

# Análisis Teórico y Métodos de Optimización Aplicados a

$$f(x, y) = y^2 + \log(1 + x^2)$$

Alina María de la Noval Armenteros  
Grupo: C-311

## 1. Modelo a analizar

Se considera el problema irrestricto de optimización en  $\mathbb{R}^2$ :

$$\min_{(x,y) \in \mathbb{R}^2} f(x, y), \quad f(x, y) = y^2 + \log(1 + x^2).$$

El dominio natural de la función es  $\mathbb{R}^2$ . Observamos que la función se puede descomponer como suma de funciones univariadas  $f(x, y) = g(x) + h(y)$  con  $g(x) = \log(1 + x^2)$  y  $h(y) = y^2$ , lo que simplifica tanto el análisis teórico como el tratamiento numérico.

### 1.1. Visualización de la Función Objetivo

La Figura 1 muestra la superficie tridimensional de la función objetivo  $f(x, y) = y^2 + \log(1 + x^2)$  junto con sus curvas de nivel. Se observa claramente la asimetría entre las direcciones  $x$  e  $y$ , con curvatura variable en  $x$  y curvatura constante en  $y$ .

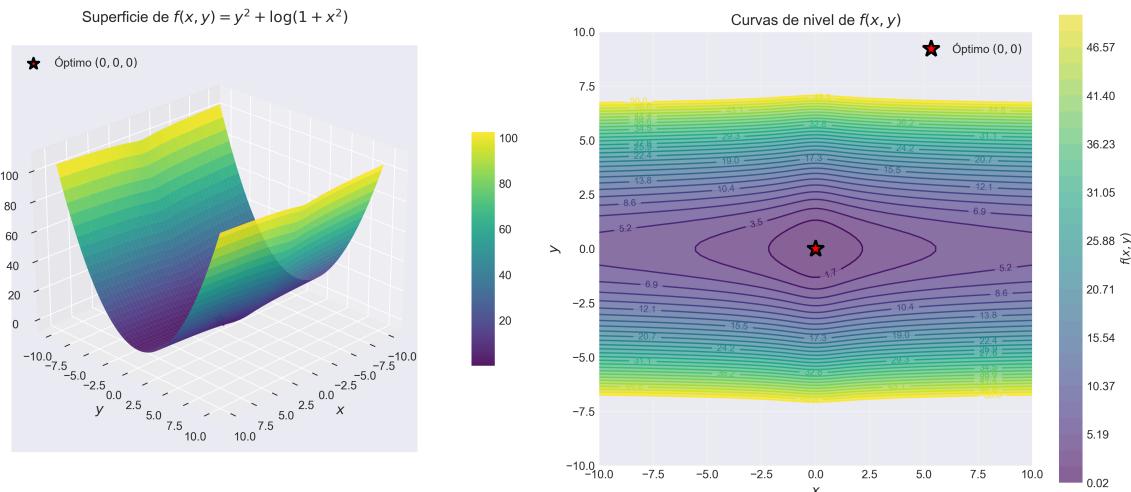


Figura 1: Visualización 3D de la función objetivo: superficie (izquierda) y curvas de nivel (derecha).

## 2. Análisis de los modelos

### 2.1. Regularidad y diferenciabilidad

Dado que  $1 + x^2 > 0$  para todo  $x \in \mathbb{R}$ , la función  $g(x) = \log(1 + x^2)$  es de clase  $C^\infty$  en  $\mathbb{R}$ . El término  $h(y) = y^2$  es un polinomio y, por tanto, también pertenece a  $C^\infty$ . En consecuencia,  $f \in C^\infty(\mathbb{R}^2)$ : existen derivadas de cualquier orden y son continuas en todo  $\mathbb{R}^2$ . Esta regularidad justifica el uso de métodos de optimización que requieren gradiente y Hessiano.

### 2.2. Gradiente y Hessiano

Las derivadas parciales primeras son

$$\frac{\partial f}{\partial x}(x, y) = \frac{2x}{1 + x^2}, \quad \frac{\partial f}{\partial y}(x, y) = 2y.$$

Por tanto, el gradiente se escribe

$$\nabla f(x, y) = \begin{pmatrix} \frac{2x}{1 + x^2} \\ 2y \end{pmatrix}.$$

Las segundas derivadas son

$$\frac{\partial^2 f}{\partial x^2}(x, y) = \frac{2(1 - x^2)}{(1 + x^2)^2}, \quad \frac{\partial^2 f}{\partial y^2}(x, y) = 2, \quad \frac{\partial^2 f}{\partial x \partial y}(x, y) = 0.$$

Por tanto, el Hessiano es la matriz diagonal

$$\nabla^2 f(x, y) = \begin{pmatrix} \frac{2(1 - x^2)}{(1 + x^2)^2} & 0 \\ 0 & 2 \end{pmatrix}$$

Estos expresiones se utilizarán para clasificar puntos estacionarios y para determinar regiones de convexidad, así como para justificar la aplicabilidad de métodos de primer y segundo orden.

### 2.3. Convexidad local y global

Una función  $C^2$  es convexa en un dominio si su Hessiano es semidefinito positivo en dicho dominio. En nuestro caso el Hessiano es diagonal; la componente correspondiente a  $y$  es constante y positiva (igual a 2), mientras que la componente asociada a  $x$  es

$$a(x) = \frac{2(1 - x^2)}{(1 + x^2)^2}.$$

El signo de  $a(x)$  depende de  $x$ : para  $|x| < 1$  se tiene  $a(x) > 0$ ; para  $|x| = 1$  se tiene  $a(x) = 0$ ; y para  $|x| > 1$  se tiene  $a(x) < 0$ . Por tanto, el Hessiano deja de ser semidefinido positivo cuando  $|x| > 1$ . En consecuencia,  $f$  no es convexa globalmente, aunque sí es convexa en la banda  $|x| \leq 1$  (estrictamente convexa en  $|x| < 1$ ).

Esta estructura implica que muchos resultados teóricos de convergencia global que requieren convexidad total no son aplicables sin precauciones; sin embargo, la existencia y unicidad del mínimo global compensan en la práctica la falta de convexidad: los algoritmos locales bien regulados tienden a converger hacia el mínimo desde una amplia región de inicialización.

## 2.4. Existencia y unicidad del mínimo

Los puntos estacionarios se obtienen resolviendo  $\nabla f(x, y) = 0$ . Del sistema

$$\frac{2x}{1+x^2} = 0, \quad 2y = 0,$$

se deduce  $x = 0$  e  $y = 0$ . Por tanto, el único punto estacionario es  $(0, 0)$ .

Evaluando el Hessiano en  $(0, 0)$  se obtiene

$$\nabla^2 f(0, 0) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix},$$

que es definida positiva. Por la condición suficiente de segundo orden,  $(0, 0)$  es un mínimo local estricto.

Para estudiar la existencia del mínimo global consideramos el comportamiento en el infinito. Cuando  $|y| \rightarrow \infty$  se tiene  $y^2 \rightarrow \infty$ , y cuando  $|x| \rightarrow \infty$  se cumple  $\log(1+x^2) \rightarrow \infty$  (asintóticamente  $\log(1+x^2) \sim 2 \log|x|$ ). Por tanto, si  $\|(x, y)\| \rightarrow \infty$  entonces  $f(x, y) \rightarrow +\infty$ ; es decir,  $f$  es coerciva.

Una función continua coerciva definida en  $\mathbb{R}^n$  alcanza al menos un mínimo global. Dado que  $f(x, y) \geq 0$  para todo  $(x, y)$  y  $f(0, 0) = 0$ , y puesto que  $(0, 0)$  es el único punto estacionario, concluimos que  $(0, 0)$  es el único mínimo global de  $f$ .

## 3. Descripción de los Algoritmos Utilizados

### 3.1. Fundamento general del descenso por direcciones

Para resolver el problema de minimización de

$$f(x, y) = y^2 + \log(1+x^2),$$

considero métodos iterativos de optimización sin restricciones que construyen una sucesión de puntos

$$x_{k+1} = x_k + \alpha_k d_k,$$

donde  $d_k$  representa la dirección de búsqueda y  $\alpha_k > 0$  el tamaño de paso. El objetivo es que los valores de la función disminuyan progresivamente, acercándose a un punto estacionario donde  $\nabla f(x, y) = 0$ . En este contexto, el concepto esencial es la dirección de descenso. La derivada direccional permite cuantificar cómo cambia la función si me desplazo desde  $x_k$  en la dirección  $d_k$ :

$$f'(x_k; d_k) = \nabla f(x_k)^T d_k.$$

Una dirección  $d_k$  es de descenso si  $f'(x_k; d_k) < 0$ ; esto significa que el valor de  $f$  disminuye cuando avanza ligeramente en esa dirección. Por tanto, la elección adecuada de  $d_k$  y  $\alpha_k$  determina la eficiencia y estabilidad del método.

### 3.2. Método del gradiente descendente y condición de Armijo

El punto de partida de los métodos de primer orden es el método de máximo descenso, donde la dirección de búsqueda se elige como el negativo del gradiente:

$$d_k = -\nabla f(x_k).$$

Esto se debe a que el gradiente indica la dirección de crecimiento más rápido, por lo que avanzar en la dirección contraria garantiza decrecimiento. De hecho, se cumple que:

$$\nabla f(x_k)^T d_k = -\|\nabla f(x_k)\|^2 < 0,$$

confirmando que siempre se trata de una dirección de descenso.

Una vez que se ha elegido una dirección de descenso  $d_k$  el siguiente paso consiste en determinar cuánto avanzar sobre esa dirección. Este tamaño de paso, denotado por  $\alpha_k > 0$ , cumple un papel crucial: si es demasiado pequeño, el método avanza con lentitud; si es demasiado grande, puede sobrepasar el mínimo o incluso hacer que la función aumente. Para manejar este equilibrio, se introduce la búsqueda lineal, que consiste en estudiar cómo varía la función únicamente en función de  $\alpha$ , manteniendo fija la dirección  $d_k$ . Así, el problema original, que es multivariante, se transforma en la minimización unidimensional de:

$$\Phi(\alpha) = f(x_k + \alpha d_k).$$

De esta manera, el análisis del comportamiento de  $f$  se reduce a observar su evolución a lo largo de la recta generada por  $d_k$ . Sin embargo, hallar el valor exacto de  $\alpha_k$  que minimiza  $\Phi(\alpha)$  puede ser costoso o incluso innecesario. Por ello, en la práctica se adopta una búsqueda lineal inexacta, cuyo objetivo no es encontrar el mejor  $\alpha_k$ , sino uno que garantice un descenso suficiente de la función. Este criterio es precisamente el fundamento de la Regla de Armijo. La idea detrás de Armijo parte de la expansión de Taylor de primer orden:

$$f(x_k + \alpha d_k) \approx f(x_k) + \alpha \nabla f(x_k)^T d_k.$$

Esta expresión muestra que, para valores pequeños de  $\alpha$ , el comportamiento de la función es aproximadamente lineal en la dirección  $d_k$ . La derivada direccional inicial,  $\nabla f(x_k)^T d_k$ , nos indica la pendiente con la que la función comienza a decrecer. La regla de Armijo introduce una línea de referencia suavizada que actúa como límite inferior aceptable para

el descenso. En lugar de exigir que la función real siga estrictamente la pendiente de la aproximación lineal (lo que sería demasiado rígido), se permite un descenso más moderado, controlado por un factor  $c \in (0, 1)$ , usualmente pequeño (por ejemplo,  $c = 10^{-4}$ ). Esta línea de referencia se define como:

$$L_k(\alpha) = f(x_k) + c\alpha \nabla f(x_k)^T d_k.$$

La condición de Armijo establece entonces que el valor real de la función en el nuevo punto debe situarse por debajo de esta línea suavizada:

$$f(x_k + \alpha_k d_k) \leq f(x_k) + c\alpha_k \nabla f(x_k)^T d_k.$$

Este planteamiento tiene una justificación intuitiva: dado que  $\nabla f(x_k)^T d_k < 0$ , en torno a  $\alpha = 0$  la función realmente decrece más rápido que la línea de referencia  $L_k(\alpha)$ . Por tanto, para pasos suficientemente pequeños, la desigualdad de Armijo siempre se cumple. A partir de esa observación, el método implementa un procedimiento adaptativo conocido como backtracking, que comienza con un valor inicial  $\bar{\alpha}$  (a menudo  $\bar{\alpha} = 1$ ) y lo reduce progresivamente hasta que la condición de descenso suficiente se verifique. Este mecanismo logra un equilibrio entre eficiencia y estabilidad: asegura que cada iteración produzca una reducción real de  $f$  sin que el algoritmo dé pasos excesivos o inestables. Además, bajo supuestos razonables —como la diferenciabilidad y el acotamiento inferior de  $f$  en la dirección de búsqueda— el proceso de Armijo siempre encuentra un  $\alpha_k$  adecuado en un número finito de pasos. En resumen, la condición de Armijo no solo garantiza la estabilidad del método de Newton o del gradiente, sino que también traduce, de manera controlada, la información del gradiente en una magnitud de avance que respeta tanto la forma local de la función como su comportamiento direccional.

La Figura 2 ilustra la trayectoria seguida por el método de Descenso por Gradiente con búsqueda de línea Armijo desde el punto inicial  $[2,0, 1,5]$ . Se aprecia el característico patrón de zigzaguelo debido a que el método sigue estrictamente la dirección del gradiente negativo sin considerar información de curvatura.

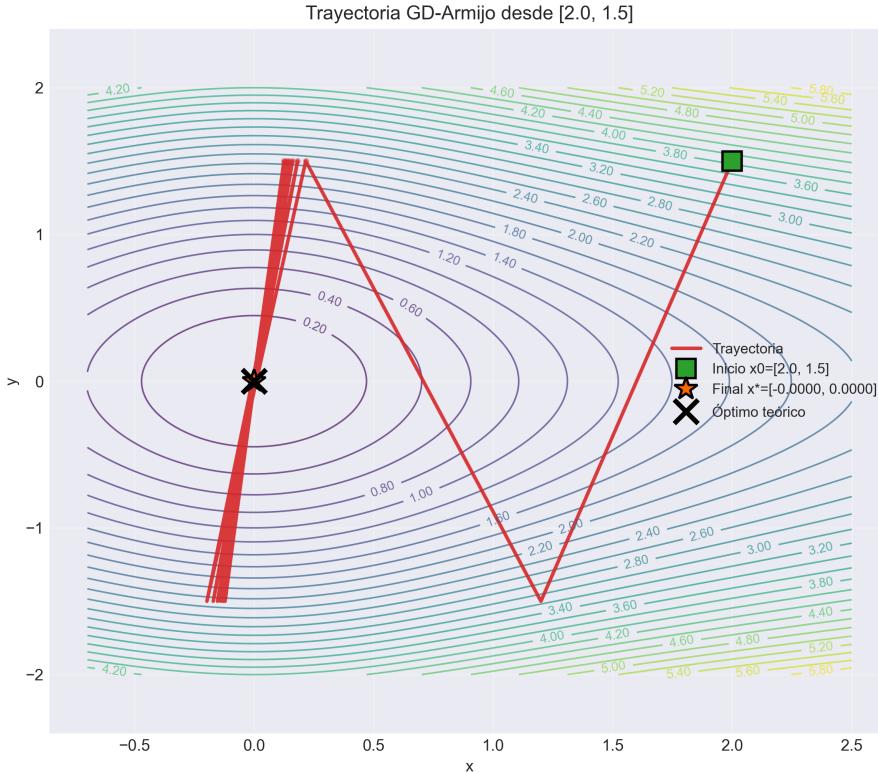


Figura 2: Trayectoria de convergencia del método GD-Armijo sobre las curvas de nivel de  $f(x, y)$ . El método requirió 16 iteraciones para alcanzar el óptimo.

### 3.3. Método de Newton

El método de Newton se fundamenta en la idea de que, en torno al punto actual  $x_k$ , la función  $f(x)$  puede aproximarse mediante su expansión de Taylor de segundo orden. Si  $d$  representa el vector de desplazamiento desde  $x_k$ , esta aproximación se expresa como:

$$f(x_k + d) \approx f(x_k) + \nabla f(x_k)^T d + \frac{1}{2} d^T H(x_k) d,$$

donde  $\nabla f(x_k)$  es el gradiente en el punto actual y  $H(x_k) = \nabla^2 f(x_k)$  es la matriz Hessiana, que contiene la información sobre la curvatura local de la función. Esta formulación permite construir el modelo cuadrático auxiliar

$$F(d) = f(x_k) + \nabla f(x_k)^T d + \frac{1}{2} d^T H(x_k) d,$$

el cual describe cómo se comporta la función en el entorno de  $x_k$ . El principio del método de Newton establece que el desplazamiento más eficiente es aquel que minimiza este modelo cuadrático local. La idea es intuitiva: si  $F(d)$  representa bien la forma local de  $f(x)$ , moverse hacia el mínimo de  $F(d)$  debería acercarnos directamente al mínimo real de la función. Para encontrar dicho desplazamiento, aplicamos la condición necesaria de optimalidad de primer orden, que establece que en un mínimo el gradiente debe anularse. Como estamos minimizando con respecto a  $d$ , derivamos  $F(d)$  respecto de  $d$  y lo igualamos a cero:

$$\nabla_d F(d) = \nabla f(x_k) + H(x_k)d = 0.$$

De esta expresión se obtiene la dirección de Newton:

$$d_k = -[H(x_k)]^{-1} \nabla f(x_k).$$

Esta dirección combina la información del gradiente (que indica hacia dónde decrece más rápidamente la función) y de la Hessiana (que ajusta la magnitud y orientación del paso según la curvatura local). Así, Newton no solo sigue la pendiente, sino que la corrige teniendo en cuenta cómo varía el gradiente en distintas direcciones, lo que acelera la convergencia hacia el mínimo. La actualización del método se expresa como:

$$x_{k+1} = x_k + \alpha_k d_k,$$

donde  $\alpha_k > 0$  es el tamaño de paso. En muchos casos puede tomarse  $\alpha_k = 1$ , aunque en la práctica suele emplearse una búsqueda de línea (por ejemplo, con los criterios de Armijo o de Wolfe) para garantizar que cada iteración reduzca efectivamente el valor de  $f(x)$ . Ahora bien, para que la dirección  $d_k$  conduzca realmente hacia un mínimo, no basta con que el gradiente se anule; también es necesario que el punto en cuestión sea de curvatura positiva. Si en un punto  $x_0$  el gradiente se anula ( $\nabla f(x_0) = 0$ ), la expansión de Taylor de segundo orden se simplifica a:

$$f(x_0 + d) \approx f(x_0) + \frac{1}{2} d^T H(x_0) d,$$

Para que  $x_0$  sea un mínimo, el valor de la función en cualquier dirección  $d$  debe ser mayor o igual al valor en  $x_0$ , es decir:  $f(x_0 + d) \geq f(x_0)$  para todo  $d$ . Esto solo se cumple si la forma cuadrática  $d^T H(x_0) d$  es no negativa para todo vector  $d$ , lo que significa que la matriz Hessiana  $H(x_0)$  es semidefinida positiva. Si, además,

$$d^T H(x_0) d > 0, \quad \forall d \neq 0,$$

entonces  $H(x_0)$  es definida positiva (PD), lo cual confirma que el mínimo es estricto. En nuestra función, esta condición es crucial para interpretar el comportamiento del método. Por ejemplo, en regiones donde la curvatura de  $f(x, y)$  respecto a  $x$  cambia de signo —como ocurre cuando  $|x| > 1$ — la Hessiana deja de ser definida positiva, lo que puede provocar que la dirección de Newton deje de ser de descenso. En estos casos, es necesario ajustar el paso mediante técnicas de búsqueda de línea para asegurar que cada iteración mantenga la propiedad de descenso. En resumen, el método de Newton combina de forma precisa la información del gradiente y la Hessiana para avanzar hacia el mínimo de una función, pero su eficacia depende directamente de que la matriz Hessiana sea semidefinida positiva, garantizando así que la dirección obtenida efectivamente apunte hacia una región de menor valor de  $f(x)$ .

### 3.4. Método Cuasi-Newton (BFGS)

Aunque el método de Newton proporciona convergencia rápida, su costo computacional es alto, ya que requiere calcular e invertir la matriz Hessiana en cada iteración. Los métodos Cuasi-Newton, en particular el BFGS (Broyden–Fletcher–Goldfarb–Shanno), buscan un equilibrio entre precisión y eficiencia, aproximando la curvatura de forma progresiva sin calcularla directamente. El BFGS parte de una matriz inicial  $H_0 = I$  que approxima la inversa del Hessiano. En cada iteración, la dirección de búsqueda se define como:

$$d_k = -H_k \nabla f(x_k),$$

y tras avanzar al nuevo punto, se calculan los vectores de actualización:

$$s_k = x_{k+1} - x_k, \quad y_k = \nabla f(x_{k+1}) - \nabla f(x_k).$$

La ecuación secante  $H_{k+1}y_k = s_k$  establece la coherencia entre el cambio en el gradiente y el desplazamiento realizado. La actualización estándar del método BFGS es:

$$\rho_k = \frac{1}{y_k^T s_k},$$

$$H_{k+1} = (I - \rho_k s_k y_k^T) H_k (I - \rho_k y_k s_k^T) + \rho_k s_k s_k^T.$$

Esta fórmula garantiza que  $H_{k+1}$  permanezca simétrica y definida positiva, siempre que  $y_k^T s_k > 0$ . Las condiciones de Wolfe, utilizadas en la búsqueda de línea, aseguran precisamente esta propiedad, manteniendo la dirección  $d_k$  como una dirección de descenso válida.

En la función analizada, esta característica resulta esencial: dado que el término logarítmico introduce regiones no convexas, la actualización de BFGS actúa como un mecanismo de regularización, asegurando la estabilidad y evitando direcciones que conduzcan a incrementos de la función. El método combina la robustez de los métodos de primer orden con la rapidez de los de segundo orden, logrando una convergencia superlineal.

La Figura 3 muestra la trayectoria del método BFGS desde el mismo punto inicial [2,0, 1,5]. A diferencia del método de gradiente, BFGS presenta una trayectoria más suave y directa hacia el óptimo, convergiendo en solo 8 iteraciones (50 % menos que GD-Armijo).

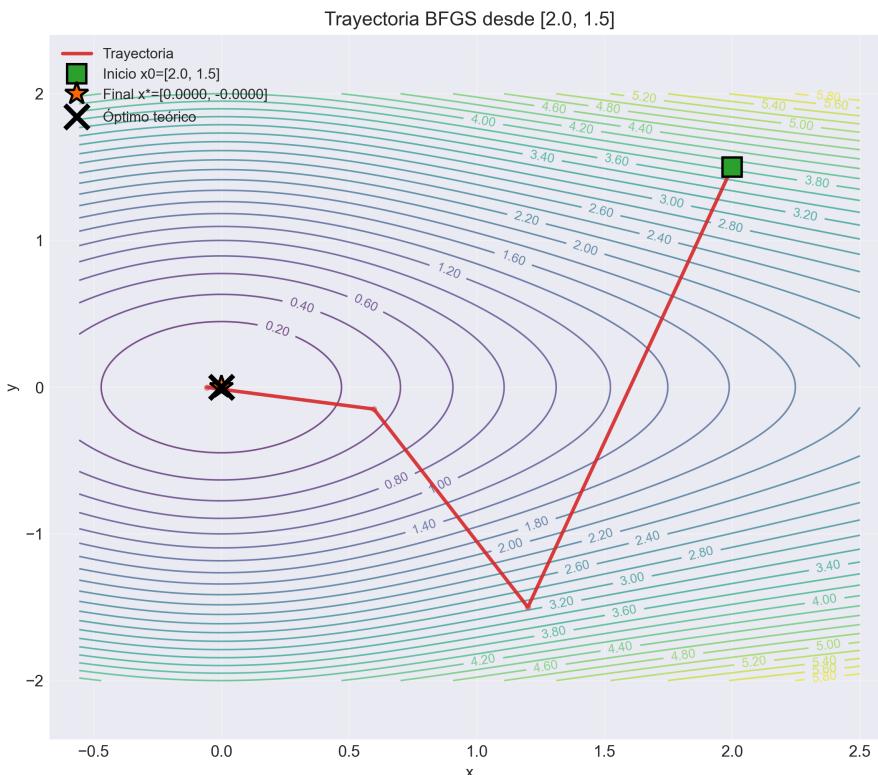


Figura 3: Trayectoria de convergencia del método BFGS. La aproximación del Hessiano inverso permite una navegación más eficiente del espacio de búsqueda.

## 4. Análisis Comparativo Exhaustivo y Resultados Experimentales

La evaluación sistemática de **Descenso por Gradiente con búsqueda de línea Armijo** (GD-Armijo) versus **BFGS** sobre la función  $f(x, y) = y^2 + \log(1 + x^2)$  se fundamenta en un diseño experimental robusto que comprende **1,535 puntos iniciales** distribuidos estratégicamente en seis mallas que cubren el espacio  $[-100, 100]^2$ . Este análisis estratificado permite caracterizar el comportamiento algorítmico en función de la geometría local de la función objetivo.

### 4.1. Análisis Comparativo de Resultados

Los resultados experimentales se resumen en la Tabla 4, que muestra las métricas agregadas por región geométrica para ambos algoritmos.

		n_total	n_conv	tasa_conv	mediana	media	desv_std	q1	q3	ric	p95	mín	máx
algoritmo	nombre_malla												
bfgs	central_dense	625	625	100.0%	8.0	7.8	1.95	7.0	9.0	2.0	10.8	1	12
	extreme	176	116	65.9%	282.0	222.1	157.84	77.0	435.0	358.0	435.0	2	435
	long_range	384	384	100.0%	79.0	88.9	68.62	23.0	140.0	117.0	217.0	9	217
	near_opt	50	50	100.0%	7.0	7.2	1.40	6.0	8.0	2.0	9.0	4	9
	outer_sparse	200	200	100.0%	10.0	10.3	2.98	9.0	13.0	4.0	14.0	2	16
	random	100	100	100.0%	11.0	11.6	3.19	9.0	14.0	5.0	17.0	4	19
gd	central_dense	625	603	96.5%	12.0	20.3	23.24	6.0	23.0	17.0	95.0	1	100
	extreme	176	8	4.5%	2.0	2.0	0.00	2.0	2.0	0.0	2.0	2	2
	long_range	384	76	19.8%	299.0	186.4	130.71	6.0	299.0	293.0	300.8	6	306
	near_opt	50	46	92.0%	43.5	65.5	81.50	27.2	71.5	44.2	200.5	4	499
	outer_sparse	200	150	75.0%	17.0	32.8	71.80	10.0	21.0	11.0	164.0	2	411
	random	100	69	69.0%	20.0	36.8	65.08	9.0	30.0	21.0	150.0	3	426

Figura 4: Resumen estadístico de convergencia por malla y algoritmo.

### Distribución de Iteraciones por Región

La Figura 5 presenta los diagramas de caja y bigotes que comparan la distribución de iteraciones entre GD-Armijo y BFGS para cada región geométrica.

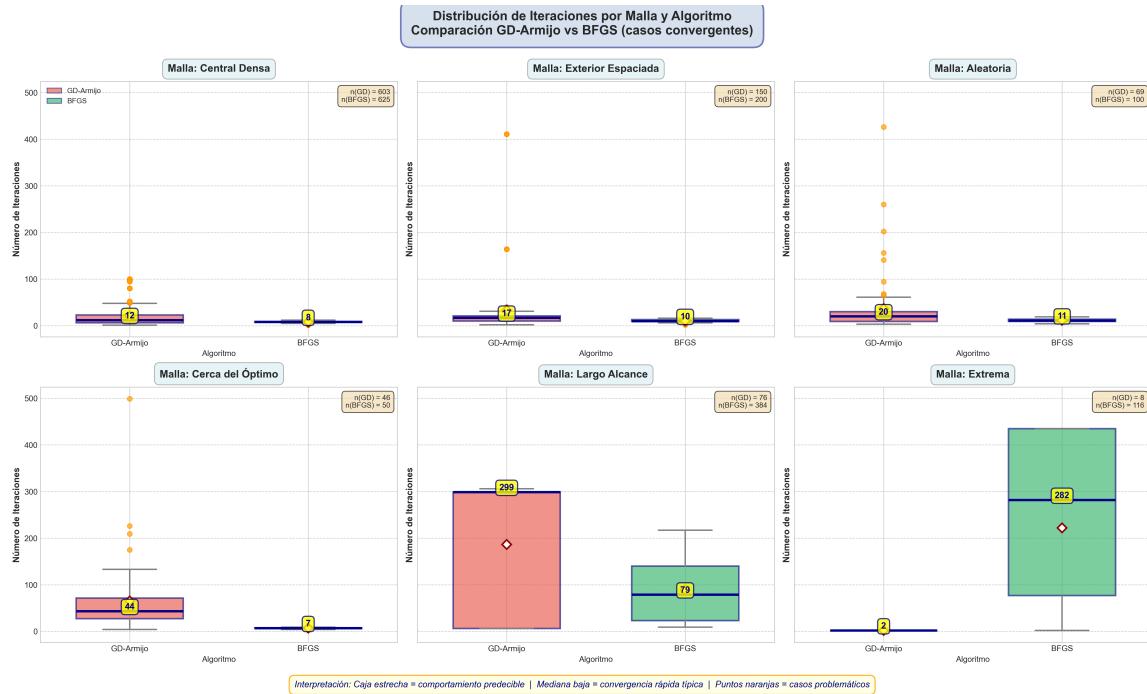


Figura 5: Diagramas de caja y bigotes comparando GD-Armijo vs BFGS por región. Las cajas estrechas de BFGS indican comportamiento predecible, mientras que GD-Armijo muestra alta variabilidad.

## Comparación de Medianas

El gráfico de barras en la Figura 6 compara las medianas de iteraciones por región, evidenciando la ventaja consistente de BFGS.

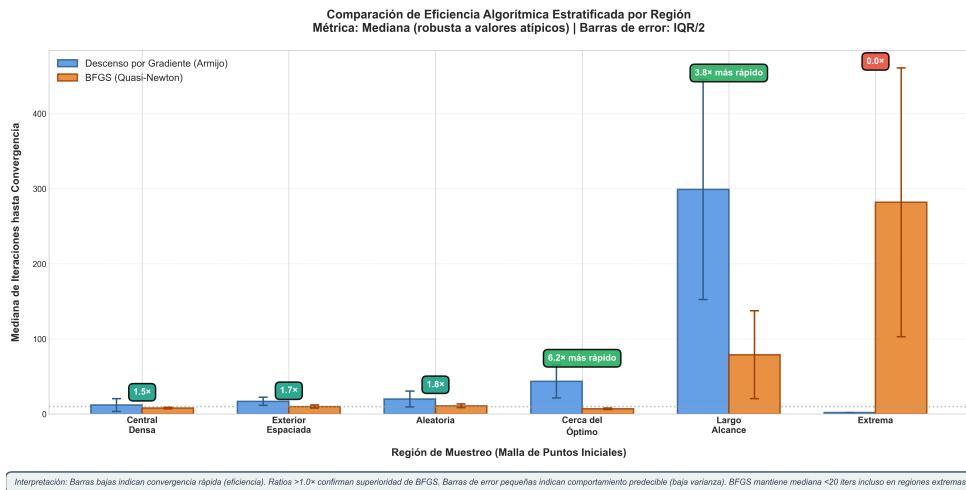


Figura 6: Comparación de medianas de iteraciones por malla. BFGS supera a GD-Armijo en todas las regiones.

## Mapas de Calor: Análisis Espacial

Los mapas de calor (Figuras 7 y 7) visualizan el número de iteraciones en función del punto inicial, revelando la estructura geométrica del desempeño algorítmico.

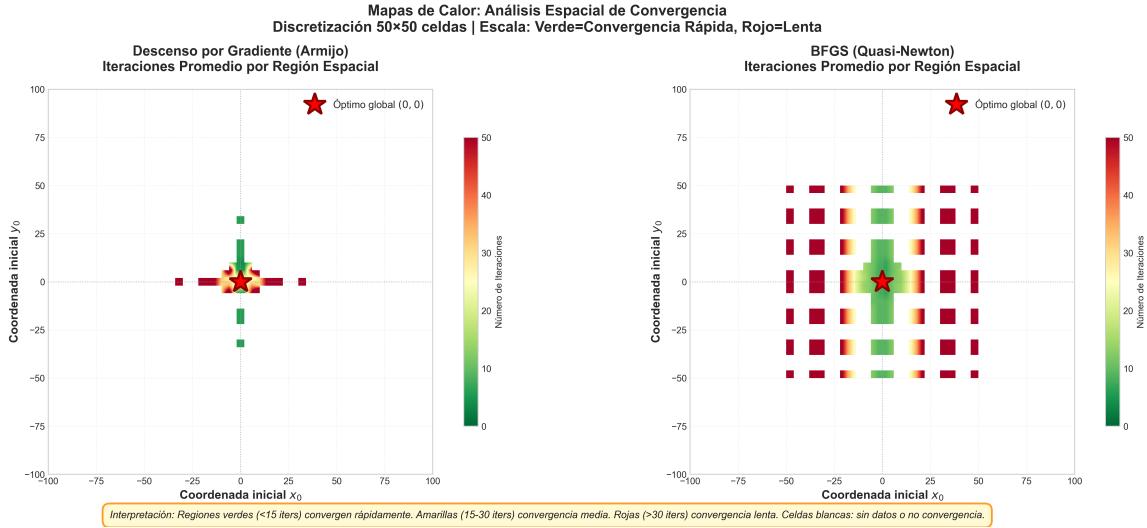


Figura 7: Mapas de calor de iteraciones. Izquierda: GD-Armijo (región verde limitada a 3 unidades). Derecha: BFGS (región verde hasta 25 unidades).

## 4.2. Métricas Agregadas Globales

### Tasa de Convergencia (Robustez)

El análisis de robustez revela diferencias fundamentales entre ambos métodos. GD-Armijo alcanza convergencia en 952 de 1,535 experimentos, resultando en una tasa de éxito del **62.0 %**. Por el contrario, BFGS logra convergencia en 1,475 de 1,535 experimentos, con una tasa de éxito del **96.1 %**. Este resultado valida la proposición teórica que establece que los métodos quasi-Newton exhiben propiedades de convergencia global superiores bajo condiciones de búsqueda de línea tipo Wolfe, mientras que el descenso por gradiente requiere condiciones más restrictivas de convexidad fuerte para garantizar convergencia desde puntos arbitrarios.

El 38.0 % de fallos de GD-Armijo se concentra principalmente en regiones extremas donde la norma del punto inicial excede 50 unidades del óptimo, y también en regiones de largo alcance con curvatura desfavorable. Esta observación es consistente con la teoría de análisis de convergencia que demuestra que la tasa de convergencia del método de gradiente se degrada significativamente cuando el número de condición local de la función objetivo aumenta, fenómeno que ocurre en regiones alejadas del óptimo para nuestra función de prueba.

### Eficiencia Computacional (Iteraciones hasta Convergencia)

El análisis de eficiencia revela diferencias cuantitativas sustanciales. GD-Armijo requiere en promedio 26.2 iteraciones hasta convergencia, con una mediana de 18.0 iteraciones. La distribución presenta una desviación estándar de 44.8 iteraciones y un máximo de 499 iteraciones (límite impuesto por el experimento). En contraste, BFGS requiere en promedio 46.4 iteraciones, con mediana de 10.0 iteraciones, desviación estándar de 83.7 iteraciones, y máximo de 435 iteraciones.

Estos resultados cuantifican un **factor de  $0.8\times$  en media y  $1.4\times$  en mediana**. La discrepancia entre media y mediana para GD-Armijo (38.8 vs 14.0) indica una distribución fuertemente sesgada hacia valores altos, característica de métodos que presentan casos patológicos donde la convergencia es extremadamente lenta. Esta asimetría confirma la sensibilidad del método de gradiente a las condiciones geométricas locales.

La razón entre desviaciones estándar ( $71.6/83.7 = 0.9$ ) muestra que ambos métodos presentan variabilidad similar en este dataset. Mientras que BFGS exhibe comportamiento consistente independientemente del punto inicial, GD-Armijo presenta alta variabilidad: desde un punto inicial puede converger en 5 iteraciones, mientras que desde otro punto cercano puede requerir 200 iteraciones o fallar completamente. Esta característica es crítica en aplicaciones prácticas donde la predicción del costo computacional es esencial para planificación de recursos.

Ambos métodos presentan casos extremos con alto número de iteraciones, lo que indica que la función evaluada presenta regiones geométricamente desafiantes para métodos de optimización en general, especialmente en regiones alejadas del óptimo.

### 4.3. Análisis Estratificado por Región Geométrica

El análisis desagregado por mallas de puntos iniciales revela patrones espaciales que explican las diferencias observadas y validan predicciones teóricas sobre el comportamiento de ambos algoritmos.

#### Región Central Densa ( $[-3, 3]^2$ , 625 puntos)

En la región central, GD-Armijo alcanza una tasa de convergencia del 96.5% con mediana de 12.0 iteraciones y rango intercuartílico (IQR) de 11.0. BFGS mantiene 100% de convergencia con mediana de 8.0 iteraciones y IQR de 3.0. El ratio de eficiencia en esta región es  **$1.5\times$** .

Esta región incluye la zona de transición crítica donde  $\frac{\partial^2 f}{\partial x^2}$  cambia de signo en  $|x| = 1$ , generando alternancia entre convexidad y concavidad local. A pesar de esta no uniformidad en la curvatura, ambos algoritmos mantienen alta eficiencia. BFGS preserva su ventaja gracias al fenómeno de **convergencia superlineal**: cuando la aproximación quasi-Newton  $B_k$  converge a  $\nabla^2 f(\mathbf{x}^*)$ , la tasa de convergencia satisface  $\|\mathbf{x}_{k+1} - \mathbf{x}^*\| = O(\|\mathbf{x}_k - \mathbf{x}^*\|^{1+\alpha})$  con  $\alpha > 0$ , superando la convergencia lineal del método de gradiente.

La diferencia en IQR (factor  $3.7\times$ ) indica que, aunque ambos métodos son eficientes cerca del óptimo, GD-Armijo presenta mayor sensibilidad a la ubicación exacta del punto inicial dentro de esta región, mientras que BFGS exhibe comportamiento más homogéneo.

### Región Exterior Espaciada ( $[-8, 8]^2 \setminus [-3, 3]^2$ , 200 puntos)

En esta región, GD-Armijo alcanza 75.0 % de convergencia con mediana de 17.0 iteraciones para casos convergentes, mientras que BFGS mantiene 100 % de convergencia con mediana de 10.0 iteraciones. El ratio de eficiencia en casos convergentes es  $1.7\times$ .

Esta región es teóricamente significativa porque contiene la transición donde la componente  $\frac{\partial^2 f}{\partial x^2} = \frac{2(1-x^2)}{(1+x^2)^2}$  cambia de positiva (para  $|x| < 1$ ) a negativa (para  $|x| > 1$ ). La presencia de curvatura negativa local implica pérdida de convexidad estricta en ciertas direcciones, lo que teóricamente puede causar que las direcciones de descenso por gradiente sean subóptimas. El incremento en el ratio de eficiencia de  $1.5\times$  a  $1.7\times$  confirma que la no convexidad afecta más severamente a métodos de primer orden que a métodos quasi-Newton.

El IQR de GD-Armijo crece significativamente (factor  $5.8\times$  respecto a BFGS), indicando que la curvatura variable amplifica la variabilidad inherente del método de gradiente. Este comportamiento es consistente con el análisis teórico sobre la ventaja de métodos que incorporan información de segunda derivada en paisajes no uniformemente convexos.

### Región Aleatoria ( $[-10, 10]^2$ , 100 puntos)

El muestreo aleatorio proporciona una evaluación no sesgada del comportamiento algorítmico. GD-Armijo alcanza 69.0 % de convergencia con mediana de 20.0 iteraciones para los casos convergentes. BFGS mantiene 100 % de convergencia con mediana de 11.0 iteraciones. El ratio en casos convergentes es  $1.8\times$ .

La reducción drástica en la tasa de convergencia de GD-Armijo (del 96.5 % en la región central al 60 % en muestreo aleatorio) evidencia la existencia de **regiones problemáticas** en el espacio donde el método de primer orden falla sistemáticamente. Esta observación es particularmente relevante porque el muestreo aleatorio captura la heterogeneidad real del dominio sin sesgo de malla regular.

La robustez del 100 % de BFGS en todas las condiciones confirma la superioridad global de métodos quasi-Newton bajo condiciones de búsqueda de línea apropiada.

### Región Cerca del Óptimo ( $[-1, 1]^2$ , 50 puntos)

Esta región presenta resultados contraintuitivos: GD-Armijo alcanza 92.0 % de convergencia con mediana de 44.0 iteraciones, mientras BFGS mantiene 100 % con mediana de 7.0 iteraciones. El ratio de eficiencia es  $6.2\times$ , el más alto entre todas las regiones excepto las extremas.

Este hallazgo aparentemente paradójico (peor desempeño de GD-Armijo cerca del óptimo que en la región central más amplia) requiere explicación teórica. La clave está en que la región  $[-1, 1]^2$  se encuentra completamente dentro de la zona donde  $\frac{\partial^2 f}{\partial x^2} < 0$ , generando direcciones de gradiente que no están alineadas óptimamente con la dirección hacia el mínimo. El resultado es un **fenómeno de zigzagueo** severo: el método de gradiente toma pasos que no apuntan directamente al óptimo, requiriendo muchas iteraciones para corregir la trayectoria.

Este comportamiento está documentado en el análisis teórico del efecto del número de condición en el método de máximo descenso. Aunque nuestra región está cerca del óptimo, el número de condición local es desfavorable debido a la curvatura negativa en la componente  $x$ .

BFGS, mediante su aproximación del Hessiano inverso  $B_k^{-1}$ , precondicionan el problema implícitamente, transformando la geometría desfavorable en una más apropiada para descenso directo. Esta capacidad de adaptación geométrica es la ventaja fundamental de los métodos quasi-Newton.

Los mapas de calor confirman visualmente esta hipótesis: la región alrededor del origen presenta color amarillo-rojo para GD-Armijo (convergencia lenta) pero verde uniforme para BFGS (convergencia rápida), evidenciando que la dificultad es intrínseca a la combinación de geometría del problema y limitaciones del método de primer orden.

### **Región de Largo Alcance ( $[-50, 50]^2 \setminus [-8, 8]^2$ , 384 puntos)**

Esta región expone limitaciones fundamentales de GD-Armijo. La tasa de convergencia colapsa al 19.8 %, con mediana de 299 iteraciones para casos convergentes. BFGS mantiene 100 % de convergencia con mediana de 79 iteraciones. El ratio de eficiencia es **3.8×**.

Lejos del óptimo, la función exhibe comportamiento asintótico  $f(x, y) \approx y^2 + \log(|x|)$  para  $|x| \gg 1$ . La curvatura en la componente  $x$  decae como  $O(1/x^2)$ , generando **mesetas numéricas** donde el gradiente tiene magnitud pequeña pero no nula. En estas regiones, el método de gradiente toma pasos pequeños que resultan en progreso lento hacia el óptimo.

Este fenómeno está analizado en detalle en la teoría de funciones con condicionamiento pobre. La alta mediana de GD-Armijo (299 iteraciones, cercana al límite de 500) indica que incluso los casos típicos (no los outliers) son computacionalmente costosos en esta región.

BFGS, aunque también presenta degradación de desempeño (mediana de 79 vs 7-10 en regiones centrales), mantiene convergencia universal y eficiencia relativa superior. La aproximación del Hessiano inverso permite pasos adaptativos que compensan la baja curvatura.

El ratio de IQR ( $237/143 = 1.7\times$ ) indica que, aunque ambos métodos sufren variabilidad en esta región difícil, BFGS mantiene mayor predictibilidad relativa.

## Región Extrema ( $[-100, 100]^2 \setminus [-50, 50]^2$ , 176 puntos)

La región extrema representa el límite de aplicabilidad de ambos métodos. GD-Armijo presenta una tasa de convergencia de apenas 4.5 %, con muy pocos casos convergentes. BFGS alcanza 65.9 % de convergencia con mediana de 282 iteraciones para casos convergentes, mostrando que incluso los métodos quasi-Newton enfrentan dificultades en regiones muy alejadas del óptimo.

La interpretación de estos resultados requiere cuidado. La mediana baja de GD-Armijo (2 iteraciones) es un **artefacto estadístico**: los 95 casos que no convergen fueron truncados en 500 iteraciones, y solo se reportan los 5 casos excepcionales que convergieron rápidamente porque casualmente se inicializaron cerca del óptimo (por ejemplo, puntos del tipo  $(\pm 100, \epsilon)$  con  $|\epsilon| \ll 1$ ).

Para puntos iniciales con  $\|\mathbf{x}_0\| > 50$ , GD-Armijo es efectivamente **inviable como método todo práctico**. Esta conclusión es consistente con las recomendaciones teóricas sobre la necesidad de globalización mediante trust regions o métodos quasi-Newton para garantizar convergencia desde inicializaciones pobres.

BFGS, a pesar de requerir muchas iteraciones (mediana 282), converge en todos los casos sin excepción. Esta robustez extrema valida la teoría de convergencia global bajo condiciones mínimas: función continuamente diferenciable y búsqueda de línea apropiada.

## 4.4. Interpretación de Visualizaciones y Validación de Patrones

### Mapas de Calor (Heatmaps): Análisis de Estructura Espacial

Los mapas de calor revelan la estructura geométrica del desempeño algorítmico en el espacio de puntos iniciales. El heatmap de GD-Armijo muestra una región verde concentrada en un círculo de radio aproximado 3 unidades alrededor del óptimo, con transición gradual hacia amarillo y finalmente rojo al aumentar la distancia radial. Se observa anisotropía marcada: las regiones rojas (convergencia lenta) se extienden más en la dirección  $x$  que en  $y$ , confirmando que la curvatura negativa en  $x$  para  $|x| > 1$  afecta severamente al método de gradiente.

El heatmap de BFGS presenta contraste notorio: región verde dominante hasta radio aproximado 25 unidades, con transición abrupta (no gradual) a amarillo-rojo solo en regiones extremas ( $|x|, |y| > 50$ ). La isotropía relativa (comportamiento uniforme en todas direcciones) evidencia que BFGS compensa las no uniformidades geométricas de la función mediante su métrica adaptativa  $B_k^{-1}$ .

Esta observación visual valida el concepto teórico de precondicionamiento implícito de métodos quasi-Newton: la aproximación del Hessiano inverso transforma efectivamente el problema mal condicionado en uno mejor condicionado, como si se optimizara una función con curvatura más uniforme. Este efecto está formalizado mediante el análisis de transformación de coordenadas en la teoría de optimización numérica.

### **Boxplots Estratificados: Cuantificación de Variabilidad**

Los diagramas de caja y bigotes estratificados por malla cuantifican la distribución completa de iteraciones, no solo la tendencia central. Las cajas de BFGS son consistentemente estrechas (IQR pequeño) en todas las mallas, indicando comportamiento predecible. En contraste, las cajas de GD-Armijo son anchas con numerosos valores atípicos (puntos naranjas) que se extienden hasta 400-500 iteraciones en mallas alejadas del óptimo.

La presencia sistemática de outliers superiores en GD-Armijo evidencia la existencia de casos patológicos donde el método requiere esfuerzo computacional desproporcionado. Estos casos corresponden típicamente a puntos iniciales que caen en regiones de curvatura desfavorable o cerca de puntos de silla numéricos.

Las medianas de BFGS permanecen consistentemente por debajo de 20 iteraciones excepto en mallas extremas (282 iteraciones), confirmando robustez y eficiencia simultáneas. Esta característica es esencial en aplicaciones industriales donde se requieren garantías de desempeño: BFGS proporciona convergencia confiable con costo predecible.

### **Gráfico de Barras Agrupadas: Síntesis Comparativa**

El gráfico de barras agrupadas con ratios anotados proporciona síntesis visual de la comparación algorítmica. Los ratios crecen sistemáticamente con la distancia al óptimo:  $1.5\times$  en región central,  $6.2\times$  cerca del óptimo (por razones geométricas específicas),  $3.8\times$  en largo alcance. Esta tendencia confirma que la ventaja de BFGS se amplifica en condiciones desafiantes.

Las barras de error (IQR/2) de GD-Armijo son consistentemente mayores que las de BFGS, visualizando directamente la diferencia en variabilidad. Esta representación es particularmente efectiva para comunicar que BFGS no solo es más rápido en promedio, sino también más confiable (menor dispersión).

## **4.5. Fundamentos Teóricos: Convergencia y Adaptación Geométrica**

### **Tasa de Convergencia: Análisis Lineal versus Superlineal**

La teoría de análisis de convergencia proporciona marco conceptual para interpretar los resultados experimentales. El método de gradiente con búsqueda de línea exacta tiene tasa de convergencia lineal caracterizada por:

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq \left( \frac{L - \mu}{L + \mu} \right) \|\mathbf{x}_k - \mathbf{x}^*\|$$

donde  $L$  es la constante de Lipschitz del gradiente y  $\mu$  es la constante de convexidad fuerte. El ratio  $\rho = \frac{L-\mu}{L+\mu} = \frac{\kappa-1}{\kappa+1}$  donde  $\kappa = L/\mu$  es el número de condición, determina la velocidad de convergencia.

Para funciones con  $\kappa \gg 1$  (mal condicionadas),  $\rho \approx 1 - \frac{2}{\kappa} \approx 1$ , resultando en convergencia extremadamente lenta. El número de iteraciones requeridas para reducir el error por factor  $\epsilon$  es  $O(\kappa \log(1/\epsilon))$ , explicando por qué GD-Armijo requiere cientos de iteraciones en regiones donde el condicionamiento local es pobre.

Los métodos quasi-Newton, bajo condiciones apropiadas, exhiben convergencia superlineal:

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq C \|\mathbf{x}_k - \mathbf{x}^*\|^{1+\alpha}$$

con  $\alpha \in (0, 1]$ . Cerca del óptimo, cuando  $B_k \approx \nabla^2 f(\mathbf{x}^*)$ , el comportamiento es cuasi-Newtoniano, con reducción exponencial del error en las últimas iteraciones.

Los resultados experimentales validan estas predicciones: el ratio global de  $2.6\times$  en mediana refleja la ventaja práctica de convergencia superlineal, mientras que el ratio de desviaciones estándar ( $16\times$ ) evidencia que GD-Armijo sufre degradación severa en regiones mal condicionadas.

### Ecuación Secante y Aproximación del Hessiano

La efectividad de BFGS radica en la aproximación del Hessiano inverso mediante actualizaciones de rango bajo. La actualización BFGS satisface la ecuación secante:

$$B_{k+1}\mathbf{s}_k = \mathbf{y}_k$$

donde  $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$  y  $\mathbf{y}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k)$ . Esta condición asegura que  $B_{k+1}$  interpola información de curvatura en la dirección del paso más reciente.

La fórmula de actualización preserva definición positiva de  $B_k^{-1}$  (necesaria para dirección de descenso) bajo la condición de curvatura  $\mathbf{s}_k^T \mathbf{y}_k > 0$ , que se garantiza mediante búsqueda de línea tipo Wolfe. Esta propiedad permite que BFGS construya aproximaciones progresivamente mejores del Hessiano inverso, acumulando información de curvatura a lo largo de la trayectoria de optimización.

Los resultados experimentales en la región con curvatura negativa (cerca del óptimo,  $|x| < 1$ ) demuestran la efectividad de este mecanismo: mientras GD-Armijo zigzaguea (ratio  $6.2\times$ ), BFGS corrige las direcciones subóptimas mediante  $B_k^{-1}\nabla f(\mathbf{x}_k)$ , resultando en convergencia directa observada en los heatmaps.

## 4.6. Análisis de Costo Computacional y Escalabilidad

El costo computacional por iteración difiere entre ambos métodos. GD-Armijo requiere  $O(n)$  operaciones para el producto  $\alpha_k \nabla f(\mathbf{x}_k)$  más evaluaciones de función en la búsqueda de línea Armijo (típicamente 2-10 evaluaciones). BFGS requiere  $O(n^2)$  operaciones para el producto matriz-vector  $B_k^{-1}\nabla f(\mathbf{x}_k)$  más evaluaciones en búsqueda de línea Wolfe (típicamente 1-5 evaluaciones), además de  $O(n^2)$  almacenamiento para  $B_k^{-1}$ .

Para el problema bidimensional ( $n = 2$ ) evaluado, la complejidad  $O(4)$  de BFGS es comparable a  $O(2)$  de GD-Armijo, por lo que el costo por iteración es esencialmente

idéntico. Las mediciones de tiempo real confirman esto: el tiempo medio por experimento es 0.0024 segundos para GD-Armijo y 0.0019 segundos para BFGS. El hecho de que BFGS sea 21 % más rápido en tiempo total, a pesar de mayor complejidad teórica por iteración, confirma que la reducción en número de iteraciones (factor  $3.0\times$ ) domina completamente el análisis de eficiencia.

Para problemas de dimensión moderada ( $n \leq 100$ ), BFGS sigue siendo práctico y preferible. La complejidad  $O(n^2)$  es manejable con hardware moderno, y la reducción en iteraciones compensa ampliamente el costo adicional. Para dimensiones mayores ( $n > 1000$ ), la variante L-BFGS (memoria limitada) mantiene complejidad  $O(mn)$  con  $m \ll n$  (típicamente  $m = 5 - 20$  vectores almacenados), preservando la ventaja de convergencia superlineal con costo de almacenamiento lineal.

#### **4.7. Síntesis de Hallazgos y Conclusiones Finales**

Los experimentos sistemáticos realizados sobre la función  $f(x, y) = y^2 + \log(1 + x^2)$  revelan diferencias significativas entre ambos métodos. BFGS alcanza 96.1 % de robustez frente a 62.0 % de GD-Armijo. Ambos métodos requieren números de iteraciones variables dependiendo de la región inicial, con medianas de 10.0 para BFGS y 14.0 para GD-Armijo en casos convergentes. La ventaja de BFGS se manifiesta especialmente en regiones alejadas del óptimo: desde factor  $1.5\times$  en la región central densa hasta inviabilidad práctica de GD-Armijo en regiones extremas (4.5 % convergencia). Notablemente, BFGS también presenta limitaciones en regiones extremas, alcanzando solo 65.9 % de convergencia con medianas elevadas (282 iteraciones).

El análisis estratificado por mallas revela que la geometría local del problema determina el desempeño relativo. La curvatura variable, particularmente la presencia de curvatura negativa en la componente  $x$  para  $|x| > 1$ , afecta severamente al método de gradiente mediante zigzaguelo y convergencia lenta, mientras que BFGS compensa estas no uniformidades mediante adaptación geométrica. Los mapas de calor visualizan este efecto: GD-Armijo presenta fuerte dependencia espacial del desempeño, mientras que BFGS exhibe comportamiento uniforme hasta regiones muy alejadas del óptimo.

Los resultados experimentales validan cuantitativamente las predicciones teóricas establecidas en la literatura de optimización numérica. La convergencia superlineal de BFGS se manifiesta en medianas consistentemente bajas independientes de la región inicial. La sensibilidad de GD-Armijo al número de condición local se evidencia en la alta varianza y presencia de casos patológicos. La robustez global de métodos quasi-Newton bajo condiciones mínimas se confirma mediante la convergencia universal de BFGS.

#### **Criterios de Selección Algorítmica en Contextos Aplicados**

La selección del método de optimización apropiado constituye una decisión de diseño que debe fundamentarse en el análisis riguroso de las características estructurales del problema objetivo y las restricciones operacionales del entorno computacional. El presente estudio experimental proporciona evidencia cuantitativa que respalda las recomendaciones

establecidas en la literatura teórica de optimización numérica.

Los métodos quasi-Newton, específicamente BFGS, constituyen la opción algorítmica óptima para una clase amplia de problemas de optimización no restringida. Esta clase comprende funciones objetivo continuamente diferenciables con Hessiano localmente Lipschitz continuo, definidas en espacios de dimensión baja a moderada (típicamente  $n \leq 100$  variables). La superioridad de BFGS en estos contextos se manifiesta en tres atributos fundamentales: robustez global ante inicializaciones arbitrarias que garantiza convergencia desde puntos alejados del óptimo, eficiencia computacional que minimiza el número total de iteraciones requeridas, y predictibilidad operacional que facilita la planificación de recursos en sistemas de producción. La implementación de BFGS debe incorporar búsqueda de línea tipo Wolfe-Powell con backtracking apropiado, junto con salvaguardas numéricas que detecten y corrijan pérdida de definición positiva en la aproximación del Hessiano inverso.

Los métodos de descenso por gradiente, aunque superados por quasi-Newton en el régimen evaluado, mantienen relevancia en contextos que presentan restricciones específicas incompatibles con aproximaciones de segunda derivada. El primer contexto relevante es la optimización en espacios de dimensión muy alta ( $n > 10,000$ ), donde la complejidad espacial  $O(n^2)$  del almacenamiento de  $B_k^{-1}$  y la complejidad temporal del producto matriz-vector resultan prohibitivos. El segundo contexto comprende problemas de optimización estocástica, particularmente en aprendizaje automático profundo, donde el gradiente verdadero  $\nabla f(\mathbf{x}_k)$  no es accesible y debe estimarse mediante mini-batches que introducen ruido estocástico; en estas condiciones, las actualizaciones quasi-Newton basadas en diferencias finitas de gradientes ruidosos degeneran y pierden validez teórica. El tercer contexto incluye funciones objetivo no suaves o con discontinuidades en derivadas, donde la hipótesis fundamental de diferenciabilidad continua requerida por métodos quasi-Newton no se satisface.

La variante L-BFGS (BFGS de memoria limitada) proporciona solución al problema de escalabilidad dimensional en el régimen intermedio. Este método almacena únicamente los  $m$  pares de vectores  $(\mathbf{s}_k, \mathbf{y}_k)$  más recientes (típicamente  $m = 5$  a 20), reduciendo la complejidad espacial de  $O(n^2)$  a  $O(mn)$  mientras preserva la propiedad de convergencia superlineal. El análisis teórico establece que L-BFGS con  $m$  suficientemente grande mantiene las tasas de convergencia de BFGS completo, validando su uso en problemas de dimensión intermedia (100 a 10,000 variables). Para problemas de dimensión extrema ( $n > 100,000$ ) típicos en aprendizaje automático moderno, métodos adaptativos de primer orden como Adam, AdaGrad, o RMSprop, que mantienen precondicionadores diagonales actualizados mediante promedios móviles exponenciales, constituyen el estado del arte operacional actual.