Tarea 11

Eduardo Navarro

Noviembre 2021

1. Introducción

En esta práctica se analizó el porcentaje de soluciones de Pareto en función del número de funciones objetivo. Después se analizaron los resultados obtenidos.

2. Desarrollo

Con las instrucciones de la tarea [3] y lo visto en clase [4] se le hicieron modificaciones al código para obtener el porcentaje a determinados números de funciones. Se añadió un `for` para las funciones `"k"` dadas de 2, 3, 4 y 5 además de otro `for` para las repeticiones.

Listing 1: Código para la obtención del porcentaje de soluciones de Pareto a determinados números de funciones.

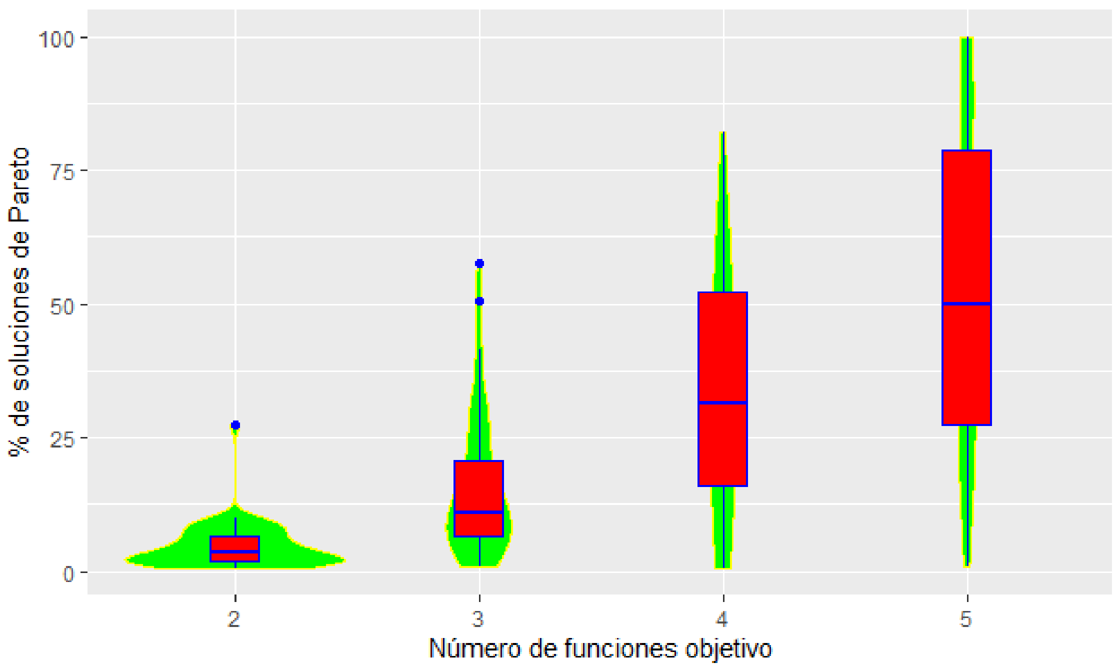
```
datos = data.frame()

vc <- 4
md <- 3
tc <- 5
ka <- c(2,3,4,5) # cuantas funciones objetivo
obj <- list()
je<- 1:50


for (k in ka) {
for (repe in je) {

for (i in 1:k) {
  obj[[i]] <- poli(md, vc, tc)
}

...

frente <- subset(val, no.dom) # solamente las no dominadas

porcentaje = (length(frente[,1])/n)*100



resultado = c(k, repe, porcentaje)
datos = rbind(datos, resultado)
names(datos) = c("k", "Replica", "Porcentaje")
}
}
```

Con esto se generaron los datos de la tabla 1 y se procedió a graficar 1.

Tabla 1: Ejemplo de datos obtenidos.

k	**Réplica**	**Porcentaje**
2	1.0	0.5
2	2.0	6.0
2	3.0	3.5
2	4.0	7.5
2	5.0	5.0
2	6.0	2.0
2	7.0	2.5

Gráfica 1: % de soluciones a funciones dadas.



Listing 2: Código para la obtención de la gráfica y análisis estadístico.

```
library(ggplot2) # recordar instalar si hace falta

datos$k = as.factor(datos$k)
gr <- ggplot(datos, aes(x=k, y=Porcentaje)) + geom_violin(fill="green", color="
    yellow")
gr + geom_boxplot(width=0.2, fill="red", color="blue", lwd=0.5) +
  labs(x = "Numero de funciones objetivo", y = "% de soluciones Pareto")

library(tidyverse)

funcionporc<-datos%>%
  group_by(k) %>%
  summarise(

    promedio = mean(Porcentaje, na.rm = TRUE),
    desviacion_std = sd(Porcentaje, na.rm = TRUE),
    varianza = sd(Porcentaje, na.rm = TRUE)^2,
    mediana = median(Porcentaje, na.rm = TRUE),
```

```
  rango_intercuartil = IQR(Porcentaje, na.rm = TRUE)
 )

pshapiro<-tapply(datos$Porcentaje, datos$k, shapiro.test)

kruskal.test(Porcentaje~k, data=datos)

pwilcox<-pairwise.wilcox.test(datos$Porcentaje, datos$k)
```

De la gráfica 1 podemos ver que hay diferencias considerables conforme aumenta el número de funciones objetivo. Para comprobarlo se realizaron las pruebas de Shapiro–Wilk [5] y Kruskal-Wallis [1] junto con la prueba Wilcox [2] para el análisis de datos.

Tabla 2: Datos estadísticos obtenidos.

k	**Promedio**	**Desviacion std**	**Varianza**	**Mediana**	**Rango intercuartil**
2	4.66	4.34	18.82	3.50	4.50
3	15.57	12.96	168.00	11.00	14.25
4	33.75	23.00	528.92	31.50	36.00
5	53.90	31.36	983.48	49.75	51.38

Tabla 3: Resultados de la prueba Shapiro–Wilk.

Funciones	**W**	**P**
2	0.7145	$1{,}52 \times 10^{-8}$
3	0.8696	$5{,}48 \times 10^{-5}$
4	0.9535	0.0474
5	0.9234	0.0031

Tabla 4: Resultados de la prueba Kruskal-Wallis.

H(3)	**P**
99.15	$2{,}20 \times 10^{-16}$

Tabla 5: Resultados de la prueba por parejas de Wilcox.

Funciones	**2**	**3**	**4**
3	$5{,}60 \times 10^{-8}$	-	-
4	$4{,}40 \times 10^{-10}$	0.0001	-
5	$8{,}50 \times 10^{-15}$	$7{,}30 \times 10^{-10}$	0.0014

3. Conclusiones

De la práctica podemos concluir que mientras más funciones se tienen se alcanza un mayor rango de porcentaje de soluciones de Pareto. A un número de funciones bajas se concentra el % de soluciones en la parte baja del rango intercuartil. Al aumentar el número de funciones se observa una concentración casi uniforme a lo largo del rango intercuartil. De las pruebas estadísticas de Shapiro–Wilk se tiene que no tienen una distribución normal pero se observa que esta aumenta conforme el número de funciones aumenta hasta cierto punto, luego vuelve a decrecer. De la prueba de Kruskal-Wallis se rechaza la hipótesis nula y se concluye que las medianas no son todas iguales y para observar mejor estas diferencias, de la prueba por parejas de Wilcox se observa como todos los grupos son significativamente diferentes. Se cree que el comportamiento se debe a que al haber más rutas dimensionales se pueden tener más soluciones óptimas y por ende puede aumentar el porcentaje.

Referencias

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

[2] Thomas Pernet. 9 Non Parametric tests, 2020. URL https://bookdown.org/thomas_pernet/Tuto/non-parametric-tests.html.

[3] Elisa Schaeffer. Práctica 11: frentes de Pareto. https://elisa.dyndns-web.com/teaching/comp/par/p11.html/, 2021. [Online; accessed 1-Noviembre-2021].

[4] Elisa Schaeffer. Simulación p11: Frentes de pareto, 2021. URL https://www.twitch.tv/videos/1195144993.

[5] El Tío Estadístico. Cómo hacer la Prueba de Normalidad en R, 2020. URL https://www.youtube.com/watch?v=LAzSb6jCFbs.