Predicción de la Tasa de Corrosión Usando un Modelo de Regresión Lineal

Autores: Alexi Caballero Esteban y Carlos Quiñonez

7 de mayo de 2025

índice

[1. Introducción a la Inteligencia Artificial 3](#_Toc197599637)

[2. Modelos básicos de IA 3](#_Toc197599638)

[3. Modelos seleccionado: Regresión Lineal 3](#_Toc197599639)

[4. Descripción del Experimento y Conjunto de Datos 4](#_Toc197599640)

[Tabla de Datos del Experimento 4](#_Toc197599641)

[5. Significado de la Velocidad de Corrosión y Estado del Arte 5](#_Toc197599642)

[6. Código Fuente 6](#_Toc197599643)

[4. Resultados 7](#_Toc197599644)

[5. Evaluación de los resultados 7](#_Toc197599645)

[9. Discusión de los Resultados 8](#_Toc197599646)

[10. Conclusión 8](#_Toc197599647)

# 1. Introducción a la Inteligencia Artificial

La Inteligencia Artificial (IA) es un campo de la informática que se centra en la creación de sistemas capaces de realizar tareas que normalmente requieren inteligencia humana. Estas tareas incluyen el aprendizaje, el razonamiento, la resolución de problemas, la percepción y la comprensión del lenguaje natural. La IA busca desarrollar algoritmos y modelos que permitan a las máquinas simular la inteligencia humana.

Su importancia radica en su capacidad para automatizar procesos, analizar grandes cantidades de datos de manera eficiente, mejorar la toma de decisiones y generar nuevas soluciones en diversos campos como la medicina, la ingeniería, las finanzas y la tecnología.

# 2. Modelos básicos de IA

Existen diversos modelos fundamentales en el campo de la Inteligencia Artificial, entre los que destacan:

* Aprendizaje Supervisado: Modelos que aprenden a partir de datos etiquetados, donde se conocen las entradas y las salidas deseadas. Ejemplos incluyen la regresión lineal, la regresión logística, las máquinas de vectores de soporte (SVM) y los árboles de decisión
* Aprendizaje No Supervisado: Modelos que aprenden patrones a partir de datos sin etiquetas. El clustering (como k-means) y la reducción de dimensionalidad (como PCA) son ejemplos comunes.
* Aprendizaje por Refuerzo: Modelos que aprenden a través de la interacción con un entorno, recibiendo recompensas o castigos por sus acciones. Se utiliza en áreas como la robótica y los juegos.
* Redes Neuronales: Modelos inspirados en la estructura del cerebro humano, compuestos por capas de nodos interconectados. Son la base del aprendizaje profundo (Deep Learning) y son muy efectivas en tareas complejas como el reconocimiento de imágenes y el procesamiento del lenguaje natural.

# 3. Modelos seleccionado: Regresión Lineal

En este ejercicio, se ha aplicado un modelo de regresión lineal. La regresión lineal es un algoritmo de aprendizaje supervisado que busca modelar la relación lineal entre una variable dependiente (o variable objetivo) y una o más variables independientes (o predictoras). El objetivo es encontrar la línea recta (en el caso de una sola variable independiente) o el hiperplano (en el caso de múltiples variables independientes) que mejor se ajuste a los datos.

Matemáticamente, la regresión lineal simple se puede expresar como: y^=b0+b1x*y*^​=*b*0​+*b*1​*x*, donde y^*y*^​ es el valor predicho de la variable dependiente, x*x* es la variable independiente, b0*b*0​ es la intersección con el eje y (bias), y b1*b*1​ es la pendiente (el coeficiente que indica cómo cambia y*y* por cada unidad de cambio en x*x*). El modelo aprende los valores óptimos de b0*b*0​ y b1*b*1​ a partir de los datos de entrenamiento.

En este caso particular, la variable independiente utilizada es el número de 'Days' y la variable dependiente es la 'Weight Loss (g)', que se calcula como la diferencia entre el peso inicial y el peso final del espécimen. Se busca modelar cómo la pérdida de peso se relaciona linealmente con el tiempo de exposición.

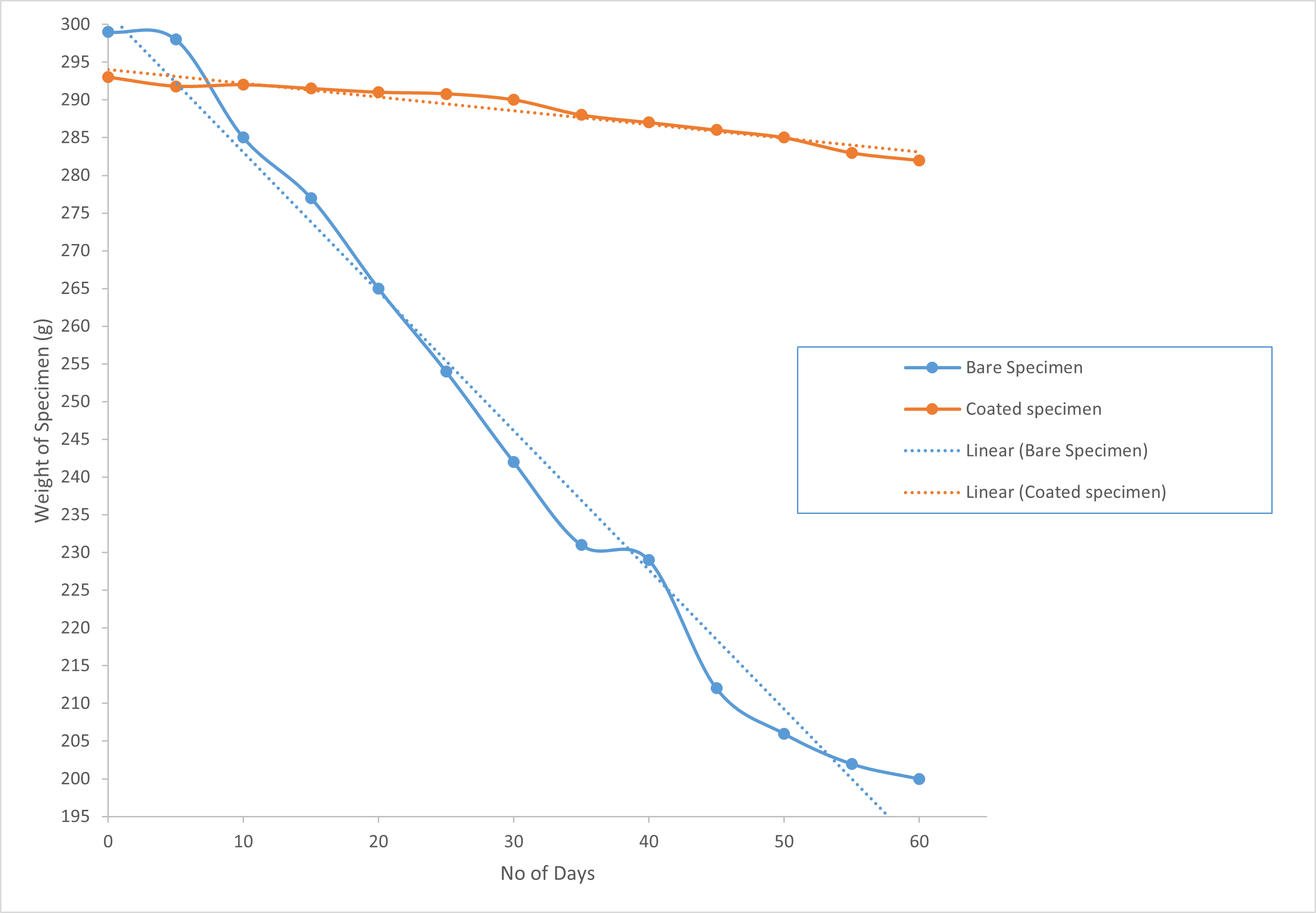
# 4. Descripción del Experimento y Conjunto de Datos

El conjunto de datos utilizado en este análisis proviene de un experimento diseñado para estudiar la velocidad de corrosión en materiales recubiertos y no recubiertos. A continuación, se detalla el contexto y el contenido del experimento:

* **Contexto**: Se recubrió una varilla metálica con cobre, níquel y cromo mediante la técnica de galvanoplastia. Posteriormente, se comparó la velocidad de corrosión de esta varilla recubierta con la de una varilla metálica desnuda (sin recubrimiento). El objetivo fue evaluar la efectividad del recubrimiento para proteger el metal base contra la corrosión.
* **Contenido**: El experimento se llevó a cabo en un tanque que contenía 2.0 litros de agua de mar sustitutiva, simulando un ambiente marino. Se colocaron en el tanque un espécimen desnudo y un espécimen recubierto, ambos con dimensiones de 22 mm de diámetro y 100 mm de longitud. Los especímenes se mantuvieron en el tanque durante un período de 60 días. Durante este tiempo, se realizaron observaciones de la corrosión a intervalos regulares, y se tomaron fotografías para documentar el proceso.

## Tabla de Datos del Experimento

|  |  |  |
| --- | --- | --- |
| Days | Weight of Bare Specimen (g) | Weight of Coated Specimen (g) |
| 0 | 299 | 293 |
| 5 | 298 | 291.8 |
| 10 | 285 | 292 |
| 15 | 277 | 291.5 |
| 20 | 265 | 291 |
| 25 | 254 | 290.8 |
| 30 | 242 | 290 |
| 35 | 231 | 288 |
| 40 | 229 | 287 |
| 45 | 212 | 286 |
| 50 | 206 | 285 |
| 55 | 202 | 283 |
| 60 | 200 | 282 |



Los datos recopilados durante el experimento incluyen mediciones del peso de los especímenes a lo largo del tiempo, lo que permite cuantificar la pérdida de material debido a la corrosión. Estos datos se utilizan para analizar cómo el recubrimiento afecta la velocidad de corrosión en comparación con el metal desnudo.

La URL de referencia para este conjunto de datos es: <https://www.kaggle.com/datasets/bobinsajigeorge/corrosion-rate>

# 5. Significado de la Velocidad de Corrosión y Estado del Arte

La velocidad de corrosión es una medida de cuánto material se pierde por unidad de tiempo debido a la corrosión. Es un parámetro crítico en ingeniería y ciencia de materiales, ya que determina la vida útil y la integridad estructural de los componentes. Comprender y predecir la velocidad de corrosión es fundamental para seleccionar materiales adecuados, diseñar sistemas de protección y prevenir fallos catastróficos.

En el estado del arte, la evaluación de la corrosión ha evolucionado desde métodos tradicionales (como la pérdida de peso, que se utiliza en este estudio) hacia técnicas electroquímicas avanzadas, microscopía electrónica, espectroscopía y modelado computacional. Se están aplicando técnicas de machine learning e IA para predecir la corrosión en tiempo real, optimizar recubrimientos y desarrollar materiales resistentes a la corrosión. La capacidad de integrar datos de diversas fuentes y simular escenarios complejos ha mejorado significativamente la precisión y la aplicabilidad de los modelos de corrosión.

# 6. Código Fuente

|  |
| --- |
| # Importamos las bibliotecas necesarias para realizar el análisis  import pandas as pd # Para manejar datos en forma de tablas (como hojas de cálculo)  import numpy as np # Para realizar cálculos matemáticos y operaciones numéricas  import matplotlib.pyplot as plt # Para crear gráficos y visualizar los resultados  from sklearn.model\_selection import train\_test\_split # Para dividir los datos en entrenamiento y prueba  from sklearn.linear\_model import LinearRegression # Para crear un modelo de regresión lineal  from sklearn.metrics import mean\_squared\_error, mean\_absolute\_error, r2\_score # Para evaluar el rendimiento del modelo  # Cargamos los datos desde un archivo llamado 'corrosion-rate.csv'  # Este archivo contiene información sobre cómo el peso de un material cambia con el tiempo debido a la corrosión  data = pd.read\_csv('corrosion-rate.csv') # Asegúrate de que el archivo esté en el mismo directorio que este script  # Mostramos las primeras filas del conjunto de datos para entender su estructura  print("Primeras filas del conjunto de datos:")  print(data.head()) # Esto nos da una idea de cómo están organizados los datos  # Verificamos si hay valores faltantes en el conjunto de datos  # Los valores faltantes pueden causar problemas en el análisis, por lo que debemos identificarlos  print("\n¿Hay valores faltantes?")  print(data.isnull().sum()) # Muestra cuántos valores faltan en cada columna  # Eliminamos las filas que tienen valores faltantes  # Esto asegura que solo trabajemos con datos completos  data = data.dropna()  # Creamos una nueva columna llamada 'Weight Loss (g)' (pérdida de peso en gramos)  # Calculamos esta pérdida como la diferencia entre el peso inicial y el peso final del material  data['Weight Loss (g)'] = data['Weight of Bare Sepcimen (g)'] - data['Weight of Coated Specimen (g)']  # Definimos las variables para el análisis:  # - 'Days' (días) será la variable independiente (X), es decir, el tiempo que el material estuvo expuesto  # - 'Weight Loss (g)' será la variable dependiente (y), es decir, cuánto peso perdió el material  X = data[['Days']] # Variable independiente (entrada del modelo)  y = data['Weight Loss (g)'] # Variable dependiente (salida del modelo)  # Dividimos los datos en dos partes:  # - Datos de entrenamiento (80%): para que el modelo aprenda  # - Datos de prueba (20%): para evaluar qué tan bien funciona el modelo  X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)  # Creamos un modelo de regresión lineal  # Este modelo intenta encontrar una línea recta que relacione el tiempo (días) con la pérdida de peso  model = LinearRegression()  # Entrenamos el modelo usando los datos de entrenamiento  # Aquí, el modelo aprende la relación entre el tiempo y la pérdida de peso  model.fit(X\_train, y\_train)  # Usamos el modelo entrenado para hacer predicciones sobre los datos de prueba  # Esto nos dice qué tan bien el modelo puede predecir la pérdida de peso en datos que no ha visto antes  y\_pred = model.predict(X\_test)  # Calculamos métricas para evaluar el rendimiento del modelo:  # - MSE (Error Cuadrático Medio): mide el promedio de los errores al cuadrado (valores predichos vs. reales)  # - MAE (Error Absoluto Medio): mide el promedio de las diferencias absolutas entre valores predichos y reales  # - R² (Coeficiente de Determinación): mide qué tan bien el modelo explica la variabilidad de los datos  mse = mean\_squared\_error(y\_test, y\_pred)  mae = mean\_absolute\_error(y\_test, y\_pred)  r2 = r2\_score(y\_test, y\_pred)  # Mostramos las métricas en la consola para entender el rendimiento del modelo  print("\nMétricas de evaluación del modelo:")  print(f"Error cuadrático medio (MSE): {mse:.2f}")  print(f"Error absoluto medio (MAE): {mae:.2f}")  print(f"Coeficiente de determinación (R²): {r2:.2f}")  # Creamos un gráfico para comparar los valores reales (observados) con los valores predichos por el modelo  plt.figure(figsize=(10, 6)) # Configuramos el tamaño del gráfico  plt.scatter(y\_test, y\_pred, color='blue', edgecolor='k', alpha=0.7) # Puntos que representan las predicciones  plt.plot([y\_test.min(), y\_test.max()], [y\_test.min(), y\_test.max()], 'r--', linewidth=2) # Línea ideal (predicción perfecta)  plt.xlabel('Valores reales (Pérdida de peso)') # Etiqueta del eje X  plt.ylabel('Valores predichos (Pérdida de peso)') # Etiqueta del eje Y  plt.title('Comparación de valores reales vs. predichos') # Título del gráfico  plt.grid(True) # Agregamos una cuadrícula para facilitar la lectura  plt.show() # Mostramos el gráfico |

# 4. Resultados

* Después de entrenar el modelo con los datos de corrosión, se obtuvieron las siguientes métricas de evaluación:
* Error cuadrático medio (MSE): 20.62
* Error absoluto medio (MAE): 3.98
* Coeficiente de determinación (R²): 0.99

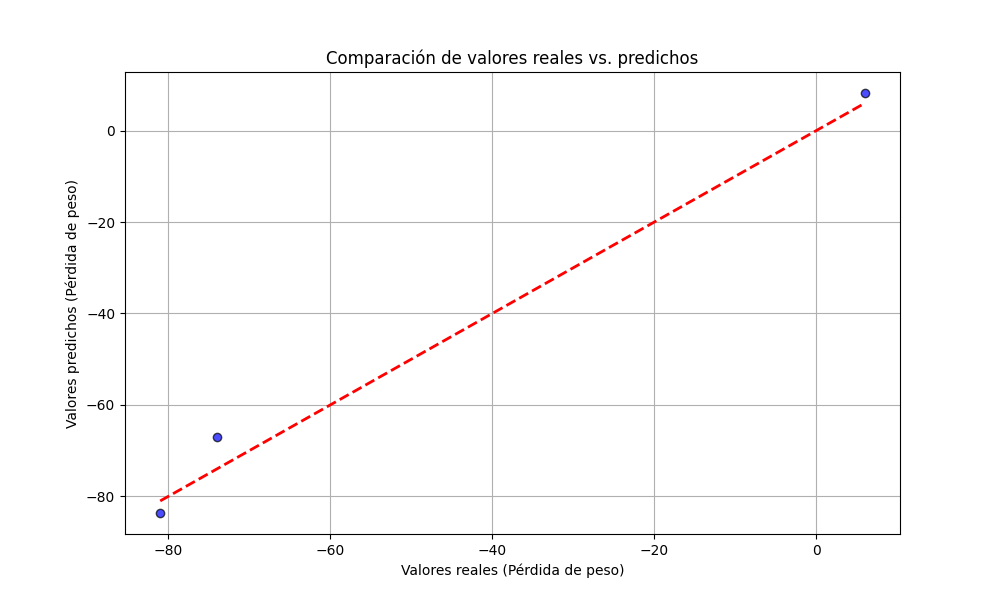
# 5. Evaluación de los resultados

Las métricas de evaluación proporcionan una indicación del rendimiento del modelo de regresión lineal sobre los datos de prueba.

Error cuadrático medio (MSE): 20.62. Este valor representa el promedio de los errores al cuadrado entre los valores reales y los valores predichos. Un valor más bajo indica un mejor ajuste del modelo.

Error absoluto medio (MAE): 3.98. Este valor representa el promedio de las diferencias absolutas entre los valores reales y los valores predichos. Al igual que el MSE, un valor más bajo es deseable.

Coeficiente de determinación (R²): 0.99. Este valor indica la proporción de la varianza en la variable dependiente que es predecible a partir de la variable independiente. Un valor de R² cercano a 1 sugiere que el modelo explica una gran parte de la variabilidad de los datos.

****

La gráfica muestra la dispersión de los valores predichos en función de los valores reales. Idealmente, los puntos deberían estar cerca de la línea roja discontinua, que representa una predicción perfecta. La dispersión observada indica la magnitud de los errores de predicción.

# 9. Discusión de los Resultados

El modelo de regresión lineal presenta un alto coeficiente de determinación (R² = 0.99), lo que sugiere que el modelo explica una proporción significativa de la variabilidad en la pérdida de peso en función del tiempo. Esto indica una fuerte relación lineal entre las dos variables, lo cual es consistente con la observación de que la corrosión a menudo progresa de manera relativamente constante en ciertos entornos.

Sin embargo, los valores de MSE (20.62) y MAE (3.98) indican la presencia de cierto error en las predicciones. Aunque el R² es alto, estos errores pueden ser importantes en aplicaciones donde la precisión es crítica. Es posible que la linealidad no sea perfecta en todo el rango de datos, o que otros factores (no incluidos en el modelo) influyan en la corrosión.

La gráfica de valores reales vs. predichos confirma visualmente la fuerte correlación, pero también revela la dispersión de los puntos alrededor de la línea de predicción perfecta. Esta dispersión sugiere que el modelo podría beneficiarse de la inclusión de variables adicionales o de un modelo no lineal para capturar mejor la complejidad del fenómeno de corrosión.

# 10. Conclusión

El modelo de regresión lineal aplicado a los datos de pérdida de peso en función del tiempo ('Days') proporciona una buena aproximación de la relación entre estas variables, como lo demuestra el alto valor de R². Esto sugiere que el tiempo es un factor significativo en la predicción de la corrosión, al menos en las condiciones experimentales de este estudio.

No obstante, la presencia de errores (indicados por MSE y MAE) y la dispersión en la gráfica sugieren que el modelo no es perfecto. Para mejorar la precisión, se podrían considerar los siguientes pasos:

Incorporar otras variables: Factores como la temperatura, la concentración del electrolito, el pH y la composición del material podrían influir en la velocidad de corrosión. Utilizar modelos no lineales: Si la relación entre la pérdida de peso y el tiempo no es estrictamente lineal, modelos como la regresión polinómica o modelos exponenciales podrían ser más apropiados. Aumentar el tamaño de la muestra: Un conjunto de datos más grande y diverso podría mejorar la generalización del modelo. En conclusión, aunque el modelo lineal proporciona información valiosa, se recomienda un análisis más exhaustivo para desarrollar un modelo predictivo más robusto y preciso de la corrosión.