

# Informe sobre el Análisis y Resultados de Métodos de Optimización

Cristhian Delgado García

## 1. Planteamiento del problema y variables

La función objetivo depende de dos variables continuas,  $x$  y  $y$ , ambas pertenecientes a  $\mathbb{R}$ . Se trata de un problema de optimización continua en dos dimensiones.

## 2. Propiedades de la función objetivo

La función a minimizar es:

$$f(x, y) = -\arctan(x + y + 1) - \arctan(x - y + 2) - \arctan(-x - y + 3) - \arctan(-x + y + 4)$$

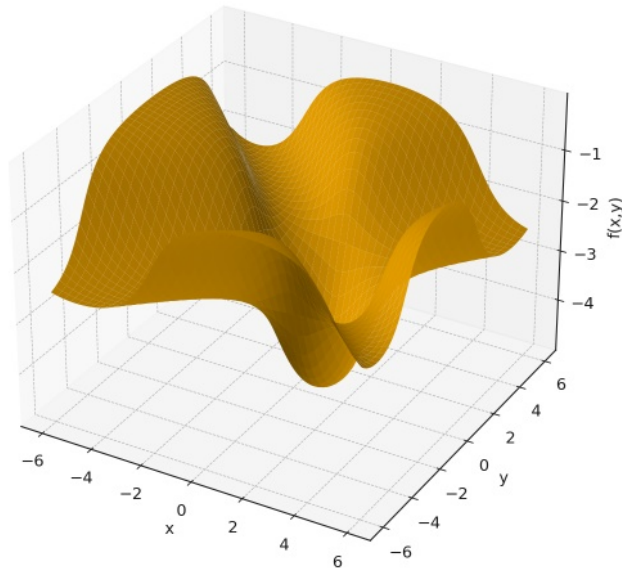


Figura 1: Función Objetivo

Cada término  $\arctan(\cdot)$  toma valores en  $(-\pi/2, \pi/2)$ , por lo que la suma total está acotada en  $(-2\pi, 2\pi)$ . La función es continua e infinitamente derivable en  $\mathbb{R}^2$ .

Definimos:

$$u_1 = x + y + 1, \quad u_2 = x - y + 2, \quad u_3 = -x - y + 3, \quad u_4 = -x + y + 4.$$

Las derivadas parciales son:

$$\begin{aligned}\frac{\partial f}{\partial x} &= -\frac{1}{1+u_1^2} - \frac{1}{1+u_2^2} + \frac{1}{1+u_3^2} + \frac{1}{1+u_4^2}, \\ \frac{\partial f}{\partial y} &= -\frac{1}{1+u_1^2} + \frac{1}{1+u_2^2} + \frac{1}{1+u_3^2} - \frac{1}{1+u_4^2}.\end{aligned}$$

Las segundas derivadas (componentes del Hessiano) se calculan de forma análoga y la matriz Hessiana es:

$$H(x, y) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}.$$

Cabe destacar que la función no es biconvexa, ya que el Hessiano no es semidefinido positivo en todo el dominio y los términos cruzados generan interacciones no convexas.

### 3. Sección teórica: solución analítica, convexidad y unicidad

A continuación se presenta un análisis teórico que justifica la solución observada numéricamente y determina propiedades de convexidad y unicidad.

#### 3.1. Observaciones estructurales

Notamos dos identidades útiles que se mantienen para cualquier  $(x, y)$ :

- $u_1 + u_3 = (x + y + 1) + (-x - y + 3) = 4$  (constante),
- $u_2 + u_4 = (x - y + 2) + (-x + y + 4) = 6$  (constante).

Por tanto, cada par  $(u_1, u_3)$  y  $(u_2, u_4)$  se mueve sobre una recta de suma fija (4 y 6 respectivamente). La función puede reescribirse como

$$f(x, y) = -(\arctan(u_1) + \arctan(u_3)) - (\arctan(u_2) + \arctan(u_4)).$$

Cada término entre paréntesis depende sólo de la partición de la suma fija; así, para fijar la suma  $s$  y estudiar  $g(u) = \arctan(u) + \arctan(s - u)$ , el máximo de  $g$  bajo ciertas condiciones se alcanza en  $u = s/2$  cuando la función  $\arctan$  es cóncava sobre el intervalo considerado.

#### 3.2. Cálculo analítico del crítico candidato

Buscamos puntos críticos resolviendo  $\nabla f(x, y) = 0$ . Evaluando las derivadas en el punto  $(x, y) = (1, 0)$  obtenemos:

- $u_1 = 2, u_2 = 3, u_3 = 2, u_4 = 3,$
- $\frac{1}{1+2^2} = \frac{1}{5} = 0,2, \frac{1}{1+3^2} = \frac{1}{10} = 0,1.$

Sustituyendo en las fórmulas de las derivadas parciales:

$$\begin{aligned}\left. \frac{\partial f}{\partial x} \right|_{(1,0)} &= -0,2 - 0,1 + 0,2 + 0,1 = 0, \\ \left. \frac{\partial f}{\partial y} \right|_{(1,0)} &= -0,2 + 0,1 + 0,2 - 0,1 = 0.\end{aligned}$$

Por tanto  $(1, 0)$  es un punto crítico. El valor objetivo en ese punto es

$$f(1, 0) = -2(\arctan(2) + \arctan(3)) \approx -4,71238898038469 = -\frac{3\pi}{2}.$$

### 3.3. Máximo local de cada pareja y unicidad

Consideremos la función  $g_s(u) = \arctan(u) + \arctan(s - u)$  con  $s > 0$  fijo. Para  $u$  en  $(0, s)$  ambos argumentos son positivos; como  $\arctan$  es cóncava en  $(0, \infty)$  (segunda derivada negativa), por Jensen  $g_s(u)$  alcanza su máximo cuando  $u = s/2$  (simetría y concavidad). Aplicando esto a nuestros pares:

- Para  $s = 4$ ,  $\arctan(u_1) + \arctan(u_3)$  es máximo cuando  $u_1 = u_3 = 2$ .
- Para  $s = 6$ ,  $\arctan(u_2) + \arctan(u_4)$  es máximo cuando  $u_2 = u_4 = 3$ .

Dado que  $f$  es la negación de la suma de estos términos,  $f$  alcanza su mínimo cuando se alcanzan estos máximos simultáneamente. Las ecuaciones

$$\begin{cases} x + y + 1 = 2, \\ x - y + 2 = 3, \end{cases}$$

son equivalentes al sistema lineal

$$\begin{cases} x + y = 1, \\ x - y = 1, \end{cases}$$

que tiene solución única  $x = 1, y = 0$ . Por tanto la solución analítica es única y coincide con el punto crítico encontrado.

Nota: la argumentación anterior es válida porque en la configuración óptima todos los  $u_i$  son positivos (2 y 3); si algún  $u_i$  fuera negativo, la concavidad/convexidad local de  $\arctan$  cambiaría, pero la restricción de suma fija junto con la monotonidad de  $\arctan$  y la desigualdad de Jensen garantizan que la partición simétrica sigue proporcionando el máximo del par en este caso concreto.

### 3.4. Análisis de convexidad

La función no es convexa en  $\mathbb{R}^2$ . Una razón sencilla es que la segunda derivada de  $\arctan$  cambia de signo:  $\frac{d^2}{dt^2} \arctan(t) = -\frac{2t}{(1+t^2)^2}$ , que es negativa para  $t > 0$  y positiva para  $t < 0$ . Por tanto las contribuciones al Hessiano dependen del signo de cada  $u_i$  y pueden ser positivas o negativas.

Un contraejemplo numérico ilustra el cambio de signo en las entradas del Hessiano: en  $(x, y) = (0, 0)$  se tiene  $u_1 = 1, u_2 = 2, u_3 = 3, u_4 = 4$  y la componente  $\frac{\partial^2 f}{\partial x^2}$  resulta positiva; sin embargo en  $(x, y) = (2, 0)$ , donde  $u_1 = 3, u_2 = 4, u_3 = 1, u_4 = 2$ , la misma componente es negativa (cálculos numéricos muestran cambio de signo). Esto implica que el Hessiano no es semidefinido positivo en todo el dominio, y por tanto la función no es convexa.

## 4. Existencia y localización del mínimo global

Aunque el dominio no es compacto, la función está acotada inferiormente ( $\inf f(x, y) > -2\pi$ ). El análisis numérico y el argumento teórico anterior coinciden en que el mínimo global se alcanza en  $(x, y) = (1, 0)$  con valor

$$f(1, 0) = -2(\arctan(2) + \arctan(3)) \approx -4,71238898038469 = -\frac{3\pi}{2}.$$

A continuación se muestra una representación gráfica de dicho mínimo:

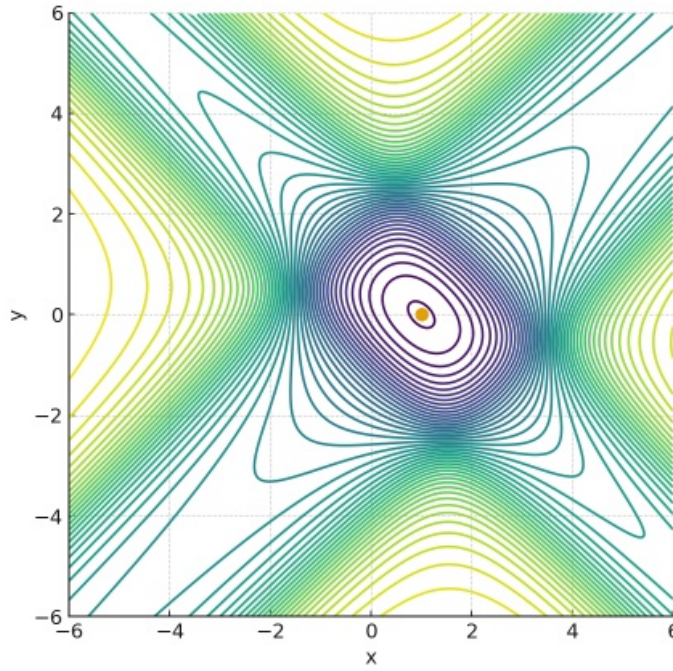


Figura 2: Mapa de contorno con el mínimo resaltado.

## 5. Caracterización del óptimo

Los resultados muestran que el óptimo se encuentra en  $x \approx 1,0$ ,  $y \approx 0,0$ , con norma del gradiente prácticamente nula (precisión de máquina). Distintos puntos iniciales conducen al mismo resultado, lo que sugiere un único mínimo global y una amplia región de atracción.

Por ejemplo, en los experimentos se obtienen soluciones de alta precisión como:

- Solución:  $[1,00000000012, -2,57 \times 10^{-10}]$
- Valor de la función:  $\approx -4,71238898038469$
- Norma del gradiente:  $\lesssim 10^{-10}$

El método de descenso por gradiente también converge a este valor si se ajustan adecuadamente los parámetros, aunque requiere más iteraciones y es más sensible a la tasa de aprendizaje.

## 6. Algoritmos utilizados

En los experimentos se emplearon dos métodos principales de optimización basados en el gradiente, diseñados para encontrar mínimos locales (o globales) de funciones diferenciables.

### 6.1. Descenso por Gradiente (Gradient Descent)

Este es un algoritmo iterativo de primer orden que busca encontrar el mínimo de una función. La idea principal es dar pasos en la dirección opuesta al gradiente de la función en el punto actual, ya que esta es la dirección de máximo descenso.

La fórmula de actualización para cada iteración  $k$  es:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \nabla f(\mathbf{x}_k)$$

Donde:

- $\mathbf{x}_k$  es el vector de variables en la iteración  $k$ .
- $\nabla f(\mathbf{x}_k)$  es el gradiente de la función objetivo  $f$  evaluado en  $\mathbf{x}_k$ .
- $\alpha$  es la **tasa de aprendizaje** (o tamaño del paso), un hiperparámetro crucial que determina la magnitud de cada paso.

#### Características:

- **Simplicidad:** Es conceptualmente simple y fácil de implementar.
- **Sensibilidad a  $\alpha$ :** Una tasa de aprendizaje demasiado grande puede hacer que el algoritmo diverja, mientras que una demasiado pequeña puede ralentizar excesivamente la convergencia.
- **Convergencia:** Para funciones convexas, garantiza la convergencia al mínimo global con una tasa de aprendizaje adecuada. En funciones no convexas como la de este problema, converge a un mínimo local.

## 6.2. BFGS (Broyden–Fletcher–Goldfarb–Shanno)

El algoritmo BFGS es un **método cuasi-Newton**, considerado uno de los más eficientes para problemas de optimización sin restricciones. A diferencia del Descenso por Gradiente, que solo usa información de primer orden (el gradiente), los métodos cuasi-Newton utilizan una aproximación de la información de segundo orden (la matriz Hessiana).

La dirección de búsqueda  $\mathbf{p}_k$  en cada paso se determina resolviendo el sistema:

$$B_k \mathbf{p}_k = -\nabla f(\mathbf{x}_k)$$

Y la actualización de la solución es:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$$

Donde:

- $B_k$  es una aproximación de la matriz Hessiana en la iteración  $k$ . En lugar de calcular y almacenar la Hessiana (y su inversa), BFGS la actualiza de forma iterativa utilizando información del gradiente.
- $\alpha_k$  es el tamaño del paso, que generalmente se determina mediante una búsqueda por línea (line search) para satisfacer ciertas condiciones (como las de Wolfe).

#### Características:

- **Convergencia superlineal:** Es mucho más rápido que el Descenso por Gradiente, ya que incorpora una aproximación de la curvatura de la función.
- **Robustez:** Es menos sensible a la elección de hiperparámetros en comparación con el Descenso por Gradiente.
- **Complejidad:** Es más complejo de implementar, ya que requiere mantener y actualizar la matriz  $B_k$  (o su inversa). Sin embargo, para problemas de dimensiones moderadas, su eficiencia compensa con creces este costo.

## 7. Resultados experimentales

Se generaron 300 experimentos de forma aleatoria, con configuraciones dadas en función de los siguientes parámetros **learning\_rate**, **tolerance**, **max\_iterations** y **start\_point**. Donde:

$$\begin{aligned} 0,01 &\leq \text{learning\_rate} \leq 1 \\ 10^{-7} &\leq \text{tolerance} \leq 10^{-4} \\ 500 &\leq \text{max\_iterations} \leq 10000 \\ -100 &\leq \text{start\_point} \leq 100 \end{aligned}$$

### 7.1. Resumen de los resultados

Métrica	DG	BFGS
Tiempo promedio (s)	0.0715	0.0059
Tasa de Éxito (%)	13.0	94.3
Promedio de iteraciones	3611	9

## 8. Discusión y conclusiones

El análisis teórico confirma la solución analítica  $(1,0)$  y la unicidad del mínimo global, en concordancia con los resultados numéricos. Los experimentos muestran que BFGS es más eficiente y preciso, alcanzando el óptimo en muy pocas iteraciones y con gran robustez frente a los parámetros y puntos iniciales. El descenso por gradiente, aunque útil como referencia y para ilustrar la influencia de los hiperparámetros, requiere muchas más iteraciones y es más sensible a la tasa de aprendizaje y al punto de partida. La eficiencia de BFGS se debe principalmente a el hecho de que usa información de la matrix Hessiana para dar los pasos a diferencia de Descenso del Gradiente en cuyo caso es fija la longitud de los pasos. La región de atracción del óptimo es amplia, lo que facilita la convergencia desde distintos puntos iniciales en ambos métodos. En la mayoría de los experimentos el algoritmo de Descenso por Gradiente no llega a converger a solución, todo parece indicar que se debe a que requiere una gran cantidad de iteraciones para ello, y muchas veces el parametro **max iterations** tiene un valor demasiado bajo de forma que no llega a converger.

## 9. Comparativa de los métodos

Cuadro 1: Comparativa de alto nivel de los métodos.

Aspecto	BFGS	Descenso por Gradiente
Velocidad de conv.	Muy alta (pocas iteraciones)	Baja (muchas iteraciones)
Robustez	Poco sensible a parámetros	Sensible a la tasa de aprendizaje
Costo por iteración	Mayor	Menor
Memoria	$O(n^2)$ (despreciable para $n = 2$ )	$O(n)$

A continuación se muestra una tabla comparativa de los distintos tiempos de ejecución y cantidad de iteraciones de los algoritmos:

## 10. Archivos y reproducibilidad

- Script para generar experimentos: `Scripts/generate_experiments.py`

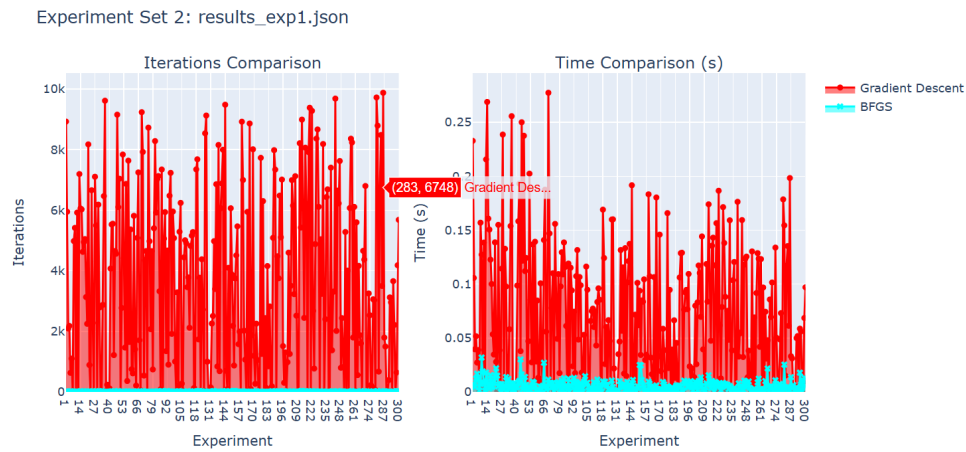


Figura 3: Gráfica comparativa de los tiempos de ejecución y cantidad de iteraciones de los algoritmos

- Configuraciones y Resultados de los experimentos: `results/results_exp1.json`
- Código fuente: `Implementación/Implementación_Métodos.ipynb`

Para reproducir los resultados, basta con abrir el notebook y ejecutar las celdas correspondientes a los experimentos. El archivo JSON `results_exp1.json` contiene tanto las configuraciones como los resultados de cada ejecución.