



FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA
UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

Nombre de la materia

Grupo: 001 | Equipo: 09

Investigación: Algoritmos Supervisados y Librerías en Python

Actividad Fundamental 5

Fecha: 07/11/2025

Profesor: Dr. Erick de Jesús Ordaz Rivas

Integrantes

Nombre	Matrícula	Hora clase
Orlando Alvarado Vargas	2226968	V1
Diego Alonso Carrillo Castillo	2144556	V1

Índice

1. Introducción	3
2. Algoritmos Principales de Aprendizaje Supervisado	3
2.1. Regresión lineal	3
2.1.1. Supuestos sobre las Variables Explicativas:	4
2.1.2. Supuestos sobre los Términos de Error:	4
2.1.3. Ventajas	5
2.1.4. Desventajas	5
2.1.5. Aplicaciones Típicas	5
2.2. Regresión logística	6
2.2.1. Ventajas	8
2.2.2. Desventajas	8
2.2.3. Aplicaciones Típicas	8
2.3. Árboles de decisión	8
2.4. Redes neuronales	12
2.4.1. Principales métricas de evaluación	14
2.4.2. Ventajas	15
2.4.3. Desventajas	15
2.5. Comparativo de librerías y algoritmos	16
2.6. Buenas prácticas y consideraciones	16
2.7. Conclusiones	17

1. Introducción

El aprendizaje supervisado es una categoría del aprendizaje automático que se utiliza para entrenar algoritmos para predecir resultados y reconocer patrones. Con este método se facilita la creación de modelos complejos que pueden hacer predicciones precisas, los cuales se usan ampliamente en varias industrias y campos como la salud, marketing, servicios financieros, etc.

Los algoritmos basados en el aprendizaje supervisado necesitan una gran cantidad de datos etiquetados, que consisten en pares de entradas y salidas correctas. Su función principal es analizar estos conjuntos de datos para inferir un modelo que pueda predecir un valor de resultado deseado cuando se le ingresan datos nuevos.

El aprendizaje supervisado se divide generalmente en dos categorías:

- **Clasificación:** Se utilizan para agrupar datos mediante la predicción de una etiqueta categórica o una variable de salida basada en los datos de entrada. Se utiliza cuando las variables de salida son categóricas (por ejemplo, "spam." o "no spam").
- **Regresión:** Se utiliza para predecir un valor real o continuo (por ejemplo, el precio de una acción).

En esta investigación se abordarán los principales algoritmos de aprendizaje supervisado y las librerías de Python que los implementan, así como sus aplicaciones, ventajas y desventajas, entre otras características.

2. Algoritmos Principales de Aprendizaje Supervisado

2.1. Regresión lineal

La regresión lineal es una técnica de modelado estadístico que se emplea para describir la relación entre dos variables y realizar predicciones. Su objetivo es predecir el valor de una variable desconocida utilizando el valor de otra variable conocida y relacionada, generando una ecuación lineal modelando la variable dependiente y la variable independiente.

Este algoritmo se emplea comúnmente para convertir datos crudos en conocimiento práctico y predecir tendencias futuras, como estimar ventas, predecir el precio de una vivienda o analizar el impacto de la publicidad. La regresión lineal es fundamental en el machine learning y el análisis de datos debido a su simplicidad interpretativa y su facilidad para implementarse en software.

En machine learning, una función de pérdida cuantifica las pérdidas generadas por errores cometidos. Es crucial minimizar la pérdida esperada. La función de pérdida, también conocida como Pérdida Cuadrática, mide el error para un único punto elevando al cuadrado la diferencia entre el valor real (y) y el valor predicho por el modelo:

$$L(y, \hat{y}) = (y - \hat{y})^2 \quad (1)$$

La función de elevar al cuadrado es penalizar mucho más los errores grandes que los pequeños y se asegura de que todos los errores sean positivos.

El criterio de optimización para todo el modelo es minimizar el promedio de estas pérdidas, conocido como el Error Cuadrático Medio (MSE). El método más común para lograr esto se llama Mínimos Cuadrados Ordinarios (OLS), y su objetivo es encontrar la linea que produce la mínima suma total de los errores al cuadrado.

La regresión lineal tiene unos supuestos o condiciones de uso que, cuando se cumplen, el modelo garantiza resultados correctos. Estas condiciones son:

2.1.1. Supuestos sobre las Variables Explicativas:

- **Linealidad:** Es el supuesto más básico y se refiere a que la relación de los parámetros debe ser lineal
- **No Multicolinealidad:** Este supuesto es solo aplicable a la regresión múltiple. Indica que las variables independientes no deben estar altamente correlacionadas entre sí, de otra manera, al modelo le resultaría difícil aislar el impacto individual de cada variable, afectando así la precisión de los coeficientes.

2.1.2. Supuestos sobre los Términos de Error:

- **Homocedasticidad:** Dicta que la varianza de los términos de error debe ser constante para todas las observaciones.
- **No Autocorrelación:** Los errores deben ser independientes entre sí. Puede causar que la precisión del modelo parezca exagerada.
- **Distribución Gaussiana:** Los errores deben seguir una distribución normal, como una campana de Gauss

Los errores de predicción son muy comunes en machine learning, por lo que es muy importante conocer las métricas que se utilizan para entender su comportamiento y saber qué tan correcto es el algoritmo. Existen diferentes métricas para poder conocer el error:

- **Error Absoluto Medio (MAE):** Esta métrica es la medida de la diferencia entre dos valores. Permite saber que tan diferente es el valor predicho y el valor real.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

- **Error Cuadrático Medio (MSE):** Previamente mencionado. Es una métrica de evaluación útil para saber que tan cerca es la linea de ajuste de la regresión a la observación. Lo que lo diferencia de otras métricas es que penaliza mucho más los errores grandes.

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

- **Raíz del Error Cuadrático Medio (RMSE):** Es la raíz cuadrada del MSE, es interpretable en las mismas unidades que la variable Y.

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

- **Coeficiente de Determinación (R², R cuadrada):** Indica que tanta variación tiene la variable dependiente que se puede predecir desde la variable independiente.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

Por último, cabe resaltar las principales ventajas y desventajas para poder determinar cuándo utilizar la regresión lineal:

2.1.3. Ventajas

- **Intepretabilidad:** Es uno de los modelos más fáciles de entender. Los coeficientes indican de forma clara y directa cómo impacta la variable independiente a la dependiente.
- **Facilidad de Implementación:** Es un modelo sencillo de entender e implementar en código.
- **Base Fundamental:** Sirve como punto de partida para entender modelos de machine learning mucho más avanzados.

2.1.4. Desventajas

- **Solo captura relaciones lineales:** Si los datos tienen un comportamiento curvo o exponencial, el modelo no se ajustará adecuadamente.
- **Sensibilidad a Outliers:** Dado que el modelo se basa en minimizar los errores al cuadrado, un solo punto de dato que esté muy lejos del resto puede mover la linea de regresión y distorsionar significativamente los resultados.
- **Correlación no implica causalidad:** Aunque el modelo muestre que x es un gran predictor de y, esto no demuestra que x cause el cambio en y

2.1.5. Aplicaciones Típicas

La regresión lineal se emplea comúnmente para convertir datos crudos en conocimiento práctico. Sus aplicaciones se pueden clasificar en tres grandes tareas:

- **Pronóstico y Proyección de Tendencias:** Se utiliza para predecir valores numéricos futuros basándose en datos históricos. Esto permite a las organizaciones anticipar resultados, como la demanda de un producto, indicadores económicos o el precio futuro de un activo.

- **Cuantificación de Relaciones (Inferencia):** Se utiliza para entender y medir la *fuerza* del efecto que tiene una variable sobre otra. Permite responder preguntas como: "¿Cuánto cambia *z*" por cada unidad que cambia "*x*", lo cual es crucial para la toma de decisiones.
- **Estimación de Valores Desconocidos:** Sirve para estimar un valor numérico actual que es desconocido o difícil de medir directamente. Por ejemplo, estimar el riesgo crediticio de un nuevo cliente, la eficiencia de un proceso industrial basándose en sus parámetros, o el valor de un bien.

2.2. Regresión logística

La Regresión Logística es un método de aprendizaje supervisado utilizado fundamentalmente para problemas de clasificación binaria. A diferencia de la regresión lineal, que predice un valor numérico continuo, la regresión logística estima una probabilidad de que una instancia pertenezca a una categoría. Para lograrlo, utiliza una o más variables independientes y transforma su combinación lineal usando una función sigmoide. Esta función garantiza que la salida sea siempre una probabilidad (un valor entre 0 y 1), la cual se utiliza para la clasificación final.

El núcleo de la regresión logística es cómo transforma la salida de una regresión lineal estándar en una probabilidad. Mientras que un modelo lineal ($w_0 + w_1x$) puede producir valores desde $-\infty$ hasta $+\infty$, una probabilidad debe estar estrictamente contenida entre 0 y 1. Para lograr esta transformación, el modelo aplica la función logística al resultado de la combinación lineal. Esta función $p(\text{pass}) = \frac{\exp(w_0 + w_1x)}{1 + \exp(w_0 + w_1x)}$ mapea cualquier valor de entrada, sin importar qué tan grande o pequeño sea, y lo mapea al rango [0, 1]. El resultado de esta función se interpreta como la probabilidad de que la instancia pertenezca a la clase "1" (la clase positiva). La fórmula para calcular esta probabilidad, $p(y = 1)$, es la siguiente:

$$p(\text{pass}) = \frac{\exp(w_0 + w_1x)}{1 + \exp(w_0 + w_1x)}$$

Donde:

- $p(\text{pass})$ representa la probabilidad de que un estudiante pase el examen.
- La función devuelve los valores entre 0 y 1, lo que permite interpretar el resultado como una probabilidad

Umbral de decisión:

- Si $p(\text{pass}) \geq 0.5$ el estudiante aprobará.
- Si $p(\text{pass}) < 0.5$ el estudiante reprobará.

La regresión logística necesita una función de pérdida diferente a la de la regresión lineal. Este algoritmo necesita una función llamada Entropía Cruzada, la cual mide lo preciso que es el modelo para predecir las probabilidades correctas.

$$J(\mathbf{w}) = -\frac{1}{N} \sum_{i=1}^N [y_i \log(p_{\text{pass}}) + (1 - y_i) \log(1 - p_{\text{pass}})]$$

Tabla 1: Matriz de Confusión

	Predicción: Pasa	Predicción: No pasa
Real: Pasa	Verdadero Positivo (TP)	Falso Negativo (FN)
Real: No pasa	Falso Positivo (FP)	Verdadero Negativo (TN)

Donde, si el estudiante pasa ($y_i = 1$), la segunda parte de la suma se anula y solo se considera $\log(p_{\text{pass}})$. Si no pasa ($y_i = 0$), la primera parte se anula y solo se considera $\log(1 - p_{\text{pass}})$.

Para garantizar que el modelo sea preciso y sus resultados sean fiables, los datos deben cumplir con ciertos supuestos estadísticos:

- **Independencia de las Observaciones:** Cada fila de datos debe ser independiente a las demás y no deben influir en el resultado de otra.
- **Ausencia de Multicolinealidad:** Establece que las variables predictoras no deben estar altamente correlacionadas entre sí, de otra manera, el modelo no podría determinar cual de ellas es la que realmente influye en la predicción.
- **Linealidad en el Logit:** Es el supuesto más importante. No asume que la relación entre las variables sea lineal. En cambio, asume que la relación lineal es entre las variables independientes x y la función logit.
- **Tamaño de Muestra Grande:** La regresión logística funciona mejor y produce resultados más estables y fiables cuando se entrena con un conjunto de datos suficientemente grande.

Dado que la regresión logística es un modelo de clasificación, el rendimiento se mide evaluando qué tan bien el modelo clasifica las instancias. La herramienta principal para esto es la Matriz de Confusión:

A partir de los cuatro valores de esta matriz, se calculan las métricas de evaluación principales:

- **Precisión:** Mide el porcentaje de las predicciones positivas fueron correctas.

$$\text{Precisión} = \frac{TP}{TP + FP}$$

- **Recall:** Mide la cantidad de casos positivos fueron detectados correctamente.

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **F1-Score:** Es la media armónica entre la Precisión y Recall. Es especialmente útil cuando existe un desbalance en las clases.

$$F1\text{-Score} = 2 \times \frac{\text{Precisión} \times \text{Recall}}{\text{Precisión} + \text{Recall}}$$

Por último, cabe resaltar las principales ventajas y desventajas para poder determinar cuándo utilizar la regresión logística:

2.2.1. Ventajas

- **Interpretabilidad:** Es un modelo muy fácil de interpretar. Los coeficientes indican cómo influye cada variable en la probabilidad del resultado
- **Eficiencia:** Es un modelo matemáticamente menos complejo que otros. Requiere menos capacidad computacional, por lo que puede entrenar rápidamente con grandes volúmenes de datos.
- **Flexibilidad:** Además de clasificación, también se puede usar para preprocesar datos.
- **Base Fundamental:** Es un pilar para entender modelos más avanzados.

2.2.2. Desventajas

- **Solo crea Límites de Decisión Lineales:** El modelo solo puede separar las 2 clases con una linea recta. Si no son linealmente separables, el modelo fallará.
- **Propensión al Underfitting:** Al ser un modelo simple, no siempre puede capturar relaciones coplejas en los datos.
- **Sensibilidad a Outliers:** El modelo sigue siendo sensible a puntos de datos atípicos que pueden distorsionar la decisión.

2.2.3. Aplicaciones Típicas

La regresión logística se usa en cualquier problema donde se necesite predecir la probabilidad de un resultado binario:

- **Sector Financiero:** Es un estándar de la industria para la detección de fraudes y la evaluación de riesgo crediticio.
- **Marketing y Ventas:** Ayuda a predecir si un usuario hará click en un anuncio o si un visitante de un sitio web completará una compra.
- **Industria:** Estimar la probabilidad de fallo de una pieza en maquinaria para planificar mantenimientos.

2.3. Árboles de decisión

Los árboles de decisión son algoritmos de aprendizaje supervisado que se utilizan tanto para clasificación como para regresión. Su funcionamiento se basa en dividir el conjunto de datos en subconjuntos más pequeños a través de una estructura jerárquica de decisiones, donde cada nodo interno representa una pregunta o prueba sobre una característica, cada rama una posible respuesta y cada hoja una clase o valor de salida final.

El objetivo principal del árbol es partitionar el espacio de características de forma que los subconjuntos resultantes sean lo más puros posible, es decir, que contengan ejemplos mayoritariamente pertenecientes a una misma clase.

La función de pérdida global de un árbol busca minimizar la suma ponderada de la impureza en los nodos terminales:

$$L = \sum_{m=1}^M N_m \cdot \text{Impureza}(R_m)$$

donde N_m es el número de muestras en el nodo m .

A diferencia de otros algoritmos de aprendizaje supervisado como la regresión lineal o logística, los árboles de decisión presentan una gran flexibilidad porque no requieren supuestos estadísticos estrictos sobre la distribución de los datos o las relaciones entre las variables. Sin embargo, existen ciertas condiciones y recomendaciones que deben considerarse para garantizar su correcto desempeño:

- **No linealidad y no parametricidad:** Los árboles no asumen que la relación entre las variables predictoras (X) y la variable respuesta (Y) sea lineal, aditiva o monótona.
- **Requisitos de independencia parcial:** Aunque los árboles no requieren independencia estricta entre variables, un alto grado de colinealidad puede causar divisiones redundantes o sesgadas, afectando la interpretabilidad del modelo.
- **Manejo de datos faltantes y categóricos:** Los árboles pueden trabajar con variables categóricas y valores faltantes, ya sea mediante la creación de ramas específicas o utilizando imputaciones internas.
- **Escalado y normalización:** No es necesario escalar ni normalizar las variables, ya que las divisiones se basan en umbrales relativos dentro de cada característica.
- **Tamaño de muestra:** Los árboles funcionan bien con conjuntos de datos pequeños o medianos, pero en bases de datos extremadamente grandes pueden volverse ineficientes en el tiempo de entrenamiento.
- **Estabilidad y sensibilidad a los datos:** Los árboles son sensibles a pequeñas variaciones en los datos de entrada; una ligera modificación en el conjunto de entrenamiento puede producir un árbol estructuralmente distinto.
- **Equilibrio de clases:** En problemas de clasificación, si las clases están desbalanceadas, el algoritmo puede sesgarse hacia la clase mayoritaria.
- **Representación jerárquica de decisiones:** Los árboles funcionan mejor cuando las reglas de decisión subyacentes en los datos son jerárquicas y pueden expresarse mediante comparaciones sucesivas.

Las métricas de evaluación permiten medir el desempeño de un árbol de decisión según el tipo de tarea: **clasificación** o **regresión**. A continuación se resumen las más utilizadas en cada caso.

Clasificación

El rendimiento en clasificación se analiza con base en la **matriz de confusión**, formada por los valores verdaderos y predichos:

$$\begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix}$$

A partir de ella se derivan las métricas principales:

- **Exactitud (Accuracy):** proporción de aciertos totales.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precisión (Precision):** mide qué tan confiables son las predicciones positivas.

$$Precision = \frac{TP}{TP + FP}$$

- **Exhaustividad o Recall:** indica la capacidad de detectar los casos positivos.

$$Recall = \frac{TP}{TP + FN}$$

- **Medida F1:** balance entre precisión y recall.

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

- **AUC-ROC:** mide la capacidad del modelo para distinguir entre clases; un valor cercano a 1 indica excelente desempeño.

Regresión

En regresión, se comparan los valores reales (y_i) con los predichos (\hat{y}_i):

- **Error Cuadrático Medio (MSE):**

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Penaliza más los errores grandes.

- **Error Absoluto Medio (MAE):**

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

Más robusto ante valores atípicos.

- **Coeficiente de Determinación (R^2):**

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

Mide qué proporción de la variabilidad de los datos es explicada por el modelo.

Una evaluación completa debe equilibrar la precisión, la simplicidad y la interpretabilidad del modelo.

Los árboles de decisión presentan diversas ventajas que los hacen especialmente atractivos en tareas de análisis predictivo y toma de decisiones:

Ventajas

- **Interpretabilidad:** su estructura jerárquica permite visualizar y comprender fácilmente el proceso de decisión.
- **No linealidad:** pueden modelar relaciones complejas y no lineales entre las variables sin necesidad de transformaciones matemáticas.
- **Versatilidad:** son aplicables tanto a problemas de clasificación como de regresión.
- **Escasa necesidad de preprocesamiento:** no requieren normalización ni estandarización de los datos.
- **Manejo de datos faltantes y categóricos:** pueden trabajar con variables mixtas sin necesidad de codificación compleja.

Pese a sus ventajas, los árboles de decisión también presentan limitaciones importantes que deben considerarse:

Desventajas

- **Sobreajuste:** tiende a ajustarse demasiado a los datos de entrenamiento si no se aplican técnicas de poda o regularización.
- **Inestabilidad:** pequeñas variaciones en los datos pueden generar estructuras de árbol muy diferentes.
- **Baja capacidad de generalización:** un árbol demasiado complejo pierde precisión con nuevos datos.
- **Sesgo hacia variables con muchos niveles:** las variables categóricas con muchos valores posibles pueden dominar las divisiones del árbol.

Aplicaciones típicas

Los árboles de decisión se utilizan ampliamente en distintos ámbitos de la ciencia de datos y la inteligencia artificial, gracias a su interpretabilidad y flexibilidad:

- **Finanzas:** evaluación de riesgo crediticio, predicción de morosidad o detección de fraude.
- **Medicina:** diagnóstico asistido, clasificación de enfermedades y análisis de factores de riesgo.
- **Marketing y ventas:** segmentación de clientes y análisis de comportamiento del consumidor.
- **Industria y manufactura:** control de calidad, detección de fallos y optimización de procesos.
- **Educación y recursos humanos:** predicción del rendimiento académico o desempeño laboral.

2.4. Redes neuronales

Las **redes neuronales artificiales (RNA)** son modelos inspirados en el funcionamiento del cerebro humano, diseñados para reconocer patrones, clasificar datos y aprender relaciones no lineales. Dentro de este campo, el **Perceptrón Multicapa (MLP, Multi-Layer Perceptron)** es una de las arquitecturas más comunes del aprendizaje supervisado.

Un MLP es una red, donde la información fluye en una sola dirección: desde la capa de entrada hacia la de salida, sin retroalimentación directa. Está compuesto por tres tipos de capas:

- **Capa de entrada:** recibe las características del conjunto de datos.
- **Capas ocultas:** procesan la información mediante combinaciones lineales y funciones de activación no lineales.
- **Capa de salida:** produce la predicción o clasificación final.

Cada neurona realiza una operación matemática del tipo:

$$z^{(l)} = W^{(l)}x^{(l-1)} + b^{(l)}, \quad a^{(l)} = \phi(z^{(l)})$$

donde $W^{(l)}$ representa los pesos, $b^{(l)}$ los sesgos, ϕ la función de activación y $a^{(l)}$ la salida activada.

El aprendizaje se lleva a cabo mediante el **algoritmo de retropropagación del error**, el cual calcula los gradientes de la función de pérdida con respecto a los parámetros del modelo. Estos gradientes se utilizan para actualizar los pesos usando métodos de optimización como el **descenso del gradiente estocástico (SGD)**, buscando minimizar el error global.

El entrenamiento del MLP se basa en minimizar una función de pérdida que mide la discrepancia entre las salidas reales y las predichas. Dependiendo del tipo de tarea, se aplican diferentes funciones:

- **Regresión:**

- Error Cuadrático Medio (MSE):

$$L = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

- Error Absoluto Medio (MAE):

$$L = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

- **Clasificación binaria:** Entropía cruzada binaria:

$$L = -\frac{1}{n} \sum_{i=1}^n [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

- **Clasificación multiclasa:** se utiliza la Entropía Cruzada Categórica:

$$L = -\frac{1}{n} \sum_{i=1}^n \sum_{k=1}^K y_{i,k} \log(\hat{y}_{i,k})$$

Aunque las redes neuronales no requieren cumplir supuestos estadísticos estrictos existen ciertas condiciones prácticas que influyen significativamente en su rendimiento, estabilidad y capacidad de generalización. Estas condiciones deben considerarse antes y durante el entrenamiento del modelo.

- **Cantidad y calidad de los datos:** Las redes neuronales necesitan un volumen considerable de datos para capturar patrones representativos y evitar el sobreajuste.
- **Normalización o estandarización de las entradas:** Los pesos se actualizan mediante gradientes, por lo que las escalas de las variables afectan directamente la convergencia.
- **No linealidad de las relaciones:** Las redes neuronales se diseñan para modelar relaciones no lineales complejas mediante funciones de activación como ReLU, Sigmoid o Tanh.
- **Adeuada arquitectura de red:** El número de capas y neuronas debe ajustarse a la complejidad del problema.
- **Inicialización apropiada de los pesos:** Una mala inicialización puede hacer que los gradientes desaparezcan o exploten, impidiendo el aprendizaje.
- **Separación de datos:** Es indispensable dividir el conjunto de datos en entrenamiento, validación y prueba, garantizando una evaluación objetiva del modelo.

Estas condiciones no son requisitos formales, pero su cumplimiento determina en gran medida la calidad del aprendizaje.

2.4.1. Principales métricas de evaluación

La evaluación del desempeño de una red neuronal permite verificar si el modelo ha aprendido correctamente y si generaliza adecuadamente a datos no vistos. Las métricas dependen del tipo de problema (clasificación o regresión) y deben seleccionarse de acuerdo con los objetivos del análisis.

Clasificación

Estas métricas se utilizan para medir el rendimiento de la red cuando la salida esperada corresponde a categorías o clases discretas.

- **Exactitud:** mide la proporción de aciertos totales sobre el conjunto de datos, comparando las predicciones correctas con el total de observaciones. Se define como:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precisión:** indica qué proporción de las predicciones positivas son realmente correctas.

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Sensibilidad:** mide la capacidad del modelo para identificar correctamente los casos positivos (verdaderos).

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **Puntaje F1:** combina precisión y sensibilidad mediante su media armónica, proporcionando un equilibrio entre ambas métricas.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Regresión

En los problemas de regresión, la salida del modelo es continua, y las métricas cuantifican el grado de error entre los valores predichos y los reales.

- **Error Cuadrático Medio (MSE):** evalúa la media de los errores al cuadrado, penalizando más los errores grandes. Es una métrica muy común en el entrenamiento de redes neuronales.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

donde y_i es el valor real, \hat{y}_i el valor predicho, y n el número de observaciones.

- **Error Absoluto Medio (MAE):** mide el promedio de los errores absolutos, siendo más robusto ante valores atípicos que el MSE.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

- **Coeficiente de determinación (R^2):** indica qué proporción de la variabilidad de los datos es explicada por el modelo. Valores cercanos a 1 reflejan un buen ajuste, mientras que valores negativos indican un mal desempeño.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

donde \bar{y} representa la media de los valores observados.

En la práctica, durante el entrenamiento de redes neuronales, se monitorean simultáneamente la pérdida y alguna métrica de desempeño como la precisión o el F1-score, para asegurar que el modelo no solo minimice el error sino que también generalice correctamente en datos nuevos.

Después de analizar las métricas que permiten evaluar el rendimiento de una red neuronal, resulta importante comprender sus implicaciones prácticas. Cada modelo presenta características que influyen en su desempeño, eficiencia y aplicabilidad según el tipo de problema. Por ello, a continuación se presentan las principales ventajas, desventajas y aplicaciones típicas de las redes neuronales, con la finalidad de ofrecer una visión más completa sobre sus alcances y limitaciones en la resolución de tareas reales.

2.4.2. Ventajas

- **Modelado de relaciones no lineales complejas:** gracias a sus múltiples capas y funciones de activación, las redes neuronales pueden aproximar funciones altamente no lineales, superando el desempeño de modelos lineales.
- **Capacidad de aprendizaje jerárquico:** las capas internas aprenden representaciones progresivamente abstractas de los datos, eliminando la necesidad de ingeniería manual de características.
- **Versatilidad y adaptabilidad:** se aplican en tareas de clasificación, regresión, detección, segmentación, predicción y generación de datos.

2.4.3. Desventajas

- **Alto costo computacional:** requieren procesadores especializados (GPU o TPU) y grandes tiempos de entrenamiento.
- **Riesgo de sobreajuste:** si el modelo es muy complejo o los datos insuficientes, la red puede memorizar el conjunto de entrenamiento.

- **Difícil interpretación:** la naturaleza de “caja negra” dificulta entender las decisiones internas del modelo.
- **Necesidad de ajuste de hiperparámetros:** la elección de la tasa de aprendizaje, número de capas y neuronas, y funciones de activación requiere experimentación cuidadosa.

Aplicaciones típicas

Las redes neuronales se aplican ampliamente en áreas donde existen grandes volúmenes de datos y relaciones no lineales:

- **Visión por computadora:** clasificación y detección de objetos, reconocimiento facial, segmentación de imágenes médicas.
- **Procesamiento del lenguaje natural:** análisis de sentimientos, traducción automática, chatbots, y generación de texto.
- **Predicción y análisis temporal:** proyecciones financieras, pronósticos meteorológicos y mantenimiento predictivo.
- **Sistemas de recomendación y detección de anomalías:** recomendación de productos o películas, y detección de fraudes financieros.

2.5. Comparativo de librerías y algoritmos

Se realizó una tabla comparativa para contrastar los modelos, destacando sus librerías de implementación, parámetros clave, ventajas, limitaciones y casos de uso típicos. El objetivo es ofrecer una visión panorámica que facilite la selección del modelo más adecuado para un problema determinado.

Algoritmo	Librería	Clase o función	Principales Parámetros	Ventajas	Limitaciones	Caso de uso
Regresión lineal	scikit-learn	LinearRegression()	fit_intercept, normalize	Simple e interpretable	Supone linealidad	Predicción de precios
Regresión logística	scikit-learn	LogisticRegression()	solver, max_iter	C, Buena para clasificación binaria	No capta relaciones no lineales	Clasificación de correos
Árboles de decisión	scikit-learn	DecisionTreeClassifier()	criterion, max_depth	Interpretable, flexible	Propenso al sobre-ajuste	Segmentación de clientes
Red neuronal (MLP)	Keras / TensorFlow	Sequential(), Dense()	activation, epochs	Modela relaciones complejas	Requiere más datos	Reconocimiento de voz/ímagenes

Tabla 2: Comparativo de librerías y algoritmos supervisados en Python.

2.6. Buenas prácticas y consideraciones

El desarrollo de un modelo supervisado exige más que solo elegir un algoritmo, este implica considerar la naturaleza del problema, la calidad de los datos y la forma en que se evaluará el desempeño. Antes de seleccionar el método de aprendizaje, es recomendable aplicar técnicas de

validación cruzada y regularización que aseguren que el modelo no solo se ajuste a los datos de entrenamiento, sino que también genere predicciones confiables en nuevos conjuntos de datos.

La elección del algoritmo depende directamente del tipo de variable objetivo. Cuando se trata de valores continuos, se emplean modelos de regresión, como la regresión lineal o las redes neuronales multicapa configuradas para regresión. En cambio, si el propósito es clasificar observaciones en diferentes categorías, se utilizan modelos de clasificación, como la regresión logística, los árboles de decisión o las redes neuronales orientadas a clasificación.

Otro aspecto esencial es la complejidad y la interpretabilidad del modelo. Los modelos simples facilitan comprender la relación entre variables y resultados, lo que resulta útil para la toma de decisiones explicables. De modo contrario, los modelos más complejos, como las redes neuronales profundas, pueden ofrecer mejor rendimiento, aunque a costa de una menor transparencia.

Finalmente, una evaluación rigurosa del desempeño es indispensable. En problemas de regresión se analizan métricas como el error cuadrático medio (MSE) o el coeficiente de determinación (R^2); mientras que en clasificación destacan la precisión y la recuperación. Estas prácticas permiten construir modelos equilibrados y precisos.

2.7. Conclusiones

Luego de analizar los algoritmos fundamentales del aprendizaje supervisado, se demostró que no existe un algoritmo universal, sino que cada modelo es bueno en su ámbito. Las principales diferencias están en la complejidad y la interpretabilidad.

La regresión Lineal y la logística son muy simples, eficientes y fáciles de interpretar. Sin embargo tienen un alcance limitado a problemas que asumen una relación lineal. Por otro lado, los árboles de decisión ofrecen un punto medio: Son altamente interpretables gracias a su estructura y no requieren supuestos estadísticos. Sin embargo, su principal limitación es la tendencia al overfitting y su inestabilidad ante pequeñas variaciones en los datos.

En el otro extremo, las Redes Neuronales ofrecen el mayor poder de modelado y alcance, siendo capaces de aprender relaciones no lineales extremadamente complejas y jerárquicas. Esta capacidad las hace indispensables en aplicaciones de alta complejidad, como el reconocimiento de imágenes o voz. No obstante, este rendimiento se obtiene a costa de una alta demanda computacional, la necesidad de grandes volúmenes de datos y la pérdida casi total de interpretabilidad, convirtiéndolas en modelos de “caja negra”.

En conclusión, la selección del algoritmo adecuado es un ejercicio de balance. Depende directamente de la naturaleza del problema, de los supuestos que los datos pueden cumplir y del objetivo final.

Fotografía de los participantes



Orlando Alvarado Vargas



Diego Alonso Carrillo Castillo

Referencias

- [1] Abd Rahman, M. S., Jamaludin, N. A. A., Zainol, Z., and Sembok, T. M. T. (2023). The application of decision tree classification algorithm on decision-making for upstream business. Disponible en:
url<https://thesai.org/Downloads/Volume14No8/Paper73-TheApplicationofDecisionTreeClassificationA06/11/2025>.
- [2] Amazon Web Services (s.f.). ¿qué es la regresión lineal? Disponible en: <https://aws.amazon.com/es/what-is/linear-regression/>. Consulta: 06/11/2025.
- [3] Arsov, N., Pavlovski, M., and Kocarev, L. (2019). *Stability of decision trees and logistic regression*. arXiv, 1 edition.
- [4] Blockeel, H., Devos, L., Frénay, B., Nanfack, G., and Nijssen, S. (2023). Decision trees: from efficient prediction to responsible ai. Disponible en:
url<https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2023.1124553/full>. Consulta: 06/11/2025.
- [5] Breiman, L., Friedman, J., Olshen, R. A., and Stone, C. J. (1984). Classification and regression trees. Disponible en:
urlhttps://books.google.com/books/about/Classification_and_Regression_Trees.html?id=JwQx-WOmSyQC. Consulta : 06/11/2025.
- [6] Contributors, G. (2024). Comprehensive guide to classification models in scikit-learn. <https://www.geeksforgeeks.org/machine-learning/comprehensive-guide-to-classification-models-in-scikit-learn/>.
- [7] Cosio, N. A. L. (2021, 20 de diciembre). Métricas en regresión. Disponible en: <https://medium.com/@nicolasarrioja/m%C3%A1tricas-en-regresi%C3%B3n-5e5d4259430b>. Consulta: 06/11/2025.
- [8] Google Cloud (s.f.). ¿qué es el aprendizaje supervisado? Disponible en: <https://cloud.google.com/discover/what-is-supervised-learning?hl=es-419>. Consulta: 04/11/2025.
- [9] Géron, A. (2023). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, 3 edition.
- [10] Hassan, E. E. and Zhang, D. (2021). The usage of logistic regression and artificial neural networks for evaluating and predicting insurers' solvency in egypt. Disponible en:
url<https://www.aimspress.com/article/id/617bd07eba35de0eb0deb183/>. Consulta: 06/11/2025.
- [11] Issitt, R. W. (2022). Classification performance of neural networks versus logistic regression: A meta-analysis. Disponible en:
url<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8942139/>. Consulta: 06/11/2025.

- [12] Krishnamoorthy, V. and Alok, U. (2022, 18 de abril). Linear regression: Assumptions and limitations. Disponible en: <https://blog.quantinsti.com/linear-regression-assumptions-limitations/>. Consulta: 06/11/2025.
- [13] Kumar, S. (2023, 14 de noviembre). Assumptions and limitations of logistic regression: Navigating the nuances. Disponible en: <https://medium.com/@skme20417/4-assumptions-and-limitations-of-logistic-regression-navigating-the-nuances-8ef249cc7a0>. Consulta: 06/11/2025.
- [14] Mawlud, D. and Mohsin, A. (2020). *A review on linear regression comprehensive in machine learning*. JASTT Path Journal, 1 edition.
- [15] Menéndez, E. M. (2024). Artificial neural networks for classification tasks: A systematic literature review. Disponible en: <urlhttps://www.redalyc.org/journal/5722/572279095001/html/>. Consulta: 06/11/2025.
- [16] Mienye, I. D. and Jere, N. (2017). A survey of decision trees: Concepts, algorithms, and applications. Disponible en: urlhttps://www.researchgate.net/publication/381564302_AsurveyofDecisionTreesConceptsAlgorithmsand 06/11/2025.
- [17] Mohammadi, F. and et al. (2021). Artificial neural network and logistic regression modelling of covid-19 infected patients' health outcomes. Disponible en: <urlhttps://www.sciencedirect.com/science/article/pii/S2319417021000123>. Consulta: 06/11/2025.
- [18] scikit-learn Developers (2025). *Scikit-learn User Guide: Supervised Learning*. scikit-learn, 1 edition.
- [19] Taboga, M. (s.f.). Loss function. Disponible en: <https://www.statlect.com/glossary/loss-function#:~:text=In%20statistics%20and%20machine%20learning,regression%2C%20to%20predict%20a%20variable>. Consulta: 06/11/2025.
- [20] Tzougas, G. and et al. (2023). Enhancing logistic regression using neural networks for insurance claims prediction. Disponible en: <urlhttps://www.mdpi.com/1999-4893/16/2/99>. Consulta: 06/11/2025.
- [21] Wen, Z. (2023). *Feature analysis and model comparison of logistic regression and decision tree for customer churn prediction*. ResearchGate, 1 edition.

Las imágenes utilizadas en este documento son propiedad de sus respectivos autores y se incluyen únicamente con fines académicos.