

Resumen:

Este estudio tiene como objetivo identificar los factores clave que influyen en el valor de mercado de los vehículos en América Latina, con el propósito de brindar recomendaciones a una automotriz china que busca expandirse en esta región. Se realiza un análisis exhaustivo de un conjunto de datos de vehículos, considerando variables como características técnicas, condiciones del mercado y factores socioeconómicos. A través de técnicas de preprocesamiento de datos, como estandarización y reducción de dimensionalidad mediante PCA, se preparan los datos para el análisis de correlación. Utilizando la matriz de correlación de Pearson y visualizaciones como mapas de calor, se exploran las relaciones entre las variables y se identifican aquellas que tienen un mayor impacto en el precio de los vehículos. Los resultados de este estudio permitirán a la automotriz china ajustar su estrategia de precios y productos para maximizar su participación en el mercado latinoamericano.

Introducción:

El mercado automotor es un ecosistema dinámico y altamente competitivo, donde los gustos y preferencias de los consumidores varían significativamente entre regiones y culturas. En América del Norte, por ejemplo, los vehículos de gran tamaño y alto rendimiento son muy populares, mientras que en Europa se privilegia la eficiencia en el consumo de combustible. Estas diferencias culturales y socioeconómicas influyen directamente en la demanda y el valor de mercado de los vehículos.

Una importante automotriz china busca expandirse a nuestro mercado y ha solicitado nuestra experticia en ciencia de datos para comprender mejor las características de los vehículos que tienen éxito en nuestra región. El objetivo es identificar las características clave de los vehículos de gama alta y baja, con el fin de adaptar su oferta a las preferencias locales y establecer precios competitivos.

El valor de un vehículo en América Latina es un reflejo de una compleja interacción de factores culturales, económicos y sociales. Más allá de las especificaciones técnicas, elementos como el prestigio de la marca, el precio final accesible, el consumo de combustible, el tamaño y el equipamiento juegan un papel crucial. En teoría, el valor de un vehículo se determina por algunos de los siguientes factores:

1. Marca y modelo: Algunas marcas y modelos mantienen su valor mejor que otros debido a su reputación, calidad y demanda en el mercado.

2. Año de fabricación: Los autos más nuevos suelen tener un valor más alto, aunque esto puede cambiar con el tiempo debido a la depreciación.

3. Kilometraje: Menos kilómetros generalmente indican menos desgaste, lo que puede aumentar el valor del vehículo.

4. Condición del vehículo: El estado general del auto, incluyendo el interior, exterior y el funcionamiento mecánico, afecta su valor.

5. Historial de mantenimiento: Un buen historial de mantenimiento puede aumentar la confianza del comprador y, por ende, el valor.

6. Características y opciones: Equipamiento adicional, como sistemas de seguridad, tecnología avanzada y acabados de lujo, puede aumentar el valor.

7. Demanda del mercado: La oferta y la demanda en el mercado local influyen en el precio. Modelos populares pueden tener un valor más alto.

8. Ubicación geográfica: En algunas áreas, ciertos tipos de vehículos son más deseables, lo que puede afectar su precio.

9. Condiciones económicas: Factores como tasas de interés, inflación y el estado general de la economía pueden influir en el valor de los autos.

Objetivos del estudio:

General:

Identificar las características clave de los vehículos de gama alta y baja, con el fin de adaptar su oferta a las preferencias locales y establecer precios competitivos.

Específicos:

-Identificar patrones dentro de la base de datos suministrada por medio de técnicas de análisis exploratorio de datos.

-Elaborar un modelo de clasificación con aprendizaje supervisado que categorice los vehículos por gama. ~~entre gama alta y baja con el uso de la mediana de los precios como punto de corte.~~

-Implementar un modelo de regresión con aprendizaje supervisado que permita predecir el precio final de los vehículos.

Exploración y preparación de los datos.

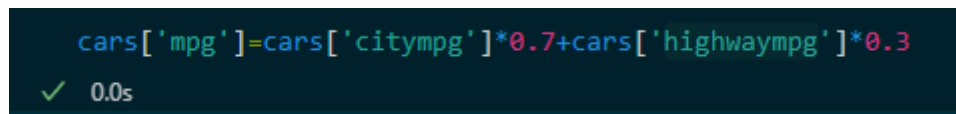
Para la elaboración del presente estudio se cuenta con una base de datos que contiene 205 observaciones y 26 atributos, es decir, una dimensión de 205 x 26.

Figura 1. Diccionario de datos.

El diccionario de datos asociado permite una mejor explicación de los atributos que se expresan en la base de datos.

Como primera medida, se verifica la integridad de la base de datos, por medio de la identificación de valores faltantes y nulos. Posteriormente, se identifican los valores únicos dentro de cada atributo y al no aportar al objetivo general y tener la base de datos otros atributos más explicativos para el caso actual, se descartan las siguientes variables: car_ID, carName, aspiration, drivewheel, enginelocation, wheelbase, enginetype, cylindernumber, fuelsystem, compressionratio.

Se resumió en una variable el consumo estimado de combustible en millas por galón, tanto en ciudad como en carretera, a través de un promedio ponderado al cual se le asignó el 70% al consumo de combustible en ciudad y un 30% al consumo en carretera. Esto dió como resultado la variable 'mpg', la cual se agregó al dataframe.



```
cars['mpg'] = cars['citympg'] * 0.7 + cars['highwaympg'] * 0.3
```

✓ 0.0s

Figura 2. Promedio ponderado de las variables 'citympg' y 'highwaympg' que genera la variable 'mpg'

Luego, se seleccionaron los atributos que contenían medidas de los vehículos y se estandarizaron los datos empleando StandardScaler para garantizar que todas las características tengan una escala similar. Después, se realiza un análisis de componentes principales (PCA) con n_components=1 para extraer la primera componente principal, que captura la mayor varianza en los datos. Esto dió como resultado el atributo 'summary measures', que resume todas las medidas de los vehículos y puede asociarse con las características estéticas del vehículo, por lo que se agregó al dataframe y se eliminaron las columnas que proveían los datos que se computaron.

Nombre de la variable	Descripción
car_ID	Número de Identificación del vehículo en la base de datos
symboling	Calificación de riesgo asociada al vehículo, +3 es riesgoso poco seguro, -3 es poco riesgoso muy seguro
CarName	Nombre de fantasía del vehículo
fueltype	Tipo de combustible
aspiration	Tipo de aspiración del motor
doornumber	Número de puertas
carbody	Tipo de carrocería del vehículo
drivewheel	Ubicación del volante del conductor
enginelocation	Ubicación del motor en el vehículo
wheelbase	Distancia entre ejes
carlength	Longitud del vehículo
carwidth	Ancho del vehículo
carheight	Altura del vehículo
curbweight	Peso del vehículo sin carga ni ocupantes
enginetype	Tipo de motor
cylindernumber	Número de cilindros del motor
enginesize	Tamaño del motor
fuelsystem	Sistema de administración de combustible del motor
boreratio	Relación diámetro/carrera de los pistones del motor
stroke	Volumen de cilindrada
compressionratio	Relación de compresión del aire dentro del motor
horsepower	Potencia del vehículo, en caballos de fuerza (HP)
peakrpm	Revoluciones máximas que soporta el motor
citympg	Consumo en ciudad, en millas por galón de combustible
highwaympg	Consumo en ruta, en millas por galón de combustible
price	Precio del vehículo

```
medidas=cars[['carlength','carwidth','carheight','curbweight']]

scaler=StandardScaler()
scaled_data=scaler.fit_transform(medidas)

pca=PCA(n_components=1)
comp_principales=pca.fit_transform(scaled_data)

componente_medidas=pd.DataFrame(data=comp_principales, columns=['PC1'])
```

Figura 3. Proceso que da como resultado la variable 'summary measures'

Para evaluar las relaciones lineales entre las variables numéricas seleccionadas (longitud, ancho, altura y peso del vehículo), se calculó la matriz de correlación de Pearson utilizando la función `.corr()` de Pandas. Esta matriz proporciona un resumen visual de las correlaciones entre cada par de variables. Posteriormente, se empleó la biblioteca Seaborn para generar un mapa de calor (heatmap) de la matriz de correlación. El mapa de calor, con la opción `annot=True`, muestra los valores numéricos de las correlaciones directamente en cada celda, y la paleta de colores `coolwarm` permite una fácil visualización de correlaciones positivas (en tonos rojos) y negativas (en tonos azules). Esta representación gráfica facilita la identificación de variables altamente correlacionadas, lo que puede ser útil para la selección de variables en modelos posteriores y para comprender las relaciones subyacentes entre las características de los vehículos.

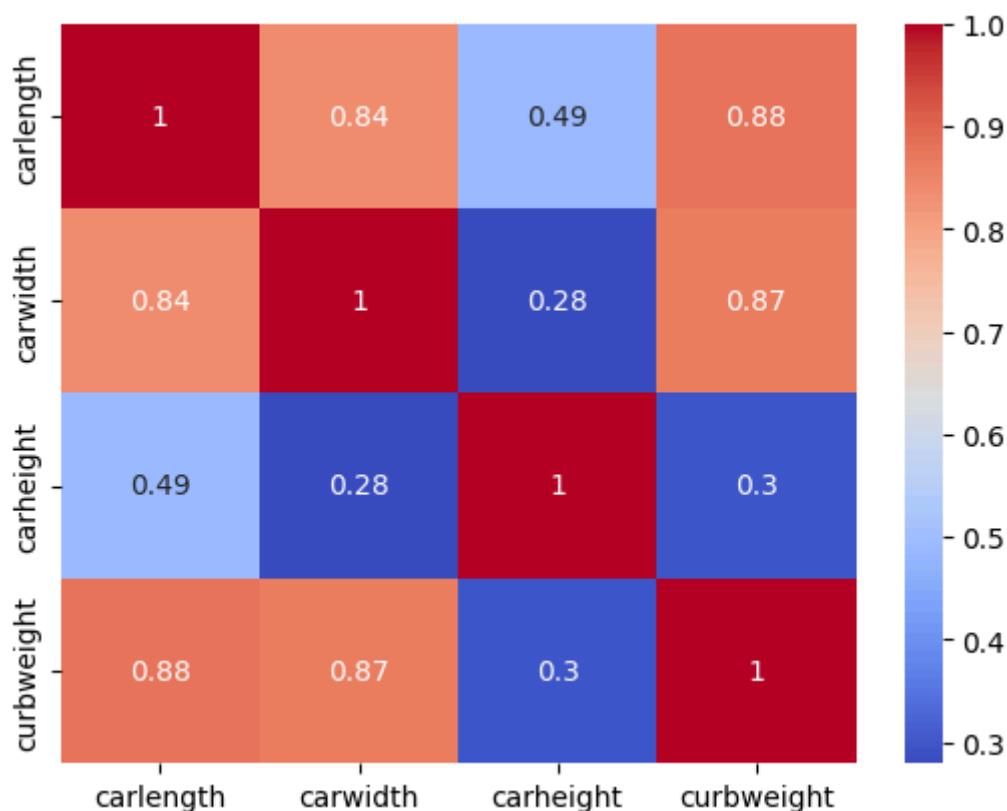


Figura 4. Representación gráfica de las variables que componen 'summary measures'

En cuanto al tipo de combustible empleado por el vehículo, se utilizó una variable dummy donde 1 indica que el vehículo emplea diesel y 0 indica que emplea gas para su funcionamiento.

```
cars=pd.get_dummies(cars, columns=['fueltype'], dtype=int)
```

Figura 5. Elaboración de la variable dummy para 'fueltype'

Para unificar la representación de la variable doornumber y facilitar su análisis cuantitativo, se procedió a normalizar los valores textuales a sus equivalentes numéricos. Se utilizaron expresiones regulares para identificar y reemplazar las cadenas 'two' y 'four' por sus correspondientes números '2' y '4', respectivamente, sin distinción de mayúsculas o minúsculas. A continuación, se convirtió la variable doornumber en tipo numérico entero, asegurando una representación consistente y adecuada para los cálculos posteriores.

```
cars=pd.get_dummies(cars, columns=['fueltype'], dtype=int)

cars.loc[cars['doornumber'].str.contains('two', case=False), 'doornumber']=cars['doornumber'].str.replace('two', '2', case=False)
```

```
cars.loc[cars['doornumber'].str.contains('four', case=False), 'doornumber']=cars['doornumber'].str.replace('four', '4', case=False)

puertas=cars['doornumber'].astype(int)

cars['doornumber']=puertas
```

Figura 6. Proceso asociado a la variable 'doornumber'

El tipo de carrocería de los automóviles es una variable cualitativa no ordinal, por lo que se le asignó un número en el mismo orden que venían los datos en la columna. A continuación, se muestran los valores asignados a cada categoría.

```
array(['convertible', 'hatchback', 'sedan', 'wagon', 'hardtop'], dtype=object)

cars['carbody'].unique()

array([0, 1, 2, 3, 4], dtype=int64)
```

Figura 7. Valores asignados a cada categoría de la variable 'carbody'

Para categorizar los vehículos como de alta gama o no, se calculó la mediana del precio de todos los automóviles en el conjunto de datos. Posteriormente, se creó una nueva variable binaria `high_end` que toma el valor 1 si el precio del vehículo es superior a la mediana, indicando que pertenece a la categoría de alta gama, y 0 en caso contrario.

```
mediana=cars['price'].median()

mediana

cars['high_end']=(cars['price']>mediana).astype(int)
```

Figura 8. Variable binaria 'high_end'

El dataframe quedó conformado por los siguientes atributos, todos descritos en variables numéricas de tipo integer y float.

```
automoviles.columns

Index(['symboling', 'doornumber', 'carbody', 'enginesize', 'boreratio',
      'stroke', 'compressionratio', 'horsepower', 'peakrpm', 'price', 'mpg',
      'fueltype', 'high_end', 'Summary_measures'],
      dtype='object')
```

Figura 9. Composición del dataframe.

Para explorar las relaciones lineales entre las variables del conjunto de datos, se calculó la matriz de correlación de Pearson utilizando la función `.corr()` de Pandas. Esta matriz ofrece una visión

general de las correlaciones entre cada par de variables. Se imprimió la matriz para su análisis detallado. Adicionalmente, se generó un mapa de calor con la biblioteca Seaborn para visualizar la matriz de manera más intuitiva. El tamaño del gráfico se estableció en 12 por 8 pulgadas para una mejor visualización. La paleta de colores coolwarm se empleó para representar las correlaciones: los tonos cálidos (rojos) indican una relación positiva fuerte y los tonos fríos (azules) señalan una relación negativa fuerte. Los valores numéricos de las correlaciones se añadieron directamente en cada celda con la opción `annot=True`. Este mapa de calor permite identificar rápidamente las variables altamente correlacionadas, lo cual es valioso para comprender las interrelaciones subyacentes entre las características de los automóviles y para la toma de decisiones en etapas posteriores del análisis.

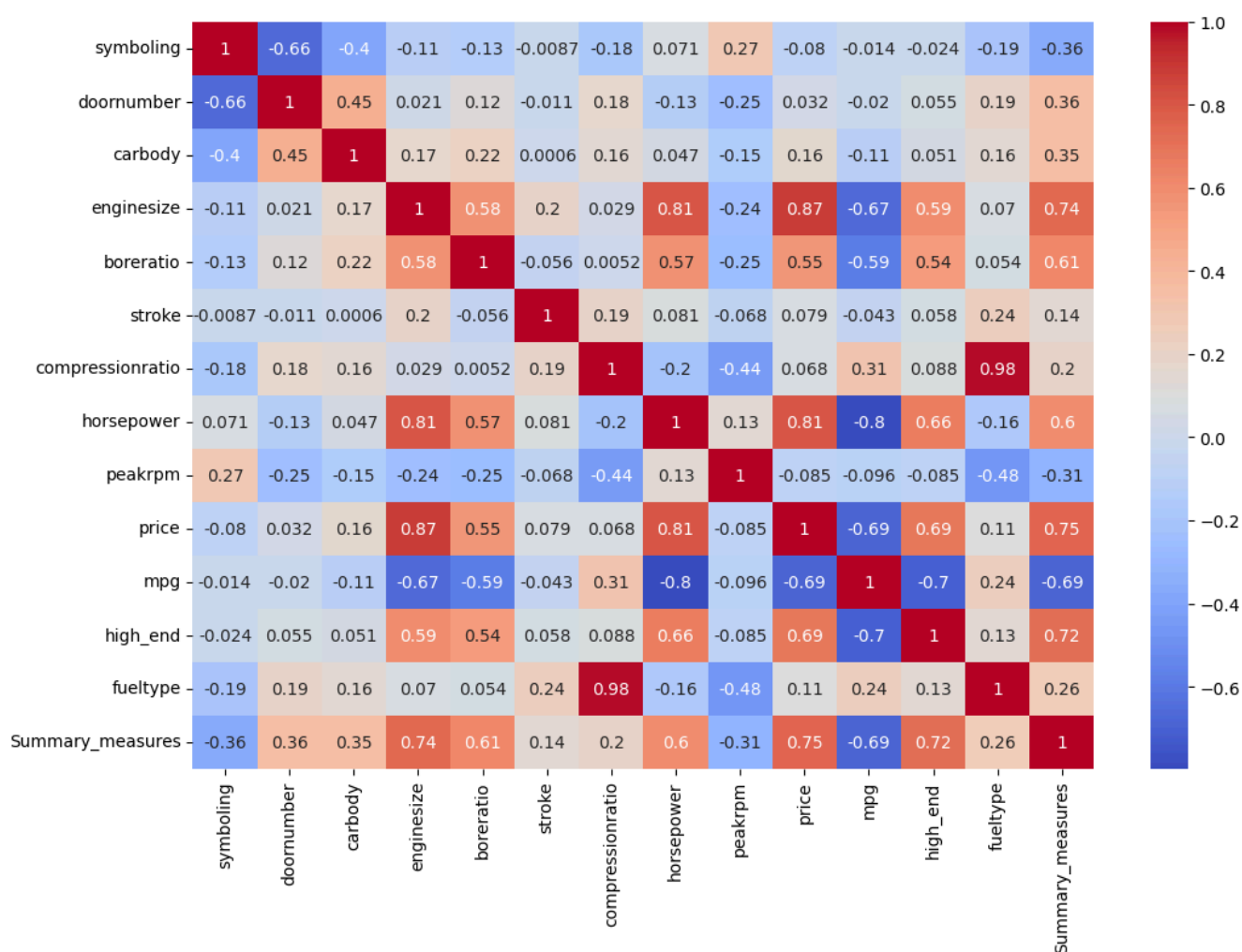


Figura 10. Matriz de correlación del dataframe.

Planteamiento de los modelos a emplear.

De acuerdo con la matriz de correlación, las variables que están más correlacionadas con el precio son las que se adjuntan a un nuevo dataset denominado 'X', como se muestra a continuación.

```
X=automoviles[['enginesize','boreratio','horsepower','mpg','high_end','Summary_measures']]
```

Figura 11. Variables incluidas en el dataset 'X'

Este dataset será tomado como el conjunto de variables independientes para explicar la variable dependiente 'price'.

$$Price = \beta_{price} + \beta_{enginesize} + \beta_{boreratio} + \beta_{horsepower} + \beta_{mpg} + \beta_{summary-measures} + \beta_{high-end} + \mu$$

donde β_{price} representa el parámetro de posición y μ el término de error de la función de regresión.

Los resultados que arrojó el modelo se muestran a continuación.

OLS Regression Results						
=====						
Dep. Variable:	price	R-squared:	0.829			
Model:	OLS	Adj. R-squared:	0.824			
Method:	Least Squares	F-statistic:	160.4			
Date:	Fri, 04 Oct 2024	Prob (F-statistic):	3.23e-73			
Time:	17:08:28	Log-Likelihood:	-1951.2			
No. Observations:	205	AIC:	3916.			
Df Residuals:	198	BIC:	3940.			
Df Model:	6					
Covariance Type:	nonrobust					
=====						
	coef	std err	t	P> t	[0.025	0.975]

const	-1878.2343	4658.689	-0.403	0.687	-1.11e+04	7308.782
enginesize	97.7152	11.964	8.167	0.000	74.121	121.309
boreratio	-1673.0982	1169.661	-1.430	0.154	-3979.691	633.495
horsepower	51.3596	13.684	3.753	0.000	24.374	78.345
mpg	61.1580	68.522	0.893	0.373	-73.968	196.284
high_end	2680.8897	762.465	3.516	0.001	1177.296	4184.483
Summary_measures	798.4413	267.258	2.988	0.003	271.403	1325.479
=====						
Omnibus:	23.256	Durbin-Watson:	0.767			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	59.708			
Skew:	0.458	Prob(JB):	1.08e-13			
...						
Notes:						
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.						

Figura 12. Resultados del modelo de regresión.

Se evidencia un coeficiente de correlación de 0.82 y sólo cuatro variables estadísticamente significativas, las cuales son:

- Enginesize, que hace referencia al tamaño del motor.
- Horsepower, que hace referencia a la potencia del vehículo expresada en caballos de fuerza.
- High_end, que hace referencia a la gama a la que pertenece el vehículo.
- Summary_measures, que hace referencia a las medidas del vehículo y está asociada a la estética del modelo específico.

Se plantea un nuevo modelo usando sólo estas cuatro variables, descartando además el parámetro de posición, el cual no resultó estadísticamente significativo.

$$Price = \beta_{enginesize} + \beta_{horsepower} + \beta_{summary-measures} + \mu$$

Los resultados del nuevo modelo son los siguientes:

OLS Regression Results						
Dep. Variable:	price	R-squared (uncentered):		0.948		
Model:	OLS	Adj. R-squared (uncentered):		0.947		
Method:	Least Squares	F-statistic:		1231.		
Date:	Sun, 06 Oct 2024	Prob (F-statistic):		1.70e-129		
Time:	10:09:41	Log-Likelihood:		-1965.3		
No. Observations:	205	AIC:		3937.		
Df Residuals:	202	BIC:		3947.		
Df Model:	3					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
enginesize	60.5878	8.950	6.770	0.000	42.940	78.235
horsepower	55.5330	10.693	5.193	0.000	34.448	76.618
Summary_measures	1643.8614	149.372	11.005	0.000	1349.334	1938.389
Omnibus:	52.273	Durbin-Watson:		0.696		
Prob(Omnibus):	0.000	Jarque-Bera (JB):		123.095		
Skew:	1.145	Prob(JB):		1.86e-27		
Kurtosis:	6.028	Cond. No.		104.		
Notes:						
[1] R ² is computed without centering (uncentered) since the model does not contain a constant.						
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.						

Figura 13. Resultados del nuevo modelo de regresión.

Se puede apreciar un aumento significativo en el coeficiente de correlación y significancia estadística en todos los parámetros de este último modelo, se procede a guardar el término de error μ .

```
residuals_model1=model1.resid
```

Figura 14. Variable que guarda el término de error μ .

Este modelo se plantea como el primer candidato para pronosticar el precio

Para la clasificación de los vehículos se plantearon dos modelos: uno logit (regresión logística) y otro probit, se observó que el dataset está correctamente balanceado.

Para entrenar el modelo se importaron los paquetes correspondientes para evaluar el reporte de clasificación y la precisión.

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report
from sklearn.metrics import accuracy_score
```

Figura 15. Paquetes correspondientes para evaluar el modelo.

Posteriormente se dividió la base de datos en un 80% para entrenamiento y el 20% restante para la evaluación de la precisión del modelo.

```

Click to add a breakpoint depend_test, Y_train, Y_test=train_test_split(independ, Y, test_size=0.2, random_state=42)

✓ 0.2s

model_reg=LinearRegression()
model_reg.fit(independ_train, Y_train)

✓ 0.3s

LinearRegression
LinearRegression()

```

Figura 16. División de la base de datos para entrenamiento y evaluación.

A continuación se muestran las métricas

```

print(f'El error cuadrático medio es: {mse} ')
print(f'El r2 es: {r2}')

El error cuadrático medio es: 14344377.442712018
El r2 es: 0.8182968477805482

```

Figura 17. Métricas asociadas al modelo.

Al dividir el dataset para entrenamiento y evaluación, se alteró el coeficiente de correlación, sin embargo, sigue siendo aceptable.

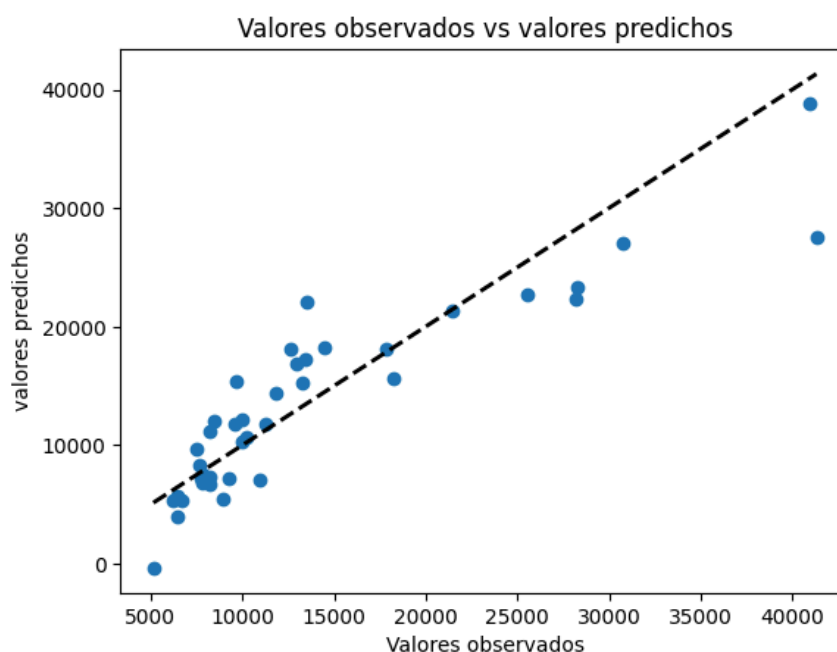


Figura 18. Discrepancia entre el precio observado y el precio pronosticado.

Para ajustar los modelos de clasificación, se plantearon cuatro modelos: una regresión logística, la cual tomó las variables que mostraron correlación significativa con la variable 'high end' referente a la gama, excluyendo el precio, dado que la gama está determinado por el mismo; un modelo probit, el

cual tomó únicamente los caballos de fuerza, el tamaño del motor y los residuos del modelo de regresión que se empleó para pronosticar el precio; posteriormente se emplearon dos modelos de árbol de clasificación que usaron las mismas variables que se tomaron tanto en la regresión logística como en el modelo probit.

El modelo probit quedó definido de la siguiente manera:

$$f(x) = \beta_{high-end} + \beta_{enginesize} + \beta_{boreratio} + \beta_{horsepower} + \beta_{mpg} + \beta_{summary-measures} + \mu$$

Donde:

$$high_end = \frac{1}{1 + e^{-f(x)}}$$

```
X_clasific_train,X_clasific_test, target_train, target_test=train_test_split(X_clasific, target, test_size=0.3, random_state=0)
```

Figura 19. División del modelo probit en datos de entrenamiento y datos de evaluación.

La precisión del modelo es de 0.90322, lo que significa que está apropiadamente ajustado, sin sobreentrenamiento y puede reflejar correctamente una progresión de datos.

```
log_reg=LogisticRegression()
log_reg.fit(X_clasific_train,target_train)
```

LogisticRegression ⓘ ?

LogisticRegression()

Se realiza el pronóstico para la evaluación del modelo

```
target_pred_log=log_reg.predict(X_clasific_test)
accuracy_log=accuracy_score(target_test, target_pred_log)
print(f'La precision (accuracy) de la regresion logistica es: {accuracy_log}')
```

La precision (accuracy) de la regresion logistica es: 0.9032258064516129

Figura 20. Precisión del modelo probit.

```

print(matriz_confusion)

[[28  2]
 [ 4 28]]

print(classification_report(target_test, target_pred_log))

```

	precision	recall	f1-score	support
0	0.88	0.93	0.90	30
1	0.93	0.88	0.90	32
accuracy			0.90	62
macro avg	0.90	0.90	0.90	62
weighted avg	0.91	0.90	0.90	62

Figura 21. Matriz de confusión y reporte de clasificación.

Para evaluar el rendimiento del modelo de regresión logística, se llevó a cabo un proceso de clasificación sobre un conjunto de datos de prueba. Inicialmente, se realizaron predicciones utilizando el modelo entrenado y se compararon con los valores reales. Posteriormente, se calculó la precisión general del modelo mediante la métrica accuracy. De forma adicional, se construyó una matriz de confusión, que generó un informe de clasificación detallado. Este último proporcionó métricas como precisión, recall y F1-score por cada clase, lo que permitió evaluar de manera más granular el desempeño del modelo y detectar posibles sesgos o desequilibrios en la clasificación.

El modelo de clasificación (probit) tiene un buen desempeño, dadas las altas métricas de precisión, recall y F1-score indican que el modelo es capaz de distinguir de manera precisa entre las dos clases. La matriz de confusión confirma esta conclusión, mostrando pocas predicciones incorrectas.

Se aplicó el modelo entrenado a nuevos datos (conjunto de prueba) para obtener predicciones. Estas predicciones se compararon con las etiquetas reales de los datos, es decir, con los valores que se buscaba predecir.

La métrica de precisión general indica la proporción total de predicciones correctas sobre el conjunto de datos de prueba. Es una medida general del rendimiento del modelo.

La matriz de confusión muestra la distribución de las predicciones correctas e incorrectas para cada clase. Permite visualizar cuántas instancias de una clase fueron clasificadas correctamente y cuántas fueron asignadas a otras clases.

El informe de clasificación proporciona métricas más específicas por clase, como la precisión, el recall y el F1-score. Estas métricas permiten evaluar el desempeño del modelo para cada clase individualmente, identificando posibles desequilibrios o dificultades en la clasificación de ciertas clases.

Árbol de decisión	Modelo logit
-------------------	--------------



Por medio de la gráfica anterior se puede observar que el modelo de árbol de decisión obtuvo mayor efectividad a la hora de clasificar los autos por gama y mostró una precisión de 0,93.

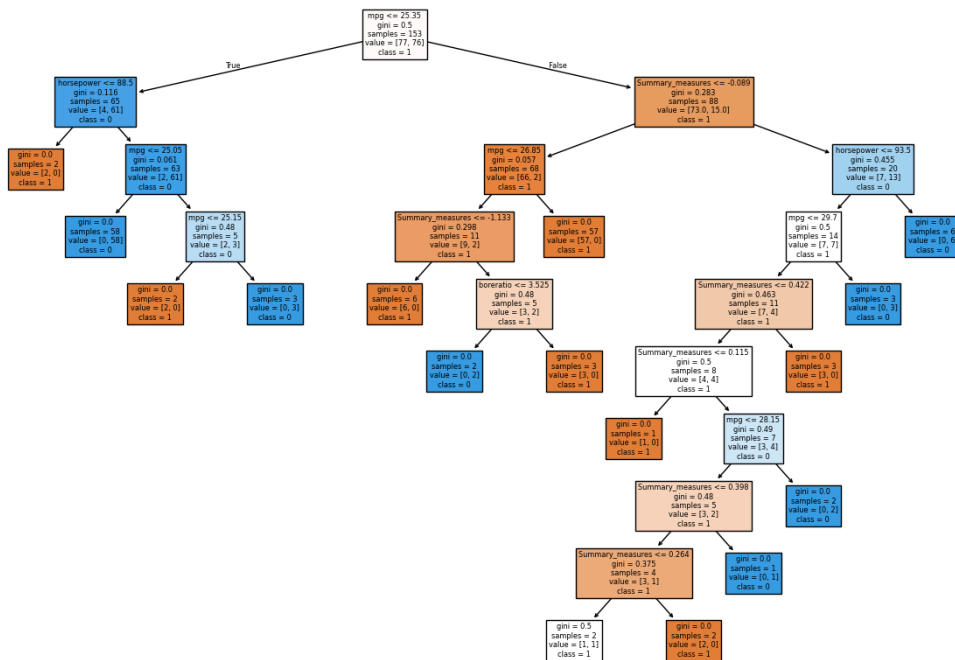


Figura 22. Árbol de decisión generado a partir del modelo anterior.

La figura resenta un árbol de decisión generado a partir del modelo anterior. Esta representación gráfica visualiza de manera clara las reglas de clasificación subyacentes al modelo. Cada nodo interno del árbol representa una prueba sobre una característica específica de los datos, y las ramas indican los posibles resultados de dicha prueba.

El planteamiento del modelo probit está expresado de la siguiente manera

$$P(\text{target} = 1 | X_{\text{probit}}) = \varphi(\beta_{\text{target}} + \beta_{\text{Summary_measures}} + \beta_{\text{horsepower}} + \beta_{\text{residuals}})$$

$$\varphi(z) = \int_{-\infty}^z \left(\frac{1}{2\pi} \right)^{1/2} + \exp\left\{ -\frac{u^2}{2} \right\} du$$

donde

es una normal estándar acumulada.

Para entrenar el modelo se dividió el dataset en unas proporciones de 75% para entrenamiento y 25% para evaluación.

```
X_probit_train,X_probit_test, target_train, target_test=train_test_split(X_probit, target, test_size=0.25, random_state=0)
```

Figura 23. Entrenamiento del nuevo modelo.

```
tree_probit_predictions=tree_model1.predict(X_probit_test)
```

```
tree1_accuracy= accuracy_score(target_test,tree_probit_predictions)
print(f'La precision (accuracy) del arbol de decision1 es: {tree1_accuracy}')
```

```
La precision (accuracy) del arbol de decision1 es: 0.9230769230769231
```

Figura 24. Precisión del modelo nuevo.

La precisión del modelo incrementó a 0.92, por lo que el reporte de clasificación es el siguiente:

```
print(probit_conf_matrix)
```

```
[[29  1]
 [ 3 29]]
```

```
print(classification_report(target_test, probit_predictions_binary))
```

	precision	recall	f1-score	support
0	0.91	0.97	0.94	30
1	0.97	0.91	0.94	32
accuracy			0.94	62
macro avg	0.94	0.94	0.94	62
weighted avg	0.94	0.94	0.94	62

Figura 25. Reporte de clasificación del nuevo modelo.

Se planteó otro árbol de decisión con las mismas variables, tomando la misma proporción para entrenamiento y evaluación que en el anterior. La precisión incrementa en comparación con el árbol anterior y se mantiene constante comparado con el modelo probit.

```

DecisionTreeClassifier
DecisionTreeClassifier()

tree_probit_predictions=tree_model1.predict(X_probit_test)

tree1_accuracy= accuracy_score(target_test,tree_probit_predictions)
print(f'La precision (accuracy) del arbol de decision1 es: {tree1_accuracy}')
```

La precision (accuracy) del arbol de decision1 es: 0.9230769230769231

Figura 26. Precisión del modelo de decisión1.

Los resultados del modelo probit y del árbol de decisión 1 se comparan en la siguiente tabla

Modelo probit	Árbol variables probit																																																												
<pre>print(probit_conf_matrix)</pre> <pre>[[29 1] [3 29]]</pre> <pre>print(classification_report(target_test, probit_predictions_binary))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>0.91</td><td>0.97</td><td>0.94</td><td>30</td></tr><tr><td>1</td><td>0.97</td><td>0.91</td><td>0.94</td><td>32</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.94</td><td>62</td></tr><tr><td>macro avg</td><td>0.94</td><td>0.94</td><td>0.94</td><td>62</td></tr><tr><td>weighted avg</td><td>0.94</td><td>0.94</td><td>0.94</td><td>62</td></tr></tbody></table>		precision	recall	f1-score	support	0	0.91	0.97	0.94	30	1	0.97	0.91	0.94	32	accuracy			0.94	62	macro avg	0.94	0.94	0.94	62	weighted avg	0.94	0.94	0.94	62	<pre>treeprobit_conf_matrix=confusion_matrix(target_test, tree_probit_predictions) print(treeprobit_conf_matrix)</pre> <pre>✓ 0.0s</pre> <pre>[[23 3] [1 25]]</pre> <pre>tree1_class_report=classification_report(target_test, tree_probit_predictions) print(tree1_class_report)</pre> <pre>✓ 0.0s</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>0.96</td><td>0.88</td><td>0.92</td><td>26</td></tr><tr><td>1</td><td>0.89</td><td>0.96</td><td>0.93</td><td>26</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.92</td><td>52</td></tr><tr><td>macro avg</td><td>0.93</td><td>0.92</td><td>0.92</td><td>52</td></tr><tr><td>weighted avg</td><td>0.93</td><td>0.92</td><td>0.92</td><td>52</td></tr></tbody></table>		precision	recall	f1-score	support	0	0.96	0.88	0.92	26	1	0.89	0.96	0.93	26	accuracy			0.92	52	macro avg	0.93	0.92	0.92	52	weighted avg	0.93	0.92	0.92	52
	precision	recall	f1-score	support																																																									
0	0.91	0.97	0.94	30																																																									
1	0.97	0.91	0.94	32																																																									
accuracy			0.94	62																																																									
macro avg	0.94	0.94	0.94	62																																																									
weighted avg	0.94	0.94	0.94	62																																																									
	precision	recall	f1-score	support																																																									
0	0.96	0.88	0.92	26																																																									
1	0.89	0.96	0.93	26																																																									
accuracy			0.92	52																																																									
macro avg	0.93	0.92	0.92	52																																																									
weighted avg	0.93	0.92	0.92	52																																																									

Conclusiones y recomendaciones:

El análisis de selección de variables reveló que el tamaño del motor, los caballos de fuerza y el resumen de las medidas (como indicador del modelo) son los principales determinantes del precio de los vehículos en el conjunto de datos. Dado que el precio influye directamente en la gama del vehículo, se omitió esta variable en los modelos para evitar problemas de multicolinealidad.

Los resultados del modelo probit, que utiliza los residuos del modelo de regresión para predecir el precio, muestran una mejora en la precisión de clasificación. Esto sugiere que los residuos capturan el efecto de variables no incluidas en el modelo original, mejorando así el poder predictivo.

Es fundamental evaluar la significancia estadística de los parámetros de un modelo de regresión para garantizar que solo las variables verdaderamente relevantes sean incluidas. Además, debido al tamaño limitado del conjunto de datos, se recomienda ampliarlo para obtener resultados más robustos y confiables.

De acuerdo con los datos analizados, el mercado local valora especialmente los caballos de fuerza, el tamaño del motor y el modelo del vehículo. Los vehículos de gama alta presentan un precio

promedio de 18,810.55 dólares, un tamaño de motor promedio de 151.42 centímetros cúbicos y una potencia promedio de 130 caballos de fuerza.

	price	enginesize	horsepower	high_end
count	102.000000	102.000000	102.000000	102.0
mean	18810.555559	151.421569	130.215686	1.0
std	8094.151207	45.785431	39.102209	0.0
min	10345.000000	70.000000	64.000000	1.0
25%	13223.750000	121.000000	101.000000	1.0
50%	16509.000000	141.000000	116.000000	1.0
75%	20652.250000	171.000000	155.750000	1.0
max	45400.000000	326.000000	288.000000	1.0

Figura 27. Análisis descriptivo de los resultados de vehículos de gama alta.

Se muestra a continuación el listado de los automóviles de gama alta con mayor demanda en el mercado local.

	Model	conteo
47	peugeot 504	6
57	saab 99gle	2
54	porsche cayenne	2
30	mazda 626	2
35	mazda rx-7 gs	2
50	peugeot 604sl	2
58	saab 99le	2
11	bmw x3	2

Figura 28. Ranking con los autos de gama alta más demandados.

Los resultados obtenidos indican que los vehículos de gama baja, dirigidos a un segmento de mercado sensible al precio, presentan características técnicas acordes a sus necesidades. El tamaño de motor de 102.63 centímetros cúbicos y la potencia de 78 caballos de fuerza sugieren vehículos compactos y eficientes, ideales para la conducción urbana y aquellos que priorizan un bajo consumo de combustible.

	price	enginesize	horsepower	high_end
count	103.000000	103.000000	103.000000	103.0
mean	7796.592233	102.631068	78.271845	0.0
std	1349.437052	14.014691	16.057003	0.0
min	5118.000000	61.000000	48.000000	0.0
25%	6790.000000	92.000000	68.000000	0.0
50%	7788.000000	98.000000	70.000000	0.0
75%	8916.500000	110.000000	88.000000	0.0
max	10295.000000	146.000000	116.000000	0.0

Figura 29. Análisis descriptivo de los resultados de vehículos de gama baja.

Se muestra a continuación el listado de los automóviles de entrada de gama con mayor demanda en el mercado local.

	Model	conteo
67	toyota corona	5
62	toyota corolla	4
56	subaru dl	3
15	honda civic	2
48	plymouth fury iii	2
35	mitsubishi outlander	2
53	subaru	2
33	mitsubishi mirage g4	2

Figura 30. Ranking con los autos de gama baja más demandados.