'])

    axs[1,0].plot(x_vals[1:], varianza_por_corrida)
    axs[1,0].hlines(varianza_esperada, 2, num_tiradas, colors='r')
    axs[1,0].set_xlabel('Número de tirada')
    axs[1,0].set_ylabel('Varianza')
    axs[1, 0].legend(['Progreso de la varianza' + str(num_elegido), 'Varianza esperada'])

    axs[1,1].plot(x_vals[1:], desvio_por_corrida)
    axs[1,1].hlines(desvio_esperado, 2, num_tiradas, colors='r')
    axs[1,1].set_xlabel('Número de tirada')
    axs[1,1].set_ylabel('Desvío')
    axs[1, 1].legend(['Progreso del desvio' + str(num_elegido), 'Desvio esperado'])

    plt.tight_layout()
    plt.show()

#-------------- Definicion de argumentos --------------
parser = argparse.ArgumentParser(description="Simulación de ruleta")

parser.add_argument("-c", type=int, required=True, help="Cantidad de tiradas (entero positivo)")
parser.add_argument("-n", type=int, required=True, help="Cantidad de corridas (entero positivo)")
parser.add_argument("-e", type=int, required=True, help="Número elegido (entre 0 y 36)")

#-------------- Parseo de argumentos --------------
args = parser.parse_args()

#-------------- Validaciones después del parseo --------------
if (args.c <= 0):
    print("Error: la cantidad de tiradas (-c) debe ser un entero positivo.")
    sys.exit(1)

if (args.n <= 0):
    print("Error: la cantidad de corridas (-n) debe ser un entero positivo.")
    sys.exit(1)

if args.c > MAX_TIRADAS:
    print(f"Error: la cantidad de tiradas no debe superar {MAX_TIRADAS}.")
    sys.exit(1)

if args.n > MAX_CORRIDAS:
    print(f"Error: la cantidad de corridas no debe superar {MAX_CORRIDAS}.")
    sys.exit(1)

if (args.e < 0 or args.e > 36):
    print("Error: el número elegido (-e) debe estar entre 0 y 36.")
    sys.exit(1)
```

```python

print(f"Tiradas: {args.c}, Corridas: {args.n}, Número elegido: {args.e}")

num_tiradas = args.c
num_corridas = args.n
num_elegido = args.e

#-------------- Inicio de la simulación --------------
for i in range(num_corridas):
    print("Corrida numero", i + 1)
    valores_por_corrida.clear()
    promedio_por_corrida.clear()
    desvio_por_corrida.clear()
    varianza_por_corrida.clear()
    freq_relativa_por_corrida.clear()
    aciertos_por_corrida = 0

    for j in range(num_tiradas):
        valor = random.randint(0, 36)
        valores_por_corrida.append(valor)
        promedio_por_corrida.append(stats.mean(valores_por_corrida))
        if j > 0:
            varianza_por_corrida.append(stats.variance(valores_por_corrida))
            desvio_por_corrida.append(stats.stdev(valores_por_corrida))
        if (valor == num_elegido):
            print("El numero elegido", num_elegido,  "tuvo acierto en la tirada numero", j + 1)
            aciertos_por_corrida += 1
        freq_relativa_por_corrida.append(aciertos_por_corrida/(j+1))

    print("Valores: ",valores_por_corrida)
    print("Cantidad de aciertos: ", aciertos_por_corrida)
    graficar_corrida(freq_relativa_por_corrida, promedio_por_corrida, varianza_por_corrida, desvio_por_corrida)
    print("---------------------------------------------")
```

4 Descripción del Experimento

Se ejecuta un programa en Python que:

1. Realiza un número configurable de tiradas por corrida.
2. Repite este proceso por una cantidad definida de corridas.
3. En cada tirada, se registra si el número elegido apareció.
4. Se calculan y grafican los valores de interés para cada corrida.

Los valores esperados teóricos para una ruleta uniforme de 37 números son:

- **Frecuencia esperada de un número:** $\frac{1}{37} \approx 0.0270$
- **Promedio esperado:** $\mu = \frac{0+36}{2} = 18$
- **Varianza esperada:** $\sigma^2 = \frac{(n^2-1)}{12} = \frac{36^2-1}{12} \approx 114.75$
- **Desvío estándar:** $\sigma = \sqrt{\sigma^2} \approx 10.71$

5 Resultados

A continuación se presentan los gráficos generados en cada corrida. Cada uno muestra el comportamiento de las métricas estadísticas a lo largo de las tiradas:

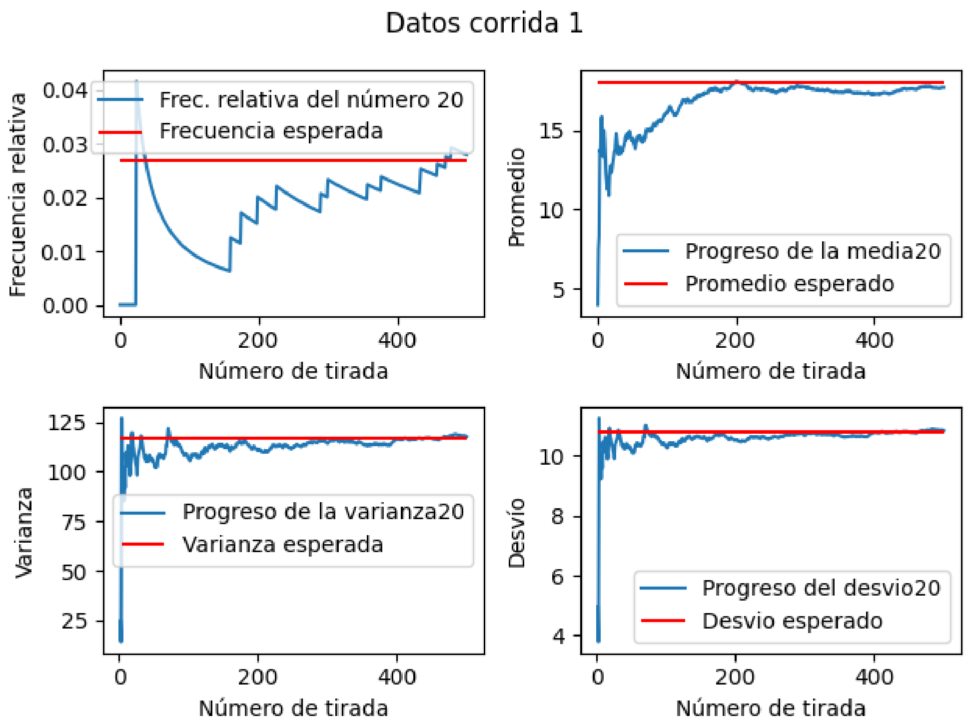

Figure 1: Gráficas corrida numero 1

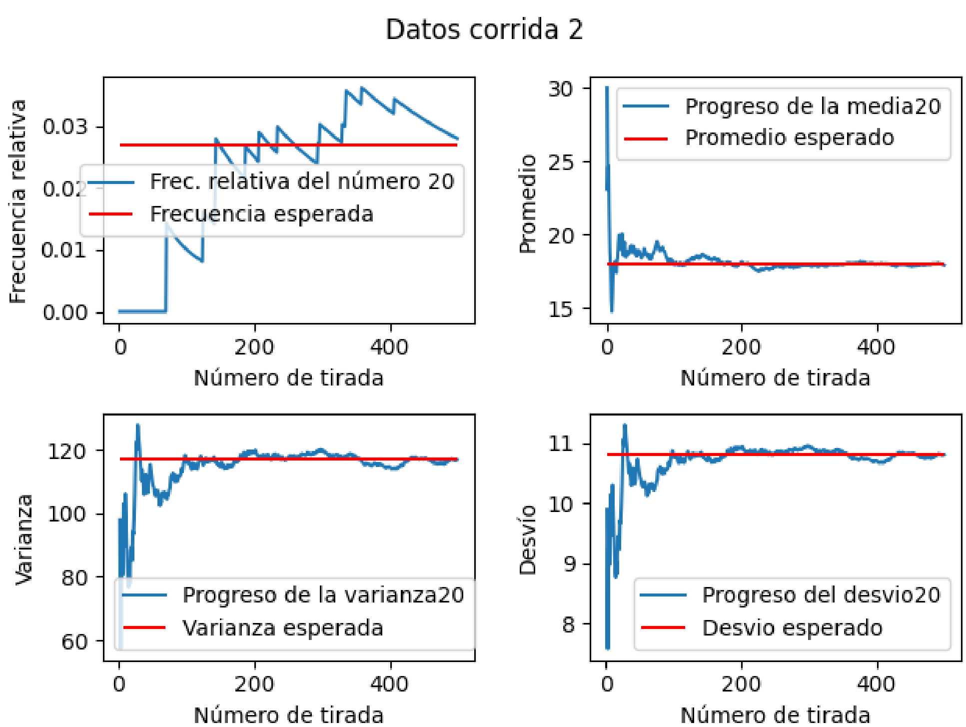

Figure 2: Gráficas corrida numero 2

6 Conclusión

Los resultados muestran una clara convergencia de las métricas estadísticas hacia sus valores teóricos a medida que aumenta el número de tiradas, confirmando el comportamiento esperado de una ruleta justa bajo el Teorema Central del Límite. A saber:

- La frecuencia relativa del número elegido tiende a estabilizarse en ≈ 0.027.
- El promedio se estabiliza alrededor de 18.
- La varianza y el desvío estándar convergen a los valores teóricos con pequeñas fluctuaciones al principio.

Esto valida el modelo de ruleta uniforme y muestra que, estadísticamente, el sistema simulado se comporta como una ruleta justa y aleatoria.

7 Fórmulas empleadas

La frecuencia relativa es la proporción o porcentaje que representa un valor o resultado particular dentro de un conjunto de datos.

$$\text{frecuencia relativa} = \frac{f}{n} \tag{1}$$

El desvío estándar indica qué tan dispersos están los datos con respecto a la media. Mientras mayor sea la desviación estándar, mayor será la dispersión de los datos.

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{n}} \tag{2}$$

El promedio es una medida de tendencia central que representa el valor esperado de una variable aleatoria

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \tag{3}$$

donde x_i es la i-ésima observación.

La varianza es una medida de dispersión o dispersión de una variable aleatoria alrededor de su valor esperado.

$$\sigma^2 = \frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{n} \tag{4}$$

8 Enlaces de Interés

Repositorio GitHub `https://github.com/franciscag714/simulacion_utn.git`

References

[1] Material Teórico V.A.D 2023.
`https://drive.google.com/file/d/1NZe9ZwLu6crIObS8Ydp1UYkIuT25CjxD/view`

[2] Teorema central del límite (TCL)
`https://economipedia.com/definiciones/teorema-central-del-limite.html`

[3] Las Matemáticas de la Ruleta
`https://www.math4all.es/las-matematicas-de-la-ruleta/`