

PRÁCTICA 7: BÚSQUEDA LOCAL

FABIOLA VÁZQUEZ

4 de noviembre de 2020

1. Introducción

El objetivo de esta práctica [3] es implementar una heurística para encontrar los máximos locales de la función,

$$f(x, y) = 3e^{-(0.8y+1)^2-(0.8x)^2}(0.8x-1)^2 - \frac{1}{3} \cdot e^{-(0.8x+1)^2-(0.8y)^2} + e^{-(0.8x)^2-(0.8y)^2}(10(0.8x)^3 - 2(0.8x) + 10(0.8y)^5).$$

La figura 1a muestra la gráfica de la función f , pero en este tipo de gráficas es complicado la visualización de un punto moviéndose por la superficie, se opta por utilizar una representación diferente, la cual se muestra en la figura 1b. El experimento se realiza con el software R versión 4.0.2 [2] en un cuaderno de Jupyter [1].

2. Experimento

En este caso se considera el dominio de la función como $D_f = \{(x, y) \mid -3 \leq x \leq 3, -3 \leq y \leq 3\}$, entonces a la función f se le pone la restricción de si se llega a evaluar en valores diferentes al dominio de f , regrese $-\infty$, esto porque se busca maximizar la función, si se busca minimizar, utilizar ∞ .

El método utilizado para maximizar la función, consiste en posicionar un punto (x, y) en el plano y dados un Δx y un Δy se consideran ocho vecinos, de los cuales se elige el que al evaluarlo en la función f regrese el valor mayor, si ese valor es mejor que el que se tiene actualmente se considera como el nuevo mejor y se repite el proceso una cantidad t de iteraciones. Esto se realiza de forma automatizada con la función `replica` mostrada en el código 1.

Se posicionan 100 puntos al azar en el plano en color negro, el mejor valor obtenido se representa con un triángulo blanco, y se realiza el proceso en 100 iteraciones, lo cual se visualiza en una animación.

Como se aprecia en la animación, en 100 pasos los cien puntos están ya atascados en algún máximo, local o global, de la función. Se realizan ahora 10,000 iteraciones, las cuales se muestran en la figura 2.

```
replica <- function(t){  
  currx <- runif(1, -3, 3)  
  curry <- runif(1, -3, 3)  
  best <- c()  
  best[1] <- currx
```

```

best[2] <- curry

for (tiempo in 1:t) {
  a <- currx
  b <- curry
  deltax <- runif(1, 0, 0.2)
  deltay <- runif(1, 0, 0.2)
  left <- currx - deltax
  right <- currx + deltax
  up <- curry + deltay
  down <- curry - deltay

  posx <- c(left, right, a)
  posy <- c(up, down, b)

  pos_max <- max(f(left, up), f(left, down), f(left, b), f(right, up), f(right, down), f(
(right, b), f(a, up), f(a, down))

  for (x in posx){
    for (y in posy){
      if (f(x,y)==pos_max){
        currx <- x
        curry <- y
      }
    }
  }

  if (f(best[1], best[2]) <= f(currx, curry)){
    best[1] <- currx
    best[2] <- curry
  }
}
return(best)
}

```

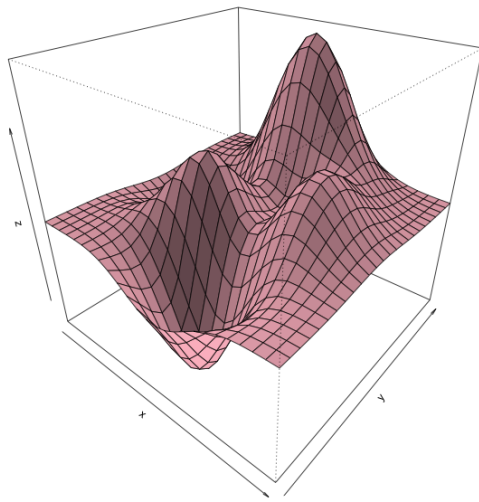
Código 1: Función `replica`.

3. Reto 1

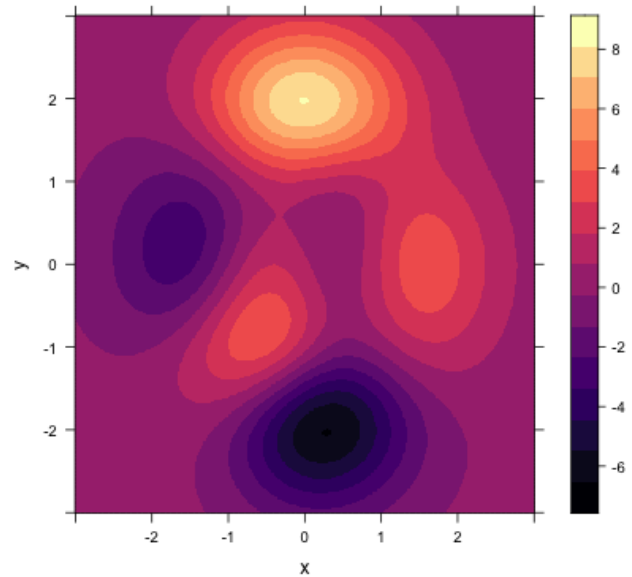
En el reto 1, a diferencia de la tarea base, se cambia el tipo de movimiento a otro llamado **recocido simulado**, el cual compara el valor **delta** definido como la diferencia entre $f(x', y')$ y $f(x, y)$, donde (x', y') es algún vecino. Si el valor de **delta** es mayor que cero, el vecino se acepta como la posición actual, ya que representa un valor mayor que el actual, pero si **delta** es menor que cero, no es una mejora, aun así se podría aceptar como vecino con una probabilidad de $\exp(-\text{delta}/T)$, donde T es la temperatura inicial. Se tienen dos parámetros iniciales, la temperatura inicial T y xi que es la rapidez con la que T disminuye si se acepta un vecino que no te da un mejor valor. Al igual que en la tarea base, se realiza una **animación** de 100 pasos ahora con este tipo de movimiento.

El objetivo es examinar los efectos estadísticos del valor inicial T y el valor xi , para ello se realiza un experimento variando dichos valores, la temperatura T se considera como $\{1, 10, 100, 1000, 10000\}$ y el valor de xi varía en $\{0.7, 0.8, 0.9, 0.95, 0.99, 0.995, 0.999\}$.

Se realiza una prueba de correlación para verificar si algún parámetro influye en si se alcanza un máximo local o no. Primero para la temperatura inicial,



(a) Gráfica en tres dimensiones de la función f .



(b) Gráfica de nivel de la función f .

Figura 1: Gráficas de la función f .

Pearson's product-moment correlation

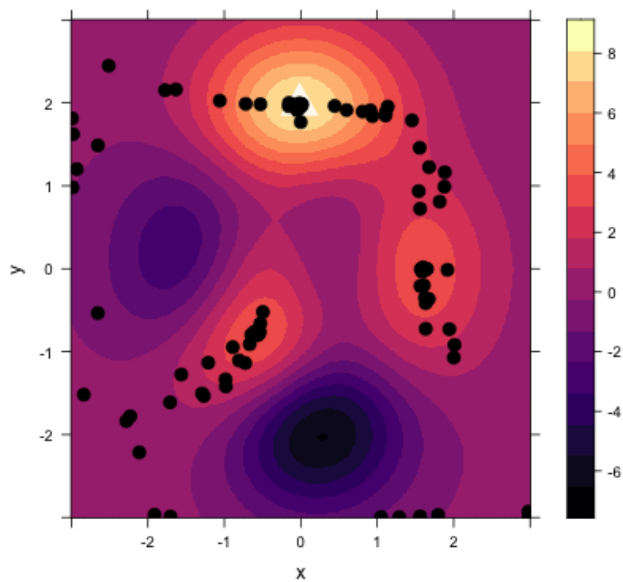
```
data:  datos$'Temp Inicial' and datos$max
t = 0.082459, df = 3498, p-value = 0.9343
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.03173876  0.03452414
sample estimates:
      cor
0.001394216,
```

se tiene un valor $p \approx 0.93$ el cual es mayor que $\alpha = 0.05$, por lo que se dice que la temperatura inicial no afecta al resultado. Para el valor de ξ ,

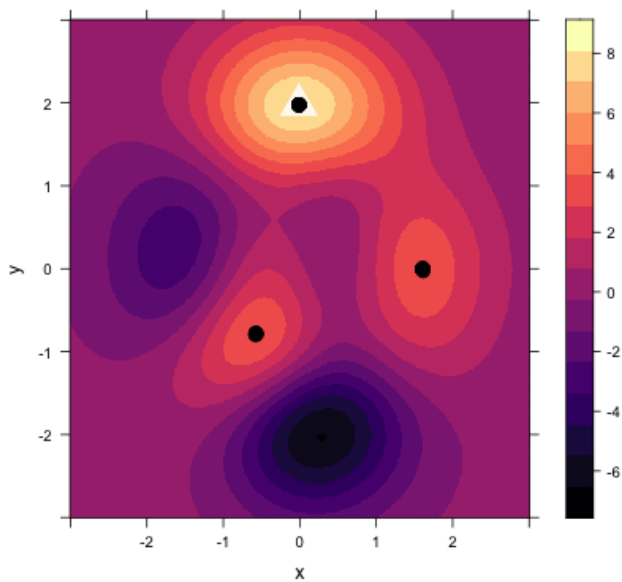
Pearson's product-moment correlation

```
data:  datos$xi and datos$max
t = 2.3218, df = 3498, p-value = 0.0203
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.006103839 0.072265015
sample estimates:
      cor
0.03922742,
```

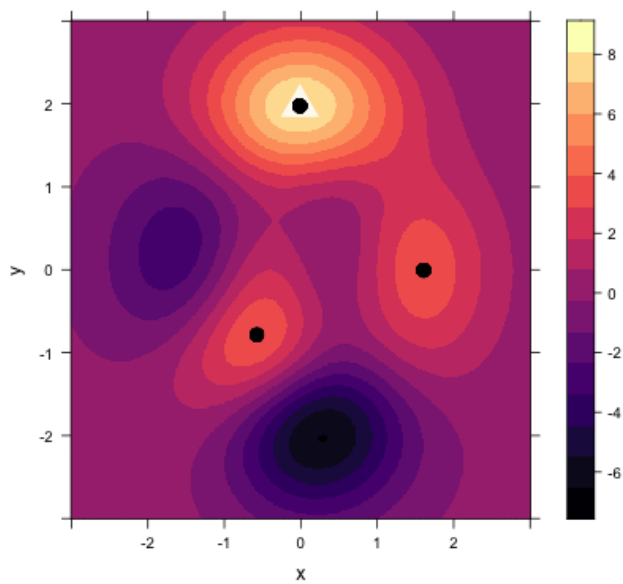
se tiene un valor $p \approx 0.02$ el cual es menor que α , por lo que se concluye que el valor de ξ sí afecta el resultado, pero como el coeficiente de correlación $\text{cor} \approx 0.039$, se concluye que sí afecta pero muy poco.



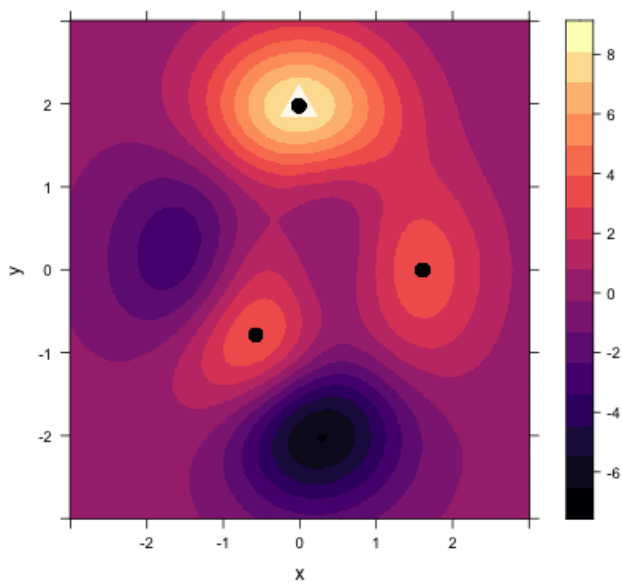
(a) 10 pasos.



(b) 100 pasos



(c) 1000 pasos.



(d) 10000 pasos.

Figura 2: Gráficas de la función f .

4. Reto 2

En el reto 1 se llega a la conclusión de que el valor del parámetro T no influye en el resultado final, así que se deja fijo como $T=1$. Se realizan tres experimentos variando el valor de $\mathbf{x_i}$ como 0.7, 0.8 y 0.9. Para los dos métodos trabajados anteriormente, se compara el valor obtenido con el valor del máximo global, los resultados se muestran en la figura 3, donde se aprecia que los valores obtenidos con los dos métodos son similares.

Se realiza una prueba Welch para ver si hay diferencia significativa con las medias de los datos obtenidos con el método base y con el método del reto 1.

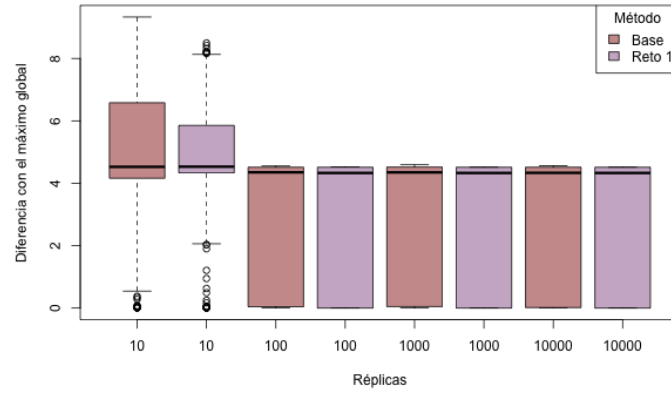
Welch Two Sample t-test

```
data: data$base and data$reto1
t = -1.1828, df = 794.08, p-value = 0.2372
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.5212880  0.1292767
sample estimates:
mean of x mean of y
 3.237430  3.433436
```

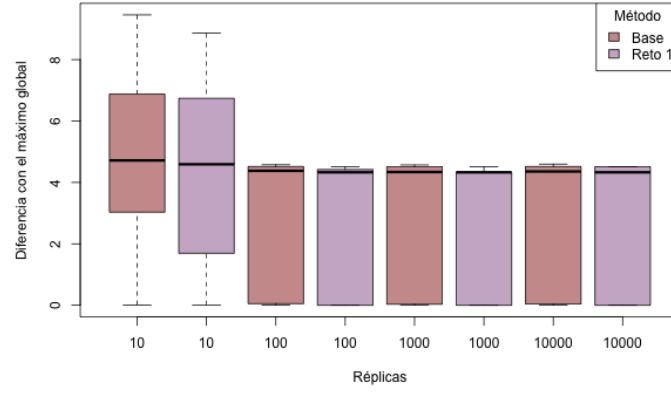
Dado que se obtiene un valor p mayor que 0.05, se dice que no hay diferencia significativa entre las medias de los conjuntos de datos.

Referencias

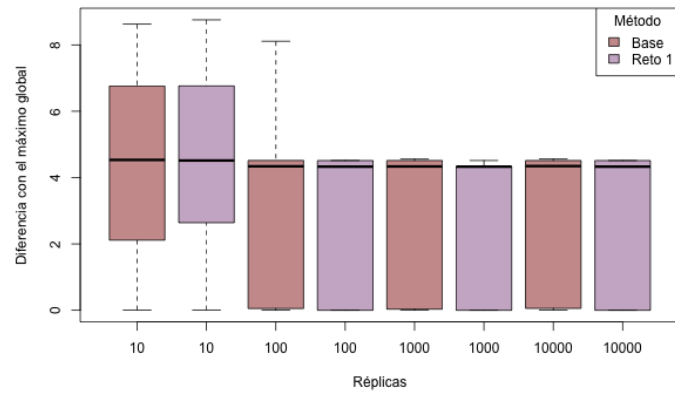
- [1] Thomas Kluyver, Benjamin Ragan-Kelley, Pérez, et al. Jupyter notebooks—a publishing format for reproducible computational workflows. In *Positioning and Power in Academic Publishing: Players, Agents and Agendas: Proceedings of the 20th International Conference on Electronic Publishing*, page 87. IOS Press, 2016.
- [2] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020.
- [3] Elisa Schaeffer. Práctica 7: búsqueda local. <https://elisa.dyndns-web.com/teaching/comp/par/p7.html>.



(a) $\xi=0.7$.



(b) $\xi=0.8$



(c) $\xi=0.9$.

Figura 3: Gráficas de caja comparando los resultados obtenidos con los métodos de la tarea base y del reto 1.