Link Google Colab: Link

Họ tên	MSSV
Nguyễn Võ Ngọc Bảo	23520131
Vũ Việt Cương	23520213
Ngô Phương Nam	23520974

BÁO CÁO BƯỚC 3 - TIỀN XỬ LÝ DỮ LIỆU

Môn học: Lập trình Python cho Máy học

1. **Xử lý Outlier**

Ta sẽ loại bỏ các dữ liệu ngoại lệ cho tất cả các cột số đã phân tích:

```
for col in numeric_cols:
  Q1 = train[col].quantile(0.25)
  Q3 = train[col].quantile(0.75)
  IQR = Q3 - Q1
  lower_bound = Q1 - 1.5 * IQR
  upper_bound = Q3 + 1.5 * IQR
  train = train[(train[col] >= lower_bound) & (train[col] <= upper_bound)]

train.describe()
```

Phương pháp sử dụng: Sử dụng IQR (Interquartile Range) để xác định và loại bỏ outliers cho tất cả các cột số.

Tính toán giá trị tứ phân vị:

- Q1 = quantile(0.25), Q3 = quantile(0.75)
- IQR = Q3 - Q1
-xác định khoảng hợp lệ: [Q1 - 1.5 x IQR, Q3 + 1.5 x IQR]

Loại bỏ các mẫu có giá trị nằm ngoài khoảng hợp lệ.

	year	motor_volume	price
count	833.000000	833.0	833.000000
mean	2018.013205	2.0	16142.196879
std	1.641138	0.0	2881.197031
min	2014.000000	2.0	8500.000000
25%	2017.000000	2.0	14100.000000
50%	2018.000000	2.0	15900.000000
75%	2019.000000	2.0	18000.000000
max	2022.000000	2.0	23700.000000

2. **Xử lý dữ liệu bị khuyết**

   Phương pháp:

   - Bộ dữ liệu gốc: Không chứa giá trị NaN (theo kiểm tra train.isnull().sum()).

   - Chuẩn bị cho trường hợp phát sinh là có chứa NaN thì ta sẽ sử dụng SimpleImputer từ thư viện Scikit-learn:
     + Cột số (numeric): Thay thế NaN bằng giá trị trung bình (strategy='mean').

     + Cột phân loại (category): Thay thế NaN bằng giá trị xuất hiện thường xuyên nhất (strategy='most_frequent').

   Lưu ý: Trong tập test, cột running_km có 262/411 giá trị non-null, điều này cho thấy dữ liệu test chứa NaN nhưng chưa được xử lý. Cần kiểm tra lại quy trình áp dụng Imputer cho tập test.

3. **Chiến thuật Encode**

Đầu tiên, dữ liệu trong cột ‘running’ không cùng đơn vị (mile, km). Do đó, cần chuyển dữ liệu về cùng một đơn vị là mile.

- Trước khi xử lý:

```
2         95000  miles
6         49000  miles
9         58000  miles
11          135800  km
13          220000  km
             ...
1635        180000  km
1637    120000  miles
1638        170000  km
1639     68900  miles
1640     31000  miles
Name: running, Length: 833, dtype: object
```

- Sau khi xử lý, do đã cùng một đơn vị nên bỏ phần đơn vị để trở thành dạng số.

```
Name: running, Length: 833, dtype: object
2       152887.300
6        78857.660
9        93341.720
11      135800.000
13      220000.000
            ...
1635    180000.000
1637    193120.800
1638    170000.000
1639    110883.526
1640     49889.540
Name: running_km, Length: 833, dtype: float64
```

Sử dụng label encoding để biến đổi dữ liệu dạng danh mục (phân loại):

- Với biến ‘model’:
  - + hyundai → 0
  - + kia → 1
  - + mercedes-benz → 2
  - + nissan → 3
  - + toyota → 4

	model	model_encoded
0	toyota	4
1	mercedes-benz	2
2	kia	1
3	mercedes-benz	2
4	mercedes-benz	2
...	...	...
1637	hyundai	0
1638	toyota	4
1639	nissan	3
1640	nissan	3
1641	toyota	4

- Với biến ‘motor_type’:
  + ‘gas’ → 1
  + ‘petrol’ → 2
  + ‘petrol and gas’ → 3

index	motor_type	motor_type_encoded
180	petrol	2
181	petrol	2
182	petrol	2
183	petrol and gas	3
187	petrol	2
188	petrol	2
190	gas	1

- Với biến ‘wheel’ do chỉ có một giá trị duy nhất là ‘left’ nghĩa là tất cả các xe đều có vô lăng nằm bên trái nên không có sự biến thiên trong dữ liệu → không mang thông tin hữu ích cho mô hình → sẽ được loại bỏ.
- Với biến ‘color’:
  + black → 1
  + blue → 2
  + brown → 3
  + cherry→ 4
  + clove → 5
  + golden → 6
  + gray→ 7
  + green → 8
  + orange → 9
  + other → 10
  + pink → 11
  + purple → 12
  + red → 13
  + sliver → 14
  + sky blue → 15
  + white → 16

	color	color_encoded
0	skyblue	15
1	black	1
2	other	10
4	black	1
6	gray	7
...	...	...
1636	black	1
1637	white	16
1638	black	1
1639	blue	2
1640	black	1

- Với biến ‘type’:
  + Universal → 1
  + Hatchback → 2
  + minivan / minibus → 3
  + Pickup → 4

	type	type_encoded
0	sedan	5
1	sedan	5
2	sedan	5
4	sedan	5
6	suv	6
...	...	...
1636	sedan	5
1637	sedan	5
1638	sedan	5
1639	suv	6
1640	suv	6

+ Sedan → 5
+ suv→ 6

- Với biến ‘status’:
  + crashed → 0
  + excellent → 1
  + good → 2
  + new → 3
  + normal → 4

	status	status_encoded
0	excellent	1
1	excellent	1
2	excellent	1
4	good	2
6	excellent	1
...	...	...
1636	excellent	1
1637	good	2
1638	good	2
1639	good	2
1640	excellent	1

4. Thêm đặc trưng mới

Giá trị running_per_year cao hơn cho biết xe đã được sử dụng nhiều hơn mỗi năm, điều này có thể cho thấy mức độ hao mòn cao hơn.

Giá trị running_per_year thấp hơn cho biết mức độ sử dụng ít hơn, điều này có thể cho thấy mức độ bảo dưỡng tốt hơn và có khả năng giá trị bán lại cao hơn.

→ Đặc trưng này năng này giúp nắm bắt mối quan hệ giữa tuổi của xe và mức độ sử dụng của xe, đây có thể là yếu tố quan trọng trong việc xác định giá của xe.