Grupo 2:
Alberico, Gabriela
Tunzi, Roberto
Lannes, Matias
Labrador Hernández, Alexis Nicolás
Silva Zuñiga, Jorge Ricardo

- **PREGUNTA 1: ¿Qué columnas has eliminado y por qué?**

Se eliminó la columna **"Marca"** porque solo contenía valores de *BMW* y tenía un 20% de nulos, sin aportar información útil.

Después no se eliminaron más columnas completas; en su lugar, se depuraron únicamente las **observaciones con datos inconsistentes** ("km" y "potencia" negativos, 1 sola fila nula, "precio" menores a 1000€ y mayores a 100,000€), siguiendo un criterio conservador para mantener la mayor cantidad de información posible (2,38% del dataset).

Tras aplicar **OHE**, se descartaron varias variables (fechas, "modelo", "tipo_coche", "color", "tipo_gasolina" y algunas características como "aire_acondicionado", "bluetooth", "gps", etc.).

- **PREGUNTA 2: ¿Cómo has manejado los valores nulos?**

Como se ha indicado en la respuesta 1, se han eliminado las filas con nulos de las variables: modelo, km, potencia, tipo_gasolina, precio, fecha_venta, volante_regulable, elevalunas_electrico, camara_trasera (menos del 3% de dataset).

Variable "color": Se obtuvo la moda de la columna "color" por cada "modelo" y se reemplazó los nulos por estas.

Variable "tipo_coche": Se obtuvo la moda de la columna "tipo_coche" por cada "modelo" y se reemplazó los nulos por estas. Para 43 observaciones nulas de "tipo_coche" que no tenían "modelo" indicado se les rellenó con la etiqueta "*no_indicado"*, ya que no tenían moda para reemplazar.

Variables con etiquetas True/False *("aire_acondicionado", "asientos_traseros_plegables", "bluetooth", "alerta_lim_velocidad"):* Se transformaron las variables en numéricas (int) siendo True = 1 y False = 0. Luego, los nulos se rellenaron con -1.

Inicialmente se valoró la eliminación de "asientos_traseros_plegables" por tener 70% de nulos, pero luego se decidió ser conservador y no eliminar la variable. Luego del OHE, estas variables se eliminaron.

Variable "fecha_registro": se siguieron los siguientes pasos:

1. Conversión a formato *datetime* de la columna *"fecha_registro"* y *"fecha_venta"*.

2. Creación de una columna nueva, *"diff_registro_venta"*, restando ambas fechas para calcular la antigüedad del coche.
3. Calculó la media de la columna *"diff_registro_venta"* según el *“modelo”*.
4. Calculó *“fecha_registro”* como (*“fecha_venta”* - media(*“diff_registro_venta"*)) según el *“modelo”*.
5. Actualizó los valores nulos para la columna *"fecha_registro"* con el cálculo anterior.
6. Borrar las filas restantes de la columna *"fecha_registro"* sin actualizar (20 filas).

- **PREGUNTA 3: Análisis univariable de las variables**

Target:
La variable *“**precio**”* presenta una distribución con **asimetría positiva**, concentrando la mayoría de los vehículos en rangos bajos y mostrando una **cola larga hacia valores altos**. Para corregir este sesgo y aproximar la normalidad, se aplicó una **transformación logarítmica**.

Variables numéricas:
Kilometraje (km): Distribución aproximadamente normal, centrada en 100,000 – 150,000 km, con pocos valores extremos. Se eliminó 1 fila con valores negativos.

Potencia (CV): Distribución sesgada a la derecha, mayoría entre 100 – 150 CV y pocos de alta potencia. Se eliminó 1 fila con potencia = 0.

Variables categóricas:
Tipo_gasolina: Valores unificados en minúsculas. Categorías poco frecuentes (hybrid_petrol, electro) agrupadas en “otros”. Dataset muy desbalanceado (96% diesel).

Color: Categorías poco frecuentes (beige, orange, red, green) agrupadas en “otros” (2.12%).

Tipo_coche: Categorías poco frecuentes (coupe, van, convertible, subcompact, no_indicado) agrupadas en “otros” (5.57%).

Modelo: Se eliminaron “Active Tourer” y “ActiveHybrid 5” por no ser modelos específicos. Se estandarizó la variable, reduciendo de 63 a 43 modelos.

- **PREGUNTA 4: ¿Existe correlación inicial entre las variables?**

Entre todas las variables existen correlaciones variadas, pero enfocando con el target están las siguientes:

potencia* vs *precio (0.69): Un BMW con más CV vale significativamente más.

tipo_coche_suv* vs *precio (0.43): Los tipos de coche SUV tienen mayor precio que los demás.

diff_registro_venta* vs *precio (-0.47): Coches más nuevos valen más.

km* vs *precio (-0.42): A menos km recorrido, mayor será el precio.

Con respecto a otras variables que no sean el target:

tipo_gasolina_otros* vs *modelo_codigo_i3 (0.82): Los BMW i3 son eléctricos/híbridos.

tipo_coche_suv* vs *modelo_codigo_X3 (0.59): Los BMW X3 son SUV.

tipo_coche_hatchback* vs *modelo_codigo_116 (0.53): Los BMW 116 son hatchbacks.

- **PREGUNTA 5: Variables vs Target (Insights adicionales)**

***Precio* vs *km*:** Tendencia negativa, a mayor kilometraje menor precio. Mayor dispersión en km bajos (corr = -0.42).

***Precio* vs *potencia*:** Relación positiva, a mayor potencia mayor precio, con más variabilidad en potencias altas (corr = 0.69).

***Log_precio* vs *fecha_registro_year*:** Correlación positiva, coches más nuevos tienen mayor precio (corr = 0.46).

***Log_precio* vs *diff_registro_venta*:** Correlación negativa, mayor antigüedad implica menor precio (corr = -0.64).

En conclusión: **potencia y año de registro aumentan el precio**, mientras que **kilometraje y antigüedad lo reducen**.

- **PREGUNTA 6: ¿Cómo vas a realizar el encoding de las variables categóricas?**

Se aplicó **One Hot Encoding** a todas las variables categóricas. En el caso de la variable ***"modelo"***, inicialmente se consideró usar **Frequency Encoding** (asignando la frecuencia de aparición, por ejemplo: *modelo 520 → 649 veces → modelo_freq=649*). Sin embargo, tras la revisión en el notebook, se optó finalmente por mantener **OHE** como método de codificación.

- **PREGUNTA 7: Escalado y correlación final**

Se aplicó **MinMaxScaler** para escalar todas las variables numéricas al rango **[0, 1]**, ya que al finalizar el pre-procesamiento todas eran de tipo numérico. **0 = valor mínimo**, **1 = valor máximo**. El método resulta **robusto frente a outliers en los extremos**. Tras la normalización, las **correlaciones mejoraron**, aportando mayor estabilidad al modelo

- **PREGUNTA 8: Pantallazo con el nombre de TODAS las columnas que tiene el DataFrame final**

```
<class 'pandas.core.frame.DataFrame'>
Index: 4727 entries, 0 to 4842
Data columns (total 77 columns):
 #   Column                                Non-Null Count  Dtype  
---  ------                                --------------  -----  
 0   km                                    4727 non-null   float64
 1   potencia                              4727 non-null   float64
 2   precio                                4727 non-null   float64
 3   diff_registro_venta                   2373 non-null   float64
 4   aire_acondicionado_number             4727 non-null   float64
 5   asientos_traseros_plegables_number    4727 non-null   float64
 6   bluetooth_number                      4727 non-null   float64
 7   alerta_lim_velocidad_number           4727 non-null   float64
 8   log_precio                            4727 non-null   float64
 9   volante_regulable_number              4727 non-null   int64  
 10  camara_trasera_number                 4727 non-null   int64  
 11  elevalunas_electrico_number           4727 non-null   int64  
 12  gps_number                            4727 non-null   int64  
 13  modelo_codigo_114                     4727 non-null   int64  
 14  modelo_codigo_116                     4727 non-null   int64  
 15  modelo_codigo_118                     4727 non-null   int64  
 16  modelo_codigo_120                     4727 non-null   int64  
 17  modelo_codigo_125                     4727 non-null   int64  
 18  modelo_codigo_135                     4727 non-null   int64  
 19  modelo_codigo_214                     4727 non-null   int64  
 20  modelo_codigo_216                     4727 non-null   int64  
 21  modelo_codigo_218                     4727 non-null   int64  
 22  modelo_codigo_220                     4727 non-null   int64  
 23  modelo_codigo_316                     4727 non-null   int64  
 24  modelo_codigo_318                     4727 non-null   int64  
 25  modelo_codigo_320                     4727 non-null   int64  
 26  modelo_codigo_325                     4727 non-null   int64  
 27  modelo_codigo_328                     4727 non-null   int64  
 28  modelo_codigo_330                     4727 non-null   int64  
 29  modelo_codigo_335                     4727 non-null   int64  
 30  modelo_codigo_418                     4727 non-null   int64  
 31  modelo_codigo_420                     4727 non-null   int64  
 32  modelo_codigo_430                     4727 non-null   int64  
 33  modelo_codigo_435                     4727 non-null   int64  
 34  modelo_codigo_518                     4727 non-null   int64  
 35  modelo_codigo_520                     4727 non-null   int64  
 36  modelo_codigo_525                     4727 non-null   int64  
 37  modelo_codigo_528                     4727 non-null   int64  
 38  modelo_codigo_530                     4727 non-null   int64  
 39  modelo_codigo_535                     4727 non-null   int64  
 40  modelo_codigo_635                     4727 non-null   int64  
 41  modelo_codigo_640                     4727 non-null   int64  
 42  modelo_codigo_730                     4727 non-null   int64  
 43  modelo_codigo_740                     4727 non-null   int64  
 44  modelo_codigo_750                     4727 non-null   int64  
 45  modelo_codigo_M13                     4727 non-null   int64  
 46  modelo_codigo_M3                      4727 non-null   int64  
 47  modelo_codigo_M4                      4727 non-null   int64  
 48  modelo_codigo_M55                     4727 non-null   int64
```

```
 49  modelo_codigo_X1                    4727 non-null   int64
 50  modelo_codigo_X3                    4727 non-null   int64
 51  modelo_codigo_X4                    4727 non-null   int64
 52  modelo_codigo_X5                    4727 non-null   int64
 53  modelo_codigo_X6                    4727 non-null   int64
 54  modelo_codigo_Z4                    4727 non-null   int64
 55  modelo_codigo_i3                    4727 non-null   int64
 56  tipo_gasolina_diesel                4727 non-null   int64
 57  tipo_gasolina_otros                 4727 non-null   int64
 58  tipo_gasolina_petrol                4727 non-null   int64
 59  color_black                         4727 non-null   int64
 60  color_blue                          4727 non-null   int64
 61  color_brown                         4727 non-null   int64
 62  color_grey                          4727 non-null   int64
 63  color_otros                         4727 non-null   int64
 64  color_silver                        4727 non-null   int64
 65  color_white                         4727 non-null   int64
 66  tipo_coche_estate                   4727 non-null   int64
 67  tipo_coche_hatchback                4727 non-null   int64
 68  tipo_coche_otros                    4727 non-null   int64
 69  tipo_coche_sedan                    4727 non-null   int64
 70  tipo_coche_suv                      4727 non-null   int64
 71  fecha_registro_Year                 4727 non-null   int32
 72  fecha_registro_Month                4727 non-null   int32
 73  fecha_registro_Weekday              4727 non-null   int32
 74  fecha_venta_Year                    4727 non-null   int32
 75  fecha_venta_Month                   4727 non-null   int32
 76  fecha_venta_Weekday                 4727 non-null   int32
dtypes: float64(9), int32(6), int64(62)
memory usage: 2.7 MB
```