

Práctica 1

Movimiento Browniano

Edson Edgardo Samaniego Pantoja

Materia: Simulación computacional

Fecha 1 de marzo de 2021

1. Introducción

En esta primera práctica se estudia el movimiento Browniano que refiere a una partícula que cambia su posición al azar de manera uniforme tomando su posición inicial el origen.

2. Objetivo

El objetivo principal de esta práctica es observar los efectos por cada dimensión, para el tiempo de regreso al origen utilizando dimensiones de uno a cinco, se modifica el numero de pasos de la caminata como potencias de dos con exponente de cuatro a nueve y se hacen repeticiones de 30. Los resultados se graficaron con diagrama caja-bigote, aparte se incluye un cuadro indicando el mínimo, promedio y máximo del tiempo de regreso por cada dimensión junto con el porcentaje de caminatas que no regresaron.

3. Simulación

Para la programación se explican ciertas partes importantes debido a que contiene líneas repetitivas y extensas, el código completo puede ser consultado en github [2]. De la mano es apoyada la programación con las practicas de Elisa Schaeffer en su github [3] Se requieren librerías como *random* para que la decisión sea al azar, *matplotlib.pyplot* utilizada para la generación de diagramas caja-bigote, *tabulate* para la obtención y visualización de datos y como extra por fines analíticos *cv2* donde se utiliza la pausa en milisegundos para ver por pasos el programa.

```
from random import random, randint
import matplotlib.pyplot as plt
from tabulate import tabulate
import cv2 as cv2 #librería sólo por las pausas en los ciclos
```

Por medio de ciclos *for* es como se hace el cambio tanto de dimensiones como de caminatas y las 30 replicas para obtener mayor conjunto de datos, dentro de las replicas se hace la decisión al azar tanto de la dimensión como el incremento a decremento unitario. También se extrae la cantidad de regresos a cero, por replicas y por caminatas que se acumulan en una variable para más adelante procesar y sacar el mínimo, máximo, promedio y porcentaje que no volvió nunca al origen. Utilizando paginas de apoyo [1] para la programación y también desde la pagina oficial [4].

```
for A in range(1, 6):
    dim = A
```

```

for pot in range(4, 10):
    saltos = 2**pot
    rep= 30
    resultados = []
    nuevos = []# porque si no la lista se hace con None y no puedo extraer números
    datos=[],[],[],[],[],[]
    pos = [0] * dim

    for replica in range(rep):
        nunca = True
        for paso in range(saltos):
            cual = randint(0, dim -1)
            pos[cual] = pos[cual] + 1 if random()<0.5 else pos[cual] -1
            if all([p== 0 for p in pos]):
                resultados.append(paso)
                nuevos.append(paso)
                nunca = False
                break

```

4. Resultados

En este apartado se observan los diferentes resultados en las caminatas de 16, 32, 64, 128, 256 y 512 pasos, mostrados en gráficos caja-bigote. Se observó una tendencia a no volver a su origen conforme aumentan las dimensiones debido al número al azar, lo que lo vuelve menos probable que coincidan los orígenes. Solo en dimensiones 1 y 2 se logra ver que se mantienen los regresos moderados como se puede ver en la tablas 1, 2 y las figuras 1, 2.

Cuadro 1: Caminatas de 16 pasos

dim	mín	máx	μ	%
1	1	15	3.66	50
2	15	15	15	96.66
3	0	0	0	100
4	0	0	0	100
5	0	0	0	100

Cuadro 2: Caminatas de 32 pasos

dim	mín	máx	μ	%
1	1	9	2.64	43.33
2	1	17	3.72	63.33
3	11	11	11	96.66
4	0	0	0	100
5	0	0	0	100

Cuadro 3: Caminatas de 64 pasos

dim	mín	máx	μ	%
1	1	33	7.21	6.66
2	3	3	3	96.66
3	0	0	0	100
4	0	0	0	100
5	0	0	0	100

Cuadro 4: Caminatas de 128 pasos

dim	mín	máx	μ	%
1	1	33	4.62	23.33
2	0	0	0	100
3	5	5	5	96.66
4	0	0	0	100
5	0	0	0	100

Cuadro 5: Caminatas de 256 pasos

dim	mín	máx	μ	%
1	1	213	15	0
2	3	149	76	93.33
3	0	0	0	100
4	0	0	0	100
5	0	0	0	100

Cuadro 6: Caminatas de 512 pasos

dim	mín	máx	μ	%
1	1	21	5.61	13.33
2	0	0	0	100
3	0	0	0	100
4	0	0	0	100
5	0	0	0	100

A continuación se muestran los gráficos caja-bigote para ver la tendencia de los resultados de las tablas anteriores siendo ya más fácil de comprender como la dimensión 1 fue la más estable en regresos a origen ya que al ser un solo plano en el que puede cambiar de posición lo hace más probable y como conforme se aumentan las dimensiones ésta probabilidad de regresar se hace más lejana tanto que hay muchos casos que nunca vuelve.

Figura 1: caja-bigote 16 pasos

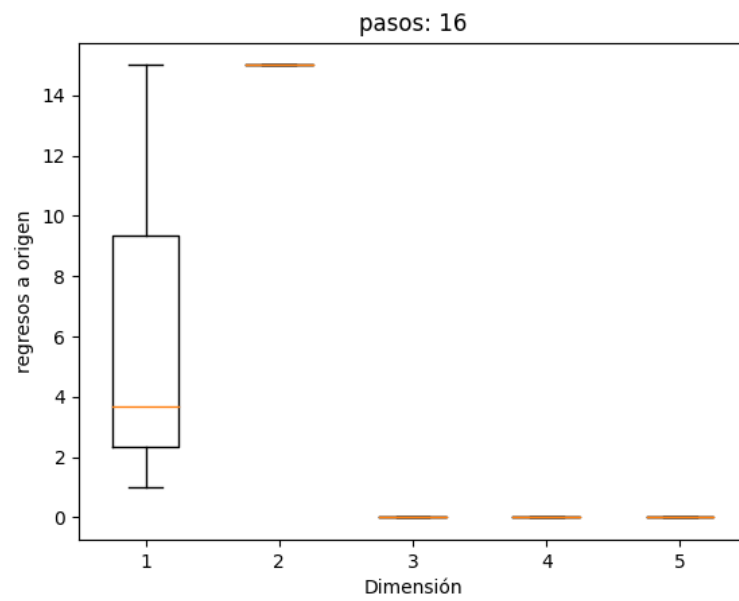


Figura 2: caja-bigote 32 pasos

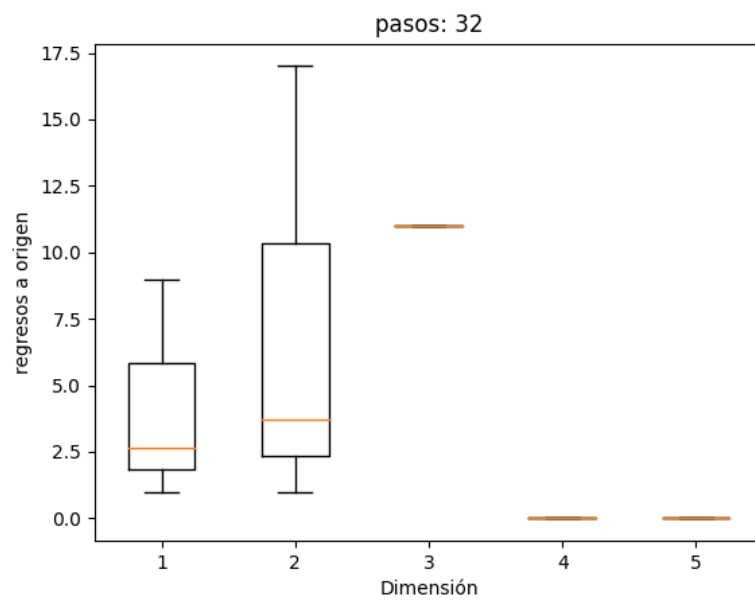


Figura 3: caja-bigote 64 pasos

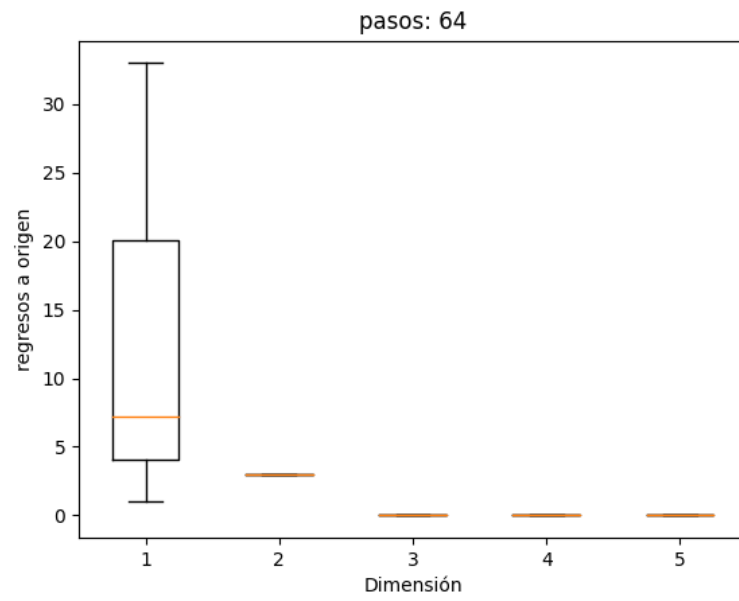


Figura 4: caja-bigote 128 pasos

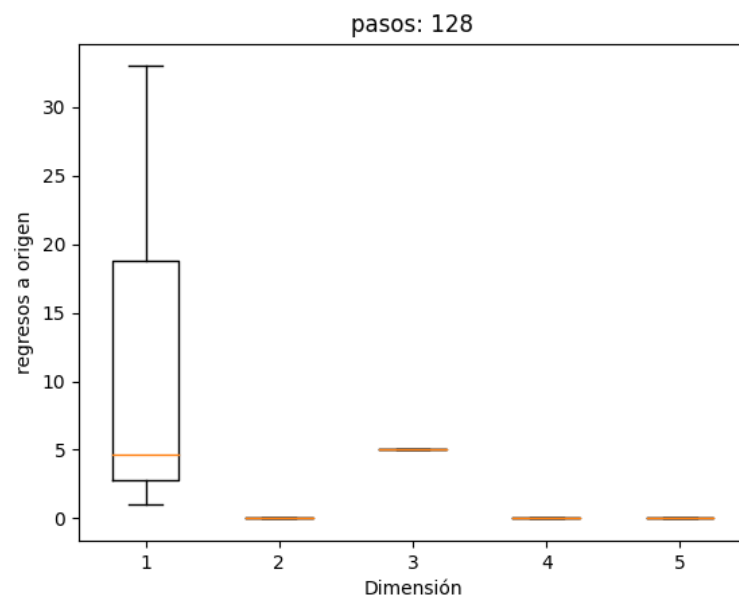


Figura 5: caja-bigote 256 pasos

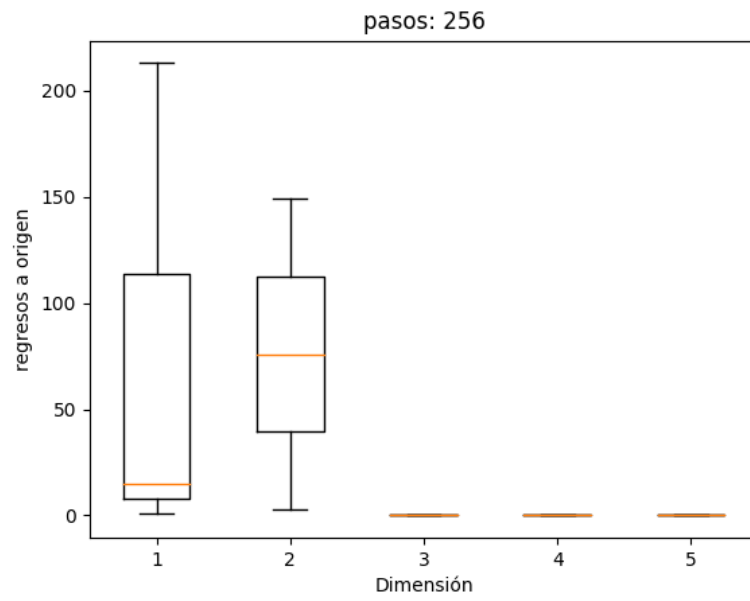
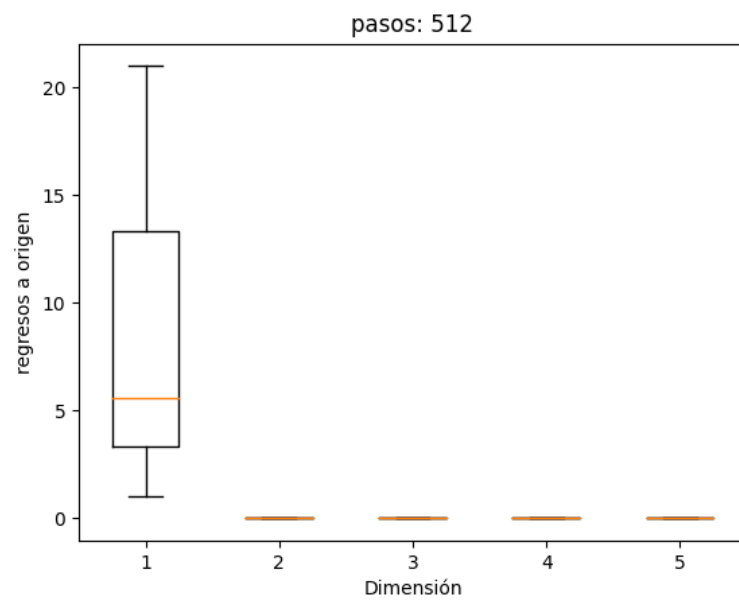


Figura 6: caja-bigote 512 pasos



Referencias

- [1] *Programación en Python-Nivel básico*. COVANTEC, 2021. URL <https://entrenamiento-python-basico.readthedocs.io/es/latest/>.
- [2] E. Samaniego. Práctica1: Movimiento Browniano, 2021. URL <https://github.com/edson-samaniego/simulation-2021/blob/main/Practica-1/practical1.py>.
- [3] E. Schaeffer. Práctica 1: Movimiento Browniano, 2021. URL <https://elisa.dyndns-web.com/teaching/comp/par/p1.html>.
- [4] Guido van Rossum. *Python*. Python Software Foundation, 2021. URL <https://www.python.org/>.