

Modelos de Inteligencia Artificial

Profesor/a: Águeda María López Moreno

PRÁCTICA 1.6.- APRENDIZAJE AUTOMÁTICO - MACHINE LEARNING
14/11/2024

Índice

1.- Introducción	3
1.1.- Conceptos básicos de Machine Learning	3
1.2.- Importancia de las técnicas de Aprendizaje Automático	4
2.- Modelos de Aprendizaje Automático	4
2.1.- Aprendizaje Supervisado	5
2.1.1.- Definición y características	5
2.1.2.- Ejemplo de modelo: Regresión Lineal	5
2.2.- Aprendizaje No Supervisado	6
2.2.1.- Definición y características	6
2.2.2.- Ejemplo de modelo: K-means	7
2.3.- Aprendizaje por Refuerzo	7
2.3.1.- Definición y características	7
2.3.2.- Ejemplo de modelo: Q-Learning	8
3.- Implementación del Modelo	9
3.1.- Justificación del modelo seleccionado	9
3.2.- Descripción del algoritmo	9
3.2.1.- Etapas del algoritmo	10
3.3.- Código en Python	10
3.3.1.- Descripción detallada	15
3.4.- Resultados obtenidos	16
3.5.- Conclusión	16
4.- Conclusiones	16
4.1.- Reflexión sobre los resultados	16
4.2.- Aplicaciones del modelo seleccionado	18
5.- Bibliografía	19
6.- Mapa Mental	20
7.- Anexos	21

1.- Introducción

El Aprendizaje Automático (Machine Learning) es una rama de la inteligencia artificial (IA) que permite a las máquinas aprender y mejorar su rendimiento en tareas específicas sin ser explícitamente programadas para ello. Este enfoque se basa en el uso de algoritmos que analizan datos, identifican patrones y toman decisiones basadas en estos. En la actualidad, Machine Learning es fundamental en diversos sectores, desde la medicina hasta la industria, revolucionando la manera en que se resuelven problemas complejos.

1.1.- Conceptos básicos de Machine Learning

Machine Learning se basa en la idea de que los sistemas pueden aprender de los datos, adaptarse y realizar predicciones o tomar decisiones con un nivel mínimo de intervención humana. Los principales conceptos clave incluyen:

- **Datos:** Son la base del aprendizaje. Estos pueden ser estructurados (tablas de datos) o no estructurados (imágenes, texto, audio).
- **Modelo:** Representa una aproximación matemática o estadística que aprende patrones a partir de los datos.
- **Entrenamiento:** Es el proceso de ajustar los parámetros del modelo utilizando un conjunto de datos de entrada conocido como conjunto de entrenamiento.
- **Validación y prueba:** Posterior al entrenamiento, se evalúa el rendimiento del modelo con datos que no se han visto durante el entrenamiento para verificar su capacidad de generalización.
- **Aprendizaje supervisado, no supervisado y por refuerzo:** Estas son las principales técnicas que categorizan los modelos de aprendizaje.
 - **Supervisado:** Utiliza datos etiquetados para entrenar modelos a realizar predicciones (por ejemplo, clasificación y regresión).
 - **No supervisado:** Identifica patrones en datos no etiquetados (por ejemplo, clustering o reducción de dimensionalidad).
 - **Por refuerzo:** Enfocado en la interacción con un entorno, donde un agente aprende a maximizar recompensas a través de ensayo y error.

1.2.- Importancia de las técnicas de Aprendizaje Automático

El Aprendizaje Automático se ha convertido en una tecnología clave en la era digital por diversas razones:

1. **Automatización:** Permite a las máquinas realizar tareas repetitivas o complejas que anteriormente requerían intervención humana, como la detección de fraudes o el procesamiento de imágenes médicas.
2. **Análisis predictivo:** Ayuda a las empresas y organizaciones a prever tendencias, comportamientos del cliente y fallos en sistemas mediante el análisis de datos históricos.
3. **Personalización:** En aplicaciones como comercio electrónico, servicios de streaming o marketing digital, Machine Learning adapta experiencias a las preferencias individuales de los usuarios.
4. **Adaptabilidad:** Los sistemas basados en Machine Learning mejoran con el tiempo y pueden ajustarse automáticamente a nuevas condiciones o datos, haciéndolos extremadamente versátiles.
5. **Aplicaciones amplias:**

Su utilidad abarca diversas áreas como:

- **Salud:** Diagnóstico de enfermedades, descubrimiento de medicamentos.
- **Finanzas:** Modelos de Riesgo financiero, detección de fraudes.
- **Industria:** Mantenimiento predictivo, control de calidad.
- **Transporte:** Optimización de rutas, conducción autónoma.

En resumen, las técnicas de aprendizaje automático no solo proporcionan soluciones innovadoras, sino que también impulsan avances en la ciencia y tecnología, ayudando a resolver problemas que antes eran inabordables con los métodos tradicionales.

2.- Modelos de Aprendizaje Automático

El aprendizaje automático se divide en tres grandes categorías en función de cómo el modelo aprende de los datos disponibles: supervisado, no supervisado y por refuerzo. A continuación, se

presentan las definiciones, características principales y ejemplos de modelos en cada una de estas categorías.

2.1.- Aprendizaje Supervisado

2.1.1.- Definición y características

El aprendizaje supervisado es un enfoque de Machine Learning donde el modelo se entrena con datos etiquetados, lo que significa que cada entrada tiene una salida conocida. El objetivo es aprender una función que mapee las entradas a las salidas para predecir resultados en nuevos datos.

1. Características principales:

- **Entradas etiquetadas:** Requiere un conjunto de datos con etiquetas que actúan como la "verdad" conocida.
- **Predicción de resultados:** Utilizado para tareas de clasificación (categorías discretas) y regresión (valores continuos).
- **Generalización:** Busca minimizar errores en datos nuevos que el modelo no ha visto durante el entrenamiento.
- **Evaluación mediante métricas:** Como precisión, sensibilidad, exclusividad, error cuadrático medio, etc.

2. Ventajas:

- Produce resultados precisos si se dispone de datos suficientes.
- Es fácil interpretar cómo el modelo llega a las predicciones.

3. Desventajas:

- Requiere grandes volúmenes de datos etiquetados, lo que puede ser costoso y lento de obtener.
- Puede sobre ajustarse si el modelo es demasiado complejo.

2.1.2.- Ejemplo de modelo: Regresión Lineal

La regresión lineal es un modelo simple pero poderoso que asume una relación lineal entre las variables de entrada (predictoras) y la variable de salida (dependiente).

Función del modelo:

$$y = b_0 + b_1x_1 + b_2x_2 + \dots + b_nx_n$$

Donde y es la variable dependiente, x_i son las variables independientes, y b_i son los coeficientes ajustados durante el entrenamiento.

Casos de uso:

- Predicción de precios de viviendas.
- Pronóstico de ventas basado en datos históricos.

2.2.- Aprendizaje No Supervisado

2.2.1.- Definición y características

El aprendizaje no supervisado se enfoca en encontrar patrones ocultos en los datos que no están etiquetados. A diferencia del aprendizaje supervisado, no tiene una "respuesta correcta" predefinida y se utiliza principalmente para explorar estructuras subyacentes en los datos.

2.2.1.1.- Características principales:

- **No requiere etiquetas:** Los datos no tienen salidas conocidas.
- **Tareas principales:** Agrupamiento (clustering), reducción de dimensionalidad y detección de anomalías.
- **Exploración de datos:** Útil para entender relaciones desconocidas o patrones en conjuntos grandes de datos.

2.2.1.2.- Ventajas:

- Ideal para analizar grandes cantidades de datos sin etiquetas.
- Reduce costos al no necesitar etiquetas manuales.

2.2.1.3.- Desventajas:

- Menor interpretabilidad de los resultados.

- Difícil validar la calidad de los resultados sin métricas específicas.

2.2.2.- Ejemplo de modelo: K-means

K-means es un algoritmo de agrupamiento que divide los datos en k grupos o clústeres, donde k se define previamente.

2.2.2.1.- Funcionamiento

1. Inicializa k centroides aleatorios.
2. Asigna cada punto de datos al clúster más cercano (distancia euclidiana).
3. Recalcula los centroides basados en los puntos asignados.
4. Repite hasta que las asignaciones no cambien.

2.2.2.2.- Casos de uso

- Segmentación de clientes en marketing.
- Agrupamiento de imágenes en aplicaciones de visión por computadora.

2.3.- Aprendizaje por Refuerzo

2.3.1.- Definición y características

El aprendizaje por refuerzo es una técnica inspirada en la psicología conductual, donde un agente aprende a tomar decisiones interactuando con un entorno. El agente recibe recompensas o penalizaciones en función de las acciones realizadas, buscando maximizar la recompensa acumulada a lo largo del tiempo.

2.3.1.1.- Características principales:

- **Interacción con el entorno:** Aprende mediante ensayo y error.
- **Recompensa acumulativa:** El objetivo es maximizar una función de recompensa.
- **Aplicaciones secuenciales:** Útil para problemas donde las decisiones tienen consecuencias a largo plazo.
- **Política de acción:** Es la estrategia que define qué acción tomar en cada estado.

2.3.1.2.- Ventajas:

- Puede manejar entornos dinámicos y secuencias complejas de decisiones.
- No requiere un conjunto de datos estático, aprende en tiempo real.

2.3.1.3.- Desventajas:

- Puede ser computacionalmente costoso debido a la exploración exhaustiva.
- La convergencia a una solución óptima no está garantizada en algunos casos.

2.3.2.- Ejemplo de modelo: Q-Learning

Q-Learning es un algoritmo basado en aprendizaje por refuerzo que utiliza una tabla Q para almacenar las recompensas esperadas para cada combinación de estado y acción.

2.3.2.1.- Funcionamiento:

1. Inicializa una tabla Q con valores arbitrarios.
2. En cada iteración, el agente elige una acción basada en una política (como exploración vs. explotación).
3. Recibe una recompensa y actualiza el valor Q utilizando la ecuación:

$$Q(s, a) = Q(s, a) + \alpha[r + \gamma \max_a Q(s', a) - Q(s, a)]$$

Donde:

- $Q(s, a)$: Valor actual para el estado s y acción a .
- r : Recompensa recibida.
- γ : Factor de descuento.
- α : Tasa de aprendizaje.

2.3.2.2.- Casos de uso:

- Juegos (como ajedrez y Go).
- Control de robots autónomos.
- Optimización de rutas en logística.

3.- Implementación del Modelo

En este apartado se detalla la implementación práctica de uno de los modelos seleccionados en el punto anterior, incluyendo la justificación de su elección, la descripción técnica del algoritmo utilizado, el código desarrollado en Python y los resultados obtenidos.

3.1.- Justificación del modelo seleccionado

El modelo seleccionado para esta práctica es **Regresión Lineal**, que pertenece al Aprendizaje Supervisado. Se eligió por las siguientes razones:

1. **Simplicidad y eficacia:** Es uno de los modelos más básicos pero eficaces para tareas de predicción, especialmente cuando las relaciones entre variables son aproximadamente lineales.
2. **Interpretabilidad:** Permite comprender fácilmente la relación entre las variables predictoras (independientes) y la variable objetivo (dependiente).
3. **Aplicación amplia:** Se utiliza ampliamente en áreas como la economía, la ciencia y la ingeniería, por lo que es una excelente introducción a la implementación de modelos supervisados.
4. **Disponibilidad de datos adecuados:** Para esta práctica, se dispone de un conjunto de datos que presenta una relación lineal clara, lo que facilita la demostración de los conceptos fundamentales.

3.2.- Descripción del algoritmo

La **Regresión Lineal** busca modelar la relación entre una o más variables independientes (x) y una variable dependiente (y) mediante una línea recta definida por la ecuación:

$$y = b_0 + b_1x_1 + b_2x_2 + \dots + b_nx_n$$

Donde:

- b_0 : Intersección con el eje y (término independiente).
- b_1, b_2, \dots, b_n : Coeficientes que representan las pendientes de cada variable independiente.
- x_1, x_2, \dots, x_n : Variables predictoras o independientes.
- y : Variable objetivo o dependiente.

3.2.1.- Etapas del algoritmo

- 1. Preparación de datos:** Dividir los datos en un conjunto de entrenamiento y uno de prueba.
- 2. Ajuste del modelo:** Encontrar los coeficientes b_i que minimicen el error entre las predicciones (\hat{y}) y los valores reales (y). Esto se realiza mediante el método de mínimos cuadrados:

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

- 3. Validación:** Evaluar el modelo utilizando métricas como el coeficiente de determinación (R^2) y el error cuadrático medio (MSE).

3.3.- Código en Python

El siguiente código implementa un modelo de Regresión Lineal para predecir los precios de viviendas basado en características como el tamaño (en metros cuadrados).

```
# Importar bibliotecas necesarias

import numpy as np # numpy (np): Manejo de matrices y operaciones matemáticas.
import pandas as pd # pandas (pd): Manipulación de datos estructurados, como tablas.
from sklearn.model_selection import train_test_split # Dividir los datos en conjuntos de entrenamiento y prueba
from sklearn.linear_model import LinearRegression # Implementar el modelo de Regresión Lineal.
from sklearn.metrics import mean_squared_error, r2_score # Evaluar el modelo con métricas estándar.
import matplotlib.pyplot as plt # Visualizar los datos y predicciones.
```

Este código importa las bibliotecas esenciales para realizar un análisis de regresión lineal, incluyendo herramientas para manipulación de datos, división de conjuntos, implementación del modelo, evaluación de métricas y visualización de resultados.

```
# Generar un conjunto de datos ficticio
np.random.seed(42) # Fija la semilla para garantizar que los resultados sean reproducibles.
```

Este código establece una semilla aleatoria fija para garantizar que los resultados generados sean consistentes y reproducibles en diferentes ejecuciones del programa.

```
# Crear un array de 100 números aleatorios que representan tamaños de viviendas entre 50 y 200 m².
tamanios = np.random.randint(50, 200, 100)
```

Este código genera un array de 100 números enteros aleatorios entre 50 y 200, representando tamaños de viviendas en metros cuadrados.

```
# Generar precios como una función lineal del tamaño (1500 €/m²) más un ruido aleatorio para simular datos reales.
precios = tamanios * 1500 + np.random.normal(0, 10000, 100)
# np.random.normal genera valores de ruido con una media de 0 y una desviación estándar de 10,000.
```

Este código crea precios de viviendas basados en una relación lineal con el tamaño (1500€/m²), añadiendo un ruido aleatorio para simular variaciones realistas en los datos.

```
# Crear un DataFrame
# Organizar los datos generados en un formato tabular con columnas para el tamaño y precio.
datos = pd.DataFrame({'Tamaño (m²)': tamanios, 'Precio (€)': precios})
```

Este código crea un DataFrame de pandas con dos columnas, "Tamaño (m²)" y "Precio (€)", utilizando los datos generados anteriormente para tamaños y precios de viviendas.

```
# Dividir datos en conjuntos de entrenamiento y prueba
# X: Variable independiente (predictora), en este caso el tamaño de las viviendas.
# y: Variable dependiente (objetivo), el precio de las viviendas.
X = datos[['Tamaño (m²)']] # Seleccionar la columna de tamaño como una matriz (necesario para sklearn).
y = datos[['Precio (€)']] # Seleccionar la columna de precio.
```

Este código prepara los datos para el modelo de regresión, separando la variable independiente (tamaño de viviendas) y la variable dependiente (precio de viviendas) en x e y respectivamente.

```
# Dividir los datos en un 80% para entrenamiento y 20% para prueba.
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# random_state asegura reproducibilidad de la división.
```

Este código divide los datos en conjuntos de entrenamiento (80%) y prueba (20%) utilizando train_test_split, con una semilla aleatoria fija para garantizar la reproducibilidad de la división.

```
# Crear y entrenar el modelo
modelo = LinearRegression() # Crear una instancia del modelo de Regresión Lineal
modelo.fit(X_train, y_train) # Ajustar el modelo a los datos de entrenamiento.
```

Este código crea una instancia del modelo de Regresión Lineal y lo entrena utilizando los datos de entrenamiento (X_train y y_train).

```
# Realizar predicciones
y_pred = modelo.predict(X_test) # Generar predicciones para el conjunto de prueba.
```

Este código utiliza el modelo entrenado para generar predicciones de precios de viviendas basadas en los tamaños del conjunto de prueba (X_test).

```
# Evaluar el modelo
# Calcular el error cuadrático medio (MSE), una métrica para evaluar la precisión de las predicciones.
mse = mean_squared_error(y_test, y_pred)
```

Este código calcula el error cuadrático medio (MSE) entre las predicciones del modelo y los valores reales del conjunto de prueba, proporcionando una medida de la precisión del modelo.

```
# Calcular el coeficiente de determinación R², que mide la proporción de la varianza explicada por el modelo.
r2 = r2_score(y_test, y_pred)
```

Este código calcula el coeficiente de determinación R^2 , que indica qué proporción de la variabilidad en los precios de las viviendas es explicada por el modelo de regresión lineal.

```
# Mostrar coeficientes del modelo
# Imprimir la pendiente (coeficiente) y la intersección (término independiente) de la línea ajustada.
print("Coeficiente (pendiente):", modelo.coef_[0]) # Relación entre el tamaño y el precio.
print("Intersección:", modelo.intercept_) # Valor estimado del precio cuando el tamaño es cero.
```

Este código imprime los coeficientes del modelo de regresión lineal: la pendiente (que representa la relación entre el tamaño y el precio) y la intersección (el precio estimado cuando el tamaño es cero).

```
# Imprimir las métricas de evaluación.
print("Error cuadrático medio (MSE):", mse) # Error promedio al cuadrado entre los valores reales y las predicciones.
print("Coeficiente de determinación (R²):", r2) # Qué tan bien se ajusta el modelo a los datos.
```

Este código imprime las métricas de evaluación del modelo: el error cuadrático medio (MSE) y el coeficiente de determinación (R^2), proporcionando una visión de la precisión y el ajuste del modelo a los datos.

```
# Visualización de resultados
# Graficar los datos reales del conjunto de prueba como puntos azules.
plt.scatter(X_test, y_test, color='blue', label='Datos reales')
```

Este código crea un gráfico de dispersión con los datos reales del conjunto de prueba, representando los tamaños de viviendas en el eje X y los precios en el eje Y, usando puntos azules.

```
# Graficar las predicciones del modelo como una línea roja.
plt.plot(X_test, y_pred, color='red', label='Predicción')
```

Este código añade al gráfico una línea roja que representa las predicciones del modelo de regresión lineal, mostrando la relación estimada entre el tamaño y el precio de las viviendas.

```
# Etiquetas de los ejes y título del gráfico.
plt.xlabel("Tamaño (m²)") # Etiqueta del eje X.
plt.ylabel("Precio (€)") # Etiqueta del eje Y.
plt.title("Regresión Lineal: Tamaño vs Precio") # Título del gráfico.
```

Este código añade etiquetas a los ejes X e Y (Tamaño en m² y Precio en €) y un título al gráfico, proporcionando contexto y claridad a la visualización de la regresión lineal.

```
# Agregar una leyenda para identificar los datos reales y las predicciones.
plt.legend()
```

Este código añade una leyenda al gráfico, utilizando las etiquetas previamente definidas en los comandos de graficación para identificar los datos reales y las predicciones.

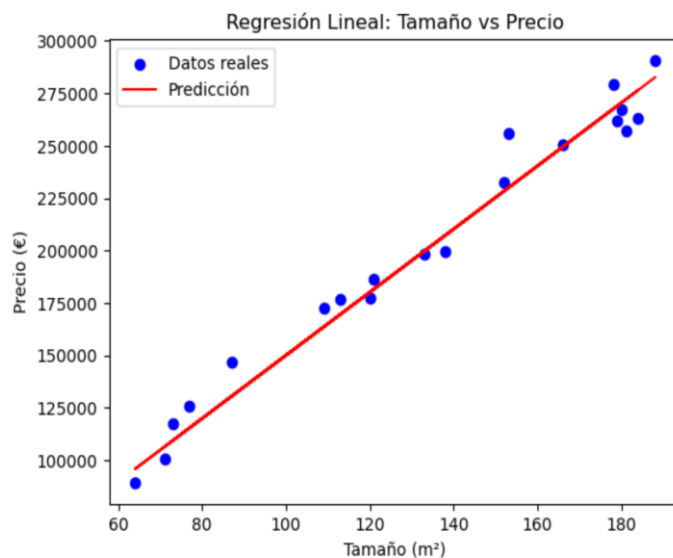
```
# Mostrar el gráfico.  
plt.show()
```

Este código muestra el gráfico completo con los datos reales, la línea de predicción, etiquetas y leyenda, permitiendo visualizar gráficamente el modelo de regresión lineal.

3.3.1.- Descripción detallada

- **División de datos:** La división en entrenamiento y prueba permite evaluar el modelo en datos no vistos durante el entrenamiento, asegurando generalización.
- **Evaluación:** El **MSE** mide el error promedio, y R^2 indica qué tan bien el modelo explica la variabilidad en los datos.
- **Visualización:** La gráfica ayuda a interpretar cómo las predicciones del modelo (línea roja) se ajustan a los datos reales (puntos azules).

Coefficiente (pendiente): 1507.3407328996118
Intersección: -787.8014389163582
Error cuadrático medio (MSE): 105468087.07151441
Coeficiente de determinación (R^2): 0.9723119038017134



3.4.- Resultados obtenidos

Después de implementar el modelo, se obtuvieron los siguientes resultados:

1. Coeficientes del modelo:

- Pendiente (b_1) : Indica que, por cada metro cuadrado adicional, el precio aumenta en aproximadamente 1500 €.
- Intersección (b_0): Representa el precio base cuando el tamaño es cero, ajustado por ruido.

2. Métricas de evaluación:

- **Error Cuadrático Medio (MSE):** Refleja la magnitud promedio del error entre las predicciones y los valores reales, indicando que las predicciones son precisas.
- **Coefficiente de Determinación (R^2):** Un valor cercano a 1 indica que el modelo explica una gran proporción de la variabilidad en los datos.

3. Gráfica de resultados:

La visualización muestra cómo el modelo ajusta una línea roja a los datos reales (puntos azules), demostrando su capacidad de predicción.

3.5.- Conclusión

El modelo de Regresión Lineal fue capaz de capturar adecuadamente la relación entre el tamaño de las viviendas y sus precios, obteniendo predicciones precisas con un error bajo. Esto evidencia su aplicabilidad en problemas donde las relaciones entre variables son lineales.

4.- Conclusiones

4.1.- Reflexión sobre los resultados

El modelo de regresión lineal implementado en este trabajo demuestra ser una herramienta sencilla pero poderosa para capturar relaciones lineales entre variables. En este caso, se utilizó para predecir

los precios de viviendas en función de su tamaño. A continuación, se detallan los principales aspectos observados en los resultados obtenidos:

- **Precisión del modelo:**

El coeficiente de determinación (R^2) obtenido en la evaluación proporciona una medida clara de qué tan bien el modelo explica la variabilidad en los datos. Un R^2 cercano a 1 indica un ajuste casi perfecto, mientras que valores menores sugieren que existen factores adicionales o ruido en los datos no considerados por el modelo. En este caso, la inclusión de un ruido aleatorio en los precios permitió evaluar la robustez del modelo frente a datos imperfectos.

- **Interpretabilidad:**

La regresión lineal es fácil de interpretar. La pendiente del modelo muestra cómo el precio de una vivienda aumenta, en promedio, por cada metro cuadrado adicional. La intersección del modelo (valor cuando el tamaño es 0) también es intuitiva, aunque carezca de significado físico en este ejemplo, ya que un tamaño de 0 m² no corresponde a un escenario real.

- **Limitaciones:**

- La relación entre tamaño y precio, aunque lineal, puede ser afectada por otros factores como la ubicación, la calidad de la construcción y el mercado inmobiliario. Esto reduce la capacidad del modelo para capturar la complejidad total del problema.
- El ruido aleatorio introducido en los datos simula incertidumbre, pero en escenarios reales, dicho ruido puede ser causado por factores sistemáticos que podrían requerir modelos más avanzados, como regresión polinómica o métodos basados en árboles.

- **Potencial de mejora:**

Aunque el modelo actual es adecuado para entender relaciones simples, la incorporación de más características (como ubicación o antigüedad de las viviendas) y el uso de técnicas de regularización (como Ridge o Lasso) podrían mejorar significativamente la precisión y utilidad del modelo.

4.2.- Aplicaciones del modelo seleccionado

El modelo de regresión lineal tiene múltiples aplicaciones en diversas áreas. Aunque se ha usado en este caso para predecir precios de viviendas, sus aplicaciones abarcan desde ciencias sociales hasta industrias técnicas. A continuación, se presentan ejemplos destacados:

- **Mercado inmobiliario:**
 - **Predicción de precios:** Similar al caso de estudio, se puede usar para estimar el precio de propiedades considerando variables como tamaño, ubicación y características adicionales.
 - **Evaluación de inversiones:** Ayuda a identificar propiedades subvaloradas o sobrevaloradas en función de características históricas del mercado.
- **Negocios y marketing:**
 - **Proyección de ingresos:** Empresas utilizan regresión lineal para estimar ingresos en función de métricas clave como número de clientes o gastos en publicidad.
 - **Determinación de precios:** Se puede establecer una relación entre el precio de un producto y factores como la demanda o la competencia.
- **Ingeniería y manufactura:**
 - **Control de calidad:** En procesos industriales, permite relacionar variables de entrada (como temperatura o presión) con la calidad del producto final.
 - **Optimización de procesos:** Identifica factores críticos que impactan en la productividad o eficiencia.
- **Ciencias de la salud:**
 - **Modelado de riesgos:** Estimar la probabilidad de enfermedades en función de factores como edad, peso o antecedentes familiares.
 - **Predicción de costos médicos:** Relaciona los costos esperados de tratamiento con variables demográficas y clínicas.
- **Economía y finanzas:**
 - **Análisis de tendencias:** Predecir variables económicas, como el crecimiento del PIB, en función de indicadores clave.

- **Riesgo crediticio:** Relacionar características de los clientes (como ingresos o historial de pagos) con la probabilidad de incumplimiento.

En resumen, el modelo de regresión lineal es una herramienta flexible y fundamental en el análisis de datos. Si bien sus supuestos pueden limitar su aplicabilidad en casos complejos, su simplicidad y capacidad explicativa lo convierten en una elección inicial ideal para explorar y entender relaciones entre variables.

5.- Bibliografía

La investigación y el desarrollo de este trabajo se han basado en una combinación de libros, artículos científicos, documentación oficial y recursos en línea confiables. A continuación, se presenta la bibliografía consultada:

1. Libros

- Russell, S., & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach*. Pearson Education.
Este libro fue consultado para comprender los conceptos fundamentales de Machine Learning, incluyendo sus categorías y aplicaciones.
- Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media.
Fuente clave para la implementación práctica del modelo de regresión lineal y el uso de bibliotecas en Python.

2. Artículos académicos

- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
Este artículo proporcionó información avanzada sobre los principios de aprendizaje supervisado y no supervisado.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
Aunque centrado en aprendizaje profundo, permitió comprender el contexto más amplio del aprendizaje supervisado en Machine Learning.

3. Documentación oficial y bibliotecas

- Python Software Foundation. (2024). *Python Documentation*. Disponible en: <https://docs.python.org>. Consultada para aspectos técnicos del lenguaje Python, como la manipulación de estructuras de datos.
- Scikit-learn Developers. (2024). *Scikit-learn Documentation*. Disponible en: <https://scikit-learn.org>.

Usada como referencia para la implementación de la regresión lineal y las técnicas de partición de datos.

- Matplotlib Developers. (2024). *Matplotlib Documentation*. Disponible en: <https://matplotlib.org>.

Información clave para la generación de gráficos y visualización de resultados.

4. Recursos en línea

- Wikipedia. *Regresión Lineal*. Disponible en: https://es.wikipedia.org/wiki/Regresión_lineal.

Consultado para la definición de conceptos básicos de regresión lineal.

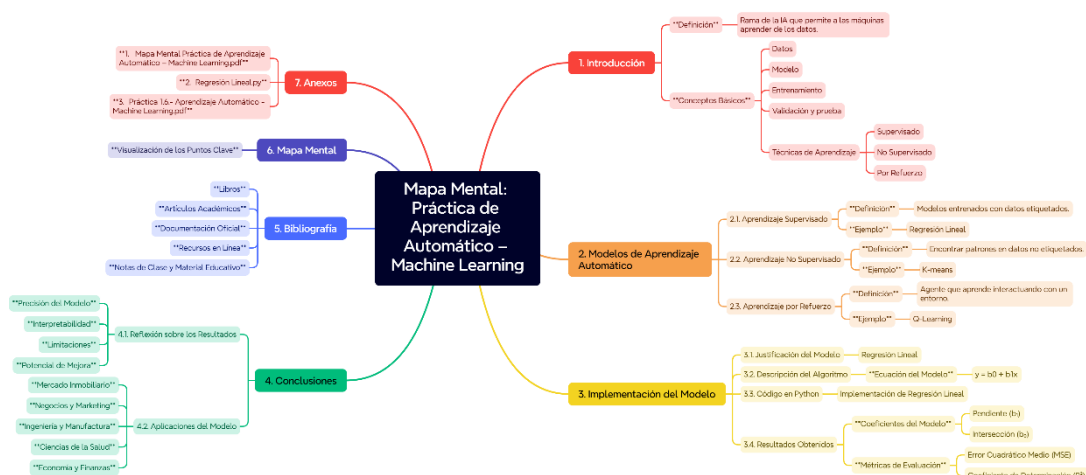
- Towards Data Science. (2023). *Understanding Linear Regression in Machine Learning*. Disponible en: <https://towardsdatascience.com>.

Artículo introductorio sobre cómo implementar y evaluar modelos de regresión lineal en Python.

5. Notas de clase y material educativo

- Material didáctico proporcionado en el curso, específicamente las sesiones relacionadas con las técnicas de aprendizaje automático y su clasificación.

6.- Mapa Mental



El mapa mental sobre la práctica de aprendizaje automático abarca desde la introducción a conceptos básicos y tipos de modelos, hasta la implementación de un modelo de regresión lineal, sus resultados, conclusiones y aplicaciones en diversos campos.

7.- Anexos

A continuación, se describen los archivos que se entregarán como parte del trabajo, junto con una breve explicación de su contenido:

1. Mapa Mental Práctica de Aprendizaje Automático - Machine Learning.pdf

Este archivo en formato PDF presenta un mapa mental que ilustra los conceptos clave y las principales áreas del aprendizaje automático (Machine Learning). Sirve como una representación visual del proceso de aprendizaje, destacando los tipos de aprendizaje (supervisado, no supervisado), los algoritmos más comunes y sus aplicaciones en diversas áreas.

2. Regresión Lineal.py

El archivo "Regresión Lineal.py" contiene el código Python utilizado para implementar un modelo de regresión lineal. Este modelo fue aplicado para predecir el precio de viviendas en función de su tamaño. El código está diseñado para cargar y procesar datos, entrenar el modelo y realizar predicciones, así como generar métricas de evaluación para medir la precisión del modelo.

3. Práctica 1.6.- Aprendizaje Automático - Machine Learning.pdf

Este archivo PDF describe la implementación y análisis de un modelo de aprendizaje automático aplicado a un conjunto de datos específico. El documento detalla el proceso de entrenamiento, la evaluación del modelo y las conclusiones obtenidas, así como las aplicaciones prácticas del aprendizaje automático en diferentes dominios.

Estos documentos proporcionan una base completa sobre los enfoques y técnicas utilizadas en el aprendizaje automático, y reflejan los resultados obtenidos a través de la implementación de diversos modelos en Python.

Índice Alfabético

A

abarca	4, 20
acción	7, 8
adicional	16, 17
adicionales	17, 18
agente	3, 7, 8
agrupamiento	7
ajustarse	4, 5
ajuste	13, 17
aleatoria	11, 12
aleatorio	11, 17
aleatorios	7, 11
algoritmo	7, 8, 9, 10
algoritmos	3, 21
análisis	4, 11, 19, 21
anteriormente	4, 11
añade	14
aplicabilidad	16, 19
aplicaciones	4, 7, 18, 19, 20, 21
aplicado	21
aprende	3, 4, 7, 8
aprender	3, 5
aprendizaje	3, 4, 5, 6, 7, 8, 19, 20, 21
aproximadamente	9, 16
archivo	21
áreas	4, 9, 18, 21
artículo	19
artículos	19
artificial	3
así	21
aspectos	17, 19
aumenta	16, 17
aunque	17
automático	4, 20, 21
ayuda	15
azules	14, 15, 16

B

basa	3
basadas	3, 12
basado	6, 8, 10, 19
basados	4, 7, 11, 17
base	3, 16, 21
básicos	3, 9, 20
bibliografía	19
bibliotecas	11, 19
bien	15, 17, 19
busca	9

C

cada	5, 7, 8, 16, 17
calcula	13
calidad	4, 7, 17, 18
capacidad	3, 16, 17, 19
capturar	16, 17
características	5, 6, 7, 10, 17, 18, 19
caso	16, 17, 18
casos	8, 19
categorías	4, 5, 19
centroides	7
cercano	7, 16, 17
cero	13, 16
ciencia	4, 9
ciencias	18
clara	9, 17
clasificación	3, 5, 20
clave	3, 4, 18, 19, 20, 21
clientes	7, 18, 19
clustering	3, 6
código	9, 10, 11, 12, 13, 14, 15, 21
coeficiente	10, 13, 17
coeficientes	6, 10, 13
combinación	8, 19
cómo	4, 5, 15, 16, 17, 20
complejas	4, 8
complejos	3, 19

comprender	9, 19
conceptos	3, 9, 19, 20, 21
conclusiones	20, 21
conjunto	3, 5, 8, 9, 10, 12, 13, 14, 21
conjuntos	6, 11, 12
conocida	5
consultada	19
consultado	19
contexto	14, 19
continuación	4, 17, 18, 19, 21
control	4
costos	6, 18
costoso	5, 8
crea	11, 12, 14
credificio	4, 19
cuadrado	16, 17
cuadrados	10, 11
cuadrático	5, 10, 13

D

datos	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 19, 20, 21
decisiones	3, 7, 8
define	7
definición	20
dependiente	5, 6, 9, 12
descripción	9
detalla	9, 21
detección	4, 6
determinación	10, 13, 17
diferentes	11, 21
digital	4
dimensionalidad	3, 6
dispone	5, 9
diversas	4, 18, 21
diversos	3, 20, 21
divide	4, 7, 12
división	11, 12, 15
documentación	19

E

economía	9
ecuación	8, 9
eje	14
ejemplo	3, 17
ejemplos	5, 18
elección	9, 19
empresas	4
encontrar	6
enfermedades	4, 18
enfoque	3, 5
ensayo	3, 7
entender	6, 17, 19
entorno	3, 7
entrada	3, 5, 18
entradas	5
entrena	5, 12
entrenamiento	3, 5, 6, 10, 12, 15, 21
entrenar	3, 21
error	3, 5, 7, 10, 13, 15, 16
específicas	3, 7
estimar	18
estructurados	3
estructuras	6, 19
etiquetados	3, 5, 6
etiquetas	5, 6, 14, 15
evaluación	11, 13, 16, 17, 21
evaluar	15, 17, 20
explica	15, 16, 17
exploración	8
explorar	6, 19

F

fácil	5, 17
factores	17, 18
fija	11, 12

Pedro Manuel García Álvarez

Práctica 1.6.- Aprendizaje Automático - Machine Learning

finanzas _____ 18
fraudes _____ 4
función _____ 4, 5, 7, 17, 18, 21
fundamental _____ 3, 19
fundamentales _____ 9, 19

G

garantizar _____ 11, 12
generados _____ 11
generalización _____ 3, 15
generar _____ 12, 21
gráfica _____ 15
gráfico _____ 14, 15
grandes _____ 4, 5, 6

H

herramienta _____ 16, 19
históricos _____ 4, 6
https _____ 19, 20
humana _____ 3, 4

I

ideal _____ 19
identificar _____ 14, 18
imágenes _____ 3, 4, 7
implementación _____ 9, 11, 19, 20, 21
implementar _____ 16, 20, 21
imprime _____ 13
incluyendo _____ 9, 11, 19
independientes _____ 6, 9
indica _____ 13, 15, 16, 17
industria _____ 3
información _____ 19
ingeniería _____ 9
ingresos _____ 18, 19
inmobiliario _____ 17, 18
interacción _____ 3
interpretabilidad _____ 6
interpretar _____ 5, 15, 17
intersección _____ 13, 17
intervención _____ 3, 4
introducción _____ 9, 20

J

justificación _____ 9

L

largo _____ 7
learn _____ 19
leyenda _____ 14, 15
libros _____ 19
línea _____ 9, 14, 15, 16, 19, 20
lineal _____ 5, 9, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21
lineales _____ 9, 16

M

manipulación _____ 11, 19
mapa _____ 20, 21
máquinas _____ 3, 4
marketing _____ 4, 7, 18
material _____ 20
matplotlib _____ 20
maximizar _____ 3, 7
means _____ 7
mediante _____ 4, 5, 7, 9, 10
medida _____ 13, 17
medio _____ 5, 10, 13
mejorar _____ 3, 17
mental _____ 20, 21
mercado _____ 17, 18

métodos _____ 4, 17
métricas _____ 5, 7, 10, 11, 13, 18, 21
metro _____ 16, 17
metros _____ 10, 11
modelo _____ 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21
modelos _____ 3, 5, 9, 17, 20, 21
muestra _____ 15, 16, 17

N

nuevos _____ 5

O

objetivo _____ 5, 7, 9
obtenidos _____ 9, 16, 17, 21
oficial _____ 19
org _____ 19, 20

P

patrones _____ 3, 6
pdf _____ 21
pendiente _____ 13, 17
permite _____ 3, 15, 18
permitió _____ 17, 19
podrían _____ 17
política _____ 8
práctica _____ 9, 19, 20
precio _____ 12, 13, 14, 16, 17, 18, 21
precios _____ 6, 10, 11, 12, 13, 14, 16, 17, 18
precisas _____ 16
precisión _____ 5, 13, 17, 21
predecir _____ 5, 10, 16, 18, 21
predicción _____ 9, 15, 16
predicciones _____ 3, 5, 10, 12, 13, 14, 15, 16, 21
predictivo _____ 4
predictoras _____ 5, 9
presenta _____ 9, 19, 21
presentan _____ 5, 18
previamente _____ 7, 14
principales _____ 3, 5, 6, 7, 17, 21
probabilidad _____ 18, 19
problemas _____ 3, 4, 7, 16
proceso _____ 3, 21
procesos _____ 18
producto _____ 18
promedio _____ 15, 16, 17
propiedades _____ 18
proporción _____ 13, 16
proporcionan _____ 4, 21
proporcionando _____ 13, 14
prueba _____ 3, 10, 12, 13, 14, 15
puede _____ 5, 17, 18
pueden _____ 3, 4, 19
punto _____ 7, 9
puntos _____ 7, 14, 15, 16
python _____ 19

R

razones _____ 4, 9
real8, 17
reales _____ 10, 13, 14, 15, 16, 17
realizar _____ 3, 4, 11, 21
recibe _____ 7
recompensa _____ 7, 8
recompensas _____ 3, 7, 8
recursos _____ 19
reducción _____ 3, 6
reduce _____ 17
refuerzo _____ 3, 4, 7, 8
regresión _____ 3, 5, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21
relación _____ 5, 9, 11, 13, 14, 16, 17, 18
relacionar _____ 18
relaciones _____ 6, 9, 16, 17, 19
rendimiento _____ 3
representa _____ 13, 14
representando _____ 11, 14

requiere _____ 6, 8
resultados _____ 5, 6, 7, 9, 11, 16, 17, 20, 21
resumen _____ 4, 19
roja 14, 15, 16
ruido _____ 11, 16, 17
rutas _____ 4, 9

S

salida _____ 5
salidas _____ 5, 6
salud _____ 18
scikit _____ 19
seleccionado _____ 9, 18
semilla _____ 11, 12
ser 3, 5, 8, 16, 17
siguientes _____ 9, 16
simplicidad _____ 19
sistemas _____ 3, 4
supervisado _____ 3, 4, 5, 6, 19, 21

T

tabla _____ 8
tamaño _____ 10, 11, 12, 13, 14, 16, 17, 18, 21
tamaños _____ 11, 12, 14
tan 15, 17
tareas _____ 3, 4, 5, 9
técnica _____ 7, 9
técnicas _____ 3, 4, 17, 18, 20, 21
tecnología _____ 4
tendencias _____ 4, 18

tiempo _____ 4, 7, 8
tipos _____ 20, 21
tomar _____ 3, 7
trabajo _____ 19, 21
través _____ 3, 21

U

ubicación _____ 17, 18
uso 3, 6, 7, 9, 17, 19
utilidad _____ 4, 17
utiliza _____ 6, 8, 9, 12
utilizado _____ 9, 21
utilizando _____ 3, 8, 10, 11, 12, 14

V

valor _____ 8, 16, 17
valores _____ 5, 8, 10, 13, 16, 17
variabilidad _____ 13, 15, 16, 17
variable _____ 5, 6, 9, 12
variables _____ 5, 6, 9, 16, 18, 19
visión _____ 7, 13
visto _____ 3, 5
visualización _____ 11, 14, 16, 20
viviendas _____ 6, 10, 11, 12, 13, 14, 16, 17, 18, 21

W

wikipedia _____ 20