Tarea 5

Eduardo Navarro

Septiembre 2021

1. Introducción

Siguiendo las indicaciones de la clase, se realizó una comparación entre un aproximado del valor del área de un integral, al cual le aplicamos pruebas estadísticas para ver el que tan mejor se aproxima nuestro resultado.

2. Desarrollo

Con las instrucciones de la tarea [4] se prosiguió a crear una instrucción para ver la similitud entre 2 cifras y que tan cercanas se encontraban a partir de sus decimales [5], agregándola al código que se nos proporcionó en [2] para ver la similitud entre la cifra generada y la calculada mostrada en [2]. Al programa se le añadieron `for` para los puntos y otro para las repeticiones dentro de éste. Se añadió un `for` adicional para la revisión de los decimales.

Listing 1: Código para la obtención de la relación entre los puntos y los decimales que coinciden.

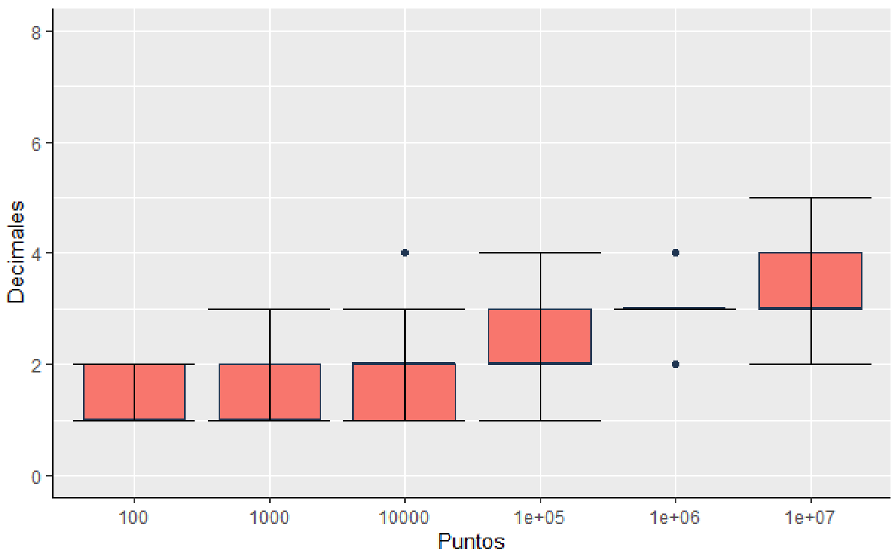
```
cuan <- c(100, 1000, 10000, 100000, 1000000, 10000000)
entero<- c(0.0488341111)
compar <- data.frame()
n = seq( 1, 10, 1)
j = c(1:50)

for (cuantos in cuan) {
  for (rep in j) {...

for (i in n) {
  e = trunc(entero*10^i)/10^i
  co = trunc(compa*10^i)/10^i
  if (co == e) {num = i
  } else {break;}
}

values<-c(cuantos, num)

compar=rbind(compar, values)
}
}
names(compar) <- c("puntos", "decimales")
```

Con los datos obtenidos se obtuvo la tabla 1, la cual nos sirvió para hacer nuestra gráfica 1 donde podemos observar un aumento en el número de decimales que coinciden en base al número de puntos.

Tabla 1: Muestra de datos obtenidos en comparación de decimales.

puntos	decimales
100	1
100	2
100	1
100	1
100	1
100	1
100	2
100	2
100	1

Gráfica 1: Decimales que coinciden a partir de detreminado número de puntos.



Listing 2: Código para la obtención de la gráfica 1.

```
compar$puntos = as.factor(compar$puntos)

ggplot(compar, aes(x=puntos , y= decimales , fill= rep)) + # fill=name allow to
    automatically dedicate a color for each group
  geom_boxplot(fill = "#F8766D", colour = "#1F3552")+
  stat_boxplot(geom = "errorbar", width = 0.9)+
  theme(axis.line = element_line(colour = "black", size = 0.25))+
  coord_cartesian(ylim = c(0,8))+
  labs(x="Puntos", y= "Decimales")
```

A los datos de la tabla 1 se les hicieron las pruebas estadísticas de Shapiro–Wilk [6] y en base a los resultados obtenidos se realizó la prueba de Kruskal-Wallis [1].

Tabla 2: Resultados de la prueba Shapiro–Wilk.

puntos	w	p
10ˆ2	0.58795	1.295*10ˆ-10
10ˆ3	0.6879	5.053*10ˆ-9
10ˆ4	0.80305	1.031*10ˆ-6
10ˆ5	0.83964	8.207*10ˆ-6
10ˆ6	0.69368	6.384*10ˆ-9
10ˆ7	0.76623	1.588*10ˆ-7

Tabla 3: Resultados de la prueba Kruskal-Wallis para todos los datos.

H(5)	p-value
180.72	2.2*10ˆ-16

Tabla 4: Resultados de la prueba Kruskal-Wallis entre los últimos 2 datos.

H(3)	p-value
3.9089	0.2715

Se intentó el primer reto donde se tenía que verificar el valor de pi [3] obteniéndose la tabla 5 a la cual se le realizarón las pruebas estadísticas y finalmente se obtuvo la gráfica 2.

Tabla 5: Ejemplos de valores de pi obtenidos.

valor pi	puntos	decimales
3.124	10000	1
3.154	10000	1
3.1212	10000	1
3.1556	10000	1
3.1532	10000	1
3.1188	10000	1
3.146	10000	2
3.1388	10000	1

Gráfica 2: Decimales que coinciden en pi.

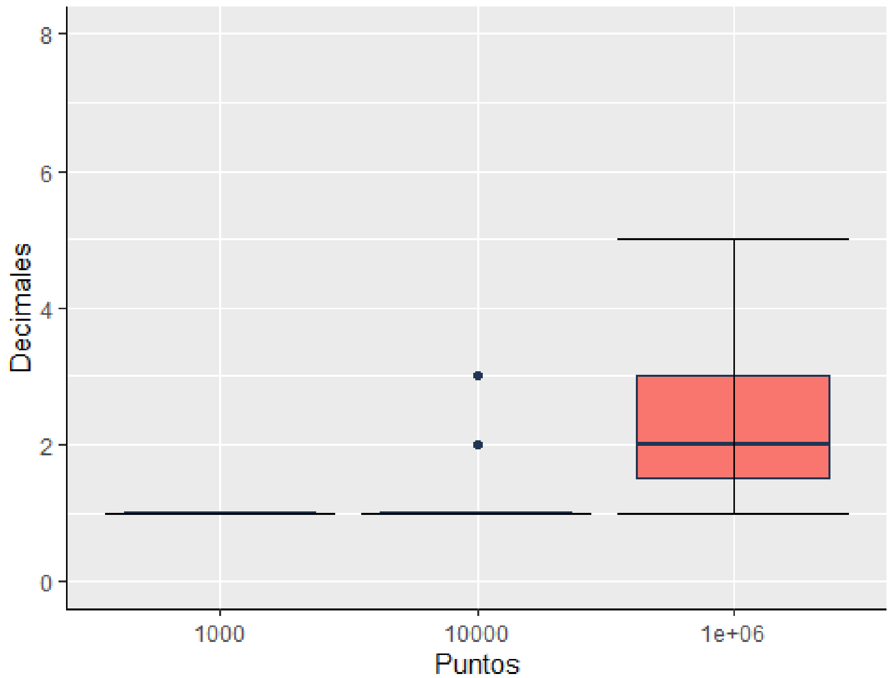


Tabla 6: Resultados de la prueba Shapiro–Wilk para pi.

w	p
0.60337	8.386*10^-10

Tabla 7: Resultados de la prueba Kruskal-Wallis para pi.

H(2)	p-value
21.239	2.44*10^-5

Listing 3: Código para la obtención de pi.

```
entero = 3.141592
n = seq(1, 6, 1)
cuan = c(1000, 10000, 1000000)
compar = data.frame()
j = c(1:50)

for (cuantos in cuan) {
  for (rep in j){
    interior = 0
    for (r in 1:cuantos) {
      x = runif(1, -1, 1)
      y = runif(1, -1, 1)
      d = sqrt(x*x + y*y)
      if (d < 1) {
        interior = interior + 1
      }
    }

    tasa = interior / cuantos
```

```
pi = 4 * tasa

print(pi)
```

3. Conclusiones

Al obtenerse una p menor a 0.05 en la prueba de Shapiro-Wilk se tiene que los datos no vienen de una distribución normal. esto se pudo observar en la grafica donde se varia la media que se iba incrementando de una forma no lineal. con los resultados de la prueba de Kruskal Wallis se obtuvo una p pequeña para la prueba con todos los puntos y se rechaza la hipótesis nula donde se concluye que no todas las medianas de población son iguales. para los últimos 2 valores se aprecia que pueden ser similares lo que indica una tendencia a la estabilización en este punto. Para el reto pi se obtuvieron resultados similares.

Referencias

[1] José Antonio: Estadística Aplicada. Kruskall-wallis en RStudio, 2020. URL `https://www.youtube.com/watch?v=WEjudFpbCcE`.

[2] Elisa Schaeffer. Práctica 5: método Monte-Carlo. `https://elisa.dyndns-web.com/teaching/comp/par/p5.html/`, 2021. [Online; accessed 28-September-2021].

[3] Elisa Schaeffer. pi.R. `https://github.com/satuelisa/Simulation/blob/master/MonteCarlo/pi.R/`, 2021. [Online; accessed 28-September-2021].

[4] Elisa Schaeffer. Simulación P5 AD21, 2021. URL `https://www.twitch.tv/videos/1155347510`.

[5] Nizamuddin Siddiqui. How to truncate a numerical vector to a specified number of decimal places in r? `https://www.tutorialspoint.com/how-to-truncate-a-numerical-vector-to-a-specified-number-of-decimal-places-in-r`, 2020. [Online; accessed 28-September-2021].

[6] El Tío Estadístico. Cómo hacer la PRUEBA DE NORMALIDAD en R, 2020. URL `https://www.youtube.com/watch?v=LAzSb6jCFbs`.