

---

# TRABAJO PRACTICO 1 - SIMULACIÓN

---

**Godoy Brenda**  
Ingeniería en Sistemas  
Universidad Tecnológica Nacional  
Legajo: 47573  
dia.godoybrenda@gmail.com

**Casey Guillermo**  
Ingeniería en sistemas  
Universidad Tecnológica Nacional  
Legajo: 47026  
guillermocasey@hotmail.com

**Strappini Liza**  
Ingeniería en Sistemas  
Universidad Tecnológica Nacional  
Legajo: 45770  
lizastrappini99@gmail.com

March 22, 2022

## ABSTRACT

En este trabajo simulamos el funcionamiento de una ruleta típica de casino con las herramientas que nos proporciona la librería gráfica matplotlib de python. Lo que hicimos fue generar una lista de números aleatorios (muestra) y analizamos gráficamente distintas métricas tales como la media, la varianza, la frecuencia relativa del número 8, el desvío y para concluir combinamos los promedios de 4 gráficas.

Utilizamos al menos 30 observaciones y vimos resultados cercanos a las magnitudes esperadas, y a medida que aumenta de tamaño la muestra, los resultados son cada vez mas cercanos a nuestras proyecciones.

## 1 Introduccion

Nuestro trabajo consiste en un juego de azar típico de los casinos, la ruleta. Simulamos el funcionamiento de la misma y analizamos algunas de las métricas estadísticas más comunes, a las cuales tienden los resultados de la ruleta a medida que realizamos las iteraciones.

## 2 Explicación del código

En nuestro código creamos una lista de números aleatorios y una lista para cada valor a graficar.

Usamos un bucle for de 30 iteraciones que son las mínimas necesarias para obtener una distribución aproximadamente normal y luego graficamos los resultados.

Algunas gráficas se acercan más al valor esperado que otras, esto se debe a la cantidad de iteraciones que utilizamos(30), en la de la frecuencia relativa sin embargo, tuvimos que utilizar más iteraciones para que los valores se estabilicen.

Para la primera parte del trabajo, en la que no debemos mostrar gráficas superpuestas tenemos un código, y cuando debamos mostrar varias al mismo tiempo, usamos un bucle FOR de 1 a 8 para generar 8 arreglos de números distintos, todo el trabajo fue hecho en el entorno de ejecución jupyter notebook.

### 2.1 Frecuencia relativa

La fórmula que utilizamos para el cálculo de la frecuencia relativa del numero 8 fue:

$$f_i = \frac{n_i}{N} \quad (1)$$

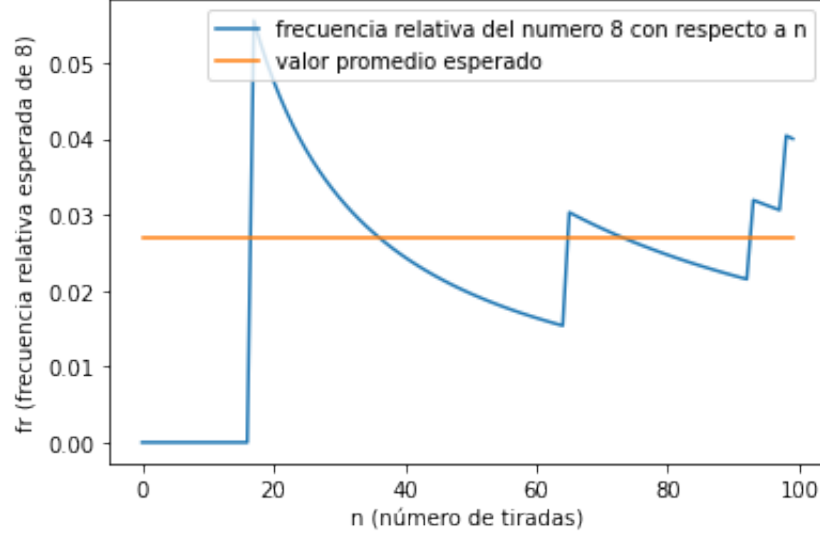


Figure 1: Frecuencia relativa del número 8

## 2.2 Valor promedio

La fórmula que utilizamos para el cálculo de la media aritmética o el promedio de las tiradas fue:

$$\bar{X} = \frac{\sum_{i=1}^n x_i}{n} \quad (2)$$

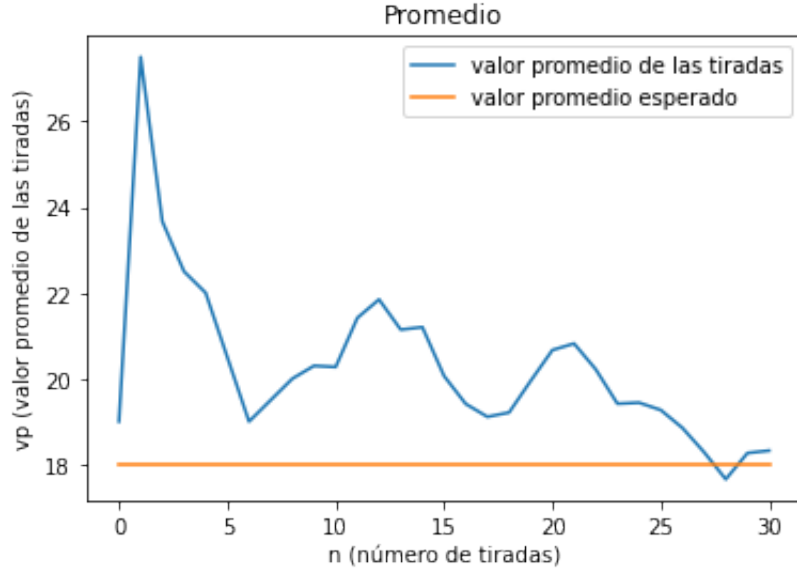


Figure 2: Valor promedio de los valores obtenidos

## 2.3 Desvío poblacional

La fórmula que utilizamos para el cálculo del desvío poblacional de las tiradas fue:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{N}} \quad (3)$$

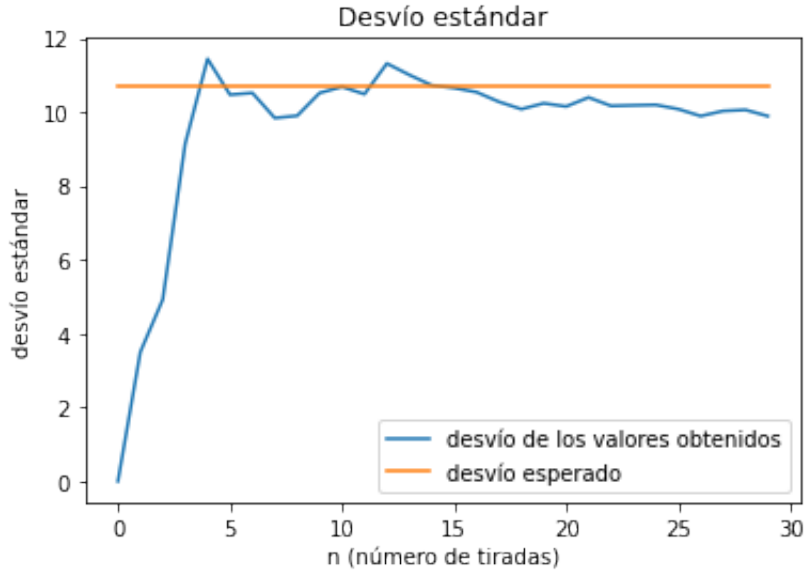


Figure 3: Desvío poblacional de los valores obtenidos

## 2.4 Varianza

La fórmula que utilizamos para el cálculo de la varianza de las tiradas fue:

$$S^2 = \sigma^2 = \frac{\sum_{i=1}^n (x - \bar{X})^2 \cdot f_i}{N} \quad (4)$$

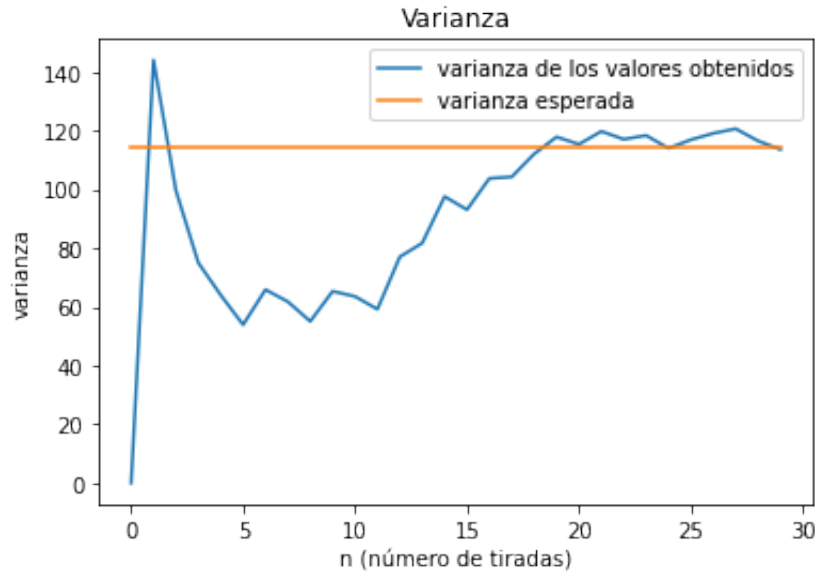


Figure 4: Varianza de los valores obtenidos

## 2.5 Código en python

```
import matplotlib.pyplot as plt
import random
from statistics import mean
import statistics

#final

def Promedio(l):
    prom = mean(l)
    return prom

def frec(arra):
    cantDe8 = arra.count(8)
    fr = cantDe8 / len(arra)
    return fr

# DESVIO
numeros = [] # arreglo de numeros aleatorios
desvio = [] # desvio acumulado
lineaDesvio = [] # desvio esperado
promedios = [] # valor promedio
lineaPromedio = [] # valor promedio esperado
lineaFrecRel = [] # Frecuencia relativa esperada
frecRel8 = [] # Frecuencia relativa
linea = [] # desvio esperado
varianza = [] # varianza
lineaVar = [] # varianza esperada

for x in range(0,31):
    num = (random.randint(0,36))
    numeros.append(num)
    #DESVIO
    desvio.append(statistics.pstdev(numeros))
```

```

lineaDesvio.append(10.677)
    #PROMEDIO
promedios.append(Promedio(numeros))
lineaPromedio.append(18.0)
    #FRECUENCIA RELATIVA
lineaFrecRel.append(1/37)
frecRel8.append(frec(numeros))
    #VARIANZA
varianza.append(statistics.pvariance(numeros))
lineaVar.append(114)

fig, axs = plt.subplots(2, 2)
axs[0, 0].plot(promedios, label="valor promedio de las tiradas")
axs[0, 0].set_title('Axis [0, 0]')
axs[0, 0].plot(lineaPromedio, label="valor promedio esperado")
axs[0, 0].set_title('Promedio')
axs[1, 0].plot(desvio, label="desvío de los valores obtenidos")
axs[1, 0].set_title('Axis [1, 0]')
axs[1, 0].plot(lineaDesvio, label="desvío esperado")
axs[1, 0].set_title('Desvio')
axs[1, 1].plot(frecRel8, label="frecuencia relativa del numero 8 con respecto a n")
axs[1, 1].set_title('Axis [1, 1]')
axs[1, 1].plot(lineaFrecRel, label="valor promedio esperado")
axs[1, 1].set_title('Frecuencia relativa')
axs[0, 1].plot(varianza, label="varianza de los valores obtenidos")
axs[0, 1].set_title('Axis [1, 1]')
axs[0, 1].plot(lineaVar, label="varianza esperada")
axs[0, 1].set_title('Varianza')

for ax in axs.flat:
    ax.set(xlabel='n = numero de tiradas', ylabel='valor')

# Hide x labels and tick labels for top plots and y ticks for right plots.
for ax in axs.flat:
    ax.label_outer()

```

### 3 Segunda parte

En esta parte lo que hicimos fue graficar varias listas de números aleatorios generados de la misma manera que en la primer parte, utilizando las métricas planteadas anteriormente para ver como se comportaban entre ellas. La cantidad de funciones que graficamos por gráfico fue de 5 puesto que nos pareció más simple a la hora de visualizarlo.

#### 3.1 Frecuencia relativa de varios conjuntos

Corrimos el experimento varias veces, generando 5 conjuntos de de números aleatorios y graficamos la frecuencia relativa del 8 para cada uno de los conjuntos.

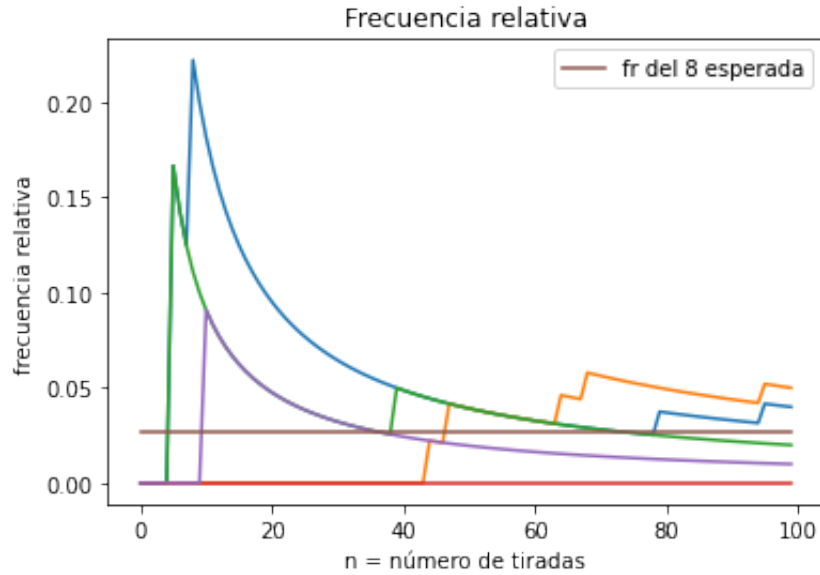


Figure 5: Frecuencias relativas del número 8

### 3.2 Valor promedio de varios conjuntos

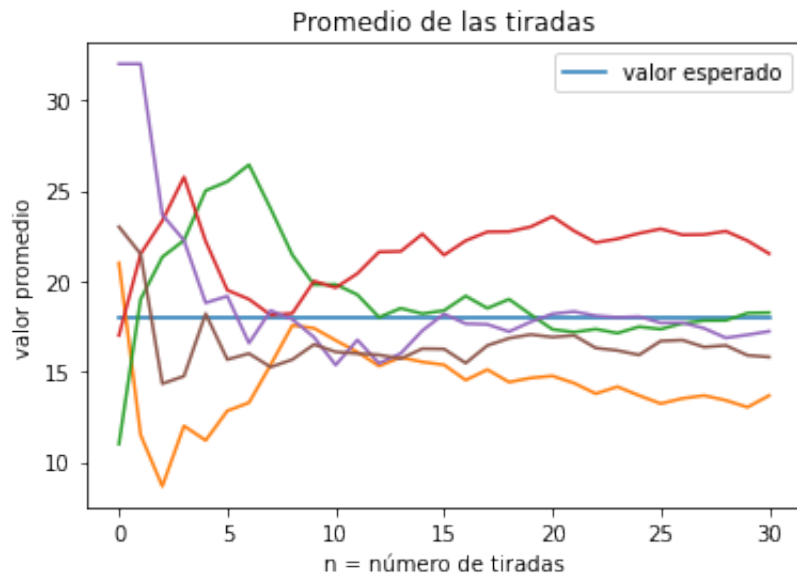


Figure 6: Valores promedios de los valores obtenidos

### 3.3 Valor promedio de los promedios

En esta gráfica se muestra el promedio de 5 conjuntos, siendo cada uno, un valor promedio para una muestra de tiradas de ruleta.

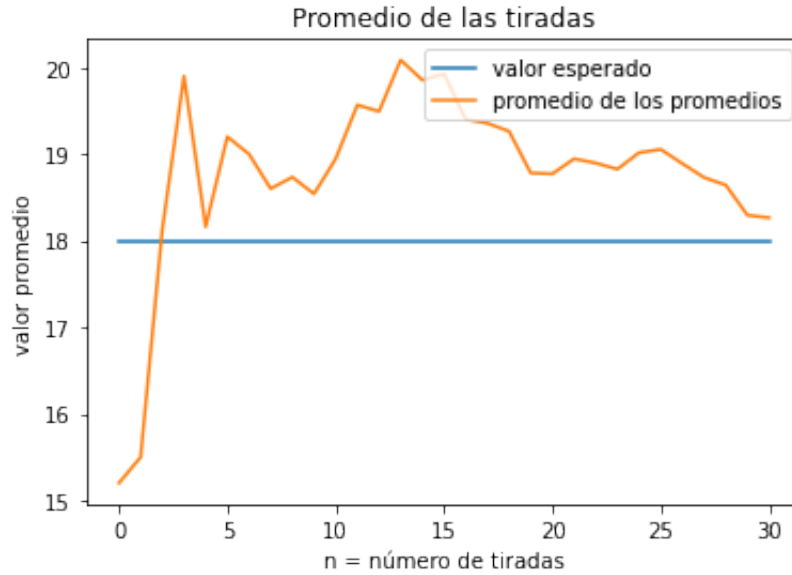


Figure 7: Valor promedio de promedios

### 3.4 Desvíos de varios conjuntos

En esta gráfica se representa el desvío de 5 tiradas de ruletas.

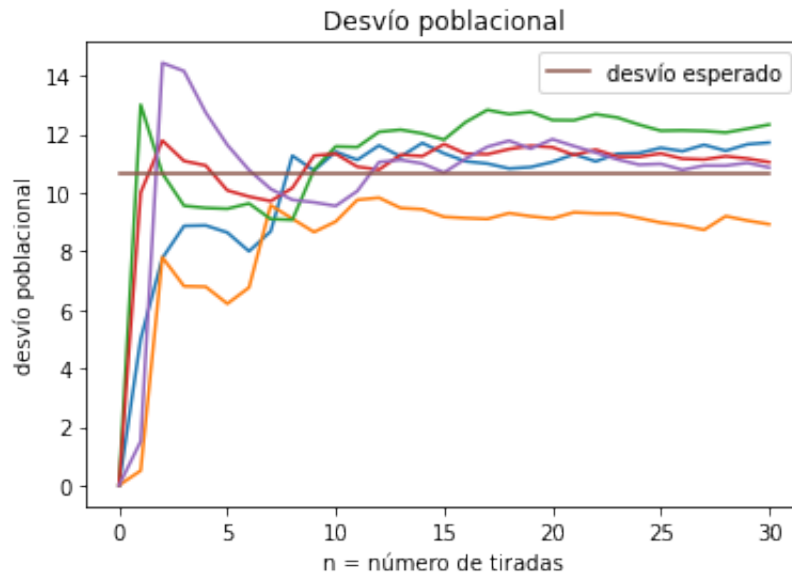


Figure 8: Valor promedio de promedios

## 4 Conclusión

Luego de correr el código varias veces pudimos concluir que la ruleta funciona correctamente, es decir que los valores generados aleatoriamente no siguen ningún patrón, ya que como vemos en las gráficas del apartado segunda parte, notamos como todas las funciones tienden a la misma línea de valor esperado respectivo de cada métrica probabilística.

## References

- [1] <https://www.w3schools.com/python/>
- [2] <https://python-para-impacientes.blogspot.com/2014/08/graficos-en-ipython.html>
- [3] <https://relopezbriega.github.io/blog/2015/06/27/probabilidad-y-estadistica-con-python/>
- [4] <https://www.overleaf.com/learn/latex/Code-Highlighting-with-minted>