### 1. Introducción

El machine learning (aprendizaje automático) es un enfoque de la inteligencia artificial que permite a las máquinas aprender de los datos y mejorar en sus tareas sin estar explícitamente programadas para cada posible escenario. En este reporte se aborda el uso machine learning para predecir el valor de una vivienda en función de varias características. Las redes neuronales, que son modelos inspirados en el funcionamiento del cerebro humano, se usan para abordar problemas de predicción complejos.

En este caso específicamente se utiliza un conjunto de datos que contiene el valor de viviendas y características de cada una que podrían ayudar a interpretar su valor

### 2. Uso de modelos de regresión

Dado que el objetivo es predecir el precio de una vivienda, que es una variable continua, es apropiado usar modelos de regresión. La regresión se enfoca en predecir valores numéricos continuos, mientras que los modelos de clasificación son más adecuados cuando se deben asignar etiquetas o categorías a los datos (p. ej., tipo de vivienda). En este caso, los modelos de regresión como la regresión lineal, el random forest y las redes neuronales son más apropiados para predecir un valor numérico.

### 3. Importación de librerías

En este proyecto, se importan varias librerías clave:

* Pandas y NumPy: Para manipulación de datos y operaciones numéricas.
* Matplotlib y Seaborn: Para visualización de datos y gráficos.
* Scikit-learn: Provee herramientas para modelado y evaluación (regresión lineal, random forest, divisiones de datos).
* TensorFlow: Utilizado para crear y entrenar una red neuronal.
* Scipy: Para detectar y eliminar valores atípicos usando puntuaciones Z.

### 4. Revisión, estructura y limpieza de datos del CSV

La calidad de los datos es un factor crucial en el éxito de cualquier modelo de machine learning. Los datos incompletos, erróneos o anómalos pueden generar un impacto negativo en la precisión de los modelos, especialmente en aquellos que realizan predicciones numéricas, como los modelos de regresión.

#### Proceso de limpieza:

El dataset original contenía columnas que no aportaban información relevante al modelo de predicción, como date, street, statezip y country. Estas columnas fueron eliminadas para reducir el ruido en los datos. Además, se observaron valores nulos en algunas filas, los cuales fueron eliminados con el método dropna(). Eliminar valores nulos en lugar de imputarlos (llenarlos con un valor, como el promedio) es una decisión que puede depender del volumen de datos disponible y del impacto que estos valores tengan en el análisis.

#### Creación de nuevas características:

* house\_age: Se creó una nueva columna que representa la edad de la casa restando el año de construcción (yr\_built) del año actual (2024). Esto ayuda a capturar cómo la antigüedad de la casa puede afectar su valor.
* renovated: Se transformó la variable yr\_renovated para indicar si la casa ha sido renovada (1 si ha sido renovada, 0 si no).

#### Eliminación de valores atípicos:

Los valores atípicos o outliers son datos que se alejan considerablemente del resto de las observaciones y pueden distorsionar los resultados de los modelos de machine learning, especialmente en modelos sensibles como la regresión lineal. Los outliers pueden ser errores de medición o representaciones de casos poco comunes que no generalizan bien en la predicción de nuevos datos.

##### Puntuación Z (Z-Score):

La puntuación Z es una medida estadística que indica cuántas desviaciones estándar un valor está alejado de la media. Se calcula como:

Z=(X−μ)/σ

Donde:

* X es el valor observado,
* μ es la media de la variable,
* σ es la desviación estándar.

Una puntuación Z alta (en términos absolutos) indica que el valor está muy alejado de la media y, por lo tanto, podría considerarse un outlier. En este proyecto, se estableció un umbral de 3 desviaciones estándar para identificar los valores atípicos. Esto significa que si una puntuación Z de un valor es mayor a 3 o menor a -3, el valor es considerado un outlier y se elimina del dataset.

El uso de la puntuación Z permite:

1. Detectar valores extremos: Valores que no siguen el patrón general de los datos.
2. Mantener la integridad del dataset: Al eliminar estos valores extremos, evitamos que distorsionen las métricas de error y sesguen los modelos.

El código relevante para esta limpieza fue:

z\_scores = np.abs(stats.zscore(data\_cleaned))

data\_cleaned = data\_cleaned[(z\_scores < 3).all(axis=1)]

Este paso asegura que los datos sean más consistentes y representativos del conjunto general, lo que mejora la capacidad de los modelos para aprender relaciones relevantes entre las características sin verse influenciados por valores atípicos que podrían desvirtuar los resultados.

### Análisis de Correlación

Una parte crucial del preprocesamiento de datos es el análisis de correlación entre las variables. La correlación mide la relación lineal entre dos variables y se expresa mediante el coeficiente de correlación de Pearson, cuyo valor varía entre -1 y 1. Un valor cercano a 1 indica una fuerte correlación positiva, mientras que un valor cercano a -1 indica una fuerte correlación negativa. Un valor cercano a 0 sugiere que no hay relación lineal entre las variables.

#### Correlación entre todas las variables:

En este proyecto, se calculó la matriz de correlación para observar las relaciones entre todas las variables. Este análisis es útil para identificar patrones entre las características que pueden influir en el valor de la vivienda o que podrían ser redundantes. Si dos variables están altamente correlacionadas, es posible que una de ellas no aporte información adicional al modelo y se considere eliminarla o combinarla.

#### Correlación específica con el precio:

Dado que el objetivo principal es predecir el precio de la vivienda, el siguiente paso fue aislar las correlaciones que cada variable tiene con el precio. Esto nos ayuda a comprender qué características tienen el mayor impacto en la predicción del valor.

Correlación con el precio:

price 1.000000

sqft\_living 0.610436

sqft\_above 0.528053

bathrooms 0.450738

bedrooms 0.295792

floors 0.279115

sqft\_basement 0.202730

view 0.182330

city 0.085463

sqft\_lot 0.081160

condition 0.049008

house\_age -0.021784

renovated -0.063463

waterfront NaN

#### Importancia del análisis de correlación:

* Identificación de variables clave: Este análisis permite centrarse en las variables más importantes que influyen en el precio. Las variables con correlación más alta con el precio suelen ser las que mayor peso tendrán en los modelos de predicción.
* Reducción de complejidad: En lugar de incluir todas las variables en los modelos de machine learning, este análisis ayuda a seleccionar solo las más relevantes, lo que puede mejorar la eficiencia y el rendimiento del modelo.
* Mejorar la interpretabilidad: Para entender el comportamiento del modelo y sus predicciones, es importante saber cuáles características influyen más en la variable objetivo (precio).

#### Normalización de los datos:

Después de limpiar los datos, se aplicó la normalización a las variables numéricas para garantizar que todas las características tengan la misma escala. Este paso es especialmente importante para algoritmos de aprendizaje como la regresión lineal y las redes neuronales, que son sensibles a la magnitud de los datos. Se utilizó el StandardScaler de Scikit-learn, que transforma las variables para que tengan una media de 0 y una desviación estándar de 1.

### 5. División de datos entre train y test

Los datos fueron divididos en conjuntos de entrenamiento y prueba en una proporción de 80% y 20%, respectivamente. Se utilizó el parámetro random\_state=42 para asegurar que los resultados sean reproducibles. Esta división es crucial para evaluar el rendimiento del modelo en datos no vistos, pero que siguen siendo “reales”.

### 6. Algoritmo de regresión lineal

La regresión lineal es uno de los algoritmos más simples y utilizados en machine learning para tareas de predicción numérica. El modelo asume que existe una relación lineal entre las variables independientes (X) y la variable dependiente (y). La idea principal es encontrar una línea que minimice la distancia (o el error) entre los puntos de datos y la predicción del modelo.

Matemáticamente, el modelo de regresión lineal puede expresarse como:

y=β0+β1X1+β2X2+...+βnXn+ϵ

Donde:

* y es el valor que se está prediciendo (en este caso, el precio de la vivienda),
* X1,X2,...,Xn son las variables predictoras,
* β0 es el término de intercepción,
* β1,β2,...,βn son los coeficientes que representan la influencia de cada variable en el precio,
* ϵ es el término de error.

El algoritmo de regresión lineal encuentra los valores de los coeficientes (β) que minimizan la suma de los errores al cuadrado entre las predicciones y los valores reales.

#### Justificación para usar la regresión lineal:

* Simplicidad y eficiencia: La regresión lineal es fácil de interpretar y computacionalmente eficiente, lo que la convierte en una buena opción cuando se desea un modelo rápido para probar una hipótesis.
* Relaciones lineales: En este caso, existe evidencia de que algunas características (como el tamaño de la casa y el número de habitaciones) tienen una relación lineal con el precio de la vivienda.

#### Resultados:

* Error cuadrático medio (MSE): Esta métrica mide la diferencia promedio al cuadrado entre las predicciones y los valores reales. Un valor más bajo indica un mejor ajuste del modelo.
* Error cuadrático medio: 0.11135722537383967
* Coeficiente de determinación (R²): Indica la proporción de la varianza en la variable dependiente que es explicada por las variables independientes. Un valor más cercano a 1 indica que el modelo explica bien los datos.
* R2: 0.46221852918388584

Claro, aquí tienes una explicación extendida de los modelos de predicción, cómo funcionan, por qué son apropiados para este caso y una sección de resultados para cada uno donde puedas incluir las gráficas y métricas. Este enfoque te permitirá agregar fácilmente las visualizaciones y estadísticas cuando las tengas listas.

### 7. Algoritmo de random forest regresión

El random forest es un modelo de aprendizaje de conjunto que utiliza múltiples árboles de decisión para mejorar la precisión de las predicciones. En lugar de construir un solo árbol profundo, el random forest genera muchos árboles (n\_estimators) y los combina, lo que reduce la varianza y mejora la robustez del modelo.

Cada árbol en el random forest es entrenado con una muestra aleatoria del dataset, y durante el proceso de construcción del árbol, solo una selección aleatoria de características es considerada en cada división del nodo. El resultado final es la media de las predicciones de todos los árboles.

El algoritmo sigue el principio de "la sabiduría de las multitudes": aunque cada árbol pueda cometer errores, la agregación de muchos árboles tiende a suavizar estos errores, mejorando la precisión general.

#### Justificación para usar random forest:

* Capacidad para capturar relaciones no lineales: A diferencia de la regresión lineal, el random forest es capaz de modelar relaciones complejas y no lineales entre las variables predictoras y el precio de la vivienda.
* Reducción del overfitting: El random forest utiliza un mecanismo de promedio de múltiples árboles, lo que reduce la posibilidad de sobreajuste (overfitting) al conjunto de entrenamiento.

#### Resultados:

* Error cuadrático medio (MSE): Al igual que en la regresión lineal, el MSE se utiliza para medir el rendimiento del modelo.
* Error cuadrático medio: 0.09130222156451673
* R²: El coeficiente de determinación también es una métrica útil aquí para evaluar la capacidad explicativa del modelo.
* R2: 0.5590708834841409

### 8. Redes neuronales con TensorFlow

Las redes neuronales son modelos inspirados en el cerebro humano y están diseñadas para capturar patrones complejos en los datos. Una red neuronal consta de capas de neuronas, donde cada neurona recibe entradas, las transforma usando una función de activación y las pasa a la siguiente capa. En una red neuronal completamente conectada, cada neurona de una capa está conectada a todas las neuronas de la capa siguiente.

En este proyecto, se utilizó una arquitectura con múltiples capas densas:

* Capa de entrada: La primera capa contiene un número de neuronas igual al número de características predictoras.
* Capas ocultas: Se utilizaron varias capas ocultas con neuronas que van de 32 a 16 unidades, con funciones de activación no lineales como ReLU para modelar relaciones complejas.
* Capa de salida: La capa de salida contiene una única neurona para predecir el valor numérico del precio.

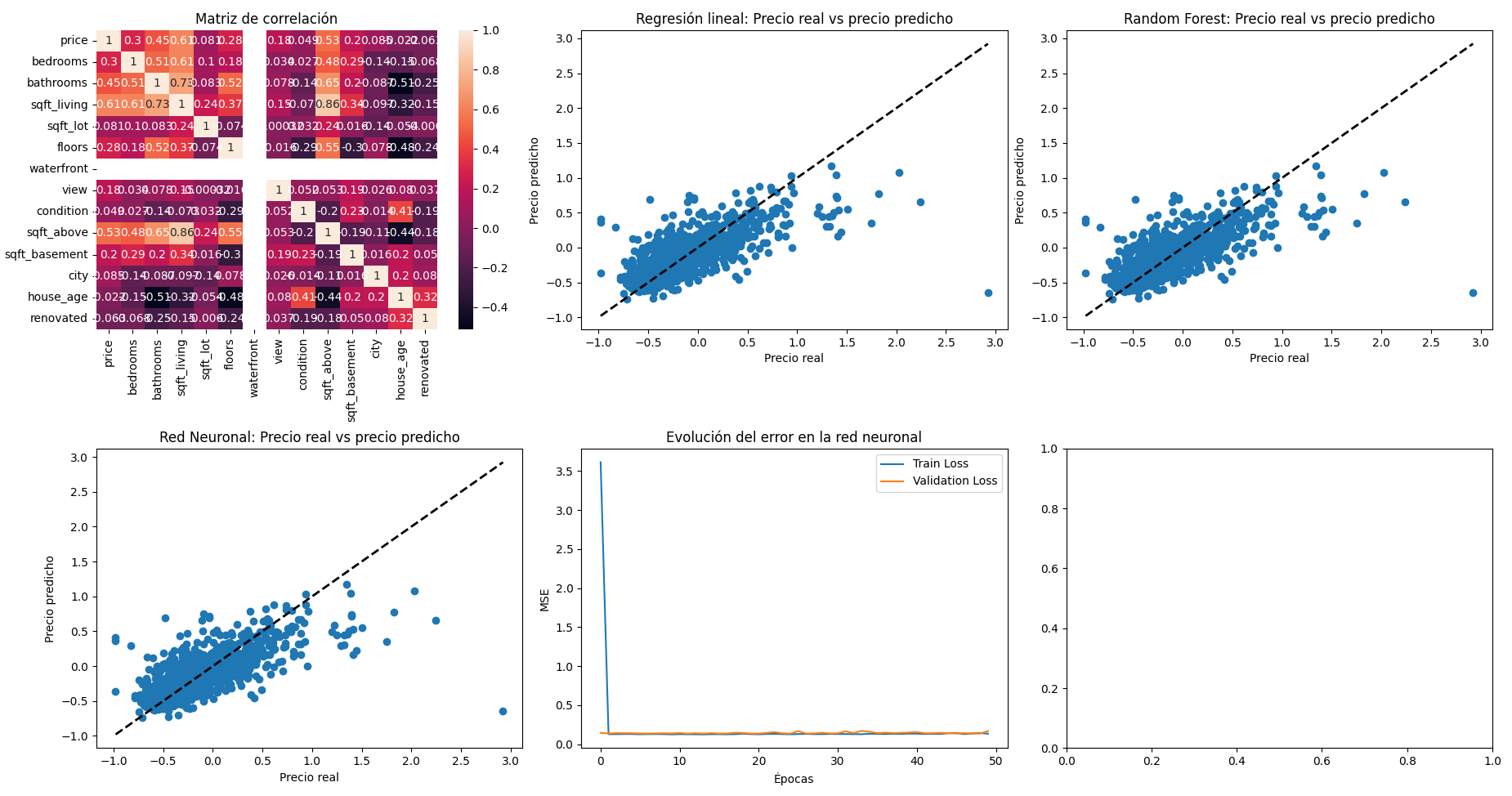
La red neuronal se entrenó utilizando el optimizador Adam y la función de pérdida MSE. El modelo ajusta sus pesos para minimizar el error en las predicciones a lo largo de varias épocas.

#### Justificación para usar redes neuronales:

* Captura de patrones complejos: Las redes neuronales son capaces de aprender patrones complejos en los datos que otros algoritmos, como la regresión lineal, no pueden modelar.
* Potencial de mejora: Si bien las redes neuronales pueden ser más difíciles de interpretar, su capacidad para mejorar con más datos y ajuste de hiperparámetros las hace una herramienta poderosa en predicciones numéricas.

#### Resultados:

* MSE y R²: Al igual que en los otros modelos, estas métricas permiten evaluar el rendimiento de la red neuronal. El entrenamiento de la red suele requerir más tiempo debido a la complejidad de los cálculos.
* Error cuadrático medio: 0.13661989526570717
* R2: 0.34021660496584527



### 9. Conclusiones y puntos a mejorar

En este proyecto se evaluaron tres modelos de predicción para estimar el precio de una vivienda: regresión lineal, random forest y redes neuronales. Cada modelo presentó ventajas y limitaciones, lo que permite hacer una evaluación comparativa sobre su rendimiento en función de la precisión, la capacidad de capturar relaciones complejas y la posibilidad de mejorarlos en el futuro.

#### Análisis de la correlación de las variables

El análisis de correlación entre las características del dataset y el precio de la vivienda permitió identificar las variables más relevantes para la predicción del valor. A continuación se muestra la correlación de las principales variables con el precio:

* Superficie habitable (sqft\_living): 0.61
* Superficie sobre el nivel del suelo (sqft\_above): 0.53
* Número de baños (bathrooms): 0.45
* Número de habitaciones (bedrooms): 0.29

Variables como la superficie habitable y el número de baños muestran una correlación significativa con el precio, lo que indica que son características importantes en la determinación del valor de una vivienda. Sin embargo, algunas variables como house\_age (edad de la casa) y renovated (si la casa ha sido renovada o no) tienen correlaciones muy bajas o negativas, lo que sugiere que su impacto en el precio es limitado o contrario a lo esperado.

Este análisis es fundamental para guiar el proceso de selección de variables para los modelos de machine learning. Las variables con correlación más fuerte, como sqft\_living y bathrooms, tendrán un mayor impacto en las predicciones y deben ser tratadas cuidadosamente en el preprocesamiento y el ajuste de los modelos.

#### Comparación y evaluación de los modelos

Al comparar los tres modelos, el random forest es el que ofrece el mejor balance entre precisión y capacidad para capturar la complejidad de los datos. Aunque la regresión lineal es eficiente y fácil de interpretar, no es capaz de manejar relaciones no lineales de manera efectiva. Las redes neuronales tienen potencial, pero su rendimiento actual sugiere que requieren un ajuste más profundo de los hiperparámetros (como el número de capas, neuronas, tasa de aprendizaje, etc.).

#### Puntos a mejorar

1. Ajuste de hiperparámetros: El modelo de redes neuronales podría beneficiarse de una búsqueda de hiperparámetros más exhaustiva, explorando diferentes arquitecturas y parámetros como el tamaño de las capas ocultas, las funciones de activación y la tasa de aprendizaje.
2. Más características y datos: Incluir más características relevantes o transformar las existentes podría mejorar la capacidad predictiva de los modelos. Por ejemplo, investigar variables adicionales como la proximidad a servicios esenciales o zonas comerciales podría proporcionar más información para la predicción del precio.
3. Preprocesamiento avanzado: Mejorar el preprocesamiento de los datos, especialmente en cuanto a la detección de outliers y la imputación de datos nulos, podría reducir el ruido en los datos y aumentar la precisión de los modelos.
4. Modelos avanzados: Explorar modelos más avanzados como el gradient boosting (XGBoost, LightGBM) podría proporcionar mejoras adicionales en la precisión, ya que estos modelos son conocidos por ser muy efectivos en problemas de predicción de valores continuos.

En resumen, el random forest demostró ser el modelo más efectivo en este análisis, pero existe un margen considerable para mejorar tanto en las redes neuronales como en otros enfoques avanzados. El siguiente paso sería afinar más los modelos, mejorar el preprocesamiento y considerar el uso de técnicas más sofisticadas para obtener mejores resultados.