Lab3

j.puig - Juan Felipe Puig - 202221336
ja.fierro - Jairo Andrés Fierro - 202226326

March 2025

1 Problema 1: Newton-Raphson en 2D para Polinomios Cúbicos

1.1 Análisis de resultados (gráfica de la función)

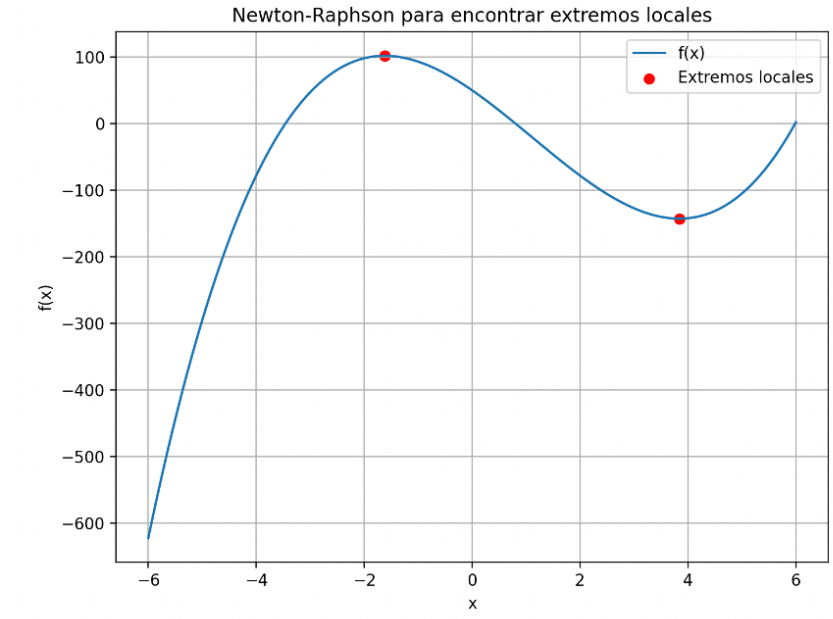

Figure 1: Gráfica de la función

La gráfica representa la función cúbica $f(x) = 3x^3 - 10x^2 - 56x + 50$ y muestra los extremos locales encontrados mediante el método de Newton-Raphson.

Se observa que la función exhibe un comportamiento característico de los polinomios cúbicos, con una tendencia a infinito positivo y negativo en los extremos del dominio. Entre estos valores, la función presenta un punto de máximo y un punto de mínimo, lo que indica la presencia de dos extremos locales.

En la gráfica se destacan dos puntos críticos marcados en rojo. El primero es un máximo local ubicado aproximadamente en $x \approx -4$, donde la pendiente de la función cambia de positiva a negativa. El segundo es un mínimo local alrededor de $x \approx 3.5$, donde la pendiente cambia de negativa a positiva. Estos puntos críticos son consistentes con los resultados obtenidos al derivar la función y resolver la ecuación $f'(x) = 0$, lo que sugiere que el método de Newton-Raphson ha convergido correctamente hacia los valores esperados.

1.2 Análisis de resultados: comportamiento de la convergencia

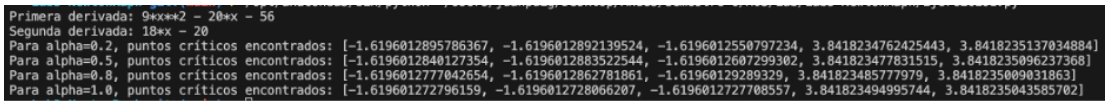

Figure 2: Convergencia

La convergencia del método depende del valor del factor de paso α. Si α es demasiado grande, el método puede oscilar o incluso divergir, dificultando la identificación de los extremos. En cambio, si α es muy pequeño, la convergencia puede volverse

2 Problema 2: Análisis de Extremos Locales y Globales

2.1 Analisis de resultados (grafica de la funcion):

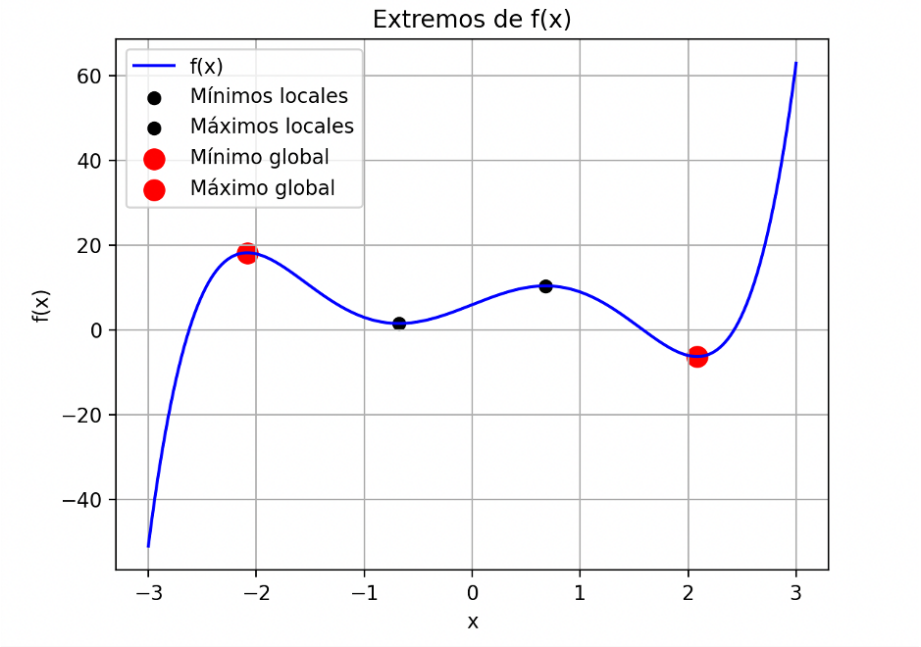

Figure 3: Gráfica de la función

La gráfica presentada muestra la función $f(x)$ y sus puntos críticos, destacando los extremos locales y globales. Se observa que la función exhibe un comportamiento oscilatorio con la presencia de varios puntos de inflexión y cambios de concavidad. La curva azul representa la función evaluada en un dominio determinado, y los puntos críticos han sido resaltados con distintos colores según su clasificación.

En la gráfica se identifican dos mínimos locales y dos máximos locales, señalados en color negro. Estos puntos representan valores donde la derivada primera de la función se anula y la segunda derivada indica un cambio en la concavidad. Adicionalmente, se han marcado el mínimo global y el máximo global en color rojo, los cuales representan los valores más bajos y más altos que la función alcanza dentro del intervalo analizado.

El máximo global se encuentra aproximadamente en $x \approx -2$, donde la función alcanza su punto más alto dentro del dominio considerado. Por otro lado, el mínimo global ocurre cerca de $x \approx 2$, indicando el valor más bajo de

la función en el intervalo observado. La correcta identificación de estos puntos sugiere que se ha aplicado un método numérico o analítico eficiente para hallar los extremos de la función.

2.2 Analisis de convergencia:

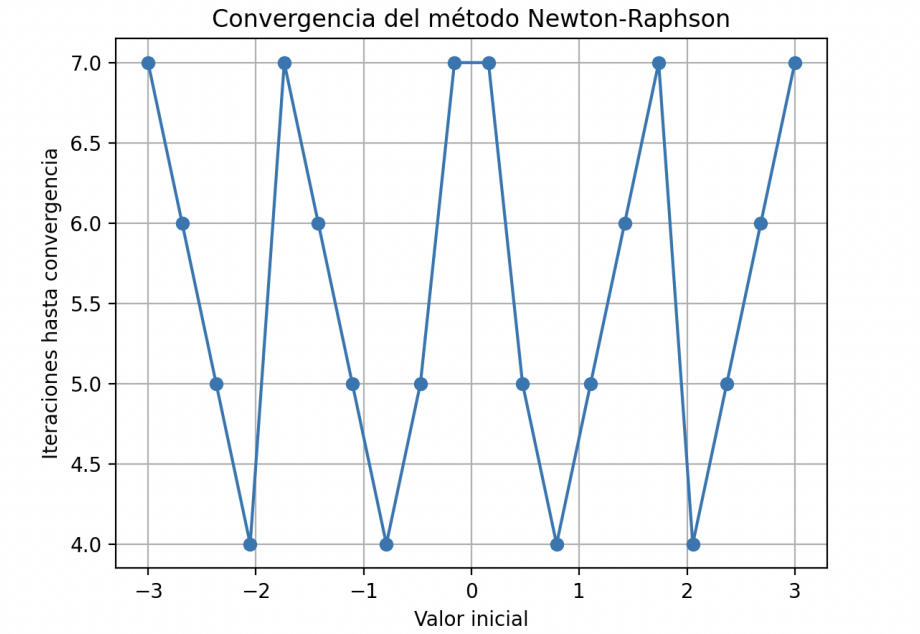

Figure 4: Grafica de convergencia

La gráfica presentada muestra la convergencia del método de Newton-Raphson en función del valor inicial elegido. En el eje horizontal se representan los valores iniciales a partir de los cuales se ejecuta el método, mientras que en el eje vertical se muestra el número de iteraciones necesarias para alcanzar la convergencia.

Se observa un comportamiento oscilatorio en el número de iteraciones requeridas, lo que sugiere que la eficiencia del método depende fuertemente del punto de partida. En ciertos valores iniciales, el método converge en un número mínimo de iteraciones (alrededor de 4), mientras que en otros requiere hasta 7 iteraciones para alcanzar la solución deseada. Esto indica la presencia de regiones donde la aproximación inicial está cerca de la raíz y otras donde la función tiene una pendiente menos favorable, afectando la rapidez de la convergencia.

El patrón observado también sugiere que los valores iniciales cercanos a ciertos puntos críticos podrían generar fluctuaciones en la convergencia, posiblemente debido a la cercanía con puntos donde la derivada se aproxima a cero,

reduciendo la eficacia del método. Este análisis es útil para determinar estrategias que optimicen la elección del punto de inicio en la aplicación del método de Newton-Raphson.

3 Problema 3: Newton-Raphson Multidimensional

3.1 Parte a

- Formulación matemática

  I. Definir la Función de Rosenbrock

$$f(x, y) = (x - 1)^2 + 100(y - x^2)^2 \tag{1}$$

  II. Calculo del gradiente

$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} 2(-1 + x + 200x^3 - 200xy) \\ 200(y - x^2) \end{bmatrix}$$

  III. Calculo de la matriz Hessiana

$$H(x, y) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} = \begin{bmatrix} 1200x^2 - 400y + 2 & -400x \\ -400x & 200 \end{bmatrix}$$

  IV. Método de Newton-Raphson para Funciones Bidimensionales

$$\mathbf{x}_{n+1} = \mathbf{x}_n - H^{-1}(x_n, y_n) \cdot \nabla f(x_n, y_n) \tag{2}$$

  donde:

  - $\mathbf{x}_n = (x_n, y_n)$ es el punto actual.
  - $H^{-1}(x_n, y_n)$ es la inversa de la matriz Hessiana.
  - $\nabla f(x_n, y_n)$ es el gradiente evaluado en (x_n, y_n).

  Iteramos hasta que el gradiente sea pequeño:

$$\|\nabla f(x, y)\| < \epsilon. \tag{3}$$

- Análisis de resultados.

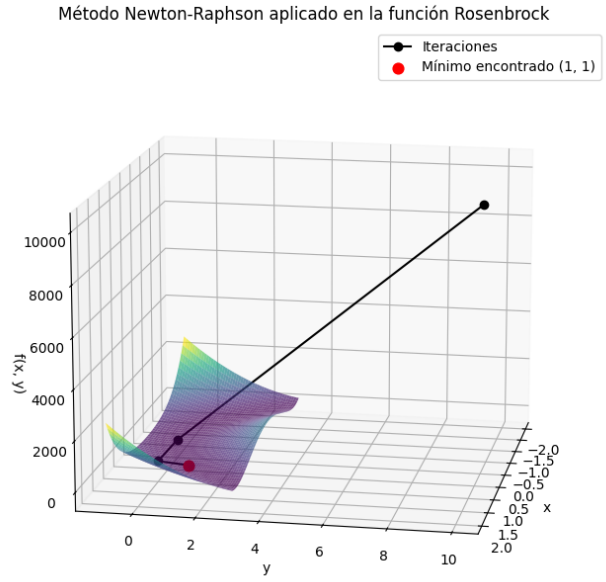

Figure 5: Representación puntos iterativos sobre la superficie y mínimo final destacado.

La superficie muestra la forma de la función de Rosenbrock con mínimo en (1,1). Los puntos negros con línea muestran las iteraciones realizadas por el algoritmo y el punto rojo muestra el mínimo encontrado en (1,1). La superficie y la información graficada muestran una trayectoria descendente que cae bien y de forma directa en el valle de la función de Rosenbrock. También se puede observar que la trayectoria luego de ser descendente se vuelve más vertical y precisa hasta encontrar el mínimo.

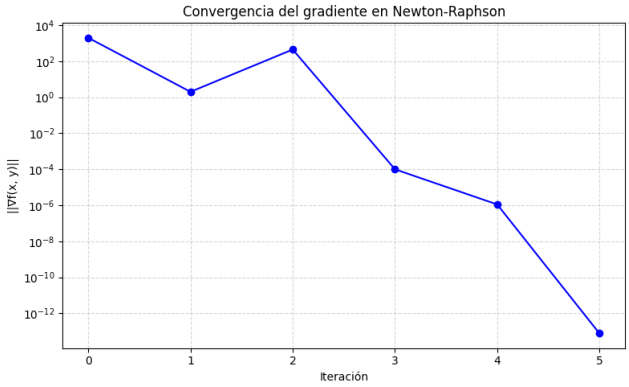

Figure 6: Analisis de convergencia hacia el mínimo conocido de la función (1,1)

Esta figura representa la norma del gradiente vs la iteración. La norma del gradiente indica la fueza de cambio en cada punto. En el gráfico se puede ver que la norma del gradiente disminuye y tiene una tendencia decreciente, pero al mismo tiempo su tendencia oscila ya que sube y baja. La tendencia oscilante indica que en vez de acercarse directamente al mínimo, el método presenta saltos que posiblemente alejan temporalmente la posibilidad de encontrar el mínimo. En contraste, la tendencia decreciente muestra que el método se aproxima de forma progresiva al mínimo. Esto quiere decir que a medida que las iteraciones avanzan, la magnitud del gradiente disminuye.

b) Parte b

I. Función de cuatro dimensiones definida

$$f(x, y, z) = x^2 + y^2 + z^2 \tag{4}$$

II. Algoritmo de Newton-Raphson en $\mathbb{R}^4$

(a) Inicializar:
- $i \leftarrow 0$
- $\mathbf{x}_0 \leftarrow$ valor inicial
- $\varepsilon \leftarrow 0.0001$

(b) **while** $\|\nabla f(\mathbf{x}_i)\| > \varepsilon$ **do**:
- Calcular gradiente: $\mathbf{g} \leftarrow \nabla f(\mathbf{x}_i)$
- Calcular Hessiana: $H \leftarrow H(\mathbf{x}_i)$
- Calcular siguiente punto:

$$\mathbf{x}_{i+1} \leftarrow \mathbf{x}_i - H^{-1}(\mathbf{x}_i)\mathbf{g}$$

- $i \leftarrow i + 1$

(c) **Retornar:** $\mathbf{x}_{\min} \leftarrow \mathbf{x}_i$

III. Calculo analítico del gradiente

$$\nabla f(x, y, z) = \begin{bmatrix} 2x \\ 2y \\ 2z \end{bmatrix}$$

IV. Calculo analítico matriz Hessiana

$$Hf(x, y, z) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

- Análisis de resultados.

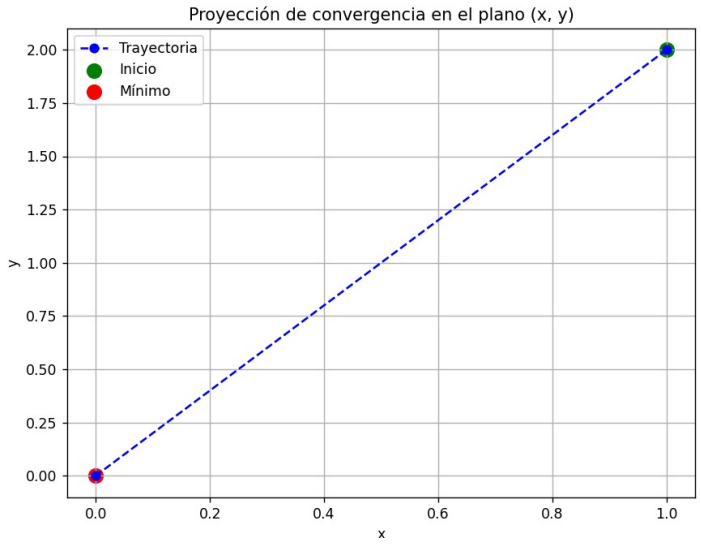

Figure 7: Proyección Bidimensional

La imagen de arriba muestra la trayectria del método Newton-Raphson proyectada sobre el plano xy. La linea azúl representa el camino seguido desde el punto de inicio hasta el mínimo(el rojo). Esta trayectoria es directa y sin oscilaciones, lo cual indica que las actualizaciones hechas por el método son estables desde el inicio. La trayectoria directa también sugiere que hubo una convergencia rápida.

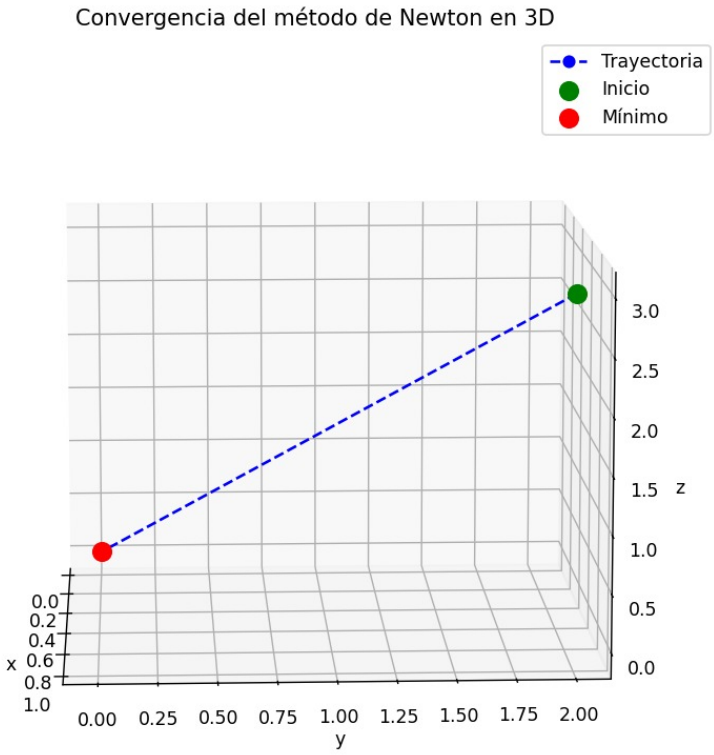

Figure 8: Proyección tridimensional

Ahora, la figura de arriba muestra una proyección tridimensional en el espacio (x,y,z). En esta figura también se puede ver el punto inicial y el mínimo unidos por una trayectoria directa lo cual indica que el algoritmo converge sin dificultad. La linea azul que representa la trayectoria muestra que el método no necesitó muchas iteraciones para llegar al mínimo y tanto el gradiente como la Hessiana proporcionan direcciones de descenso.

En cuanto a las dificultades computacionales específicas del problema en alta dimensiones encontramos el cálculo y almacenamiento de la Hessiana. Si el problema es de n dimensiones, entonces la Hessiana tendrá un tamaño de n x n lo cual aumenta la complejidad del algoritmo porque se vuelve costoso almacenarla. Esto generaría problemas de memoria y aumentaría los tiempos de ejecución. Por otro lado encontramos dificultades para analizar y visualizar los resultados ya que para funciones con mas de tres dimensiones visualizar la convergencia o trayectoria se vuelve complicado.

Finalmente, en cada iteración el método debe calcular el gradiente, la Hessiana y resolver la inversa de la Hessiana multiplicada por el gradiente.

Este proceso resulta computacionalmente costoso en problemas de altas dimensiones, ya que involucra operaciones con matrices grandes.

4 Problema 4: Gradiente Descendente en Optimización

4.1 Parte A: Analisis del gradiente descendente

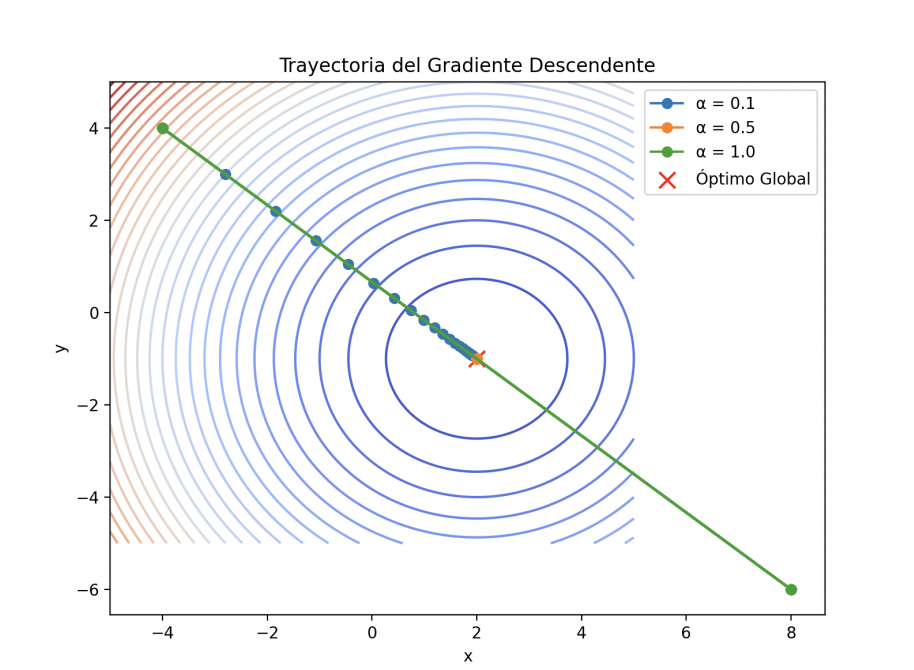

Figure 9: Gradiente descendente

La gráfica representa la trayectoria del algoritmo de Gradiente Descendente aplicado a una función de contorno. Se observan curvas de nivel que indican el comportamiento de la función objetivo, con colores que representan la magnitud de los valores de la función. En la leyenda, se muestran diferentes valores del parámetro de aprendizaje α (0.1, 0.5 y 1.0), lo que permite analizar el impacto de la tasa de aprendizaje en la convergencia hacia el óptimo global.

Las trayectorias indican cómo el algoritmo ajusta los valores de x y y en cada iteración en función del gradiente de la función. Se puede notar que, independientemente del valor de α, todas las trayectorias convergen hacia el mismo punto marcado como "Óptimo Global" con una cruz roja.

- Para $\alpha = 0.1$, la convergencia es más lenta, reflejada en pasos pequeños a

lo largo del trayecto. - Para $\alpha = 0.5$, la trayectoria es más eficiente y alcanza el óptimo en menos iteraciones. - Para $\alpha = 1.0$, el algoritmo avanza en pasos grandes pero sin sobrepasar el óptimo, lo que indica una tasa de aprendizaje efectiva.

Este análisis permite comparar el efecto de la tasa de aprendizaje en la eficiencia del método y resalta la importancia de una elección adecuada del parámetro α para evitar una convergencia excesivamente lenta o problemas de estabilidad en el descenso.

5 Problema 5: Descenso de Gradiente y Descenso de Gradiente Basado en Momento

5.1 Problema

Queremos encontrar los parámetros óptimos de la red neuronal que minimicen la función de costo, que mide la diferencia entre los valores predichos y los valores reales.

5.2 Formulación matemática descenso de Gradiente básico

I. Función seno

$$f(x) = sen(x) \tag{5}$$

II. Función de costo

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{6}$$

n: es el número de ejemplos de entrenamiento.
y_i: es el valor real de la función seno, es decir, $y_i = \sin(x_i)$.
$\hat{y}_i$: es la predicción de la red neuronal.

III. Calculo gradiente

$$\nabla J(\theta) = \frac{\partial J(\theta)}{\partial \theta} \tag{7}$$

IV. Regla de actualización del Descenso de Gradiente

$$\theta_{t+1} = \theta_t - \alpha \nabla_\theta J_\theta \tag{8}$$

V. Algoritmo Gradient Descent

(a) Inicializar:
- α: Tasa de aprendizaje.
- t $\leftarrow 0$.

(b) Mientras max_iterations hacer:
- $\theta_{t+1} = \theta_t - \alpha \nabla J(\theta)$
- $t \leftarrow t + 1$

5.3 Formulación matemática descenso de Gradiente Basado en Momento

I. Función seno

$$f(x) = sen(x) \tag{9}$$

II. Función de costo

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{10}$$

n: es el número de ejemplos de entrenamiento.
y_i: es el valor real de la función seno, es decir, $y_i = \sin(x_i)$.
$\hat{y}_i$: es la predicción de la red neuronal.

III. Calculo gradiente

$$\nabla J(\theta) = \frac{\partial J(\theta)}{\partial \theta} \tag{11}$$

IV. Regla de actualización del Descenso de Gradiente modificado con el Momentum.

$$v_t = \beta v_{t-1} - \alpha \nabla_\theta J(\theta) \tag{12}$$

$$\theta_{t+1} = \theta_t + v_t \tag{13}$$

(a) Inicializar:

- α: Tasa de aprendizaje.
- β: Coeficiente de momento.
- t $\leftarrow$ 0.

(b) Mientras $t <$ max_iterations hacer:

- $v_t = \beta v_{t-1} - \alpha \nabla_\theta J(\theta_t)$
- $\theta_{t+1} = \theta_t + v_t$

5.4 Análisis de resultados obtenidos descenso de Gradiente básico

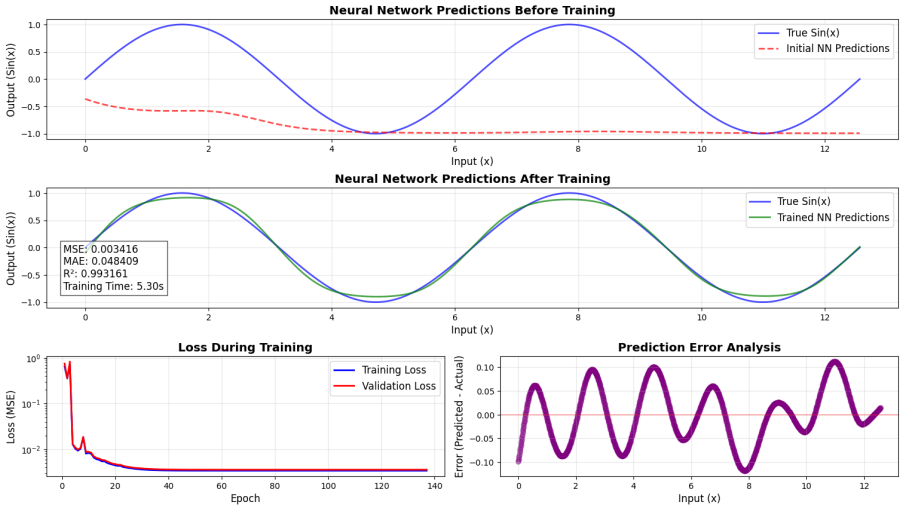

Figure 10: Descenso de Gradiente básico

La gráfica "Loss During Training" se ve una rápida disminución del error (MSE) en las primeras 40 iteraciones, indicando una convergencia inicial eficiente. Luego de esas iteraciones la curva se estabiliza y se aproxima al mínimo del error sin oscilaciones.Es importante nombrar el tiempo que duró el entrenamiento 5.30 segundos lo que es un tiempo razonable para una red simple y un conjunto de datos pequeños.

En cuanto a las métricas es importante nombrar que el error cuadrático medio es muy bajo con un error de 0.0034, un error absoluto de 0.046 y un $R^2 = 0.9916$ lo que indica que el modelo explica muy bien la variabilidad de la función sen(x).

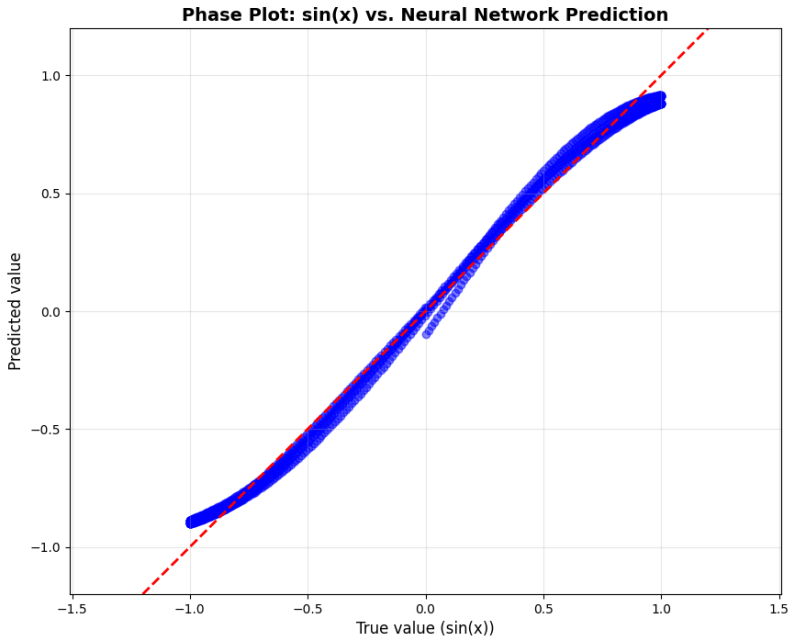

Figure 11: sin(x) vs Neural Network Prediction

El gráfico de fase sin(x) vs predicción muestra que los puntos azules se alinean bien sobre la línea roja de identidad, lo que implica una buena correspondencia entre predicción y valor real

5.5 Análisis de resultados obtenidos descenso de Gradiente Basado en Momento

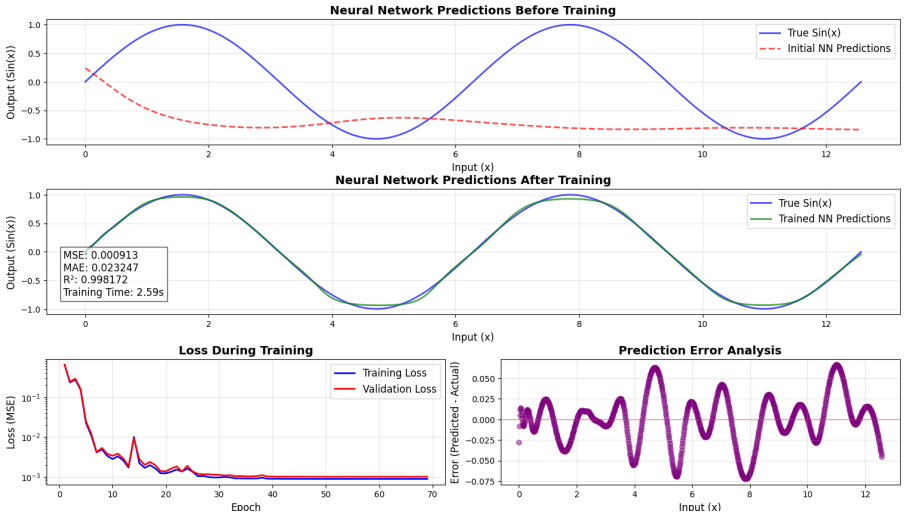

Figure 12: Descenso de Gradiente Basado en Momento

La gráfica Loss During Training muestra una disminución rápida y al mismo tiempo muy variable de la función de pérdida (MSE). Luego de las 28 iteraciones se estabiliza la función de pérdida. El tiempo total de entrenamiento es de 2.59 segundos, lo que muestra una buena eficiencia computacional. En cuanto a las métricas obtenidas el MSE es de 0.000913 el cual es un valor muy bajo. El MAE es de 0.0232 y el $R^2 = 0.9917$ indicando que tiene una buena capacidad de generalización.

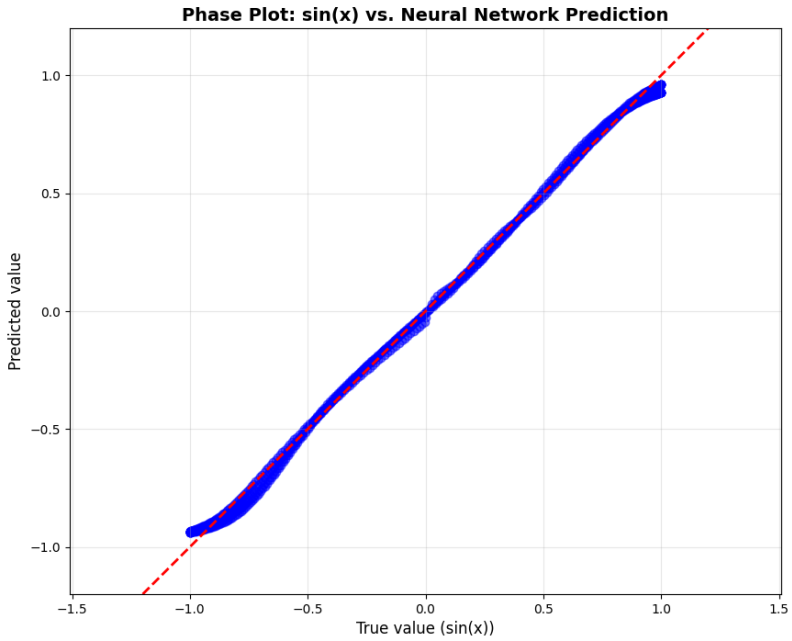

Figure 13: sin(x) vs Neural Network Prediction

El gráfico de fase sin(x) vs predicción obtenido de la implementación del GD con momento muestra que los puntos azules se alinean mucho mejor sobre la línea roja de identidad, en comparación a la implementación del GD sin momento. Se puede ver una menor variabilidad y mayor presición de los puntos azules sobre la linea roja.

5.6 Análisis y comparación

La implementación del Descenso de Gradiente Básico tiene una convergencia más lenta, ya que toma 50 iteraciones para estabilizarse mientras que el Descenso de Gradiente con Momento se estabiliza aproximadamente en 28 iteraciones. La Curva de pérdida es algo irregular al inicio mientras que con el momento la curva de pérdida es más suave y estable. Por otro lado el tiempo de entrenamiento en la implementación con momento es menor a la implementación básica. Finalmente las metricas obtenidas en la implementación con momento son mucho mejores e indican un buen rendimiento y una aproximación mas precisa y con menor error en la curva.