

# DATA PREPARATION

## DESKRIPSI DATASET



.....

Dataset yang digunakan adalah Car Price Prediction Challenge yang berasal dari Kaggle dan digunakan untuk membangun model machine learning dalam memprediksi harga jual mobil bekas berdasarkan berbagai fitur. Dataset berisi 19.237 sampel dengan 18 fitur, termasuk tahun produksi, jarak tempuh, jenis bahan bakar, tipe penjual, transmisi, jumlah kepemilikan, konsumsi bahan bakar, ukuran mesin (cc), dan tenaga mesin (BHP)

.....



xxx

# DATA UNDERSTANDING



× × × ×

## MELIHAT STRUKTUR DATA

Dengan df.head(), kita melihat struktur data, di mana setiap baris mewakili satu kendaraan dan setiap kolom merepresentasikan fitur tertentu. Kolom Levy mengandung tanda "-" yang diartikan sebagai nilai hilang. Untuk menangani hal ini, nilai "-" diganti dengan 0 dan dikonversi ke tipe float agar dapat digunakan dalam analisis numerik.

	ID	Price	Levy	Manufacturer	Model	Prod. year	Category	Leather interior	Fuel type	Engine volume	Mileage	Cylinders	Gear box type	Drive wheels	Doors	Wheel	Color	Airbags
0	45654403	13328	1399	LEXUS	RX 450	2010	Jeep	Yes	Hybrid	3.5	186005 km	6.0	Automatic	4x4	04-May	Left wheel	Silver	12
1	44731507	16621	1018	CHEVROLET	Equinox	2011	Jeep	No	Petrol	3	192000 km	6.0	Tiptronic	4x4	04-May	Left wheel	Black	8
2	45774419	8467	-	HONDA	FIT	2006	Hatchback	No	Petrol	1.3	200000 km	4.0	Variator	Front	04-May	Right-hand drive	Black	2
3	45769185	3607	862	FORD	Escape	2011	Jeep	Yes	Hybrid	2.5	168966 km	4.0	Automatic	4x4	04-May	Left wheel	White	0
4	45809263	11726	446	HONDA	FIT	2014	Hatchback	Yes	Petrol	1.3	91901 km	4.0	Automatic	Front	04-May	Left wheel	Silver	4

```
df['Levy'] = df['Levy'].replace('-', 0)
df['Levy'] = df['Levy'].astype(float)
```

## MENGECEK MISSING VALUE

	ID	Price	Prod. year	Cylinders	Airbags
count	1.923700e+04	1.923700e+04	19237.000000	19237.000000	19237.000000
mean	4.557654e+07	1.855593e+04	2010.912824	4.582991	6.582627
std	9.365914e+05	1.905813e+05	5.668673	1.199933	4.320168
min	2.074688e+07	1.000000e+00	1939.000000	1.000000	0.000000
25%	4.569837e+07	5.331000e+03	2009.000000	4.000000	4.000000
50%	4.577231e+07	1.317200e+04	2012.000000	4.000000	6.000000
75%	4.580204e+07	2.207500e+04	2015.000000	4.000000	12.000000
max	4.581665e+07	2.630750e+07	2020.000000	16.000000	16.000000

Pemeriksaan statistik dengan df.describe() menunjukkan bahwa dataset memiliki 19.237 entri tanpa missing values. Struktur dataset dari df.info() juga mengonfirmasi tidak adanya nilai null dalam setiap kolom, yang membuatnya siap untuk analisis lebih lanjut.

#	Column	Non-Null Count	Dtype
0	ID	19237 non-null	int64
1	Price	19237 non-null	int64
2	Levy	19237 non-null	object
3	Manufacturer	19237 non-null	object
4	Model	19237 non-null	object
5	Prod. year	19237 non-null	int64
6	Category	19237 non-null	object
7	Leather interior	19237 non-null	object
8	Fuel type	19237 non-null	object
9	Engine volume	19237 non-null	object
10	Mileage	19237 non-null	object
11	Cylinders	19237 non-null	float64
12	Gear box type	19237 non-null	object
13	Drive wheels	19237 non-null	object
14	Doors	19237 non-null	object
15	Wheel	19237 non-null	object
16	Color	19237 non-null	object
17	Airbags	19237 non-null	int64

× × ×

# DATA UNDERSTANDING



X X X X

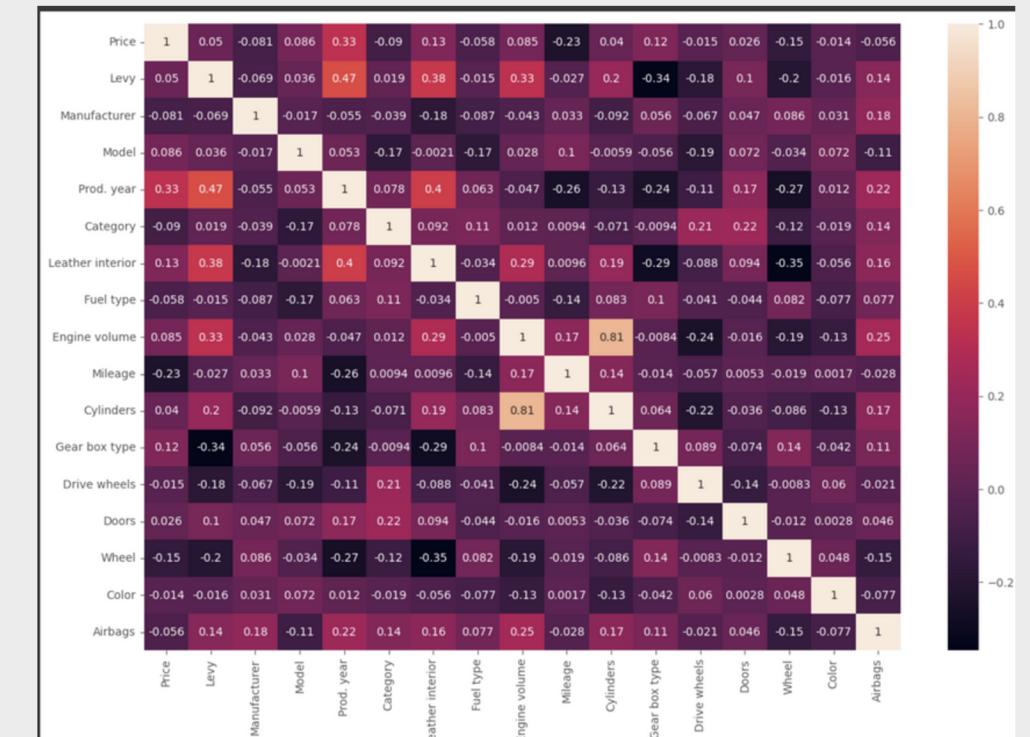
## PERIKSA DUPLIKAT DAN MENGHAPUSNYA

Dataset memiliki 313 baris duplikat, yang dapat menyebabkan bias dalam analisis dan pelatihan model. Untuk mengatasi ini, digunakan perintah `df.drop_duplicates(inplace=True)` guna menghapus data duplikat dan memastikan kualitas dataset lebih baik.

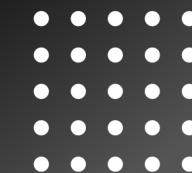
```
[ ] df.duplicated().sum()  
→ 313  
  
[ ] df.drop_duplicates(inplace=True)
```

## HEATMAP UNTUK KORELASI ANTAR FITUR

Berdasarkan heatmap korelasi di atas, kolom 'Color', 'Doors', 'Drive wheels', 'Fuel type', dan 'Airbags' memiliki korelasi yang sangat rendah terhadap harga mobil (Price).



X X X X



# DATA PREPARATION



× × × ×

## KONVERSI KOLOM KATEGORIKAL MENJADI NUMERIK

Pada tahap ini, dilakukan Label Encoding untuk mengonversi kolom kategorikal menjadi numerik, memungkinkan model Machine Learning memproses data dengan lebih baik.

```
▶ encoder = LabelEncoder()
for column in categorical_columns:
    df[column] = encoder.fit_transform(df[column])
df.sample(5)
```

	Price	Levy	Manufacturer	Model	Prod. year	Category	Leather interior	Fuel type	Engine volume	Mileage	Cylinders	Gear box type	Drive wheels	Doors	Wheel	Color	Airbags
7632	49410.3	1017.0		23	1334	2017	9	1	5	2.0	30930.0	4	0	1	1	0	14
3950	49410.3	1327.0		58	435	2017	9	1	5	2.5	6800.0	6	0	1	1	0	8
3250	314.0	0.0		5	1533	2008	4	1	5	3.0	240000.0	6	2	0	1	0	1
9427	15053.0	0.0		5	1533	2000	4	1	5	4.0	250000.0	8	2	0	1	0	4
16189	16900.0	503.0		58	262	2012	3	1	5	1.5	46756.0	4	0	1	1	0	1

## FEATURE EXTRACTION

Feature Extraction dilakukan dengan menghitung usia kendaraan (Age) berdasarkan tahun produksi (Prod. year) menggunakan rumus:

**Age = Tahun Sekarang - Prod. year**

Ini dilakukan karena Usia kendaraan lebih relevan untuk prediksi harga, mengurangi redundansi, dan meningkatkan generalisasi model karena lebih stabil dibandingkan tahun produksi.

```
[ ] now_year = dt.datetime.now().year
df['Age'] = now_year - df['Prod. year']
df.drop('Prod. year', axis=1, inplace=True)
```

# DATA PREPARATION



## CLIPPING UNTUK MENANGANI OUTLIER

Untuk menangani outlier, digunakan teknik Clipping dengan menetapkan batas bawah (persentil 5%) dan batas atas (persentil 95%) pada setiap fitur numerik. Nilai yang berada di luar batas ini akan dipotong agar tidak mempengaruhi hasil analisis dan performa model. Langkah ini penting karena outlier dapat menyebabkan model overfitting dan mempengaruhi akurasi prediksi, terutama dalam algoritma berbasis regresi. Dengan clipping, kita mempertahankan data tetapi mengurangi pengaruh nilai ekstrem, sehingga model lebih stabil dan akurat.

```
[ ] # Menentukan batas bawah dan atas berdasarkan persentil
lowerBound = df[numerical_columns].quantile(0.05) # Persentil 5%
upperBound = df[numerical_columns].quantile(0.95) # Persentil 95%

# Menerapkan clipping pada setiap kolom numerik
df[numerical_columns] = df[numerical_columns].apply(lambda x: x.clip(lower=lowerBound[x.name], upper=upperBound[x.name]))
```



## MENGHAPUS KOLOM YANG MEMPUNYAI RELASI RENDAH

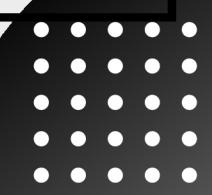
Berdasarkan heatmap korelasi, kolom 'Color', 'Doors', 'Drive wheels', 'Fuel type', dan 'Airbags' memiliki korelasi yang sangat rendah terhadap harga mobil (Price). Korelasi yang rendah menunjukkan bahwa variabel-variabel ini tidak memiliki pengaruh yang signifikan terhadap target yang ingin kita prediksi.

```
[ ] # Daftar kolom yang akan dihapus
columns_to_drop = ['Color', 'Doors', 'Drive wheels', 'Fuel type', 'Airbags']

# Menghapus kolom dari DataFrame
df.drop(columns=columns_to_drop, axis=1, inplace=True)

# Menampilkan DataFrame setelah penghapusan
df.head()
```

	Price	Levy	Manufacturer	Model	Category	Leather interior	Engine volume	Mileage	Cylinders	Gear box type	Wheel	Age
0	13328.0	1399.0		32	1242	4	1	3.5	186005.0	6	0	15
1	16621.0	1018.0		8	658	4	0	3.0	192000.0	6	2	0
2	8467.0	0.0		21	684	3	0	1.4	200000.0	4	3	19
3	3607.0	862.0		16	661	4	1	2.5	168966.0	4	0	14
4	11726.0	446.0		21	684	3	1	1.4	91901.0	4	0	11



# **TERIMA KASIH**