memoria del proyecto de Machine Learning

Predicción del Precio de Coches de Segunda Mano

Autor: Mikel Guillén Baqué

1. INTRODUCCIÓN

En el mercado automovilístico de segunda mano en España, tanto compradores como vendedores se enfrentan a la dificultad de determinar un precio justo para los vehículos. La gran variedad de marcas, modelos, años de fabricación, kilometraje, tipo de combustible, potencia, entre otros factores, hace que la tasación de un coche no siempre sea sencilla ni objetiva. Además, la falta de transparencia o la asimetría de información puede derivar en decisiones de compra o venta desfavorables.

En este contexto, un modelo de machine learning capaz de predecir de forma automatizada el precio estimado de un coche de segunda mano puede aportar un valor significativo. Para plataformas de compraventa, concesionarios, tasadores o incluso usuarios particulares, una herramienta basada en datos reales que ofrezca estimaciones precisas puede mejorar la eficiencia del mercado, generar confianza en las transacciones y ayudar a detectar precios anómalos o fuera de mercado.

El objetivo del proyecto es desarrollar un modelo de machine learning capaz de predecir el precio de coches de segunda mano en España.

2. DATASET

Se trata de datos reales de España en 2023 obtenidos en la web:

https://datamarket.es/#vehiculos-de-segunda-mano-dataset

Se ha trabajado con un dataset de muestra, mediante subscripción de pago incluye:

- Volumen estimado: 500000 registros cada 24 h
- Histórico: disponible desde 2020-11
- Integración con bases de datos y API.

El dataset cuenta con los siguientes campos:

Campo	Descripción	Non - Null Count	Dtype	Campo	Descripción	Non - Null Count	Dtype
vehicle_type	Tipo de vehículo	100000	object	make	Marca del vehículo	100000	object
model	Modelo del vehículo	100000	object	version	Versión del vehículo	99842	object
fuel	Tipo de combustible	96810	object	year	Año de fabricación	100000	int64
kms	Kilometraje	99189	float64	power	Potencia del vehículo	99709	float64
doors	Número de puertas	0	float64	shift	Tipo de cambio (Automático/Manual)	98982	object

color	Color del vehículo	0	float64	photos	Número de fotos del anuncio	100000	int64
description	Descripción del anuncio	55810	object	price	Precio de venta del vehículo	100000	int64
currency	Moneda del precio	100000	object	location	Ciudad del anuncio	100000	object
publish_date	Fecha de publicación del anuncio	100000	object	update_date	Fecha de actualización del anuncio	100000	object
dealer_name	Vendedor del vehículo	99908	object	dealer_description	Descripción del anunciante	81558	object
dealer_address	Dirección del anunciante	99908	object	dealer_zip_code	Código postal del anunciante	99908	float64
dealer_city	Ciudad del anunciante	99908	object	dealer_country_code	Código de país del anunciante	100000	object
dealer_is_professional	¿Es profesional?	100000	bool	dealer_website	Página web del anunciante	79913	object
dealer_registered_at	Fecha de registro en la plataforma	99908	object	date	Fecha de extracción de la información	100000	object

Cuenta con anuncios de vendedores profesionales, un total de 100.000 anuncios (filas) y 28 campos (columnas). Existen campos con valores nulos.

Realizamos una primera limpieza de datos:

- Observamos que hay anuncios duplicados, con la única diferencia de la localización del vendedor. Esto se debe a que los vendedores profesionales cuentan con diferentes tiendas donde ofertan el mismo coche. Con esto reducimos el dataset a 13,075 filas.
- Eliminamos filas con valores faltantes para “power” y “kms”. Los de “power” un total de 291 podrían buscarse y rellenar manualmente. Los “kms” no pueden estimarse y son un valor que podría generar mucho ruido si lo estimáramos.
- Los valores faltantes de “shift” y “fuel” los hemos reemplazado con una nueva categoría “otros/unknown”.
- Eliminamos los campos que no aportan valor al estudio: "vehicle_type", "version","doors","color","photos","description","currency","location","publish_date", "update_date","dealer_name","dealer_description","dealer_address","dealer_city", "dealer_country_code","dealer_is_professional","dealer_website","dealer_registered_at","date","dealer_zip_code"

Realizamos un EDA preliminar y mediante boxplots observamos outliners con valores erróneos del database. Este tipo de anuncios de vehículos suelen realizarse de manera manual por lo que puede haber datos erróneos. Seguimos limpiando estas filas:

- Vehículos con menos de 200 km y más de 500.000kms

- Vehículos de menos de 1000 euros. Los profesionales no suelen vender este tipo de vehículos.

Con todo ello nos quedamos con 11185 vehículos (filas) y 8 columnas:

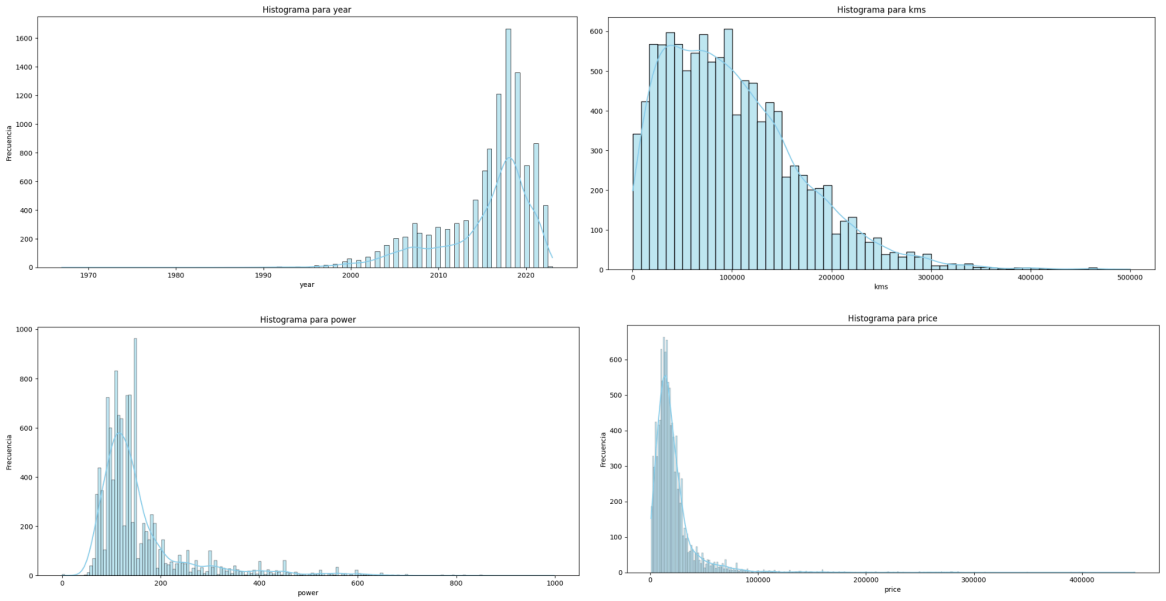
- 4 numéricas: “year”, “kms”, “power” y “price”
- 4 categóricas: “make”, “model”, “fuel” y “shift”

3. ANÁLISIS EXPLORATORIO (EDA)

En primer lugar, hemos realizado un countplot de las variables categóricas. Observamos que están desbalanceadas. Identificamos el número de categorías por variable:

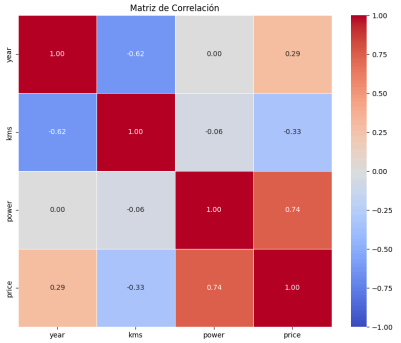
- “make”: 64 marcas
- “model”: 723 modelos
- “fuel”: 4 tipos de combustible
- “shift”: 3 tipos de cambio

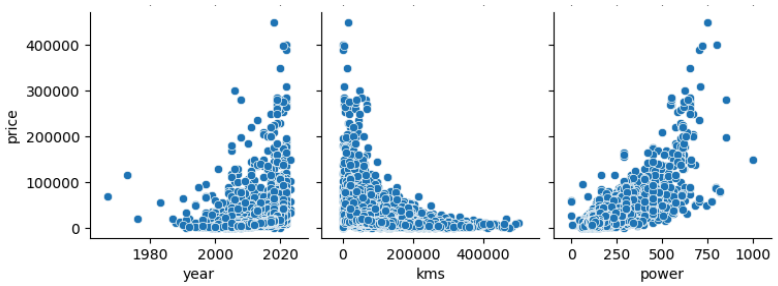
Realizamos histogramas con las variables numéricas, y observamos una distribución sesgada a la izquierda para “year” y sesgada a la derecha para “kms”, “power” y “price”.



Vemos como el “target” el precio, tiene una cola muy larga. Debido a que la mayoría de vehículos tiene un precio medio de alrededor de 20.000 euros y a partir de ahí cada vez hay menos vehículos en venta.

Creamos una matriz de correlación para observar la relación entre las variables numéricas. También representamos gráficamente la relación entre cada una de las variables numéricas con el “target” = precio. Observamos una alta correlación entre el “target” precio y “power”, “kms” y “year”.





4. PREPROCESAMIENTO DE DATOS

En primer lugar, hemos dividido el dataset en 80% para el desarrollo del proyecto y 20% para la validación final. De esta manera evitamos fuga de datos sobre todo cuando utilizamos *target encoding*.

Posteriormente,, tratamos las variables categóricas.

- “make”, “fuel”, “shift”: Directamente one hot encoding mediante get dummies. Con ello generamos 70 columnas
- “model”: 723 modelos son demasiados para realizar one hot encoding. Por lo tanto, los clasificamos en los 9 segmentos definidos por la Comisión Europea:

    - Segmento A: Minicompactos
    - Segmento B: Coches pequeños
    - Segmento C: Compactos
    - Segmento D: coches grandes
    - Segmento E: coches de prestigio
    - Segmento F: coches de lujo
    - Segmento J: SUV
    - Segmento M: Familiares grandes
    - Segmento S: Deportivos

    Generamos un segmento extra, “otros”, donde hemos agrupado los vehículos que no hemos podido clasificar en los segmentos anteriores, como por ejemplo las furgonetas. Aplicamos one hot encoding a todo ello generando 10 columnas más.

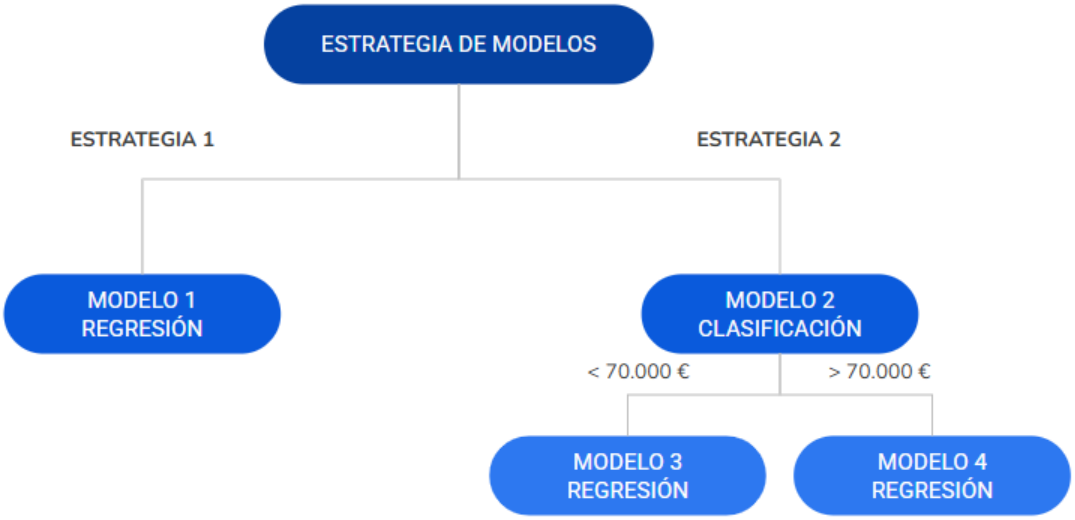
En total contamos con un dataset de 85 variables. 4 numéricas y 81 generadas mediante one hot encoding.
Los modelos utilizados son Random Forest, XGBoost, CatBoost y Gradient Boosting. Estos no son sensibles al escalado, ni necesitan transformación logarítmica en los features, pero pueden ser beneficiados con una transformación logarítmica en el target si está muy sesgado. Se han realizado diferentes pruebas y se ha observado una mejora al aplicar la transformación logarítmica al target.

Al contar con tantas variables binarias se ha estudiado utilizar las técnicas de reducción de dimensionalidad PCA (Análisis de Componentes Principales) y TruncatedSVD. Sin

embargo, no se han obtenido resultados favorables por lo que se ha descartado ya que perjudica a la interpretabilidad de los datos.

5. MODELADO

Vamos a trabajar con modelos supervisados y se van a estudiar 2 tipos de estrategias:



- Estrategia 1:
    - Se tratarán todos los datos mediante un único modelo de regresión (Modelo1).
- Estrategia 2
  :
    - Se aplica un primer modelo de clasificación (Modelo2) que clasifica los vehículos como premium (> 70.000 €) o no premium (< 70.000 €).
    - Para los vehículos no premium se aplica un modelo de regresión (Modelo3).
    - Para los vehículos premium se aplica un modelo de regresión (Modelo4).

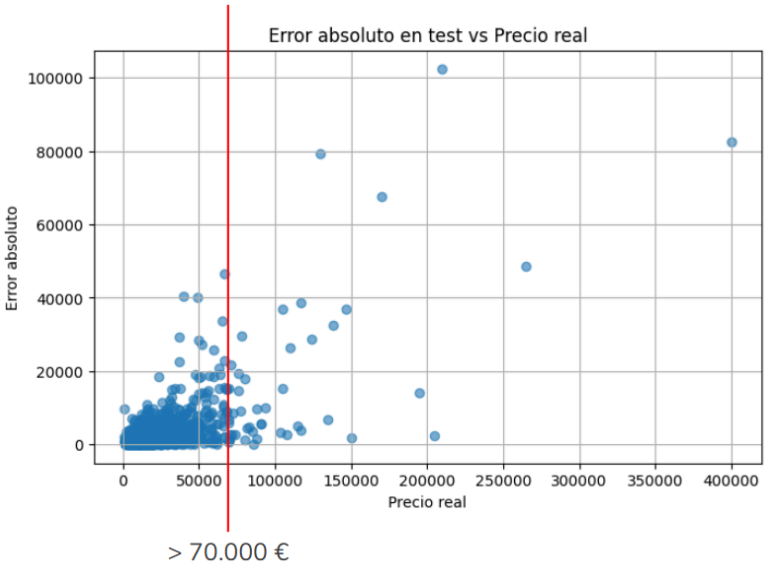
El límite de 70.000 € para diferenciar los 2 modelos de regresión se ha elegido en base a los residuos obtenidos en el entrenamiento del Modelo 1 para la estrategia 1 (ver a continuación).

De esta manera evaluaremos si mediante la estrategia 2 somos capaces de mejorar el resultado y reducir el efecto de la distribución asimétrica de nuestro “target”. Para la búsqueda del mejor modelo hemos estudiado 4 algoritmos diferentes, Random Forest, XGBoost, CatBoost y Gradient Boosting. Y nos hemos apoyado en Grid Search con cross validation para encontrar la mejor combinación de hiperparámetros.

Los modelos 1, 2, 3 y 4 se entrenan de manera independiente y se obtienen los siguientes resultados:

Estrategia 1

Modelo	Algoritmo	MAE Test	MAE Train	R^2 Test	R^2 Train
1	CatBoost	2746.74	2067.80	0.915	0.974



Estrategia 2

- Para el modelo de clasificación: No premium < 70.000 € y Premium >= 70.000 €

Matriz de Confusión

	Predicho: No premium (<70.000 €)	Predicho: Premium (≥70.000 €)
Real: No premium (<70.000 €)	1734	5
Real: Premium (≥70.000 €)	14	37

Reporte de Clasificación

Clase	Precisión	Recall	F1-score	Soporte
No premium (<70.000 €)	0.99	1.00	0.99	1739
Premium (≥70.000 €)	0.88	0.73	0.80	51

- Para los modelos de regresión:

Modelo	Algoritmo	MAE Test	MAE Train	R^2 Test	R^2 Train
3	CatBoost	2123.66	1245.45	0.921	0.977
4	Random Forest	15733.32	7854.14	0.840	0.958

6. PREDICCIÓN Y RESULTADOS FINALES

Finalmente hemos alimentado el análisis de la Estrategia de Modelos con los datos procesados en el apartado 4.

Estrategia 1

Para la estrategia 1, al contar con solo un modelo, ya tenemos el resultado en test del apartado anterior;
:

Modelo	Algoritmo	MAE Test	MAE Train	R^2 Test	R^2 Train
1	CatBoost	2746.74	2067.80	0.915	0.974

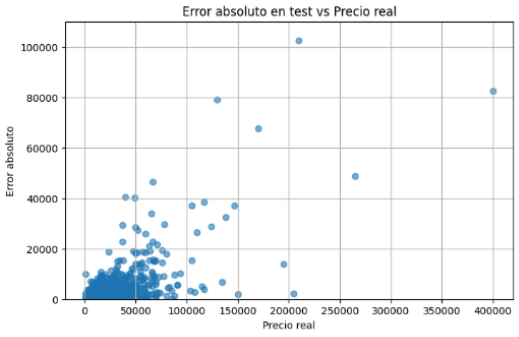
Estrategia 2

Para la estrategia 2 realizamos un pipeline en el que introducimos los datos a:

1. Modelo 2 de clasificación (premium o no premium)
2. Si los clasifica en no premium los envía al modelo de regresión 3
3. Si los clasifica como premium lo alimenta al Modelo de regresión 4.

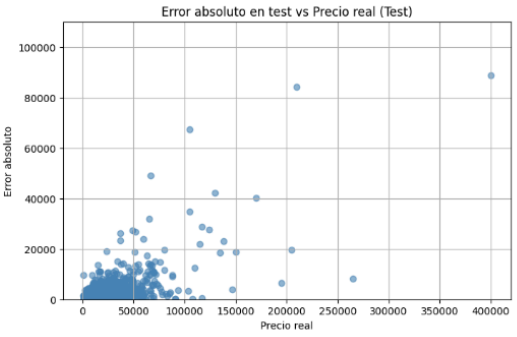
Conjunto	MAE	RMSE	R^2
Train	1524.15	3765.36	0.976
Test	2457.47	5433.45	0.937

Error absoluto en Test vs Precio Real para Estrategia 1 y 2



ESTRATEGIA 1

MAE TEST: 2746.74



ESTRATEGIA 2

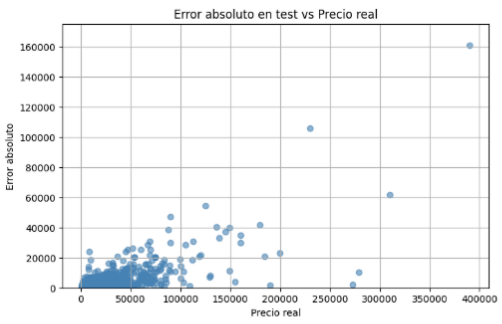
MAE TEST: 2457.47

En la estrategia 2 se consigue una leve mejora para el MAE en test respecto a la estrategia 1.

7. VALIDACIÓN

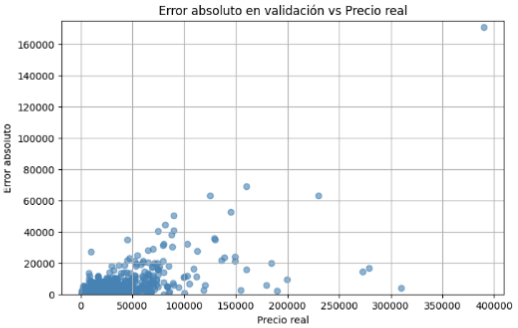
Realizamos la validación con el 20% de los datos que hemos separado al inicio del proyecto, estos datos no se han utilizado en ningún proceso hasta ahora.

Observamos que la estrategia 2 sigue sigue obteniendo mejor resultado pero la diferencia es todavía menor, siendo prácticamente inapreciable.



ESTRATEGIA 1

MAE: 2752.98



ESTRATEGIA 2

MAE: 2.691,92

8. CONCLUSIONES

Los resultados de la Estrategia 2 en la que se han realizado modelos independientes para gestionar modelos no premium y premium no han supuesto una mejora significativa en el resultado.

Posiblemente la Estrategia 2 es la correcta pero en este caso en particular se tienen muy pocos datos de vehículos premium (256 unidades) por lo que no se ha podido entrenar bien ni el modelo de clasificación (modelo2) ni el modelo de regresión para vehículos premium (modelo 4) . Por lo tanto, el resultado final es muy similar a la Estrategia 1.

De todos modos, los resultados de la estimación de precios de coches de segunda mano han sido buenos. Obteniendo un error respecto a la media del precio de venta de los vehículos de segunda mano del dataset de alrededor del 12%.

Estrategia	Conjunto	MAE (€)	% Error vs Media
Estrategia 1	Test	2.746,74	12,56 %
	Validación	2.752,98	13,01 %
Estrategia 2	Test	2.457,47	11,23 %
	Validación	2.691,92	12,72 %

9. FUTUROS PASOS

Entrenar los mejores modelos con todo el dataset

- Se puede mejorar ligeramente los resultados si los entrenamos con todos los datos.

Probar otro tipo de procesamiento de datos:

- Aplicando target encoding para reemplazar:
    - Marca: por la media del precio de todos los vehículos de la misma marca.
    - Modelo: como en el caso anterior agrupamos los modelos por segmentos y posteriormente reemplazamos el segmento por la media marca/Segmento.

- "fuel", "shift": Directamente one hot encoding mediante get dummies generando 7 columnas.

De esta manera generamos únicamente 13 variables, 6 numéricas ("year", "kms", "power" y "price", "make encoded", "segmento_make_encoded") y 7 binarias (obtenidas de fuel y shift)

Probar otros límites para la definición de la estrategia de modelos:

- Probar con otro valor de precio diferente a 70.000 €
- Clasificar la estrategia de modelos con otras métricas diferentes al precio. (se han probado métodos no supervisados como clustering sin éxito)
- Crear más de 2 estrategias para los modelos.

Conseguir más datos

- Con más datos sobre todo de los vehículos premium los resultados deberían de mejorar.
- También podría mejorar el resultado realizar una limpieza de datos más exhaustiva.