Búsqueda local

Francisco Gerardo Meza Fierro

1. Introducción

En esta práctica se encontrarán puntos máximos locales de funciones mediante una optimización heurística. Se implementará un código que buscará el punto máximo en una función bidimensional tomando como base el código propuesto, el cual minimiza una función unidimensional. Además se mostrará gráficamente la manera en que ese máximo es encontrado.

2. Maximizar la función

La función con la que se trabajará es
$g(x,y) = ((x+0{,}5)^4 - 30x^2 - 20x + (y+0{,}5)^4 - 30y^2 - 20y)/100$ y cuya gráfica se encuentra dada por la figura 1.

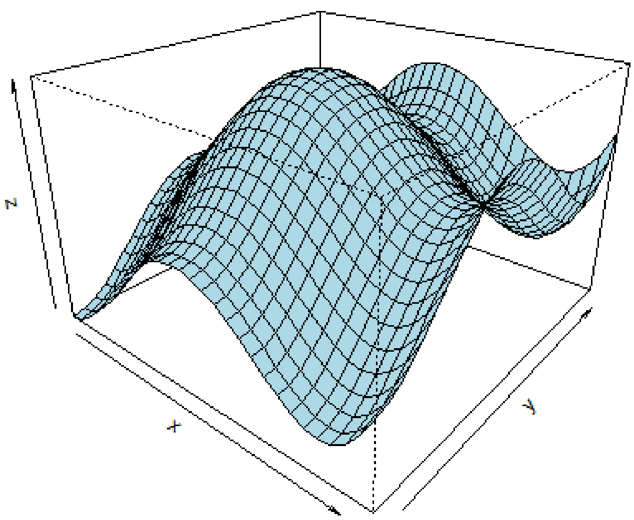


Figura 1: Gráfica de la función $g(x,y)$

El área de análisis para la búsqueda del máximo se limitó desde cuatro negativo hasta cuatro positivo en el plano xy.

Una manera sencilla de comprender cómo es que este código propuesto encuentra el máximo de la función es la siguiente: se posiciona un punto aleatoriamente sobre la superficie de la función y a partir del cual se creará una pequeña vecindad alrededor del punto, los puntos sobre esa vecindad se evaluarán en la función y el mayor obtenido es hacia donde el punto se moverá; ese proceso se repetirá hasta que no haya puntos sobre tal vecindad cuya evaluación en la función no sea mayor al de la posición actual.

La forma en que el código crea dicha vecindad es la siguiente: del punto donde se inicia, toma la coordenada en x y la aumenta y disminuye en un número entre 0 y 0,5; de manera análoga se hace con la coordenada en y, de manera que al final se tienen cuatro puntos candidatos a los cuales se podría mover. Cada uno de estos cuatro puntos se evalúan en la función $g(x, y)$ y la posición actual se reemplaza por el mayor de los cuatro puntos.

Para saber si la implementación del código es adecuada, se comparará el resultado obtenido con el propuesto por $WolframAplpha$. El siguiente cuadro muestra el máximo propuesto por $WolframAplpha$ y el máximo obtenido por el código implementado.

Cuadro 1: Máximos de la función

	Máximo	Coordenada en x	Coordenada en y
$WolframAplpha$	0.0666822	-0.333023	-0.333023
R	0.0666821	-0.332863	-0.332974

Dado que la diferencia entre el valor máximo encontrado con el código implementado en R con el valor que $WolframAplpha$ ofrece es prácticamente cero, se concluye que la implementación fue la adecuada. Se anexa al repositorio una tabla con los demás puntos obtenidos en un archivo $.csv$.

3. Visualización del proceso de búsqueda

En este tipo de problemas, la manera más fácil de poder ver la manera en que se va encontrando la solución deseada es mediante una manera gráfica, pero hacerlo sobre una gráfica como en la figura 1 sería un poco tedioso. Para ello, mediante el uso del paquete `lattice`, se obtiene la siguiente proyección en el plano xy, facilitando el análisis del comportamiento de la búsqueda:

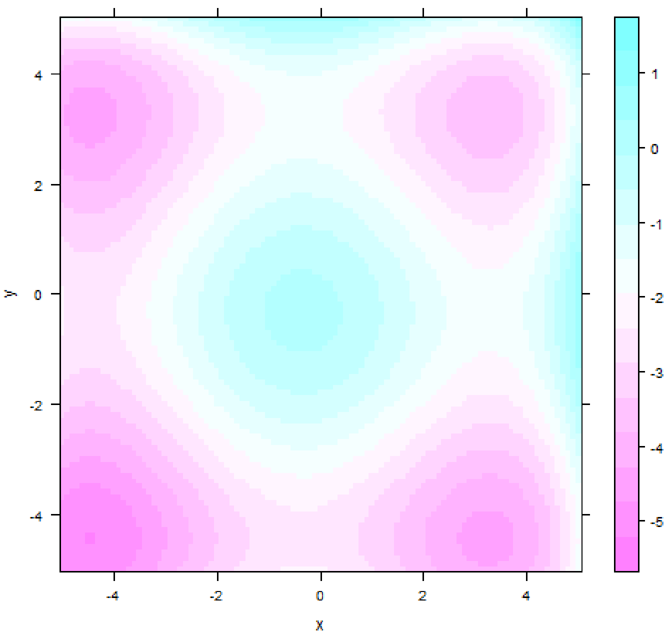
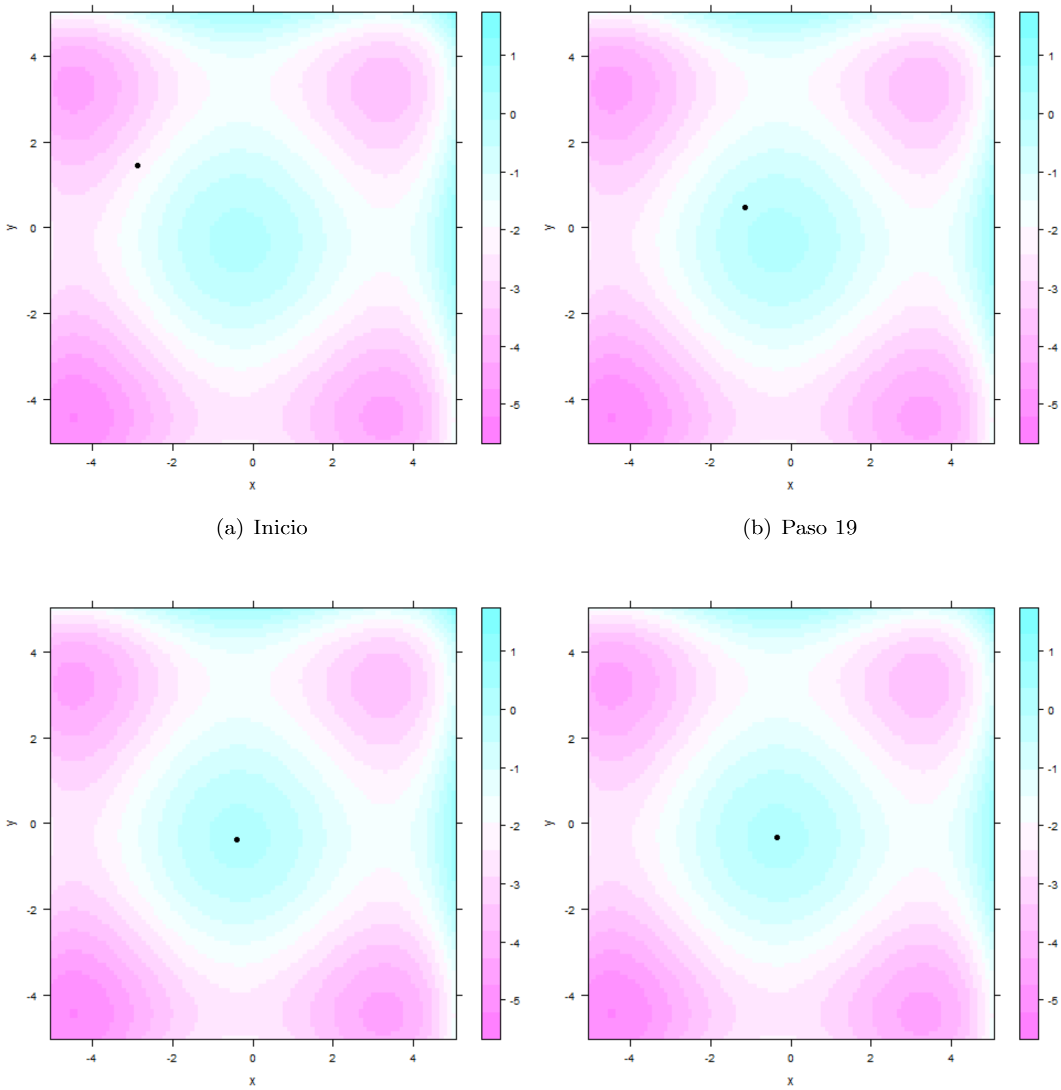


Figura 2: Proyección de la función sobre el plano xy

Teniendo esta proyección, se inicia la búsqueda del máximo. Alguno de los pasos obtenidos la localización del punto máximo se encuentran reflejados sobre la figura 3. Se anexa al repositorio un archivo $.gif$ con el movimiento de inicio a fin de la simulación.



(a) Inicio

(b) Paso 19

(c) Paso 35

(d) Localización del máximo en el paso 51

Figura 3: Visualización gráfica de la búsqueda del máximo

Sabemos por la sección anterior de la práctica que las coordenadas del punto máximo son $(-0{,}333023, -0{,}333023)$ y basándonos en el último paso mostrado en la figura 3, el punto máximo fue obtenido.