

Frentes de Pareto

1445183

30 de abril de 2019

1. Objetivo

Paralelizar el código y graficar el porcentaje de soluciones de Pareto como función del número de funciones objetivo, verificando que las diferencias observadas sean estadísticamente significativas.

2. Descripción

Se empieza paralelizando el código proporcionado por la práctica [3] desde el inicio.

```
1 library(doParallel)
2 cluster <- makeCluster(detectCores() - 1)
3 pick.one <- function(x) {
4   if (length(x) == 1) {
5     return(x)
6   } else {
7     return(sample(x, 1))
8   }
9 }
10 [...]
```

Después el código es modificado para que sean 100 soluciones aleatorias (n), con funciones objetivo (k) de 2 a 8 con 25 réplicas cada uno haciendo uso de `for`.

```
1 vc <- 4
2 md <- 3
3 tc <- 5
4 n=100
5 datos<-data.frame(funciones= integer(), replicas=integer(),
6                   soluciones= integer(), porcentaje=integer())
7
8 for(k in 2:8){
9   print(k)
10  for(replica in 1:25){
11    print(replica)
12    obj <- list()
13    for (i in 1:k) {
14      obj[[i]] <- poli(vc, md, tc)
15    }
16    minim <- (runif(k) > 0.5)
17    sign <- (1 + -2 * minim)
18    sol <- matrix(runif(vc * n), nrow=n, ncol=vc)
19    clusterExport(cluster, c("n", "k", "sol", "tc", "obj", "eval", "dim", "valor"))
```

Para las pruebas estadísticas se hizo uso de `Shapiro.test` y `Dunn.test` para este último se tuvo que instalar el paquete en R.

3. Resultados

Se puede observar en la figura 1 los porcentajes obtenidos para cada función objetivo (k), donde a mayor cantidad de k los porcentajes tienden a ser mayores.

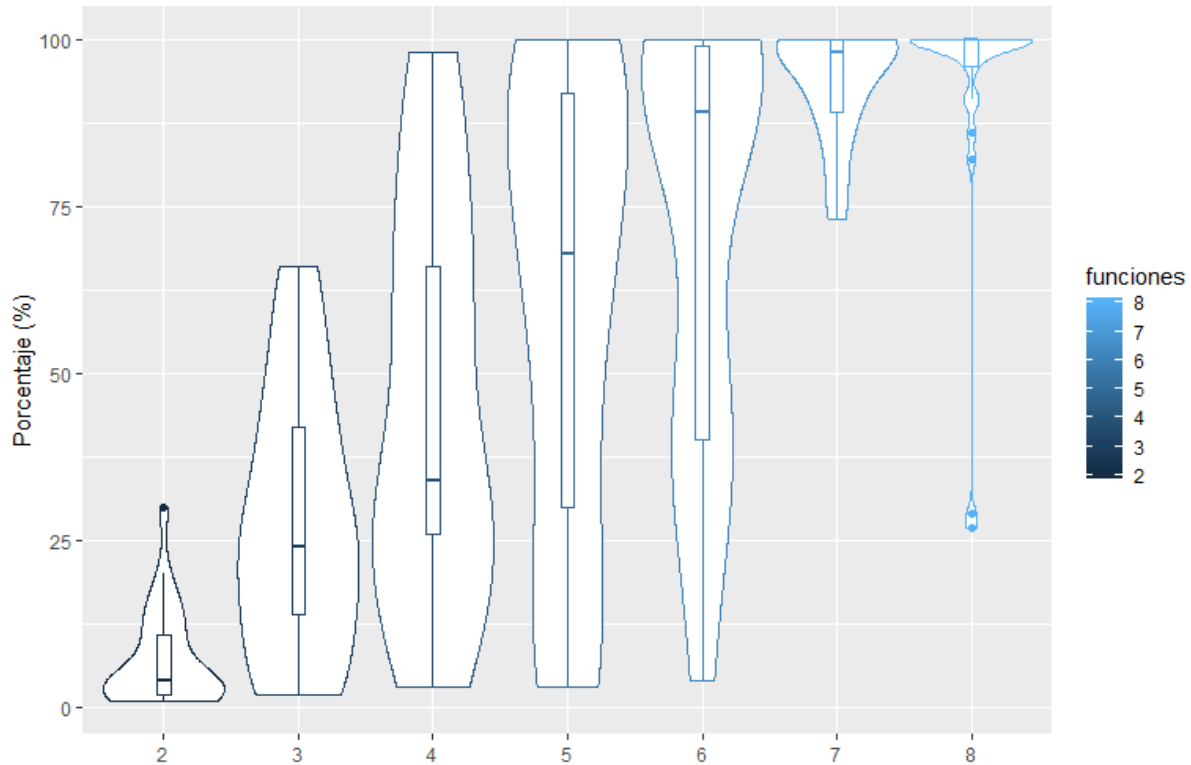


Figura 1: porcentaje de las funciones objetivo (k)

La prueba de `shapiro.test` nos da el valor p que en este caso es menor de 0.05 por lo que los datos no cumplen la normalidad, tomando de referencia el trabajo de *Saus* [2] se hace la prueba `dunn.test` [1].

```
1 shapiro.test(datos$porcentaje)
2
3 Shapiro-Wilk normality test
4
5 data:  datos$porcentaje
6 W = 0.84612, p-value = 2.668e-12
```

```

1  Kruskal-Wallis rank sum test
2
3  data: x and group
4  Kruskal-Wallis chi-squared = 106.5637, df = 6, p-value = 0
5
6
7
8      Comparison of x by group
9      (No adjustment)
10
11  Col Mean-|
12  Row Mean |      2      3      4      5      6      7
13
14      3 |      -2.190205
15          |      0.0143*
16
17      4 |      -3.499568      -1.309362
18          |      0.0002*      0.0952
19
20      5 |      -4.787924      -2.597718      -1.288356
21          |      0.0000*      0.0047*      0.0988
22
23      6 |      -5.957247      -3.767042      -2.457679      -1.169323
24          |      0.0000*      0.0001*      0.0070*      0.1211
25
26      7 |      -7.877179      -5.686973      -4.377611      -3.089254      -1.919931
27          |      0.0000*      0.0000*      0.0000*      0.0010*      0.0274
28
29      8 |      -8.134850      -5.944644      -4.635282      -3.346925      -2.177602      -0.257671
30          |      0.0000*      0.0000*      0.0000*      0.0004*      0.0147*      0.3983
31
32  alpha = 0.05
33  Reject Ho if p <= alpha/2

```

4. Conclusión

A partir de siete objetivos se empieza a observar que los porcentajes de las soluciones no dominadas son equivalentes, es decir, que no existen diferencias significativas. Al existir mayor cantidad de objetivos la comparación entre ellos aumenta por lo que si al compararse un objetivo con otro no mejora hay más posibilidades de que con otro si, de manera que hay mayor cantidad de soluciones dominantes, esto es la frente de pareto.

Referencias

- [1] Alexis Dinno. Package ‘dunn.test’, 2017. URL <https://cran.r-project.org/web/packages/dunn.test/dunn.test.pdf>.
- [2] Liliana Saus. Práctica 11: Frentes de Pareto, 2018. URL <https://github.com/pejli/simulacion/blob/master/p11/p11.pdf>.
- [3] Elisa Schaeffer. Práctica 11: Frentes de Pareto, 2019. URL <https://elisa.dyndns-web.com/teaching/comp/par/p11.html>.