

UNIVERSIDAD TECNOLÓGICA DE CHIHUAHUA

Tecnologías de la Información: Desarrollo y Gestión de Software



III. Análisis Supervisado

IDGS91N - Kevin Iván Aguirre Silva

Extracción de Conocimiento en Bases de Datos - Ing.

Luis Enrique Mascote Cano

Chihuahua, Chih., 29 de noviembre de 2025

Índice

1. Introducción	3
2. Investigación de algoritmos	4
2.2. Algoritmos de regresión	4
<i>Regresión líneal</i>	4
<i>Regresión por Máquinas de Soporte Vectorial (SVR)</i>	5
3. Caso de estudio	5
3.1. Justificación del Algoritmo Elegido: Regresión Lineal	5
4. Diseño e implementación	6
4.1. Variables del Modelo	6
4.2. Estructura de Datos.....	6
4.3. Pipeline de Entrenamiento	7
4.4. Implementación (Python).....	7
4.1. Preparación de datos	7
4.2. Entrenamiento del modelo.....	8
4.3. Cálculo de las métricas seleccionadas.....	8
5. Resultados y evaluación	9
5.1. Discusión de Limitaciones y Líneas de Trabajo.....	10
6. Conclusiones y recomendaciones	10
7. Referencias.....	12

1. Introducción

El presente reporte documenta la aplicación de técnicas de Análisis Supervisado para abordar un problema de regresión, utilizando datos simulados de precios de viviendas. El objetivo principal de la investigación es predecir el valor de venta continuo de una propiedad basándose en sus características clave (área, número de habitaciones, antigüedad y distancia al centro). Para ello, se investigaron los algoritmos de Regresión Lineal y Regresión por Máquinas de Soporte Vectorial (SVR), analizando sus principios de funcionamiento, fortalezas y limitaciones. La meta es diseñar, implementar y evaluar un modelo de regresión, seleccionando inicialmente la Regresión Lineal como modelo base por su simplicidad e interpretabilidad, y posteriormente proponer líneas de trabajo para mejorar el desempeño del modelo.

2. Investigación de algoritmos

2.2. Algoritmos de regresión

Regresión lineal

1. ¿Qué resuelve? (objetivo)

Predecir el valor continuo de una variable dependiente en función de variables independientes asumiendo una relación lineal entre ellas.

2. Principio de funcionamiento (proceso)

Ajusta una línea recta (o hiperplano en varias dimensiones) que minimice la suma de los errores al cuadrado entre las predicciones y los valores reales.

3. Métricas de evaluación

Error Cuadrático Medio (MSE), Raíz del Error Cuadrático Medio (RMSE), Error Absoluto Medio (MAE), coeficiente de determinación (R^2).

4. Fortalezas

Es sencillo, rápido de entrenar, fácil de interpretar, y eficiente para relaciones lineales.

5. Limitaciones

No modela relaciones no lineales; es sensible a valores atípicos y puede tener bajo desempeño si la relación no es lineal (InteligenciaArtificial.Tech, 2024).

Regresión por Máquinas de Soporte Vectorial (SVR)

1. ¿Qué resuelve? (objetivo)

Ajustar una función que prediga valores continuos, permitiendo un margen de error aceptable alrededor de la función ajustada.

2. Principio de funcionamiento (proceso)

Encuentra una función dentro de un margen (epsilon) que se ajusta a los datos minimizando la complejidad del modelo y el error. Puede usar núcleos para modelar relaciones no lineales.

3. Métricas de evaluación

MSE, RMSE, MAE, R^2 , similares a la regresión lineal.

4. Fortalezas

Maneja bien dimensiones elevadas y relaciones complejas no lineales, robusto a sobreajuste si se regula adecuadamente.

5. Limitaciones

Más complejo de entrenar, menos interpretable que la regresión lineal, selección de parámetros (margen, núcleo) puede ser difícil (InteligenciaArtificial.Tech, 2024).

3. Caso de estudio

El problema consiste en predecir el valor de venta (precio) de una vivienda en función de sus características. Este es un problema de regresión, ya que la variable objetivo (el precio) es continua.

3.1. Justificación del Algoritmo Elegido: Regresión Lineal

Se elige la Regresión Lineal basándose en la suposición de que el precio de una vivienda tiene una relación directa y lineal con la mayoría de sus características clave (por ejemplo, a mayor tamaño o mayor número de habitaciones, mayor precio).

- **Razón principal:** La Regresión Lineal es sencilla, rápida de entrenar y, sobre todo, fácil de interpretar (Fortaleza 4). Es un excelente punto de partida para modelar relaciones que se asumen lineales.
- **Contexto:** Antes de pasar a modelos más complejos como SVR, es fundamental establecer una línea base de desempeño y entender la contribución lineal de las características. Si el desempeño es aceptable, se prefiere la simplicidad y la interpretabilidad de la Regresión Lineal.

4. Diseño e implementación

4.1. Variables del Modelo

Tipo de Variable	Nombre	Descripción
Variable Dependiente (Salida)	Precio	Valor continuo en dólares o la moneda local.
Variables Independientes (Entrada)	Area_m2	Metros cuadrados de la vivienda.
	Num_habitaciones	Cantidad de dormitorios
	Antigüedad_anios	Años desde la construcción
	Distancia_centro_km	Distancia en km al centro de la ciudad.

4.2. Estructura de Datos

Los datos se estructurarían en un DataFrame de Pandas, donde cada fila representa una vivienda y cada columna es una de las variables mencionadas.

Precio	Area_m2	Num_habitaciones	Antigüedad_anios	Distancia_centro_km
350000	150	3	10	5.2
280000	100	2	25	1.8
...

4.3. Pipeline de Entrenamiento

1. **Carga de Datos:** Importar el dataset de viviendas.
2. **Limpieza/Preprocesamiento:** Manejar valores faltantes (si los hay) y escalar características si es necesario (aunque para Regresión Lineal simple no es estrictamente obligatorio, es buena práctica).
3. **División de Datos:** Separar el dataset en conjuntos de Entrenamiento (para ajustar el modelo) y Prueba (para evaluar el desempeño). Típicamente, 70-80% para entrenamiento.
4. **Entrenamiento:** Ajustar el modelo de Regresión Lineal a los datos de entrenamiento.
5. **Predicción:** Usar el modelo entrenado para predecir precios en el conjunto de prueba.
6. **Evaluación:** Calcular las Métricas de Evaluación (MSE, R^2 , etc.) para medir la precisión.

4.4. Implementación (Python)

Se utilizarán las librerías pandas para la gestión de datos y scikit-learn para el modelo y las métricas.

Librerías e imports

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error
```

4.1. Preparación de datos

Crear DataFrame simulado

```
np.random.seed(42)
data = {
    'Area_m2': np.random.randint(50, 200, 100),
    'Num_habitaciones': np.random.randint(1, 5, 100),
    'Antigüedad_anios': np.random.randint(1, 50, 100),
    'Distancia_centro_km': np.random.uniform(0.5, 20, 100)
}
df = pd.DataFrame(data)
```

La columna 'Precio' (Variable Dependiente) se genera como una función lineal + ruido.
$$\text{Precio} = 2000 * \text{Area} + 10000 * \text{Habitaciones} - 500 * \text{Antigüedad} - 1000 * \text{Distancia} + \text{ruido}.$$

```
df['Precio'] = (  
    2000 * df['Area_m2'] +  
    10000 * df['Num_habitaciones'] -  
    500 * df['Antigüedad_anios'] -  
    1000 * df['Distancia_centro_km'] +  
    np.random.normal(0, 50000, 100) # Añadir ruido  
)
```

Definición de variables de entrada (X) y salida (y):

```
X = df[['Area_m2', 'Num_habitaciones', 'Antigüedad_anios', 'Distancia_centro_km']]  
y = df['Precio']
```

División entrenamiento/prueba (70% entrenamiento, 30% prueba):

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,  
                                                    random_state=42)
```

4.2. Entrenamiento del modelo

```
modelo_rl = LinearRegression()  
modelo_rl.fit(X_train, y_train)
```

Predicciones sobre el conjunto de prueba:

```
y_pred = modelo_rl.predict(X_test)
```

4.3. Cálculo de las métricas seleccionadas

```
# Error Cuadrático Medio (MSE)  
mse = mean_squared_error(y_test, y_pred)  
# Raíz del Error Cuadrático Medio (RMSE)  
rmse = np.sqrt(mse)  
# Error Absoluto Medio (MAE)  
mae = mean_absolute_error(y_test, y_pred)  
# Coeficiente de Determinación (R²)  
r2 = r2_score(y_test, y_pred)
```

Ejemplo de predicción para un caso:

```
ejemplo = pd.DataFrame({
    'Area_m2': [130], 'Num_habitaciones': [3],
    'Antiguedad_anios': [5], 'Distancia_centro_km': [3.0]
})
prediccion_ejemplo = modelo_rl.predict(ejemplo)
```

5. Resultados y evaluación

```
Datos de entrenamiento: (70, 4) (70,)
Datos de prueba: (30, 4) (30,)

--- Entrenando Modelo de Regresión Lineal ---

--- Métricas de Evaluación del Modelo ---
Error Cuadrático Medio (MSE): 3,018,372,603.35
Raíz del Error Cuadrático Medio (RMSE): 54,939.72
Error Absoluto Medio (MAE): 40,934.18
Coeficiente de Determinación (R²): 0.7075

Predicción para la vivienda de ejemplo: $289,924.33
```

El modelo de Regresión Lineal obtuvo un Coeficiente de Determinación (R^2) de 0.7075, lo que indica que aproximadamente el 70.75% de la variabilidad en los precios de las viviendas es explicada por las variables de entrada, lo cual es un desempeño aceptable para un modelo base. Sin embargo, el Error Cuadrático Medio (MSE) es muy alto (\$3,018,372,603.35) y el Error Absoluto Medio (MAE) de \$40,934.18 sugiere que, en promedio, las predicciones del modelo se desvían en más de \$40,000 del precio real, una limitación significativa que podría deberse a la presencia de relaciones no lineales en los datos o a la sensibilidad a valores atípicos inherente al modelo lineal.

5.1. Discusión de Limitaciones y Líneas de Trabajo

Una de las principales limitaciones de la Regresión Lineal es su incapacidad para modelar con precisión las relaciones complejas y no lineales, que son comunes en la fijación de precios de bienes inmuebles.

Líneas de Trabajo y Propuestas de Mejora:

6. **Exploración de Modelos No Lineales:** La principal propuesta es la sustitución o complemento con algoritmos que manejen mejor la no linealidad, como la Regresión por Máquinas de Soporte Vectorial (SVR), que fue discutida en la sección 2.2 del reporte, o modelos basados en árboles como Random Forest o Gradient Boosting.
7. **Ingeniería de Características:** Incluir variables más ricas o transformaciones (p. ej., el cuadrado del área o la interacción entre antigüedad y distancia) para capturar mejor las no linealidades.
8. **Detección y Tratamiento de Outliers:** Investigar y posiblemente eliminar o transformar los valores atípicos que el MSE, al ser sensible a errores grandes, sugiere que están afectando fuertemente el desempeño.

6. Conclusiones y recomendaciones

El estudio concluyó que el modelo de Regresión Lineal, elegido por su sencillez y facilidad de interpretación, logró explicar aproximadamente el 70.75% de la variabilidad del precio de las viviendas, según lo indicado por el coeficiente R^2 . Este desempeño es aceptable para un modelo inicial o "línea base". Sin embargo, el Error Absoluto Medio (MAE) de \$40,934.18 reveló una limitación significativa en la precisión predictiva, sugiriendo que el modelo base tiene dificultades para ajustarse a las variaciones reales del mercado inmobiliario. Esta desviación es consistente con las limitaciones teóricas de la Regresión Lineal, que es sensible a valores atípicos y no es apta para modelar relaciones no lineales complejas.

Basado en la evaluación de los resultados, se recomienda implementar las siguientes líneas de trabajo para mejorar la precisión del modelo:

- **Explorar Modelos No Lineales Avanzados:** Se recomienda probar la Regresión por Máquinas de Soporte Vectorial (SVR), que tiene la capacidad de usar núcleos para modelar relaciones complejas no lineales. Alternativamente, se sugiere la evaluación de modelos basados en árboles como Random Forest o Gradient Boosting.
- **Ingeniería de Características:** Implementar transformaciones de variables (p. ej., logaritmos) e incluir términos de interacción entre las características para que el modelo capture mejor las no linealidades de los datos.
- **Análisis de Outliers:** Realizar una detección y tratamiento riguroso de valores atípicos en los datos, ya que estos inflan el Error Cuadrático Medio (MSE) y el Error Absoluto Medio (MAE), afectando la robustez del modelo lineal.

7. Referencias

InteligenciaArtificial.Tech. (27 de Agosto de 2024). *Algoritmos de regresión*. Obtenido de InteligenciaArtificial.Tech: <https://inteligenciaartificial.tech/2024/08/27/algoritmos-de-regresion/>