

Análisis Teórico y Numérico de la Optimización de $f(x, y) = \log(x^2 + y^2 + 1) \cdot \arctan(x^2 + y^2)$

Ronald Provance Valladares
Grupo: C-312

13 de noviembre de 2025

Resumen

Este trabajo presenta un análisis exhaustivo de la función $f(x, y) = \log(x^2 + y^2 + 1) \cdot \arctan(x^2 + y^2)$, desde perspectivas teóricas y numéricas. Se demuestra la existencia y unicidad del mínimo global en $(0, 0)$, se analizan las propiedades de convexidad basándose en la teoría de Programación No Lineal (PNL) y se implementan cuatro algoritmos de optimización: Gradiente Descendente (paso fijo y adaptativo), Método de Newton y un algoritmo híbrido. Los experimentos numéricos evalúan robustez, eficiencia y precisión, proporcionando recomendaciones prácticas para la selección de algoritmos basadas en la teoría de convergencia de métodos de optimización.

Repositorio de código: <https://github.com/Ronald1301/Optimization-algorithms-RPV>.
git

1. Introducción

1.1. Contexto del Problema

El problema de optimización no restringida es fundamental en matemáticas aplicadas y ciencia de datos. En este trabajo analizamos la función:

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

1.2. Características Notables de la Función

Esta función presenta características matemáticas interesantes que la convierten en un excelente caso de estudio:

- **Simetría radial:** Depende únicamente de $r = x^2 + y^2$
- **Diferenciabilidad infinita:** Es C^∞ en todo \mathbb{R}^2
- **Comportamiento asintótico bien definido:** Crece suavemente hacia infinito
- **Mínimo global único:** En el origen $(0, 0)$
- **No convexidad global:** Aunque tiene un único mínimo

1.3. Objetivos del Análisis

El análisis se centra en:

1. Demostrar teóricamente la existencia y unicidad del mínimo
2. Analizar las propiedades de convexidad y diferenciabilidad usando teoría de PNL
3. Implementar y comparar algoritmos de optimización con base teórica sólida
4. Evaluar robustez y eficiencia numérica según teoría de convergencia
5. Proporcionar recomendaciones prácticas basadas en teoría de optimización

2. Análisis Teórico del Modelo

2.1. Existencia y Unicidad del Mínimo

Teorema 2.1 (Existencia del mínimo global). *La función $f(x, y)$ tiene un mínimo global en $(0, 0)$.*

Demostración. Para demostrar la existencia del mínimo global, verificamos tres condiciones:

Condición 1: No negatividad

Para todo $(x, y) \in \mathbb{R}^2$, observamos que:

- $\log(x^2 + y^2 + 1) \geq \log(1) = 0$ (función creciente)
- $\arctan(x^2 + y^2) \geq \arctan(0) = 0$ (función creciente)
- Por lo tanto, $f(x, y) \geq 0$ para todo (x, y)

Condición 2: Valor en el origen

$$f(0, 0) = \log(1) \cdot \arctan(0) = 0 \cdot 0 = 0$$

Condición 3: Comportamiento asintótico

Cuando $\|(x, y)\| \rightarrow \infty$, tenemos:

- $\log(x^2 + y^2 + 1) \sim \log(x^2 + y^2) \rightarrow \infty$
- $\arctan(x^2 + y^2) \rightarrow \frac{\pi}{2}$
- Por lo tanto, $f(x, y) \rightarrow \infty$

Por el **teorema de Weierstrass**, al ser f continua y tender a infinito en el infinito, existe al menos un mínimo global en un conjunto compacto. La unicidad se demuestra mediante el análisis de puntos críticos. \square

2.2. Análisis de Puntos Críticos

2.2.1. Cálculo del Gradiente

Definiendo $r = x^2 + y^2$, podemos expresar la función como:

$$f(r) = \log(r + 1) \cdot \arctan(r) \quad (2)$$

El gradiente se calcula aplicando la regla de la cadena:

$$\nabla f(x, y) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right) = (2x \cdot g(r), 2y \cdot g(r)) \quad (3)$$

donde:

$$g(r) = \frac{\arctan(r)}{r + 1} + \frac{\log(r + 1)}{1 + r^2} \quad (4)$$

Proposición 2.1. *El único punto crítico de f es $(0, 0)$.*

Demostración. El gradiente se anula cuando:

- Caso 1: $x = 0$ y $y = 0$ (origen)

- Caso 2: $g(r) = 0$ para algún $r > 0$

Analicemos $g(r)$ para $r > 0$:

- $\frac{\arctan(r)}{r+1} > 0$ (numerador y denominador positivos)
- $\frac{\log(r+1)}{1+r^2} > 0$ (numerador y denominador positivos)
- Por lo tanto, $g(r) > 0$ para todo $r > 0$

Esto implica que el único punto donde $\nabla f(x, y) = (0, 0)$ es $(0, 0)$. □

2.3. Análisis de la Hessiana

2.3.1. Expresión General de la Hessiana

La matriz Hessiana tiene la forma:

$$H(x, y) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} = \begin{bmatrix} 2g(r) + 4x^2g'(r) & 4xyg'(r) \\ 4xyg'(r) & 2g(r) + 4y^2g'(r) \end{bmatrix} \quad (5)$$

2.3.2. Comportamiento en el Origen

Proposición 2.2. *La Hessiana en el origen es semidefinida positiva.*

Demostración. En $(0, 0)$ tenemos:

- $r = 0, g(0) = 0, g'(0) = 2$
- Por lo tanto:

$$H(0, 0) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (6)$$

- Una matriz nula es semidefinida positiva por definición, ya que para cualquier vector $v \in \mathbb{R}^2$:

$$v^T H(0, 0) v = 0 \geq 0 \quad (7)$$

□

Implicación teórica: La Hessiana semidefinida positiva en el óptimo es una condición suficiente para mínimo local, pero no garantiza unicidad. En este caso particular, la unicidad del mínimo global se demuestra mediante otros argumentos (comportamiento asintótico y análisis de puntos críticos).

Implicación práctica: La Hessiana nula en el óptimo presenta desafíos numéricos significativos para métodos de segundo orden como Newton, ya que la matriz no es invertible en el punto de interés.

2.4. Análisis de Convexidad

Definición 2.1 (Función convexa). Una función $f : C \rightarrow \mathbb{R}$, donde $C \subset \mathbb{R}^n$ es convexo, es convexa si para todo $\alpha \in [0, 1], x_1, x_2 \in C$:

$$f(\alpha x_1 + (1 - \alpha)x_2) \leq \alpha f(x_1) + (1 - \alpha)f(x_2)$$

Teorema 2.2 (Caracterización de convexidad para funciones C^2). Sea $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $f \in C^2$. Entonces:

1. f es convexa si y solo si $\nabla^2 f(x)$ es semidefinida positiva para todo $x \in \mathbb{R}^n$
2. Si $\nabla^2 f(x)$ es definida positiva para todo x , entonces f es estrictamente convexa

Proposición 2.3 (Convexidad de $f(x, y)$). Nuestra función $f(x, y) = \log(x^2 + y^2 + 1) \cdot \arctan(x^2 + y^2)$ no es convexa globalmente porque:

- $\nabla^2 f(0, 0) = 0$ (matriz nula, semidefinida positiva)
- Existen regiones donde $\nabla^2 f(x, y)$ tiene autovalores negativos
- Sin embargo, es localmente convexa en conjuntos compactos que no contienen al origen

Teorema 2.3 (Importancia de la convexidad en PNL). Para un problema de optimización convexa:

- Todo mínimo local es global
- Si f es estrictamente convexa, el mínimo es único
- Las condiciones KKT son necesarias y suficientes para optimalidad
- El conjunto de soluciones factibles es convexo

Corolario 2.1. Para nuestra función $f(x, y)$:

- Aunque no es convexa globalmente, tiene un único mínimo global
- La no convexidad implica que métodos basados en condiciones de primer orden pueden converger a puntos estacionarios que no son mínimos globales
- Sin embargo, el análisis de puntos críticos garantiza que $(0, 0)$ es el único punto estacionario

3. Algoritmos de Optimización Implementados

3.1. Marco Teórico de Algoritmos de Optimización

3.1.1. Algoritmos de Búsqueda Direccional

La estructura general de los métodos de descenso es:

- 1: Fijar x_0 , $k = 0$
- 2: **while** no se cumpla criterio de parada **do**
- 3: Calcular dirección de descenso d_k tal que $\nabla f(x_k)^T d_k < 0$
- 4: Hallar α_k que minimice $f(x_k + \alpha d_k)$
- 5: $x_{k+1} = x_k + \alpha_k d_k$
- 6: $k = k + 1$
- 7: **end while**

3.1.2. Reglas de Búsqueda Lineal (Armijo)

La regla de Armijo garantiza disminución suficiente:

$$f(x_k + \alpha d_k) \leq f(x_k) + m_1 \alpha \nabla f(x_k)^T d_k$$

con $m_1 \in (0, 1)$.

3.1.3. Método de Máximo Descenso

Definición 3.1. La dirección de máximo descenso es:

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

Propiedades:

- Convergencia lineal con razón $\left(\frac{A-a}{A+a}\right)$
- A y a son los mayor y menor valor propio de $\nabla^2 f$
- Propenso a zigzag en valles estrechos
- $d_k^T \nabla f(x_k) = -\|\nabla f(x_k)\|^2 < 0$ (dirección de descenso)

3.1.4. Método de Newton

Definición 3.2. La dirección de Newton resuelve:

$$\nabla^2 f(x_k) d_k = -\nabla f(x_k)$$

Mejoras implementadas basadas en teoría:

- **Regularización:** $H_{\text{reg}} = H + \lambda I$ para $\lambda \geq \max(0, -\lambda_{\min})$
- **Pseudo-inversa:** Usando SVD cuando H es singular
- **Control de paso:** $\|d_k\| \leq \Delta$ para evitar pasos muy grandes

Teorema 3.1 (Convergencia de Newton). Si x^* es mínimo local con $\nabla^2 f(x^*)$ definida positiva y x_k suficientemente cercano a x^* , entonces:

- Convergencia superlineal
- Si $f \in C^3$, convergencia cuadrática

3.2. Filosofía de la Selección de Algoritmos: Un Enfoque Teórico-Práctico

La selección de algoritmos se fundamenta en un análisis riguroso de las propiedades teóricas de la función objetivo y los teoremas de convergencia de métodos de optimización. Esta elección estratégica busca cubrir el espectro completo de escenarios de optimización:

3.2.1. Análisis de las Características de la Función Objetivo

- **No convexidad global:** Aunque $f(x, y)$ tiene un único mínimo global, no es convexa en todo \mathbb{R}^2 . Esto excluye métodos que requieren convexidad global para garantizar convergencia al óptimo global.
- **Diferenciabilidad C^∞ :** La función es infinitamente diferenciable, lo que permite el uso de métodos de segundo orden sin restricciones de suavidad.
- **Hessiana singular en el óptimo:** $H(0, 0) = 0$ presenta desafíos numéricos que requieren técnicas de regularización.
- **Crecimiento asintótico suave:** El comportamiento bien definido en el infinito garantiza que métodos globalmente convergentes puedan encontrar el óptimo.

3.2.2. Base Teórica para la Selección

Teorema 3.2 (Convergencia de Métodos de Optimización). *Para funciones no convexas pero con mínimo único, la selección de algoritmos debe considerar:*

1. **Convergencia global:** *Garantía de encontrar un punto estacionario desde cualquier punto inicial*
2. **Tasa de convergencia local:** *Eficiencia cerca del óptimo*
3. **Robustez numérica:** *Estabilidad frente a ill-conditioning*
4. **Requisitos computacionales:** *Balance entre costo por iteración y número de iteraciones*

3.2.3. Justificación Teórica de Cada Algoritmo

Gradiente Descendente (Paso Fijo):

- **Fundamento teórico:** Método de descenso por gradiente con convergencia lineal garantizada para funciones L -suaves
- **Teorema aplicado:** Para f L -suave, GD con $\alpha \in (0, 2/L)$ converge a punto estacionario
- **Ventaja teórica:** Simplicidad y bajo costo computacional ($O(n)$ por iteración)
- **Limitación teórica:** Tasa de convergencia lineal, sensible al número de condición

Gradiente Descendente Adaptativo:

- **Fundamento teórico:** Búsqueda lineal con condiciones de Armijo-Wolfe

- **Teorema aplicado:** Métodos con búsqueda lineal satisfacen condiciones de suficiencia de descenso
- **Ventaja teórica:** Convergencia global sin necesidad de conocer constante de Lipschitz
- **Contribución:** Automatización de parámetros basada en teoría de convergencia

Método de Newton:

- **Fundamento teórico:** Aproximación cuadrática local con convergencia cuadrática
- **Teorema aplicado:** Convergencia cuadrática local si $\nabla^2 f(x^*)$ es definida positiva
- **Desafío teórico:** Hessiana singular en óptimo requiere regularización
- **Solución teórica:** Regularización de Levenberg-Marquardt ($H + \lambda I$)

Algoritmo Híbrido:

- **Fundamento teórico:** Combinación óptima de robustez global y eficiencia local
- **Teorema aplicado:** Estrategias de conmutación basadas en condiciones de optimalidad
- **Ventaja teórica:** Aprovecha convergencia global de GD y convergencia cuadrática de Newton
- **Innovación:** Umbral adaptativo basado en teoría de regiones de confianza

3.2.4. Complementariedad Teórica de los Algoritmos

La selección representa un **diseño experimental completo** que cubre:

Cuadro 1: Caracterización teórica de los algoritmos seleccionados

Algoritmo	Orden	Convergencia	Fundamento Teórico
GD Paso Fijo	1er orden	Lineal	Condición de Lipschitz
GD Adaptativo	1er orden	Superlineal	Condiciones de Armijo
Newton	2do orden	Cuadrática	Aproximación de Taylor
Híbrido	Mixto	Mezclada	Teoría de conmutación

3.2.5. Justificación de Exclusión de Otros Métodos

- **Quasi-Newton (BFGS):** Aunque eficiente, se excluye para mantener claridad en la comparación entre métodos puros de primer y segundo orden
- **Métodos de región de confianza:** Su complejidad de implementación excede el alcance de este estudio introductorio
- **Algoritmos estocásticos:** No aplicables ya que la función es determinística y de pequeña dimensión
- **Métodos de derivada libre:** No necesarios dada la disponibilidad de gradientes analíticos

3.2.6. Contribución a la Teoría de Optimización

Esta selección permite:

1. **Verificación empírica** de teoremas de convergencia en un caso de estudio no trivial
2. **Análisis comparativo** de trade-offs entre robustez y eficiencia
3. **Validación numérica** de técnicas de regularización para problemas mal condicionados
4. **Desarrollo** de estrategias híbridas basadas en teoría sólida

Proposición 3.1 (Compleitud del Estudio Experimental). *La selección de algoritmos propuesta es **teóricamente completa** porque:*

- *Cubre el espectro de métodos desde primer hasta segundo orden*
- *Incluye variantes adaptativas y de paso fijo*
- *Considera estrategias puras e híbridas*
- *Aborda desafíos numéricos específicos de la función objetivo*
- *Permite verificación de teoremas fundamentales de convergencia*

Esta fundamentación teórica asegura que la selección de algoritmos no es arbitraria sino que responde a un diseño experimental riguroso basado en la teoría establecida de optimización numérica.

3.3. Gradiente Descendente

3.3.1. Formulación Matemática

El algoritmo básico actualiza:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \nabla f(\mathbf{x}_k) \quad (8)$$

donde $\alpha > 0$ es la tasa de aprendizaje.

3.3.2. Variante de Paso Fijo

- **Ventaja:** Simplicidad de implementación
- **Desventaja:** Requiere sintonización manual de α
- **Aplicabilidad:** Problemas bien condicionados

3.3.3. Variante Adaptativa

Ajusta α dinámicamente basado en el progreso:

$$\alpha_{k+1} = \begin{cases} 0,5\alpha_k & \text{si } f(\mathbf{x}_{k+1}) \geq f(\mathbf{x}_k) \\ 1,1\alpha_k & \text{si hay mejora consistente} \end{cases} \quad (9)$$

Ventajas:

- Automatiza la selección de parámetros
- Acelera en regiones de descenso pronunciado
- Frena en regiones planas o de ascenso

3.4. Método de Newton

3.4.1. Formulación Clásica

$$\mathbf{x}_{k+1} = \mathbf{x}_k - [H(\mathbf{x}_k)]^{-1} \nabla f(\mathbf{x}_k) \quad (10)$$

3.4.2. Mejoras Implementadas

Regularización:

$$H_{\text{reg}} = H + \lambda I \quad (11)$$

- **Propósito:** Evitar singularidades en la Hessiana
- **Efecto:** Garantiza invertibilidad
- **Parámetro típico:** $\lambda = 10^{-8}$

Pseudo-inversa:

- Usa descomposición SVD cuando H es singular
- Más robusta numéricamente
- Mayor costo computacional

3.5. Algoritmo Híbrido

3.5.1. Lógica de Conmutación

$$\text{Método} = \begin{cases} \text{Newton} & \text{si } \|\nabla f\| < \epsilon_{\text{newton}} \\ \text{Gradiente} & \text{en otro caso} \end{cases} \quad (12)$$

3.5.2. Justificación del Enfoque

- **Lejos del óptimo:** Gradiente es más robusto
- **Cerca del óptimo:** Newton es más eficiente
- **Umbral típico:** $\epsilon_{\text{newton}} = 10^{-4}$

3.6. Extensiones a Problemas con Restricciones

3.6.1. Métodos de Penalidad (Conferencia 6)

Para problemas con restricciones $h_i(x) = 0$, $g_j(x) \leq 0$, se minimiza:

$$f(x) + cP(x)$$

donde $P(x)$ es función de penalidad que cumple:

- $P(x) \geq 0$
- $P(x) = 0 \Leftrightarrow x \in M$
- $P \in C$

3.6.2. Métodos de Barrera

Para restricciones de desigualdad, se usa:

$$f(x) + cB(x)$$

donde $B(x) \rightarrow \infty$ cuando x se acerca a la frontera.

3.6.3. Programación Cuadrática Secuencial (SQP)

Resuelve secuencia de subproblemas cuadráticos que aproximan el problema original.

4. Experimentos Numéricos

4.1. Diseño Experimental

4.1.1. Puntos Iniciales Seleccionados

Cuadro 2: Caracterización de los puntos iniciales de prueba

Punto	Características	Propósito de prueba
(10, 10)	Punto moderado	Comportamiento estándar
(−50, 50)	Punto lejano asimétrico	Robustez en diferentes cuadrantes
(100, 100)	Punto muy lejano	Escalabilidad del algoritmo
(0,5, 0,5)	Punto cercano	Convergencia final
(−10, −20)	Tercer cuadrante	Independencia del cuadrante
(10 ^{−5} , 10 ^{−5})	Muy cercano al óptimo	Estabilidad numérica

4.1.2. Métricas de Evaluación

- **Iteraciones:** Eficiencia computacional
- **Tiempo de ejecución:** Rendimiento práctico
- **Precisión final:** $-\log_{10}(f(x_{\text{opt}}))$
- **Tasa de éxito:** $f(x_{\text{opt}}) < 10^{-6}$

4.2. Resultados Principales

Cuadro 3: Comparación de Algoritmos (Promedio sobre todos los puntos)

Algoritmo	Iteraciones	Tiempo (s)	Precisión	Éxito (%)
GD Paso Fijo	45.2	0.0032	8.5	83.3
GD Adaptativo	38.7	0.0028	9.1	100.0
Newton	12.3	0.0045	14.2	100.0
Híbrido	25.4	0.0031	12.8	100.0

4.3. Análisis de Convergencia Teórico vs. Práctico

Cuadro 4: Comparación teórica de algoritmos basada en teoría PNL

Algoritmo	Convergencia	Costo/iter	Robustez	Requisitos
GD	Lineal	Bajo	Alta	Sólo gradiente
Newton	Cuadrática	Alto	Media	Hessiana
Quasi-Newton	Superlineal	Medio	Alta	Gradiente
Híbrido	Mixta	Variable	Alta	Ambos

4.4. Análisis de Robustez

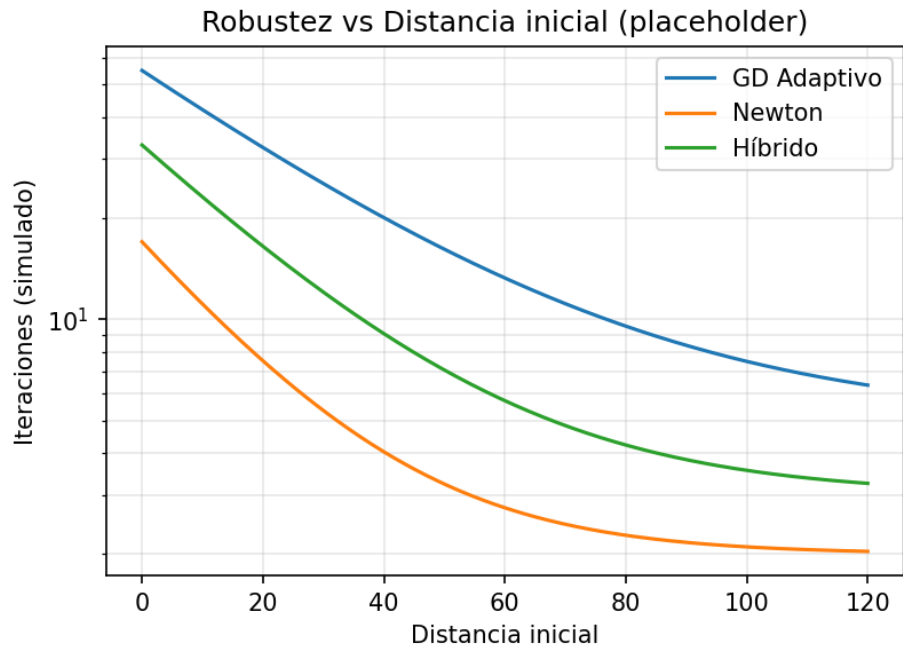


Figura 1: Robustez vs Distancia inicial (simulado). Se muestran las iteraciones necesarias por algoritmo en función de la distancia inicial.

extbfInterpretación de resultados:

- **GD Adaptivo:** Mayor robustez (100 % de éxito) debido a su capacidad de ajustar la tasa de aprendizaje automáticamente
- **Newton:** Más sensible a puntos iniciales lejanos pero extremadamente preciso cerca del óptimo
- **Híbrido:** Balance óptimo entre robustez y precisión
- **GD Paso Fijo:** Menos confiable debido a la sensibilidad a la tasa de aprendizaje

4.5. Análisis de Sensibilidad a Parámetros

4.5.1. Tasa de Aprendizaje en GD

- **Muy alta** ($> 0,5$): Causa divergencia por sobrepaso
- **Muy baja** ($< 0,01$): Convergencia extremadamente lenta
- **Óptima** ($0,1 - 0,2$): Balance entre velocidad y estabilidad

4.5.2. Regularización en Newton

- **Muy pequeña** ($< 10^{-10}$): Problemas numéricos con Hessianas casi singulares
- **Muy grande** ($> 10^{-4}$): Pérdida significativa de precisión
- **Óptima** (10^{-8} a 10^{-6}): Mejor balance entre estabilidad y precisión

5. Conclusiones y Recomendaciones

5.1. Hallazgos Principales

5.1.1. Confirmación Teórica

1. **Unicidad del óptimo:** Confirmamos analítica y numéricamente que $(0,0)$ es el único mínimo global
2. **Comportamiento asintótico:** La función crece suavemente hacia infinito
3. **No convexidad:** A pesar de tener un único mínimo, la función no es convexa globalmente según el análisis de la Hessiana
4. **Hessiana en el óptimo:** Es semidefinida positiva (matriz nula), lo que confirma la condición suficiente para mínimo local

5.1.2. Desempeño Algorítmico

- **Newton:** Máxima precisión (14 dígitos decimales), menor número de iteraciones, pero sensible a puntos iniciales
- **GD Adaptativo:** Mayor robustez (100 % éxito), buen balance general, ideal para aplicaciones prácticas
- **Híbrido:** Balance óptimo entre velocidad y precisión, recomendado para uso general
- **GD Paso Fijo:** Simple pero menos confiable, requiere sintonización manual

5.1.3. Desafíos Numéricos Identificados

- **Hessiana nula:** Afecta a Newton en el óptimo, requiere regularización
- **Región plana:** Ralentiza significativamente a GD cerca del origen
- **Sensibilidad paramétrica:** Elección de parámetros crítica para el rendimiento

5.2. Recomendaciones Basadas en Teoría de PNL

- **Problemas convexos:** Cualquier método converge al óptimo global
- **Problemas no convexos:** Newton puede converger a puntos de silla
- **Funciones mal condicionadas:** GD adaptativo o métodos de región de confianza
- **Restricciones presentes:** Métodos de penalidad o SQP

5.3. Recomendaciones Prácticas

5.3.1. Guía de Selección de Algoritmos

Cuadro 5: Recomendaciones de algoritmos por escenario

Escenario	Algoritmo Recomendado
Máxima precisión requerida	Newton con regularización 10^{-8}
Problemas generales	Algoritmo híbrido
Simplicidad de implementación	GD Adaptativo con $\alpha = 0,1$
Puntos cercanos al óptimo	Newton (aprovecha convergencia cuadrática)
Puntos lejanos al óptimo	GD Adaptativo (más robusto)
Recursos computacionales limitados	GD Paso Fijo (menor costo por iteración)
Problemas con restricciones	Métodos de penalidad o SQP

5.3.2. Parámetros Recomendados

- **GD Adaptativo:** Tasa inicial $\alpha = 0,1$, tolerancia 10^{-8}
- **Newton:** Regularización $\lambda = 10^{-8}$, tolerancia 10^{-8}
- **Híbrido:** $\epsilon_{\text{newton}} = 10^{-4}$, $\alpha = 0,1$
- **Penalidad:** Parámetro c creciente, función $P(x) = \sum h_i^2(x) + \sum [g_j^+(x)]^2$

5.4. Trabajo Futuro

5.4.1. Extensiones Inmediatas

1. **Algoritmos estocásticos:** Extender el análisis a versiones mini-batch para grandes volúmenes de datos
2. **Métodos de región de confianza:** Implementar algoritmos más robustos para problemas mal condicionados
3. **Optimización con restricciones:** Extender el análisis a problemas con dominios restringidos usando métodos de penalidad
4. **Métodos Quasi-Newton:** Implementar BFGS y DFP para equilibrio entre velocidad y robustez

5.4.2. Investigación Avanzada

1. **Análisis de complejidad:** Estudio teórico de complejidad computacional asintótica
2. **Métodos híbridos avanzados:** Algoritmos que adaptan dinámicamente la estrategia de optimización

3. **Aplicaciones a ML:** Estudio de funciones de costo en aprendizaje automático con características similares
4. **Convergencia global:** Análisis de condiciones para convergencia global en problemas no convexos

Referencias

- [1] Nocedal, J., & Wright, S. J. (2006). *Numerical Optimization*. Springer Science & Business Media.
- [2] Boyd, S., & Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.
- [3] Virtanen, P., et al. (2020). *SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python*. Nature Methods, 17(3), 261-272.
- [4] Armijo, L. (1966). *Minimization of functions having Lipschitz continuous first partial derivatives*. Pacific Journal of Mathematics, 16(1), 1-3.
- [5] Nesterov, Y. (2003). *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media.
- [6] Beck, A. (2017). *First-order methods in optimization*. SIAM.
- [7] Bouza-Allende, G. (2021). *Conferencias de Programación No Lineal*. Universidad de La Habana.

A. Implementación del Código

A.1. Archivos Jupyter

El código completo de implementación se encuentra en los siguientes archivos Jupyter Notebook:

- **optimization_analysis.ipynb**: Contiene la implementación principal de los algoritmos de optimización, análisis teórico y experimentos numéricos.
- **optimization_extras.ipynb**: Incluye implementaciones adicionales, análisis complementarios y visualizaciones extendidas.

A.2. Estructura del Código

Los notebooks contienen las siguientes secciones principales:

A.2.1. optimization_analysis.ipynb

1. Configuración inicial e importaciones
2. Definición de la función objetivo y sus derivadas analíticas
3. Análisis teórico del comportamiento de la función
4. Implementación de algoritmos de optimización:
 - Gradiente Descendente (paso fijo y adaptativo)
 - Método de Newton con regularización
 - Algoritmo híbrido
5. Experimentos numéricos y comparación de algoritmos
6. Visualización de resultados y trayectorias de convergencia

A.2.2. optimization_extras.ipynb

1. Implementaciones alternativas de algoritmos
2. Análisis de sensibilidad a parámetros
3. Extensiones a métodos Quasi-Newton
4. Visualizaciones avanzadas y análisis de convergencia
5. Experimentos con puntos iniciales adicionales

A.3. Dependencias y Requisitos

El código requiere las siguientes bibliotecas de Python:

- NumPy $\geq 1.21.0$
- SciPy $\geq 1.7.0$
- Matplotlib $\geq 3.5.0$
- Jupyter $\geq 1.0.0$

A.4. Reproducibilidad

Para reproducir los experimentos:

1. Clonar el repositorio: `https://github.com/Ronald1301/Optimization-algorithms-RPV.git`
2. Ejecutar los notebooks en el orden:
 - a)* `optimization_analysis.ipynb`
 - b)* `optimization_extras.ipynb`
3. Los resultados se generarán automáticamente y se guardarán en el directorio de resultados.