

P8

Ismael Crespo

6 de abril de 2022

1. Introducción

Esta práctica introduce la simulación de los modelos de urnas para analizar los fenómenos donde nos interesa más conocer cuánto hay de algo pero no su tratamiento individual [2]. Este modelo se utilizará para simular un filtrado de todas aquellas partículas de mayor tamaño a un valor crítico c a partir de la agregación y segregación de cúmulos con un número de partículas individuales inicial.

2. Objetivos

- 1.- Estudiar el efecto de la tasa k/c , usando por lo menos cinco valores distintos para ella, suponiendo que partículas con tamaño mayor a c se filtraran y analizar el porcentaje de las partículas que se lograría filtrar por iteración.
- 2.-Determinar cómo el momento idóneo de filtrado depende del valor de c . ¿Qué todo cambia y cómo si c ya no se asigna como la mediana inicial sino a un valor menor o mayor?
- 2.-Estudiar el efecto del parámetro suavizante d en el desempeño de filtrado si la meta es recuperar la mayor cantidad posible de partículas en el proceso, identificar la mejor iteración para realizar el filtrado.

3. Programación en R

La agregación y segregación de las partículas se realiza utilizando las rutinas presentadas por E.Schaeffer [2] a partir de la generación de k cúmulos de diferentes tamaños con distribución normal a partir de un total de n partículas. Una partícula se va a fraccionar con una probabilidad mayor cuando su tamaño este cerca del valor c en una curva sigmoideal y cerca de c en una curva exponencial para la unión de partículas, fomentando la agregación entre cúmulos pequeños. Para la primera parte del trabajo se aceptó la mediana de los tamaños de los cúmulos de partículas como el valor de c .

Para la primera parte del trabajo, se analiza la relación n/k es decir la densidad de cúmulos a partir de una población controlada de partículas. Para este trabajo el valor de n se mantuvo constante y se varió k para trabajar con valores de $k/c = 0.0015, 0.0050, 0.0100, 0.0175$ y 0.0250 . Al final de cada iteración de agregación y segregación de partículas se realiza una filtración hipotética para estimar el porcentaje de partículas que se hubieran filtrado manteniendo el valor de c como la mediana, se realizan 10 filtraciones en cada una de la 10 réplicas simuladas para el análisis. En el código 1 se muestra la selección de los valores de k para cada situación.

Código 1: Declaración de los valores de k y n y su uso en el ciclo `for`.

```
1 #valores de k, n se mantiene constante
2 k_i <-c(1500,5000,10000,17500,25000)
3 n_i <- c(1000000)
4 #iteracion con cada valor de k
5 for(jj in 1:length(k_i)){
```

```

6 for(j in 1:replicas){
7 k=den$k_i[jj]
8 n=den$n_i[jj]
9 originales <- rnorm(k)
10 cumulos <- originales - min(originales) + 1
11 cumulos <- round(n * cumulos / sum(cumulos))
12 }}

```

Código 2: Filtrado de cúmulos de partículas mayor a c

```

1  filtrados=numeric()
2  for(i in 1:length(freq$tam)){
3    if(freq$tam[i]>c){
4      filtrados=c(filtrados,(freq$tam[i]*freq$num[i]))
5    }
6    filtracion=c(filtracion,(sum(filtrados)/n))

```

En las rutinas presentadas en [2] los cúmulos se separan utilizando el modelo de urnas para agrupar los cúmulos del mismo número de partículas, estos agrupamientos se utilizan para realizar el filtrado como se muestra en el código 2, se multiplica la frecuencia de cada cúmulo por el número de partículas que lo constituyen para obtener el total de partículas filtradas.

Para la segunda parte del trabajo se buscó identificar el mejor paso para filtrar variando el valor de c , se utiliza la mediana de los cúmulos como valor base y posteriormente se utiliza como factor 0,95 y 1,05 para multiplicar por el valor original de c y así obtener valores por encima y por debajo de la mediana utilizada originalmente, véase el código 3. De la misma manera se realiza el tercer análisis, variando el valor de d utilizando como factores divisores de la desviación estándar de los tamaños de cúmulos a 1,2,3,4. El filtrado para ambos casos se realizó como se muestra en el código 2. Para consulta ampliada de las rutinas consulte los códigos publicados en [1]

Código 3: Variación del valor de c .

```

1 crit_value=c(0.95,1,1.05)
2 .....
3 for(jj in crit_value){
4 .....
5 cmediana <- median(cumulos) # tamaño critico de cumulos
6 c=cmediana*jj
7 .....
8 }

```

4. Resultados

El porcentaje de partículas recuperadas variando la relación k/c se muestra en la figura 1, se observa un óptimo de filtración con la relación $k/c = 0,005$ en el paso 4, manteniéndose constante posterior a este paso. Es apreciable como la relación $k/c = 0,0015$ presenta los mejores porcentajes de filtración solamente en las primeras dos iteraciones.

El análisis para la filtración variando el valor de c y d se realiza de la misma manera y se presenta en la figura 2 y 3. Se observa una dependencia importante entre estos valores para mejorar la filtración y depende de en que paso se realice la filtración el valor de d y c funcionará mejor, sin embargo la filtración observa mejoría hasta los pasos 4 y 5 para después mantenerse constantes para cada valor de d y c .

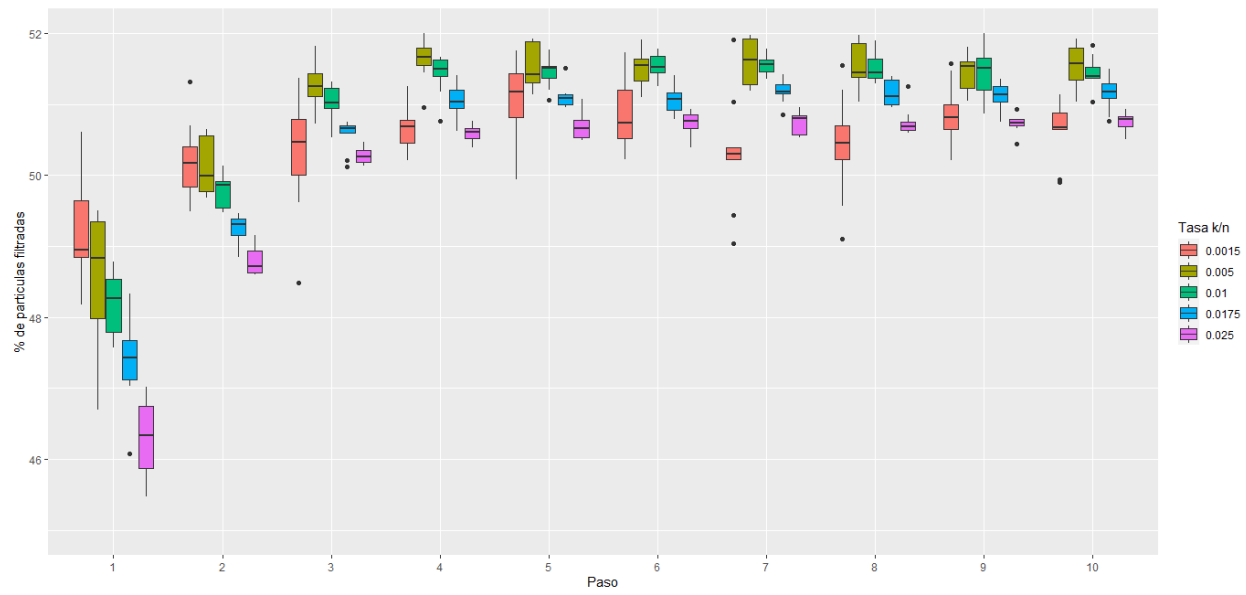


Figura 1: Porcentaje de partículas filtradas en cada paso de la réplica según el valor de k/c

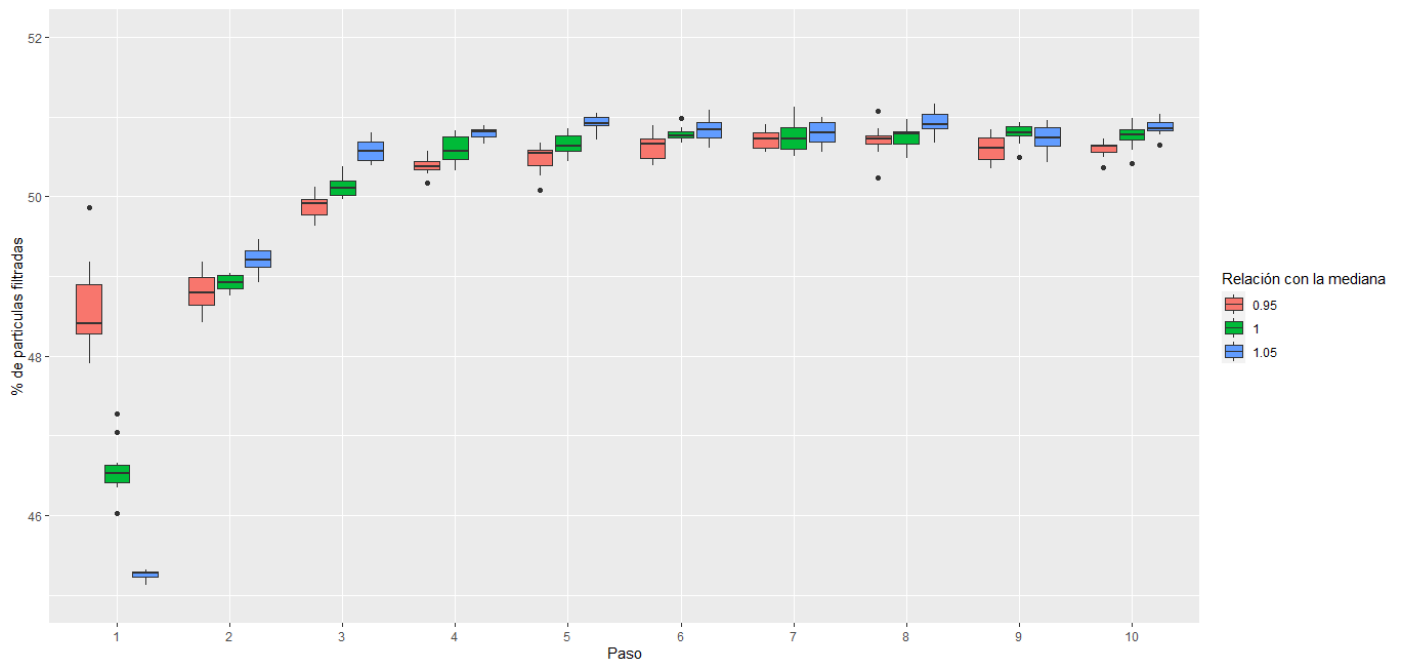


Figura 2: Porcentaje de partículas filtradas en cada paso de la réplica según el valor de c .

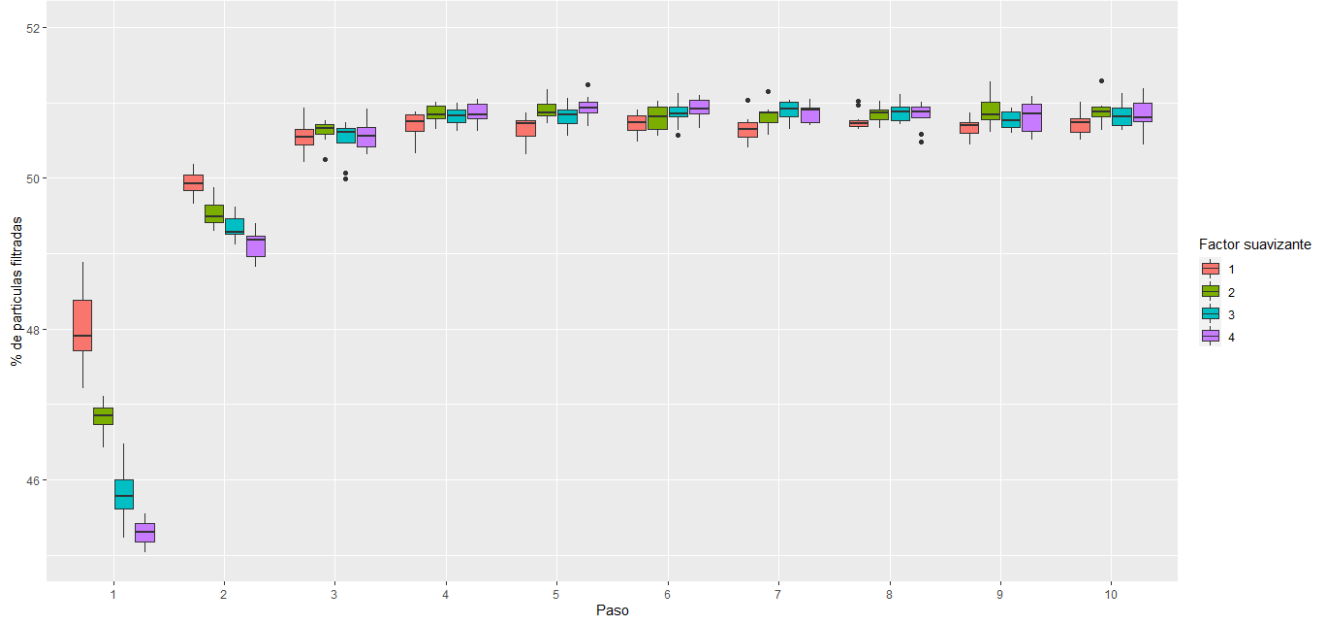


Figura 3: Porcentaje de partículas filtradas en cada paso de la réplica según el valor de d .

5. Conclusiones

La filtración de cúmulos de partículas tiene mejores rendimientos en el paso 4, manteniéndose constantes para los pasos posteriores, se toma como mejor relación $k/c = 0,005$ para esta situación, si la filtración se ve obligada a realizarse en tiempos cortos, es necesario utilizar relaciones k/c menores a $0,005$. De manera similar los resultados de la filtración en cada paso dependen del valor de d y c , si la filtración se realiza a tiempos cortos se es mejor utilizar valores de c menores a 1, por otra parte valores de d mayores nos darán mejores resultados de filtración si esta se tuviese que realizar en las primeras iteraciones.

Referencias

- [1] I. Crespo. P8: Modelo de urnas. 2022. URL <https://github.com/IsmaelHC/Simulacion-NANO-2022/tree/main/P8>.
- [2] E. Schaeffer. Simulación. 2022. URL <https://satuelisa.github.io/simulation>.