Predicción de precios de casas con algoritmos de ML

Luis Ernesto Monsalve Arango, Johnatan Andrés Gómez Monsalve
Departamento de ingeniería de sistemas
Universidad de Antioquia, Colombia
ernesto.monsalve@udea.edu.co
johnatana.gomez@udea.edu.co

***Abstract*— Este documento presenta el camino ejecutado para desarrollar un algoritmo de ML que permita predecir el precio de una casa, teniendo en cuenta las categorías más relevantes para un posible comprador, categorías tales como el vecindario, el tipo de vivienda, los materiales de construcción de la casa entre otros, en este se abordan temas como el tratamiento de datos faltantes y correlación de las variables que permiten tener un modelo bien ajustado, generando una buena predicción del precio de las casas.**

I. PROBLEMA

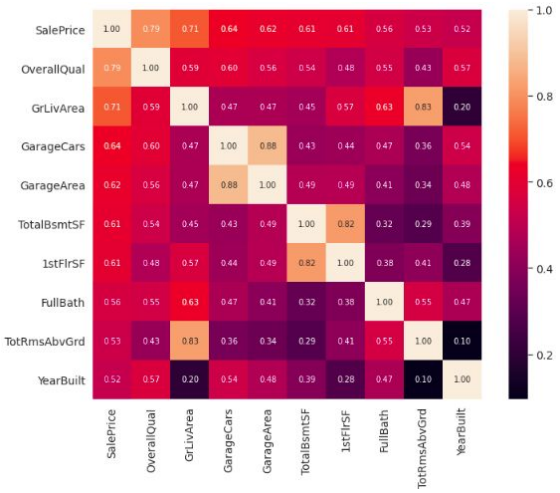
El problema abordado consiste en predecir el valor de cada casa teniendo en cuenta las características que un posible comprador tiene presente al momento de realizar la compra de su inmueble, algunas de estas variables son: tamaño del lote, vecindario, año de construcción, el sótano, entre otras.

Este problema corresponde a un modelo de regresión debido a que, teniendo en cuenta estas característica de interés de los usuarios se pretende predecir el valor de la vivienda. Se cuenta con un total de 80 variables de las cuales 43 de ellas son categóricas, podemos mencionar las siguientes, que son quienes tienen más relación con el precio de la casa, la cual es la variable a predecir, estas variables son: material de terminado, área habitable, tamaño del garaje (capacidad de carros) y área del garaje en pies cuadrados, de las 80 variables de entrada con las que cuenta el problema, se evidencia una pérdida de datos de más del 80% en las categorías de: PoolQC, MiscFeature, Alley, Fence, adicional a esto se cuenta con otras categorías con una pérdida de datos bastante significante con porcentajes de pérdida de 40%, 17% o en su mayoría del 5% o 2%. La base de datos cuenta con valores faltantes y valores atípicos, los cuales requieren de un tratamiento especial para conseguir una predicción de calidad.

Para tratar los datos faltantes se empleó la mediana, la moda, y el análisis de correlación para ciertas características, también se empleó label encoder para cambiar las variables categóricas a numéricas para que el modelo pudiera recibir los datos.

Haciendo uso de una matriz de correlación (Gráfico 1) se identifica las variables que tienen mayor correlación con el precio de venta que es la variable que nos interesa predecir. De aquí se puede identificar variables altamente correlacionadas entre sí por tal motivo se realiza una combinación de estas para solo trabajar con una variable de estas, por ejemplo, 'GarageCars' y 'GarageArea' son variables muy relacionadas, de igual forma TotalBsmtSF' y '1stFloor', en ambos casos basta con conservar una de las variables para realizar la predicción.

GRÁFICO 1. MATRIZ DE CORRELACIÓN



Para los valores faltantes se debe tener en cuenta que reemplazar los missing values por 0, injustificadamente, empeora la precisión del modelo, por tal motivo se reemplaza con valores de las muestras más cercanas correspondientes, y se evalúa cada una de las columnas para indicar que dato es el apropiado para su llenado de datos. Las variables con más del 80% de datos faltantes no son tenidas en cuenta para el modelo. Luego de esto, y conociendo las variables categóricas, se llenan los NaN con 'None', puesto que esto quiere decir que la variable no aplica para la muestra o no existe, y los valores restantes se procesan con la media o media garantizando un modelo en óptimas condiciones.

II. ESTADO DEL ARTE

Se presentan a continuación cuatro artículos que usan la misma base de datos empleada en este informe, donde se indica que modelos entrenaron, que técnicas o métricas de evaluación utilizaron y qué resultados obtuvieron.

A. Predicción de precio con regresiones lineales[1]:

Elimina las columnas que tengan más del 50% de datos faltantes. Reemplaza todos los valores faltantes por 0.

Implementa su propio método para cambiar las categorías numéricas. Si la correlación con respecto a la columna objetivo es 0.05, elimina la columna.

Modelos: LinearRegression, DecisionTreeRegressor,

Lasso, ElasticNet, Random Forest, AdaBoostRegressor, GradientBoostingRegressor, XGB.
Metodologías de validación: r2_score.
Resultados: El mejor es Gradient boost con n_estimators=50 con un score de 0.85039.

B. House Prices Notebook por Kenny Van[2]:

Elimina todas las columnas que tienen valores faltantes, simple imputation, extended imputation.
Excluye las categóricas, Label encoding, One hot encoding.
Modelos: Random Forest, XGB.
Metodologías de validación: MAE.
Resultados: XGBRegressor(n_estimators=500, learning_rate=0.05) MAE: 16802.965325342466.

C. House Price Prediction por Prosenjit123[3]:

Borra las columnas que tienen más del 80%. Llena los datos según la columna (N/A, mediana, moda, 0 o valor más frecuente).
Asigna un valor a las categóricas. Separa las categóricas en dos, ordinales y nominales. Considera solo las características con una correlación mayor a 0.2.
Modelos: RandomForestRegressor, GradientBoostingRegressor, XGBRegressor.
Resultados: Hace una transformación para normalizar los datos y queden con una gaussiana.
Al final es difícil entender los resultados, al parecer no válida.

D. House Prices using GradientBoostingRegressor 90%[4]:

Rellena cada columna con un valor que escoge como la media.
One-Hot Encoding.
Modelo: GradientBoostingRegressor.
Metodologías de validación: score, cross_val_score.
Resultados: GradientBoostingRegressor score de 90.

III. EXPERIMENTOS

La base de datos empleada para este problema es tomada de Kaggle.com, "House Prices: Advanced Regression Techniques" [5], que cuenta con 80 variables de entrada, de las cuales 43 son variables categóricas, que son aquellas que pueden describir cualidades o categorías, y 37 variables numéricas, en las cuales está incluida la variable a predecir que en este caso particular es el precio de una casa.

Se utilizó bootstrapping como metodología de validación, es esta se empleo el 80% de los datos para el entrenamiento del modelo y el otro 20% para validar el modelo de regresión.

Bootstrapping:Es una técnica empleada para validar la efectividad de un modelo predictivo, los métodos de conjunto, la estimación del sesgo y la varianza del modelo, para este fin se puede utilizar la función "grid search cv" que implementa el método de "fit" y arroja un score, tambien implementa metodos com "predict", predict proba", "decision function", "transform" y "inverse transform".

Para evaluar el rendimiento de los modelos se utilizaron; **MAE, MAPE, RMSE Y R2**

- **MAE:** Promedio de todos los errores absolutos.

$$mae = \frac{\sum_{i=1}^{n} abs\left(y_i - \lambda(x_i)\right)}{n}$$

En este se miden las diferencias entre dos variables continuas.

- **MAPE:** Error porcentual absoluto medio

$$\mathrm{M} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right|$$

es una medida de precisión de predicción de un método de pronóstico, se usa comúnmente como una función de pérdida para problemas de regresión y en la evaluación de modelos.

- **RMSE**: error cuadrático medio

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

Es de fácil interpretación y utiliza valores absolutos pequeños que facilitan los cálculos informáticos.

- **R2:** R cuadrado, se calcula usando la siguiente fórmula:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

donde SS res es la suma residual de cuadrados y SS tot es la suma total de cuadrados, entre más cercano a 1 el valor de r-cuadrado, mejor es el modelo.

Para la elección de los mejores parámetros de los modelos se utilizó la función *GridSearchCV* de sklearn para los que aplicaban, recibe un conjunto de parámetros y entrega cual es la combinación que da mejores resultados. Los modelos entrenados con su respectivos resultados se describen a continuación, los algoritmos utilizados están disponibles en el siguiente enlace HousePrices [7] .

A. Regresión Múltiple

Con este modelo se busca simular el comportamiento del conjunto de datos a través de una función, no fue necesario configurar parámetros ya que no aplica.Se obtuvieron resultados muy buenos comparados con los demás, un factor importante pudo ser el ajuste inicial que se le hizo a la variable de salida para que tuviera una distribución más uniforme. Los resultados de las métricas de evaluación se describen en la Tabla 1.

TABLA 1
RESULTADOS REGRESIÓN MÚLTIPLE

MAE	MAPE	RMSE	R2

0.0916	0.764	0.121	0.894

B. K vecinos más cercanos

Este modelo asume el comportamiento de la variable de salida basado en la similaridad y comportamiento de sus muestras 'vecinas'. Se variaron los parámetros de *n_neighbors* entre 1 y 10, y *'algorithm'* entre 'ball_tree' o 'brute', teniendo como mejores parámetros el algoritmo 'ball_tree' con 5 vecinos. Los resultados son expuestos en la Tabla 2.

TABLA 2
RESULTADOS KNN

MAE	MAPE	RMSE	R2
0.161	1.345	0.217	0.498

C. Random Forest

Este modelo consiste en una combinación de árboles de decisión basados en vectores aleatorios independientes. Se variaron los parámetros *'n_estimators'* entre [20, 50, 100], y *'max_depth'* entre [15,18,20]. Se trabajó con un 'max_depth' de 15 y 'n_estimators' de 100.

TABLA 3
RESULTADOS RANDOM FOREST

MAE	MAPE	RMSE	R2
0.083	3.507	0.117	0.894

D. Gradient boosting

En este modelo se usan árboles de decisión de forma escalonada. Los resultados de entrenar este modelo se describen en la tabla 4.

TABLA 4
RESULTADOS GRADIENT BOOSTING

MAE	MAPE	RMSE	R2
0.0757	3.559	0.101	0.925

E. Regresión con vectores de soporte RBF

Este método Utiliza el algoritmo Support Vector Machine
para predecir una variable continua. Se varía el valor de epsilon entre [0.004, 0.008, 0.0005, 0.0008], teniendo como mejor opción 'epsilon' igual a 0.008.

TABLA 4
RESULTADOS SVM RBF

MAE	MAPE	RMSE	R2
0.095	3.463	0.159	0.792

TABLA 5
RESULTADOS UNIFICADOS

MODELO	RMSE
Regresión Múltiple	0.121
KNN	0.217
Random Forest	0.117
Gradient Boosting	0.101
SVM RBF	0.159

Con los resultados obtenidos, unificados en la Tabla 5, anteriormente descritos se llega a la conclusión basados en que la medida de desempeño elegida es RMSE, que los tres mejores modelos son:

- Regresión múltiple
- Gradient Boosting
- Random Forest

IV. EXTRACCIÓN DE CARACTERÍSTICAS POR PCA

Con el fin de optimizar los modelos se opta por una reducción de dimensionalidad por medio de PCA para la extracción de características. Este análisis se realiza sobre los tres modelos que presentaron mejor desempeño, a continuación los resultados

TABLA 5
RESULTADOS KNN

	Regresión Múltiple	Random Forest	Gradient Boosting
MAE	9.537	9.518	9.526
RMSE	9.546	9.526	9.535

La extracción de características no parece mejorar el desempeño del modelo, por el contrario, aumenta el error.

V. CONCLUSIONES

Se observa una gran similitud entre los artículos elegidos como referencia y el modelo generado en este proyecto, debido a que en todos se debe hacer una limpieza o tratamiento de datos, ya que sin este, sería muy difícil obtener un modelo predictivo que logre estar ajustado a los datos reales que se pretenden entregar, por ende se deja por sentado que en un problema con tantas variables es indispensable realizar el tratamiento de datos, y definir la mejor manera de realizar dicho tratamiento para la efectividad del modelo.

Una diferencia grande fue ver modelos que mostraron un desempeño similar al Gradient Boosting, ya que en

algunos de los artículos era mencionado como uno de los mejores muy por encima de los otros. Los resultados obtenidos pueden derivar del tratamiento que se le dió a la variable de salida para tener una versión normalizada y el análisis detallado de cada variable para identificar el correcto llenado de valores vacíos.

Se observa que al usar metodologías de validación diferentes a las planteadas en los artículos de referencia, algunos resultados son muy similares, haciendo notar que cualquier metodología usada puede brindar buenos resultados y garantizar la fiabilidad del modelo predictivo.En algunos casos no se pudo comparar directamente el desempeño de los modelos debido al aplicar una transformación sobre la variable de salida las unidades de las métricas cambiaban

VI. REFERENCIAS

[1] Predicción de precios con regresiones lineales
https://www.kaggle.com/jesuscarmona12/predicci-n-de-precios-con-regresiones-lineales
[2] House Prices Notebook por Kenny Van
https://www.kaggle.com/kennyvan/house-prices-notebook
[3[House Price Prediction por Prosenjit123
https://www.kaggle.com/prosenjit123/house-price-prediction
[4] House Prices using GradientBoostingRegressor 90%
https://www.kaggle.com/mountaga/house-prices-using-gradientboostingregressor
[5] House Prices: Advanced Regression Techniques
https://www.kaggle.com/c/house-prices-advanced-regression-techniques
[6] House Prices - Algoritmos y funciones
https://colab.research.google.com/drive/1tVoHno54Z9ggYSDMDnn2WpqNQhw2S1e5?usp=sharing