
TP 1.2 - ESTUDIO ECONÓMICO-MATEMÁTICO DE APUESTAS EN LA RULETA

Juan I. Torres
Universidad Tecnológica Nacional - FRRO
Zeballos 1341, S2000, Argentina
orkuan@gmail.com

16 de abril, 2024

ABSTRACT

El siguiente documento tiene por objetivo detallar el trabajo de investigación que debe realizarse para profundizar el estudio del comportamiento de la ruleta pero desde el punto de vista del apostador y sus estrategias.

Keywords Simulación · Trabajo práctico · Ruleta · apuestas · estrategias

1 Introducción

Apostar en algún juego lleva a controversias de todos los puntos de vista en cuanto a sus resultados favorables o no, pero desde el nuestro como ingenieros debemos tener una **visión objetiva** de cualquier problema por muy ajeno que nos resulte. Este trabajo tiene como fin el empleo de nuestra primera simulación con el objetivo desmitificar estadísticamente la verdadera probabilidad de obtener ganancias con un medio ideal, como es nuestra ruleta simulada.

Las estrategias que se pueden proponer son varias, pero comenzaremos con una de las más sencillas y fáciles de implementar, dejando al alumno la posibilidad de construir o imitar otras.

2 Descripción del trabajo de investigación

El trabajo consiste en construir una programa en lenguaje Python 3.x que simule el funcionamiento del plato de una ruleta y al mismo tiempo lleve adelante diversos tipos de apuestas, monitorizando el flujo de caja y el período de ocurrencia de que la apuesta produzca o no beneficios. Para esto se debe tener en cuenta lo siguientes temas:

- Beneficios de las apuestas según la selección (color, fila, número único, etc).
- Distintos tipos de estrategias de apuestas en la ruleta.
- Gráficas de los resultados mediante el paquete Matplotlib (u otro similar).

Se pide que se detalle las estrategias empleadas y las fuentes donde las obtuvieron (si no son de elaboración propia). Se proponen analizar 3 estrategias: la martingala, D'Alembert y Fibonacci, además el grupo de investigación sugerir o crear una estrategia a elección distinta de las propuestas. Para cada estrategia se debe tener en cuenta dos supuestos mutuamente excluyentes: **capital acotado** (realidad) y **capital infinito** (ideal). En el caso de capital acotado se debe tener en cuenta las veces que se produce "la banca rota" del jugador.

Se incorporan a la ejecución anterior dos parámetros nuevos: -s -a. Siendo:

- "s" la estrategia utilizada: **m** (martingala), **d** (D'Alembert), **f** (Fibonacci) y **o** (otra estrategia a elección del grupo).
- "a" el tipo de capital: **i** (infinito), **f** (finito).

Por lo tanto la ejecución junto con el TP 1.1: `python -c XXX -n YYY -e ZZ -s -a`

Nota: El -e es solo en caso de usar un solo número. Sino no es necesario.

2.1 Exposición de los resultados y análisis de los mismos

Los resultados se deben graficar y luego concluir su comportamiento simulado y esperado. A modo de ejemplo se dejan los siguientes bocetos de gráficas, siendo estas las que, como mínimo deben de estar en el estudio:

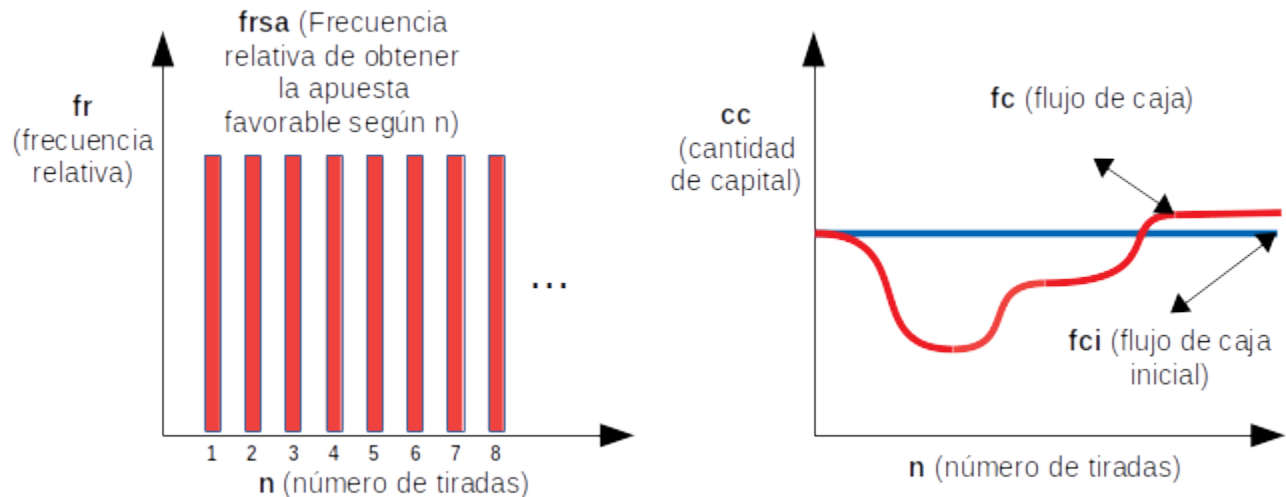


Figura 1: Conjunto de gráficas mínimas solicitadas.

Luego de finalizar lo anterior, se deben de realizar varias corridas del experimento (determinar y justificar la cantidad) y generar nuevas gráficas donde se muestren en forma simultánea sus resultados. En total son como mínimo 3 gráficas en todo el trabajo por cada estrategia (las tres propuestas y una sugerida por el grupo).

2.2 Presentación del trabajo y entrega

Este y los siguientes trabajos se presentan obligatoriamente en formato \LaTeX . Una manera cómoda de trabajar es mediante un IDE el cual puede ser online o local. La ventaja de trabajar online es la posibilidad de que el resto del grupo aporte y corrija directa y simultáneamente.

Una de las plataformas online más conocidas es Overleaf. Por el otro lado, en forma local tenemos, para los que trabajan con Linux distribución Ubuntu el muy conocido Texmaker. En Windows debe instalarse primero el compilador Miktex, y posteriormente puede instalarse Texmaker o TexStudio. VSCode también tiene utilidades para escribir en Latex, pero se deja que los grupos elijan sus herramientas, siempre y cuando respeten el formato proporcionado. Se sugiere buscar e investigar como se escriben los "papers científicos" de alguna fuente confiable para no omitir o agregar secciones incorrectas.

El contenido mínimo a entregar es:

- Código completo en Python 3.x.
- Informe en formato Latex con introducción, gráficas, fórmulas empleadas, conclusiones, referencias (hay un apartado para esto) y cualquier otra información que quieran agregar.

La fecha de entrega es el día **08/04/2020**.

3 Recursos online obligatorios

Volveremos a usar el template \LaTeX de la Cornell University por su sencillez:

<https://es.overleaf.com/latex/templates/style-and-template-for-preprints-arxiv-bio-arxiv/pkzcrhzcxdxmc>

[https://blog.sportium.es/
3-simples-estrategias-para-ganar-en-la-ruleta-que-cualquiera-puede-intentar/](https://blog.sportium.es/3-simples-estrategias-para-ganar-en-la-ruleta-que-cualquiera-puede-intentar/)

4 Recursos online sugeridos

Algunas páginas que han servido a alumnos de años anteriores en Python:

<https://youtu.be/vBgrvVY1jGo?si=m1BjUsM-lGK0IyQm>

<https://python-para-impacientes.blogspot.com/2014/08/graficos-en-ipython.html>

<https://relopezbriega.github.io/blog/2015/06/27/probabilidad-y-estadistica-con-python/>