Cars Price Prediction

Colaboradores	Nelson Andres Quiroga Ruiz
Versión	0.0.1
Fecha de revisión	v0: 08/julio/2024

Contexto

El objetivo de esta práctica es extraer datos de una página web de ventas de automóviles mediante técnicas de web scraping y utilizar tres modelos de aprendizaje automático (*Light GBM, Random Forest Regressor y XGBoost Regressor*) para predecir el valor de los vehículos. Al final de la práctica, se compararon los modelos para determinar cuál es el más efectivo en términos de precisión y rendimiento para la tarea de predicción del valor de los vehículos.

Transformación del Dataset

Se toma como análisis la información del vehículo Mazda linea CX-30 de la página de tucarro.com, para la cual se carga la información usando la librería de pandas de python:

Load data

```
[3] cols = ['model','price','year','kms','color','fueltype']
    data = pd.read_csv('/content/drive/MyDrive/Trabajos U/Seminario BigData/cars/usedCarsCol_cx-30_200624.csv', sep=',', names=cols, header=0, encoding='latin-1')
    data.head()
```

(336, 6)

	model	price	year	kms	color	fueltype
0	Mazda CX-30 2.0 Touring At	$96.900.000	2022	28.120	Gris	Gasolina
1	Mazda CX-30 2.0 Grand Touring	$110.000.000	2022	48.622	Blanco	Gasolina
2	Mazda cx-50 2.5 Grand Touring Awd	$186.600.000	2024	0.000	Blanco	Gasolina
3	Mazda CX-30 2.0 Grand Touring At	$102.000.000	2022	32.000	Gris	Gasolina
4	Mazda Cx-30 Touring 2.0 At Hibrida 2025 /127/jl	$118.990.000	2025	0.000	Blanco	HÃbrido

Se realizan las siguientes modificaciones en la información:

➔ Se modifica el valor de las columnas kms, price

```
[9]  datacc['price'] = datacc['price'].str.replace(r'[$.]', '', regex=True).astype(int)
```

```
[10] # remove regex string, the datatype kms is float not string.
     #datacc['kms'] = datacc['kms'].str.replace(r'[.]', '',regex=True).astype(int)
     datacc['kms'] = (datacc['kms']*1000).astype(int)
```

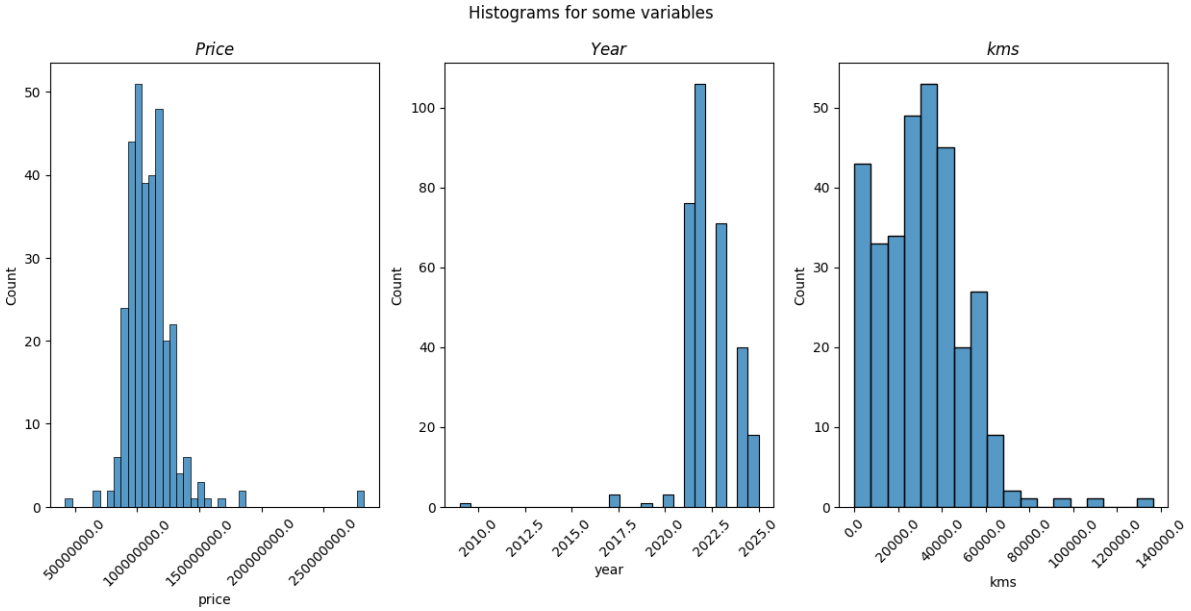
```
[13] # remove letter 'Publicado'
     datacc.replace({'kms': {'Publicado': 0}}, inplace=True)
```

```
[14] datacc['kms'] = datacc['kms'].replace('[.]','',regex=True).astype(int)
```

Análisis de la información

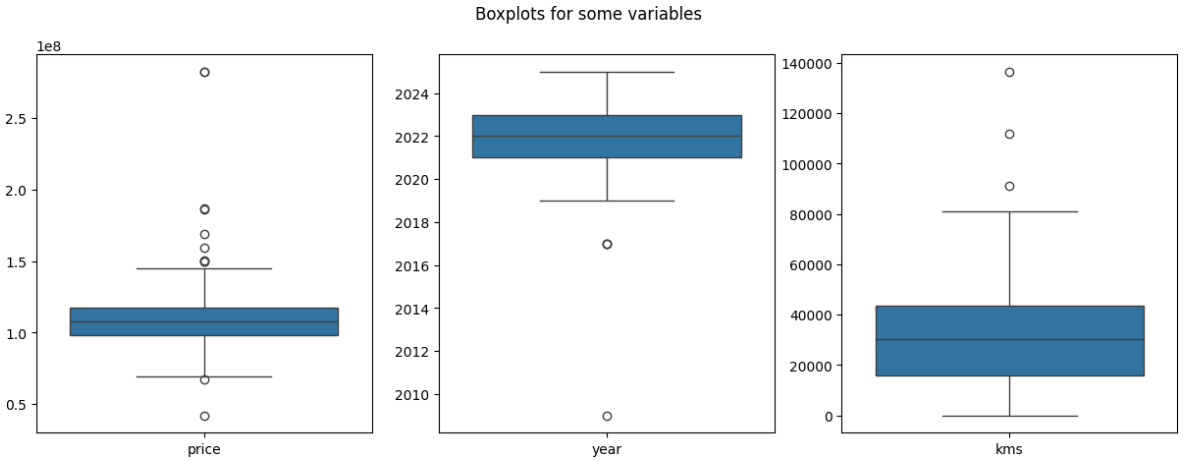
Se crea un histograma con la información, en la cual podemos deducir lo siguiente:

- ➔ **Price**: En este histograma se puede evidenciar que el valor en el cual se concentra la mayoría de la información está dentro de los $70,000,000 y los $150,000,000
- ➔ **Year**: En este histograma se concentra el valor entre 2,021 y 2,025
- ➔ **Kms**: En kilometrajes los 5,000 y los 50,000



Con el diagrama de caja podemos encontrar la desviación de los datos:

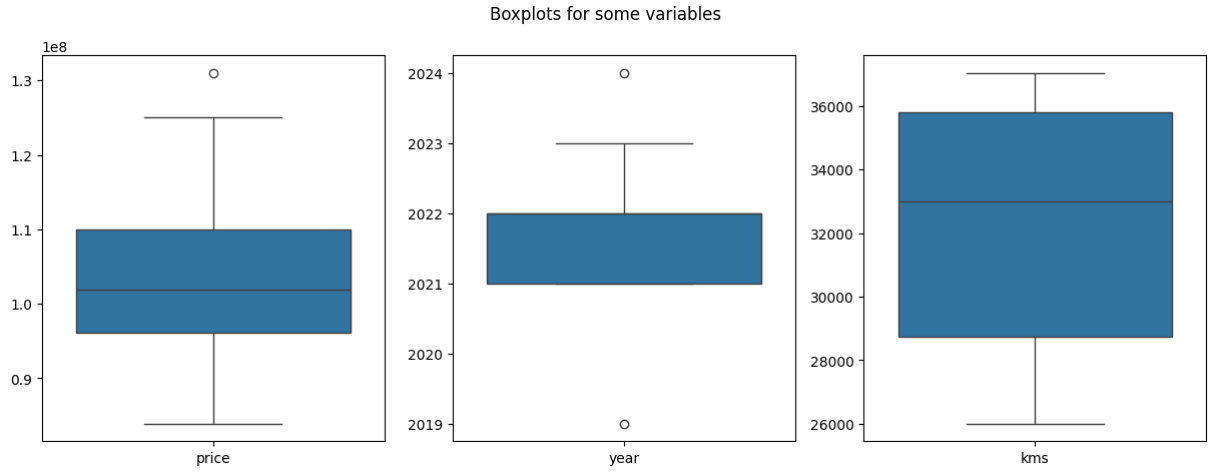
- ➔ **Price**: hay información superior a los 150 millones
- ➔ **Year**: Hay información con modelos inferiores al 2018
- ➔ **Kms**: hay información superior a los 80,000 km

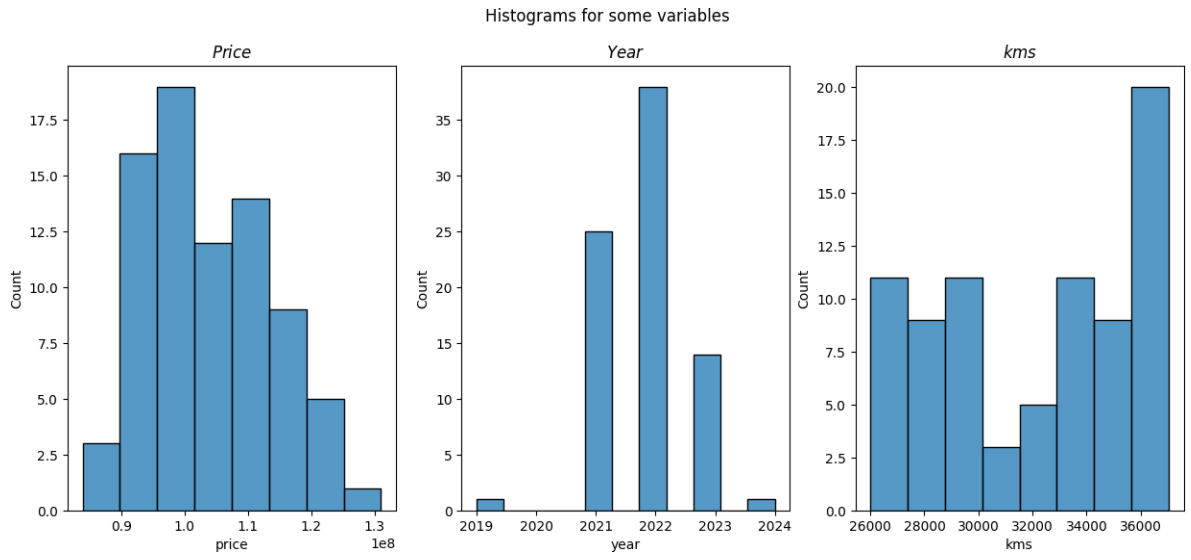


Se calcula la desviación estándar y se ajusta el dataset con la información:

```
# Z score to eliminate outliers of 'year' and 'kms'
from scipy import stats

#find absolute value of z-score for each observation of 'kms'
z2 = np.abs(stats.zscore(datacc['kms']))

#only keep rows in dataframe with all z-scores less than absolute value of 3
toremove = datacc.kms[(z2>1)].index
datacc = datacc.drop(toremove)

fig, (ax1, ax2, ax3) = plt.subplots(1, 3, figsize=(15,5))
fig.suptitle('Boxplots for some variables')
sns.boxplot(data=datacc[['price']], ax=ax1)
sns.boxplot(data=datacc[['year']], ax=ax2)
sns.boxplot(data=datacc[['kms']], ax=ax3)
```

Se observa que se mejora la información para ser procesada en los modelos de machine learning:

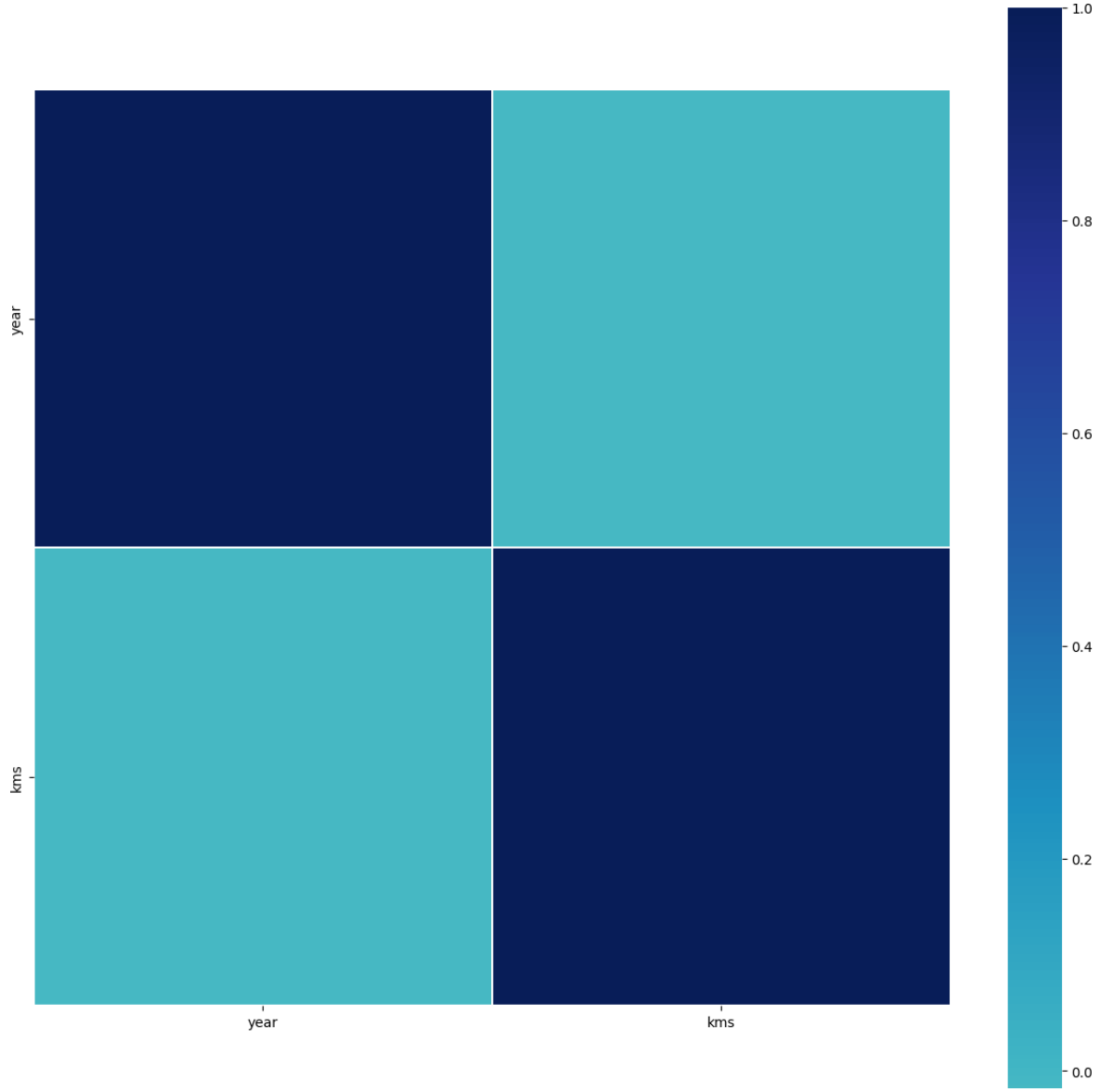




Se puede observar que la información quedó más normalizada con los siguientes rangos:

- ➔ **Price**: Toma el valor aproximado desde $90,000,000 a los $120,000,000
- ➔ **Year**: El valor se concentra entre el 2021 al 2023
- ➔ **Kms**: el valor se concentra entre los 26,000 km a los 36,000 km

Se adiciona un gráfico de correlación de variables, en donde evidenciamos que las variables [*km, year*] se correlacionan:



Machine Learning

Según referencias, Machine Learning es una disciplina del campo de la Inteligencia Artificial que, a través de algoritmos, dota a los ordenadores de la capacidad de identificar patrones en datos masivos y elaborar predicciones (análisis predictivo). Este aprendizaje permite a los computadores realizar tareas específicas de forma autónoma, es decir, sin necesidad de ser programados.

El término se utilizó por primera vez en 1959. Sin embargo, ha ganado relevancia en los últimos años debido al aumento de la capacidad de computación y al boom de los datos. Las técnicas de aprendizaje automático son, de hecho, una parte fundamental del Big Data.

Multivariate Linear Regression

La regresión lineal es un tipo de algoritmo de aprendizaje automático supervisado que calcula la relación lineal entre la variable dependiente y una o más características independientes mediante la adecuación de una ecuación lineal a los datos observados.

Cuando sólo hay una característica independiente, se conoce como Simple Regresión Lineal, y cuando hay más de una característica, se conoce como regresión lineal múltiple.

Del mismo modo, cuando sólo hay una variable dependiente, se considera regresión lineal Univariada, mientras que cuando hay más de una variable dependiente, se conoce como regresión multivariada.

Aplicación del Modelo:

Se ejecuta el modelo obteniendo las siguientes métricas:

```
[50] # accuracy check
     rmse = MSE(y_test, y_pred1, squared=False)
     mae = MAE(y_test, y_pred1)
     r2 = r2_score(y_test, y_pred1)
     print("RMSE: %.2f" % rmse)
     print("MAE: %.2f" % mae)
     print("R2: %.2f" % r2)

     RMSE: 10621371.83
     MAE: 8679832.34
     R2: 0.05
```

Light GBM

LightGBM es un método de ensamblaje de refuerzo de gradientes y se basa en árboles de decisión. Al igual que con otros métodos basados en árboles de decisión, *Light GBM* se puede utilizar tanto para la clasificación como para la regresión. LightGBM está optimizado para un alto rendimiento con sistemas distribuidos.

LightGBM crea árboles de decisión que crecen por hojas, lo que significa que, dada una condición, solo se divide una única hoja, en función de la ganancia. En ocasiones, los árboles por hojas pueden ajustarse en exceso, especialmente con datasets más pequeños. Limitar la profundidad del árbol puede ayudar a evitar el exceso de ajuste.

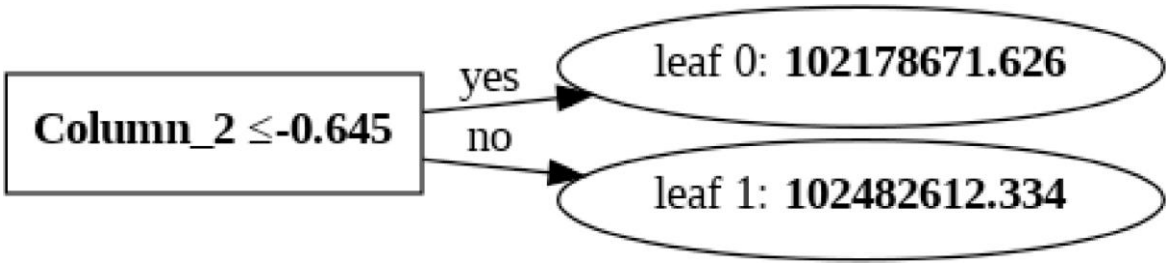
Aplicación del Modelo:

Se ejecuta el modelo obteniendo las siguientes métricas:

```
# accuracy check
rmse = MSE(y_test, y_pred2, squared=False)
mae = MAE(y_test, y_pred2)
r2 = r2_score(y_test, y_pred2)
print("RMSE: %.2f" % rmse)
print("MAE: %.2f" % mae)
print("R2: %.2f" % r2)
```

```
RMSE: 12218159.88
MAE: 9841394.69
R2: -0.26
```

Árbol de decisión:



Random Forest Regressor

Random Forest Regression es una técnica versátil de aprendizaje automático para predecir los valores numéricos. Combina las predicciones de múltiples árboles de decisión para reducir el exceso de ajuste y mejorar la precisión. También es un método de aprendizaje de conjuntos que combina las predicciones de múltiples árboles de decisión para producir una predicción más precisa y estable. Es un tipo de algoritmo de aprendizaje supervisado que se puede utilizar tanto para tareas de clasificación como de regresión.

Aplicación del Modelo:

Se ejecuta el modelo obteniendo las siguientes métricas:

```
[61] # accuracy check
     rmse = MSE(y_test, y_pred3, squared=False)
     mae = MAE(y_test, y_pred3)
     r2 = r2_score(y_test, y_pred3)
     print("RMSE: %.2f" % rmse)
     print("MAE: %.2f" % mae)
     print("R2: %.2f" % r2)

     RMSE: 13227932.89
     MAE: 10245314.58
     R2: -0.47
```

Análisis de los modelos

	Multivariate Linear Regression	**Light GBM**	**Random Forest Regressor**
RMSE	10,621,371	12,218,159	13,227,932
MAE	8,679,832	9,841,394	10,245,314
R2	0.05	-0,26	-0,47

RMSE

El RMSE es una medida de la diferencia entre los valores predichos por un modelo y los valores observados. Específicamente, es la raíz cuadrada de la media del error cuadrático medio. Un RMSE más bajo indica un mejor ajuste del modelo.

MAE

El MAE es la media de los errores absolutos entre los valores predichos y los valores observados. A diferencia del RMSE, el MAE no eleva los errores al cuadrado, lo que hace que sea menos sensible a los valores atípicos. Un MAE más bajo también indica un mejor ajuste del modelo.

R^2

El R^2 es una medida estadística que indica la proporción de la varianza en la variable dependiente que es explicada por las variables independientes en el modelo. Entre más cercano a 1 indica un mejor ajuste del modelo.

Conclusión:

Se podría interpretar que el modelo de Regresión Lineal Múltiple presenta las mejores métricas para el desarrollo del modelo en comparación con los otros métodos evaluados.
Estas métricas de rendimiento, se puede concluir que el modelo de Regresión Lineal Múltiple es el más adecuado para este conjunto de datos y objetivos específicos. No obstante, es esencial considerar otros factores como la interpretabilidad del modelo, la complejidad computacional y la aplicabilidad en el contexto práctico al momento de realizar una acción final.

Bibliografía:

https://www.geeksforgeeks.org/random-forest-regression-in-python/
https://pro.arcgis.com/es/pro-app/latest/tool-reference/geoai/how-lightgbm-works.htm
Univariate Linear Regression in Python - GeeksforGeeks