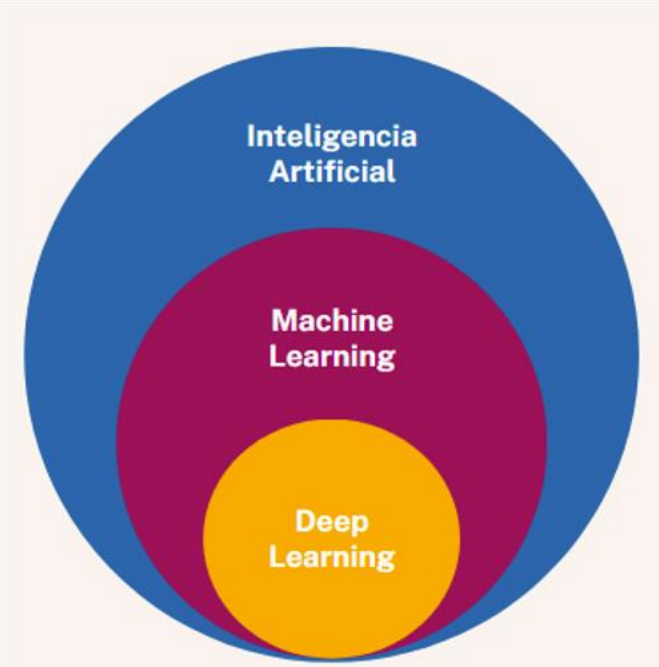
INTRODUCCIÓN A MACHINE LEARNING

Inteligencia Artificial (IA):

La Inteligencia Artificial es el campo de estudio que se centra en la creación de máquinas o sistemas capaces de realizar tareas que normalmente requieren inteligencia humana. Esto incluye cosas como razonar, aprender, percibir y tomar decisiones. La IA puede abarcar desde sistemas que responden a preguntas específicas hasta robots autónomos.



Machine Learning (ML):

El Machine Learning es un subconjunto de la inteligencia artificial. Se refiere a la técnica de utilizar algoritmos y modelos estadísticos para que las máquinas mejoren su rendimiento en una tarea específica a través de la experiencia o datos, sin ser explícitamente programadas para esa tarea. En esencia, se trata de enseñar a las máquinas a aprender de los datos.

Deep Learning (DL)

El Deep Learning es un subconjunto del Machine Learning. Utiliza redes neuronales artificiales profundas, que son algoritmos inspirados en la estructura y función del cerebro humano, para modelar y entender patrones complejos en grandes cantidades de datos. El Deep Learning es particularmente efectivo para tareas como el reconocimiento de voz e imagen, donde puede identificar y clasificar información con una precisión que a menudo supera a la humana.

Qué es Shallow Learning

El "Shallow Learning" o aprendizaje superficial en el campo del aprendizaje automático se refiere a métodos y modelos que no implican estructuras profundas como las que se encuentran en las redes neuronales profundas (Deep Learning). Estos modelos tienden a tener una arquitectura más simple y se basan en una sola capa de procesamiento o en unas pocas capas.

Características del Shallow Learning:

1. ***Arquitectura Simple***: A diferencia del Deep Learning, que utiliza múltiples capas ocultas, el Shallow Learning suele trabajar con modelos que tienen una única capa de procesamiento (como en perceptrones) o solo unas pocas capas.
2. ***Menor Necesidad de Recursos Computacionales***: Debido a su simplicidad, estos modelos suelen requerir menos potencia de cómputo y menos datos para entrenar de manera efectiva en comparación con los modelos de Deep Learning.
3. ***Amplia Aplicabilidad***: Aunque no son tan potentes como los modelos de Deep Learning para tareas complejas como el procesamiento del lenguaje natural o la visión por computadora, los modelos de Shallow Learning son muy efectivos para una amplia gama de problemas de aprendizaje automático, especialmente aquellos donde la relación entre las características y la variable objetivo es menos compleja.
4. ***Modelos Tradicionales***: Incluyen algoritmos como la regresión lineal y logística, máquinas de vectores de soporte (SVM), K vecinos más cercanos (KNN), y árboles de decisión. Estos modelos han sido la base del aprendizaje automático durante muchas décadas.
5. ***Interpretabilidad***: A menudo, los modelos de Shallow Learning son más fáciles de interpretar y explicar que los modelos de Deep Learning. Esto puede ser una ventaja en campos donde la explicabilidad es crucial, como en la medicina o en finanzas.

En resumen, el Shallow Learning abarca técnicas de aprendizaje automático que son relativamente menos complejas y profundas en términos de arquitectura de modelo, pero siguen siendo poderosas y ampliamente utilizadas para muchas aplicaciones prácticas.

Tipos de Aprendizaje en Shallow Learning

- **Aprendizaje Supervisado**: Se entrena al modelo con datos etiquetados. El objetivo es que el modelo aprenda a predecir etiquetas a partir de características. Ejemplos comunes son la **clasificación y la regresión**.

    - **Regresión**: **Se utiliza para predecir valores continuos**. Esto significa que **la salida o la predicción es un número que puede variar dentro de un rango**. Un ejemplo clásico es la predicción de precios de casas. Aquí, el modelo de regresión analiza características como el tamaño de la casa, la ubicación, etc., y predice un precio, que es un valor numérico. Los tipos comunes de regresión incluyen regresión lineal y regresión polinómica.

    - **Clasificación**: A diferencia de la regresión, **la clasificación se utiliza para predecir una clase o categoría**. En este enfoque, el modelo asigna una etiqueta a la entrada. Por ejemplo, en un sistema de clasificación de correos electrónicos, el modelo podría clasificar los correos como 'spam' o 'no spam'. Aquí, no hay valores numéricos continuos como salida, sino categorías discretas. Los tipos comunes de clasificación incluyen clasificación binaria (donde hay dos categorías posibles) y clasificación multiclase (donde hay más de dos categorías).

  Ambos métodos son cruciales en el campo del aprendizaje automático y se utilizan en una variedad de aplicaciones, desde el análisis de tendencias de mercado hasta el reconocimiento de imágenes y más allá. La elección entre regresión y clasificación depende principalmente del tipo de problema y la naturaleza de los datos disponibles.

- **Aprendizaje No Supervisado**: **Se trabaja con datos sin etiquetar**. El objetivo es explorar la estructura de los datos para extraer patrones.

    - **El "clustering" o agrupamiento** en aprendizaje automático es un método utilizado para dividir un conjunto de datos en grupos, de tal manera que los datos en cada grupo sean más similares entre sí en comparación con los de otros grupos. Es una técnica de aprendizaje no supervisado, lo que significa que el proceso se lleva a cabo sin intervención humana para definir o ajustar los grupos. Aquí, el sistema intenta identificar estructuras o patrones en los datos por sí mismo.
      Imagina que tienes un montón de frutas mezcladas y quieres organizarlas. Si las clasificas en grupos según su tipo, color, tamaño o sabor, estás haciendo algo muy similar al clustering. En el aprendizaje automático, esto se hace analizando las características de los datos, como las dimensiones de las frutas en nuestro ejemplo.
      Los algoritmos de clustering son muy útiles en diversos campos para segmentar datos en grupos naturales y descubrir patrones ocultos, como en el análisis de datos de clientes, la clasificación de genes en la biología, o incluso en el reconocimiento de patrones en imágenes y sonidos.

- **Aprendizaje por Refuerzo**: El modelo aprende a tomar decisiones a través de ensayo y error, basándose en recompensas recibidas por sus acciones.

  El aprendizaje por refuerzo en el aprendizaje automático es como enseñar a un niño a montar en bicicleta mediante un sistema de recompensas y penalizaciones. Imagina que el niño es un agente (un programa) en un ambiente (como un juego o un simulador). Cada vez que el niño (el agente) toma una decisión correcta, como mantener el equilibrio, recibe una golosina (una recompensa). Si toma una decisión incorrecta, como caerse, recibe una reprimenda leve (una penalización).

El objetivo del niño es aprender a montar la bicicleta de la manera más eficiente posible, maximizando las golosinas y evitando las reprimendas. De manera similar, en el aprendizaje por refuerzo, el agente aprende a realizar tareas o tomar decisiones efectivas en su ambiente, buscando maximizar sus recompensas a lo largo del tiempo. Este tipo de aprendizaje es muy utilizado en situaciones donde las instrucciones no son claras o son demasiado complejas para programar directamente, como juegos, navegación de robots, o en la optimización de sistemas.

Herramientas y Lenguajes Comunes

Python es el lenguaje de programación más popular en ML, gracias a su sintaxis clara y a la amplia disponibilidad de librerías.

Librerías como **NumPy**, **Pandas** para manipulación de datos; **Matplotlib** y **Seaborn** para visualización; **Scikit-learn** para modelos de ML; **TensorFlow** y **PyTorch** para aprendizaje profundo.

El Proceso de Machine Learning

- **Recolección de Datos**: Obtener un conjunto de datos relevantes para tu problema.
- **Preprocesamiento de Datos**: Limpiar y preparar los datos. Incluye manejo de valores faltantes, normalización, codificación de variables categóricas, etc.
  En el campo del aprendizaje automático (machine learning), el preprocesamiento y la limpieza de datos son pasos cruciales para asegurar la calidad y eficacia de los modelos. Aquí hay algunas tareas comunes de preprocesamiento y limpieza:
  1. ***Limpieza de Datos***:
     - **Tratar con valores faltantes**: Los datos pueden tener valores faltantes. Estos pueden ser imputados (rellenados) o eliminados, dependiendo del contexto y la cantidad de datos faltantes.
     - **Eliminar duplicados**: Los registros duplicados pueden sesgar los resultados del modelo y deben ser eliminados.
  2. ***Transformación de Datos***:
     - **Normalización/Estandarización**: Consiste en escalar los datos para que tengan un rango específico (como 0 a 1) o una distribución con una media de 0 y una desviación estándar de 1. Esto es importante para modelos que son sensibles a la magnitud de los datos, como las redes neuronales.
     - **Codificación de variables categóricas**: Los modelos de aprendizaje automático generalmente requieren entradas numéricas. Las variables categóricas (como "rojo", "azul", "verde") deben ser convertidas a números, a través de técnicas como la codificación one-hot o la codificación de etiquetas.
  3. ***Reducción de Dimensionalidad***:
     - **Análisis de Componentes Principales (PCA)**: Reduce la cantidad de variables, manteniendo la mayor cantidad de información original.
     - **Selección de Características**: Consiste en elegir las características más relevantes para el modelo, eliminando aquellas que aportan poca o ninguna información útil.
  4. ***Manejo de Datos Desbalanceados***:
     - En conjuntos de datos donde las clases están desbalanceadas (una clase es mucho más frecuente que otras), se pueden usar técnicas de sobremuestreo o submuestreo para equilibrarlas.
  5. ***Tratamiento de Outliers (Valores Atípicos)***:
     - Identificar y tratar los outliers, ya que pueden afectar negativamente el rendimiento del modelo.
       Los "outliers" en el aprendizaje automático s**e refieren a datos que son significativamente**

diferentes o se desvían mucho de la mayoría de los otros datos. Imagina que estás en una clase donde casi todos los estudiantes miden entre 1.50 y 1.80 metros, pero hay un estudiante que mide 2.10 metros. Ese estudiante sería un outlier en términos de altura.

En el contexto de los datos, los outliers pueden aparecer por varias razones, como errores de medición o entrada de datos, variabilidad en los datos, o simplemente porque representan una ocurrencia rara. Por ejemplo, en un conjunto de datos sobre el precio de las casas, una mansión extremadamente cara sería un outlier comparada con las casas ordinarias.

Es importante identificar y tratar adecuadamente los outliers en el aprendizaje automático porque pueden tener un gran impacto en los resultados de los modelos. Pueden distorsionar el análisis estadístico y los modelos predictivos, llevando a conclusiones inexactas. Dependiendo del caso, los outliers pueden ser eliminados, ajustados o analizados por separado para asegurar que el modelo de aprendizaje automático funcione correctamente y sea confiable.

6. ***Ingeniería de Características***:
    - Crear nuevas características a partir de las existentes que pueden ser más informativas y útiles para los modelos.
7. ***División de Datos***:
    - Dividir el conjunto de datos en conjuntos de entrenamiento, validación y prueba, para entrenar el modelo, ajustar parámetros y evaluar su rendimiento.

Cada uno de estos pasos es importante para asegurar que los datos sean adecuados y útiles para construir modelos de aprendizaje automático eficientes y precisos. La elección y la aplicación de estas técnicas dependen en gran medida de la naturaleza del problema y del conjunto de datos específico con el que se esté trabajando.

- **Exploración de Datos**: Analizar y visualizar los datos para encontrar patrones y tendencias.
- **División de Datos**: Dividir los datos en conjuntos de entrenamiento y prueba.
- **Construcción del Modelo**: Seleccionar un modelo de ML y entrenarlo con los datos. Esto puede incluir la selección de hiperparámetros.
- **Evaluación del Modelo**: Probar el modelo con el conjunto de prueba y evaluar su rendimiento.
- **Ajuste y Optimización**: Ajustar el modelo y sus hiperparámetros para mejorar su rendimiento.
- **Despliegue**: Implementar el modelo en un entorno de producción.

Herramientas matemáticas para Machine Learning

Para trabajar en inteligencia artificial (IA), es importante tener conocimientos en varias herramientas matemáticas. Aquí hay algunas esenciales y una explicación sencilla de cada una:

- **Álgebra Lineal**: Es el estudio de vectores, matrices, y sistemas de ecuaciones lineales. En IA, se usa para manejar y transformar conjuntos de datos. Por ejemplo, las imágenes en el procesamiento de imágenes son tratadas como matrices de píxeles.

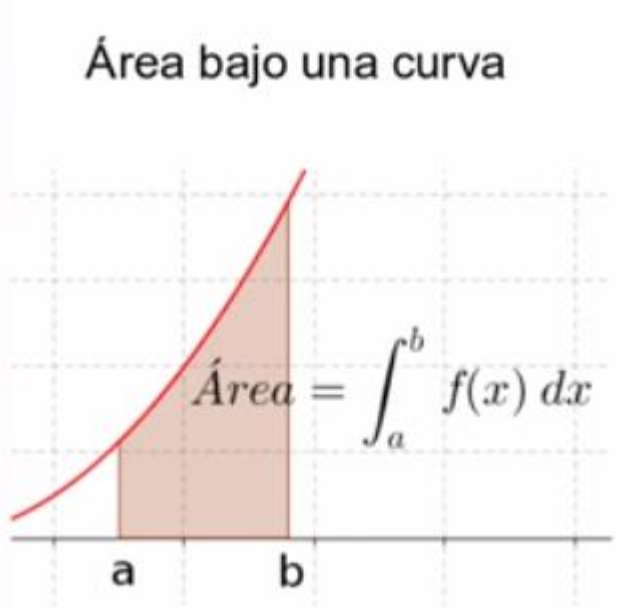
- **Cálculo**: Esencialmente, es el estudio de cómo cambian las cosas. En IA, se usa para optimizar algoritmos, especialmente en aprendizaje profundo, donde se necesitan calcular gradientes para ajustar los pesos de las redes neuronales.

El cálculo es una rama de las matemáticas que se centra en el estudio de los cambios. Hay dos áreas principales en el cálculo: el cálculo diferencial y el cálculo integral. Aquí te explico cada una de manera sencilla:

- Cálculo Diferencial:
  - **Derivadas**: Imagina que conduces un coche y quieres saber qué tan rápido estás acelerando en un punto específico del camino. La derivada te da esa información. Es como una instantánea que te dice cuán rápido cambia una cantidad en un momento particular. En matemáticas, la derivada de una función mide cómo cambia el valor de esa función (como la posición) con respecto a un cambio en otra cantidad (como el tiempo).
- Cálculo Integral:
  - **Integrales**: Siguiendo con el ejemplo del coche, ahora imagina que quieres saber qué distancia total has recorrido durante un viaje. La integral te da esa información. Es como sumar pequeñas distancias recorridas en cada instante de tiempo para obtener la distancia total. En matemáticas, las integrales suman infinitos pedazos pequeños para encontrar áreas, volúmenes, o la suma total de un conjunto de valores.

Ambas áreas están interconectadas por el Teorema Fundamental del Cálculo, que establece una relación entre la derivada y la integral. En la inteligencia artificial, el cálculo es fundamental, especialmente en el aprendizaje automático, donde se usa para optimizar modelos y algoritmos, como ajustar los pesos en las redes neuronales para mejorar su precisión y rendimiento.

La relación entre el área debajo de una curva y la integral es muy directa y fundamental en el cálculo. Cuando calculamos la integral de una función sobre un intervalo específico, estamos encontrando el área total bajo la curva de esa función en ese intervalo.



Para entenderlo mejor, imagina la gráfica de una función en un sistema de coordenadas. La región que queda entre el eje horizontal (generalmente el eje x) y la curva de la función, limitada por dos valores en el eje x, es el área que nos interesa.

La integral de la función en ese intervalo nos da un valor numérico que representa esa área. Si la función está por encima del eje x en todo el intervalo, esta área es positiva. Si está por debajo, el área es negativa. En el caso de que la función cruce el eje x, la integral suma las áreas positivas y resta las negativas.

- **Estadística y Probabilidad**: Son fundamentales para entender y modelar la incertidumbre en los datos. En IA, se usan para hacer predicciones y tomar decisiones. Por ejemplo, determinar la probabilidad de que un correo electrónico sea spam.

Parámetros e Hiperparámetros

En el aprendizaje automático, los términos "parámetro" e "hiperparámetro" tienen significados específicos y son fundamentales para entender cómo se construyen y ajustan los modelos:

Parámetro:

- Un parámetro es una variable interna de un modelo de aprendizaje automático y se aprende a partir de los datos. Por ejemplo, en un modelo de regresión lineal, los coeficientes de las variables independientes son parámetros (pesos y sesgos).

$$y = mx + b$$

- Los parámetros son la parte del modelo que se ajusta automáticamente durante el entrenamiento. El modelo los modifica para aprender patrones y hacer predicciones precisas.
- No los establecemos manualmente; en cambio, el algoritmo los encuentra durante el proceso de aprendizaje.

Hiperparámetro:

- **Un hiperparámetro es una configuración externa del modelo y no se aprende a partir de los datos**. Por ejemplo, la tasa de aprendizaje en una red neuronal o la profundidad máxima de un árbol de decisión son hiperparámetros (ejemplo: nº de clusters, test_size, etc…)
- Los hiperparámetros son como configuraciones que guían el proceso de aprendizaje del modelo. **Tienen que ser definidos antes de comenzar a entrenar el modelo**, y su ajuste se hace generalmente de forma manual o mediante técnicas de búsqueda automática como la búsqueda en cuadrícula (grid search) o la búsqueda aleatoria.
- Elegir los hiperparámetros correctos puede ser crucial para el rendimiento del modelo, ya que afectan cómo y qué tan bien el modelo aprende de los datos.

En resumen, los parámetros son aprendidos directamente por el modelo a partir de los datos, mientras que los hiperparámetros son establecidos por el usuario para optimizar el proceso de aprendizaje. La selección y ajuste de hiperparámetros es una parte importante del desarrollo de modelos de aprendizaje automático.

Algoritmos vs Modelos Algorítmicos

En el campo del aprendizaje automático (machine learning), los términos "algoritmo" y "modelo algorítmico" a menudo se usan y a veces pueden confundirse, pero tienen significados distintos:

Algoritmo:

- Un algoritmo en aprendizaje automático **es un conjunto de reglas o instrucciones diseñadas para realizar una tarea específica**. Por ejemplo, el algoritmo de regresión lineal, el algoritmo de redes neuronales, y el algoritmo K-means son todos procedimientos que detallan cómo aprender de los datos y hacer predicciones o agrupaciones.
  El algoritmo define el proceso de aprendizaje: cómo se ajustan los parámetros del modelo a partir de los datos, cómo se minimiza el error, cómo se realiza la clasificación, etc.

Modelo Algorítmico:

- **Un modelo algorítmico se refiere a la representación específica aprendida de los datos usando un algoritmo**. Por ejemplo, una vez que entrenas un algoritmo de regresión lineal con tus datos, el conjunto específico de coeficientes que se obtiene (la pendiente y la intersección, en el caso más simple) constituye tu modelo algorítmico.
  El modelo es, por lo tanto, el resultado concreto del algoritmo aplicado a un conjunto de datos. Incluye los parámetros específicos y la estructura que ha aprendido de los datos para hacer predicciones o tomar decisiones.
  En resumen, el algoritmo es como la receta general, mientras que el modelo algorítmico es el plato específico que preparas siguiendo esa receta con tus ingredientes (datos). Cada vez que usas el mismo algoritmo con

diferentes datos o diferentes configuraciones de hiperparámetros, obtendrás un modelo algorítmico diferente.

Librerías de Shallow Learning

Scikit-learn: Es una de las librerías más populares para el aprendizaje automático en Python. Proporciona una amplia gama de algoritmos supervisados y no supervisados, incluyendo clasificación, regresión, clustering, y reducción de dimensionalidad. Es conocida por su facilidad de uso y su sólida documentación.

Scikit-learn (sklearn) es una de las librerías más versátiles y ampliamente utilizadas en el campo del aprendizaje automático en Python. Ofrece una gran variedad de sublibrerías y modelos para diferentes tareas de aprendizaje automático. A continuación, detallo algunas de las más importantes:

- Modelos de Clasificación:
  - Incluyen algoritmos como Regresión Logística, Máquinas de Vectores de Soporte (SVM), K Vecinos más Cercanos (KNN), Árboles de Decisión, Bosques Aleatorios, y Naive Bayes.
  - Se utilizan para tareas donde el objetivo es categorizar las entradas en distintas clases.
- Modelos de Regresión:
  - Ofrecen algoritmos como Regresión Lineal, Ridge, Lasso, Árboles de Decisión y Bosques Aleatorios para regresión.
  - Se aplican en problemas donde se debe predecir una variable continua.
- Modelos de Clustering:
  - Proporciona métodos para el agrupamiento de datos, como K-Means, Clustering Jerárquico, DBSCAN, y Clustering Espectral.
  - Son útiles para agrupar datos en conjuntos basados en su similitud.
- Reducción de Dimensionalidad:
  - Incluye técnicas como Análisis de Componentes Principales (PCA), Análisis Discriminante Lineal (LDA), y t-SNE.
  - Estas técnicas son importantes para reducir el número de variables de entrada sin perder demasiada información.
- Selección de Modelos:
  - Ofrece herramientas para la selección de modelos y ajuste de hiperparámetros, como la búsqueda en cuadrícula (GridSearchCV) y la validación cruzada (cross-validation).
  - Son cruciales para mejorar el rendimiento del modelo y evitar el sobreajuste.
- Preprocesamiento
  - Proporciona una amplia gama de herramientas de preprocesamiento y transformación de datos, como escalado de características, codificación de variables categóricas, y generación de características polinómicas.
  - Estas herramientas son esenciales para preparar los datos antes del entrenamiento del modelo.

- Métricas:
  - Incluye una variedad de métricas para evaluar el rendimiento de los modelos, como precisión, recall, F1-score para clasificación, y error cuadrático medio para regresión.
    Estas métricas son fundamentales para evaluar y comparar diferentes modelos.

Métricas de evaluación

La evaluación de un modelo de machine learning es un paso esencial para entender cómo de bien está funcionando. Dependiendo de si tu modelo es de clasificación, regresión o algún otro tipo de modelo, hay diferentes métricas que puedes usar. Aquí te dejo algunas de las métricas más comunes:

Para Clasificación:

- **Precisión (Accuracy)**: Es la proporción de predicciones correctas sobre el total de predicciones. Es una métrica útil cuando las clases están bien balanceadas.
- **Recall (Sensitivity o True Positive Rate)**: Es la proporción de verdaderos positivos que se identificaron correctamente. Es útil en situaciones donde los falsos negativos son más preocupantes que los falsos positivos.
- **Precision**: Es la proporción de verdaderos positivos entre todas las predicciones positivas. Es útil en situaciones donde los falsos positivos son más preocupantes que los falsos negativos.
- **F1 Score**: Es la media armónica de Precision y Recall. Intenta equilibrar ambas métricas y es más útil que la precisión cuando tienes una distribución de clases desequilibrada.
- **Area Under the ROC Curve (AUC-ROC)**: Es la probabilidad de que un clasificador ordene un ejemplo positivo aleatorio más alto que un ejemplo negativo aleatorio. Un AUC de 1 indica una clasificación perfecta, mientras que un AUC de 0.5 indica un clasificador aleatorio.

Para Regresión:

- **Mean Absolute Error (MAE)**: Es la media del valor absoluto de los errores. Da una idea de cuánto te estás equivocando en tus predicciones.
- **Mean Squared Error (MSE)**: Es la media de los cuadrados de los errores. Da más peso a los errores grandes.
- **Root Mean Squared Error (RMSE)**: Es la raíz cuadrada de la media de los cuadrados de los errores. También da más peso a los errores grandes y tiene la misma unidad que la variable objetivo.
- **R-squared (Coeficiente de Determinación)**: Proporciona una medida de cuánto de la variabilidad en la variable objetivo puede ser explicada por las características del modelo. Un valor de 1 indica que el modelo explica toda la variabilidad, mientras que un valor de 0 indica que el modelo no explica nada de la variabilidad.

Para Clustering:

- **Silhouette Score**: Esta métrica se usa para medir cuán bien se han agrupado los datos en los clusters.
- **Davies-Bouldin Index**: Este índice indica la similitud media entre los clusters. Los valores más bajos indican una mejor partición.

Estas son solo algunas de las métricas disponibles. La elección de la métrica depende en gran medida de tu problema y de lo que más te importe en tus predicciones.