ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH

BÀI TẬP LỚN

Kho Dữ liệu và Hệ Hỗ trợ Quyết định

THẨM ĐỊNH GIÁ XE Ô TÔ CŨ: XÂY DỰNG MÔ HÌNH ƯỚC TÍNH SỬ DỤNG CÁC PHƯƠNG PHÁP HỒI QUY TUYẾN TÍNH

Giảng viên hướng dẫn:	Bùi Tiến Đức	
Sinh viên thực hiện:	Từ Huy Bảo	2112887
	Nguyễn Doãn Hoàng	2111238
	Nguyễn Quang Minh	2111753
	Tô Hòa	1910198
	Đồng Minh Thiện	2110555

Thành phố Hồ Chí Minh, Tháng 4 năm 2024

Mục lục

1 Giới thiệu đề tài

Theo nghiên cứu thị trường, thị trường xe ô tô cũ đạt giá trị 1.2 nghìn tỷ USD vào năm 2020 và được dự đoán sẽ vượt qua 1.5 nghìn tỷ USD vào năm 2027 với tốc độ tăng trưởng kép hàng năm ($CAGR$) khoảng 3.2%. Bởi vậy, việc tổ chức và điều tiết thị trường này là điều cần thiết. Mô hình ước tính giá xe ô tô cũ có thể giải quyết các vấn đề quan trọng cho cả người bán và người mua, chẳng hạn như hỗ trợ ra quyết định sáng suốt, phân tích thị trường, tối ưu hóa chiến lược giá, cải thiện trải nghiệm khách hàng, giảm thiểu rủi ro và phục vụ nghiên cứu phát triển.

Việc xây dựng mô hình ước tính giá xe ô tô cũ này mang lại lợi ích cho nhiều bên:

- Đối với người bán: Mô hình có thể giúp người bán định giá xe một cách chính xác và cạnh tranh, từ đó rút ngắn thời gian bán xe và tối đa hóa lợi nhuận.
- Đối với người mua: Người mua có thể sử dụng mô hình để tham khảo giá thị trường, tránh mua phải xe với giá quá cao.
- Đối với thị trường: Mô hình góp phần minh bạch hóa thị trường, tạo ra sự công bằng cho cả người mua và người bán.

Trong dự án này, chúng tôi sẽ tập trung phát triển một mô hình ước tính giá bằng cách sử dụng ba phương pháp hồi quy tuyến tính: hồi quy tuyến tính đa biến (*Multiple Linear Regression*), hồi quy Ridge (*Ridge Regression*) và hồi quy đa thức (*Polynomial Regression*). Bằng cách sử dụng cùng một bộ dữ liệu về giá xe ô tô cũ, cả ba mô hình hồi quy sẽ được xây dựng và chạy. Kết quả dự đoán của từng mô hình sẽ được so sánh với giá bán thực tế của xe. Cuối cùng, kết quả của các mô hình này sẽ được so sánh để xác định phương pháp hiệu quả nhất. Mô hình có độ chính xác cao nhất, tức là mô hình có sai số giữa giá dự đoán và giá bán thực tế thấp nhất, sẽ được coi là mô hình hiệu quả nhất.

Vậy, tại sao sử dụng ba phương pháp hồi quy tuyến tính để xây dựng mô hình ước tính?

- *Hồi quy tuyến tính đa biến*: Đây là phương pháp cơ bản nhất, sử dụng các biến độc lập (chẳng hạn như năm sản xuất, số kilomet đã đi, thương hiệu, tình trạng xe) để dự đoán biến phụ thuộc (giá bán).
- *Hồi quy Ridge*: Phương pháp này khắc phục nhược điểm của hồi quy tuyến tính đa biến trong trường hợp có quá nhiều biến độc lập (đa tạp collinearity), giúp cải thiện độ chính xác của mô hình.
- *Hồi quy đa thức*: Phương pháp này cho phép mô hình capture được các mối liên hệ không tuyến tính giữa các biến, có thể phù hợp hơn với một số trường hợp nhất định.

2 Cơ sở lý thuyết

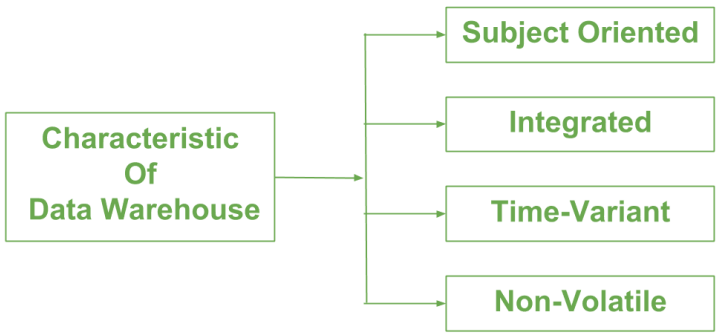
2.1 Kho dữ liệu (*Data Warehouse*)

2.1.1 Khái niệm

Kho dữ liệu là một hệ cơ sở dữ liệu khổng lồ từ vài trăm GB đến vài chục TB. Dữ liệu được tổng hợp từ đa dạng các nguồn vào một kho lưu trữ tập trung, thống nhất để hỗ trợ trong phân tích, khai phá dữ liệu, trí tuệ nhân tạo và học máy (*IBM*).

Kho dữ liệu thường được sử dụng để truy vấn, phân tích lượng lớn dữ liệu lịch sử, trích xuất thông tin có giá trị trong quá khứ nhằm cải thiện việc ra quyết định (*Oracle*).

2.1.2 Các đặc trưng của Kho dữ liệu



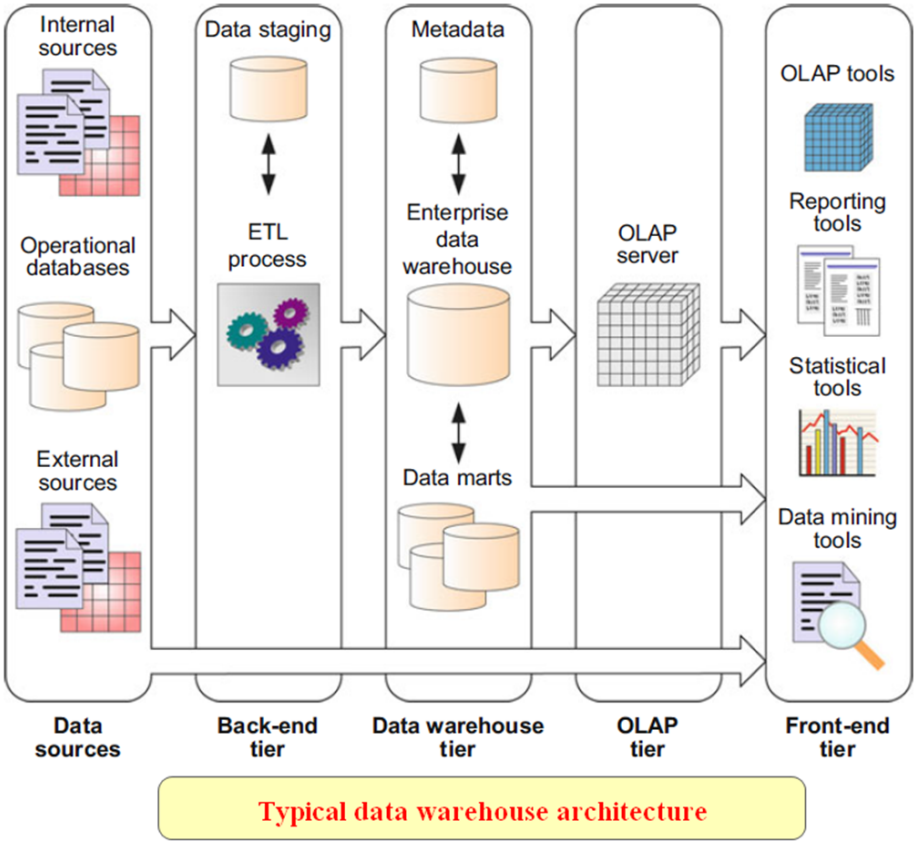
- **Theo Chủ đề (*Subject-Oriented*)**: Kho dữ liệu hoạt động theo chủ đề, cung cấp thông tin về lĩnh vực cụ thể thay vì các hoạt động hiện tại của doanh nghiệp. Nói cách khác, kho dữ liệu được dùng để xử lý tốt cho một chủ đề nhất định. Ví dụ như bán hàng, phân phối, marketing, v.v.
- **Tính tích hợp (*Integrated*)**: Kho dữ liệu tích hợp dữ liệu từ nhiều nguồn khác nhau, nhằm đảm bảo tính nhất quán và đúng đắn của dữ liệu, đồng thời duy trì tính nhất quán, giúp người dùng có thể tin tưởng vào dữ liệu và sử dụng nó để phân tích, đưa ra quyết định và tạo báo cáo.
- **Dữ liệu được gán thời gian (*Time-variant*)**: Dữ liệu trong kho dữ liệu được thu thập và lưu trữ trong khoảng thời gian dài. Kho dữ liệu lưu trữ tất cả các dữ liệu cùng sự biến động của chúng qua thời gian, không chỉ là dữ liệu ở hiện tại. Thuộc tính này cho phép phân tích xu hướng, so sánh dữ liệu theo chu kỳ thời gian, và tạo các báo cáo lịch sử.

  Ví dụ, một DW có thể lưu trữ dữ liệu về doanh thu hàng tháng của một công ty trong nhiều năm. Khi cần phân tích sự thay đổi của doanh thu theo tháng, quý, hoặc năm, thuộc tính "Time-variant" cho phép chúng ta thực hiện điều này một cách hiệu quả.
- **Không biến đổi(*Non-volatile*)**: Dữ liệu trong DW sẽ không bị thay đổi hoặc xóa bỏ, trừ khi có các hoạt động bảo trì và cập nhật. Thuộc tính này đảm bảo tính nhất quán và đáng tin cậy của dữ liệu trong DW, giúp người dùng có thể dựa vào nó để phân tích và đưa ra quyết định.

2.1.3 Kiến trúc kho dữ liệu

Một kho dữ liệu thường sẽ có cấu trúc gồm các tầng:

1. **Tầng dữ liệu đầu vào (*Back-end tier*)**: là nơi thu thập, làm sạch và chuyển đổi dữ liệu từ nhiều nguồn thông qua quy trình ETL (Trích xuất, Biến đổi, Tải).
2. **Tầng lưu trữ kho dữ liệu (*data warehouse tier*)** được cấu thành từ một kho lưu trữ dữ liệu doanh nghiệp (*enterprise data warehouse*) và/hoặc một vài kho dữ liệu nhỏ (*data marts*) cùng với một kho lưu trữ siêu dữ liệu (metadata repository) lưu trữ thông tin về kho dữ liệu.
3. **Tầng OLAP (*Online Analytical Processing - Xử lý Phân tích Trực tuyến*)** được cấu thành từ một máy chủ OLAP, cung cấp một cái nhìn đa chiều cho dữ liệu, bất kể cách thức lưu trữ dữ liệu ở hệ thống bên dưới.
4. **Tầng dữ liệu đầu ra (*front-end tier*)** được sử dụng để truy vấn, tương tác và hiển thị dữ liệu. Nó chứa các công cụ dành cho người dùng cuối như công cụ OLAP, công cụ báo cáo, công cụ thống kê và công cụ khai thác dữ liệu.



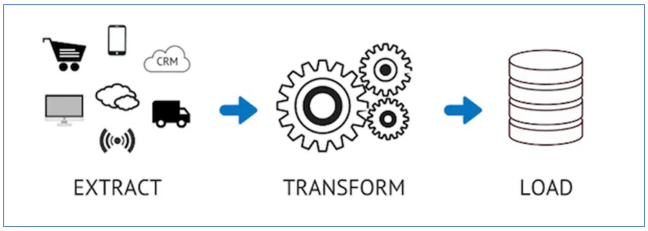
Hình 1: *Kiến trúc Kho dữ liệu thường thấy*

2.2 Một số khái niệm khác

2.2.1 ETL

ETL, viết tắt của cụm từ "Extract - Transform - Load", là một quá trình tích hợp dữ liệu kết hợp dữ liệu từ nhiều nguồn dữ liệu khác nhau vào một bộ dữ liệu đơn nhất, sau đó được tải vào kho dữ liệu hoặc một hệ thống khác.

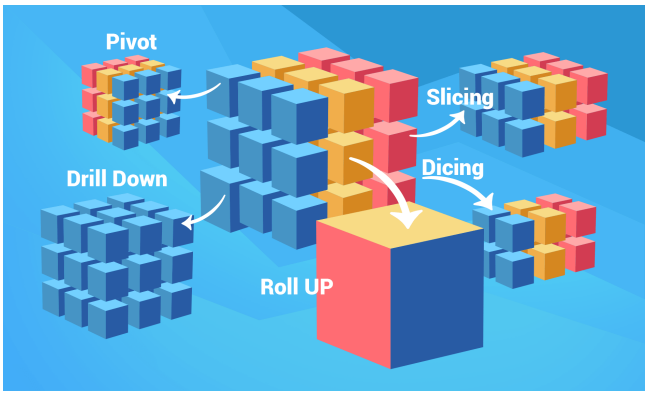
Quá trình này bắt đầu từ việc thu thập dữ liệu trích xuất từ nhiều nguồn khác nhau (*Extract*). Sau khi trích xuất, dữ liệu thường cần được biến đổi (*Transform*) thông qua các bước như lọc, sắp xếp, tính toán, nối chuỗi, xử lý điểm dữ liệu khuyết,... để đảm bảo chất lượng và thống nhất về định dạng. Cuối cùng, dữ liệu được xử lý sẽ được tải vào (*Load*) vào kho dữ liệu hoặc hệ thống đích.



ETL đóng vai trò quan trọng trong việc tích hợp dữ liệu từ nhiều nguồn vào kho dữ liệu đích; giúp chuẩn bị các tập dữ liệu thô, riêng lẻ theo cấu trúc dễ sử dụng hơn cho mục đích phân tích và nghiên cứu.

2.2.2 OLAP (*Online Analytical Processing*)

OLAP là một công nghệ phần mềm cho phép các nhà phân tích, quản lý và giám đốc điều hành hiểu rõ hơn về thông tin qua khả năng truy cập nhanh, nhất quán và có tính tương quác từ nhiều khía cạnh của dữ liệu.

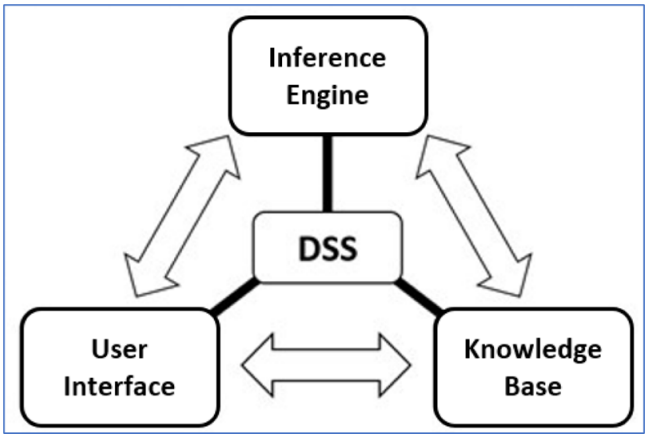


OLAP cho phép người dùng truy cập và xem dữ liệu từ nhiều góc độ (*roll-up*, *drill-down*, *slice*, *dice* và *pivot*); hỗ trợ tính toán phân tích phức tạp trên dữ liệu như phân tích biến động, so sánh dữ liệu từ các chiều khác nhau, dự đoán ngân sách hay tình hình kinh doanh trong tương lai,...

2.2.3 Hệ hỗ trợ quyết định (*Decision Support System - DSS*)

Hệ hỗ trợ quyết định là chương trình máy tính hỗ trợ tổng hợp và phân tích các tập dữ liệu lớn, được ứng dụng để đưa ra quyết định và giải quyết nhiều bài toán.

Hệ hỗ trợ quyết định có thể bao gồm nhiều mô hình, mỗi mô hình trong đó thực hiện một nhiệm vụ riêng, góp phần tối ưu hóa quyết định và đưa ra các hành động thông minh trong môi trường kinh doanh và tổ chức.



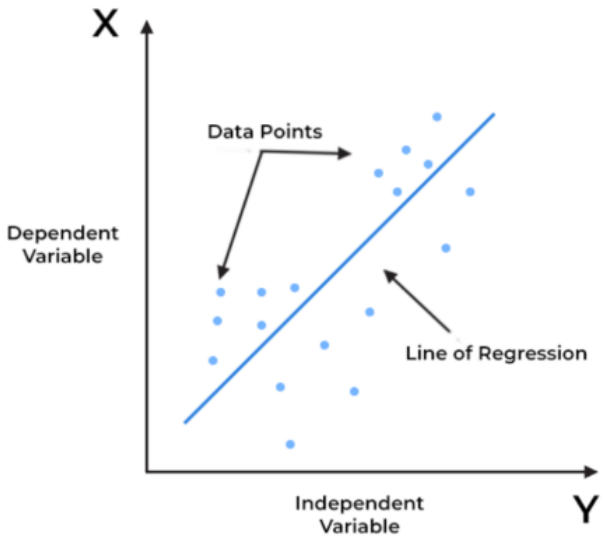
2.3 Một số phương pháp học máy

2.3.1 Linear Regression

2.3.1.1 Khái niệm

Linear Regression là một phương pháp phân tích dữ liệu dự đoán giá trị của dữ liệu không xác định bằng cách sử dụng một giá trị dữ liệu liên quan và đã biết khác. Phương pháp này mô hình hóa toán học biến không xác định hoặc phụ thuộc và biến đã biết hoặc độc lập như một phương trình tuyến tính. Ví dụ, giả sử rằng bạn có dữ liệu về chi phí và thu nhập của bạn trong năm ngoái. Kỹ thuật hồi quy tuyến tính phân tích dữ liệu này và xác định rằng chi phí của bạn là một nửa thu nhập của bạn. Sau đó, họ tính toán một chi phí trong tương lai không rõ bằng cách giảm một nửa thu nhập được biết đến trong tương lai.

Mục tiêu của linear regression là tìm ra một mối quan hệ tuyến tính giữa các biến độc lập và biến phụ thuộc.



Minh họa bài toán Linear Regression

2.3.1.2 Công thức

Linear Regression mô hình hóa mối quan hệ giữa biến đầu ra (được ký hiệu là (y)) và các biến đầu vào (được ký hiệu là $(x_1, x_2, \ldots, x_n)$).

Mối quan hệ được biểu diễn bằng một hàm tuyến tính:

$$y = w_0 + w_1x_1 + w_2x_2 + \ldots + w_nx_n \tag{1}$$

Trong đó:

- y là biến đầu ra *output*.
- $x_1, x_2, \ldots, x_n$ là các biến đầu vào *input*.
- $w_0, w_1, w_2, \ldots, w_n$ là các hệ số *weights* của mô hình.

Công thức tổng quát cho Linear Regression là:

$$y = w_0 + w_1x_1 + w_2x_2 + \ldots + w_nx_n \tag{2}$$

Trong đó:

- y là giá trị dự đoán của biến đầu ra.
- $x_1, x_2, \ldots, x_n$ là giá trị của các biến đầu vào.
- $w_0, w_1, w_2, \ldots, w_n$ là các hệ số ước lượng.

Để tìm các hệ số $w_0, w_1, w_2, \ldots, w_n$, chúng ta sử dụng dữ liệu huấn luyện (tập dữ liệu có giá trị thực của biến đầu ra). Mục tiêu là tối thiểu hóa sai số giữa giá trị dự đoán và giá trị thực tế.

Công thức tính sai số (hàm mất mát) thường là **Mean Squared Error (MSE)**:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2] \tag{3}$$

Trong đó:

- N là số lượng quan sát trong tập huấn luyện.
- y_i là giá trị thực tế của biến đầu ra.
- $\hat{y}_i$ là giá trị dự đoán của biến đầu ra.

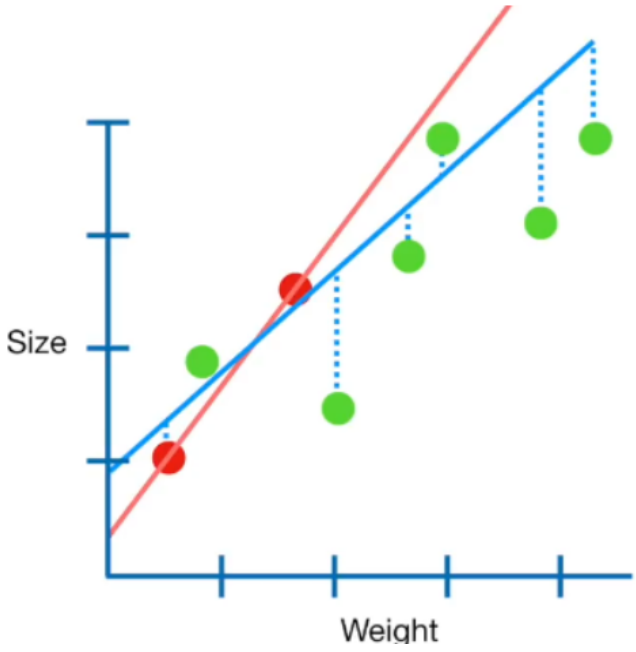
Để tìm nghiệm cho bài toán Linear Regression, chúng ta có nhiều phương pháp, bao gồm cả phương pháp dựa trên đạo hàm và phương pháp dựa trên thư viện như Scikit-learn.

2.3.2 Ridge Regression

2.3.2.1 Khái niệm

Hồi quy Ridge, còn được gọi là hồi quy Tikhonov hoặc hồi quy L2, là một phương pháp học máy được sử dụng để giảm thiểu hiện tượng quá khớp (*overfitting*) trong mô hình hồi quy tuyến tính.

Nó hoạt động bằng cách thêm một hình phạt vào hàm mất mát của mô hình, hình phạt này penalize độ lớn của các trọng số (*coefficients*).



Minh họa bài toán Ridge Regression

2.3.2.2 Công thức

Giả sử ta có mô hình hồi quy tuyến tính:

$$y = \sum(w_i * x_i) + b \tag{4}$$

Với:

- y là biến mục tiêu
- x_i là các biến dự báo
- w_i là các trọng số
- b là thuật ngữ sai số

Hàm mất mát L2 (Ridge) được định nghĩa như sau:

$$J(w, b) = \sum((y - (\sum(w_i * x_i) + b))^2) + \lambda * sum(w_i^2) \tag{5}$$

Trong đó: λ là tham số điều chuẩn, λ càng lớn, mức độ điều chuẩn càng cao và giảm *overfitting* càng mạnh.

Cách thức hoạt động:

- Giảm thiểu độ lớn của trọng số: Hình phạt L2 penalize các trọng số lớn, khiến cho mô hình có xu hướng chọn các trọng số nhỏ hơn.
- Giảm thiểu sự tương quan giữa các biến dự báo: Khi các trọng số nhỏ hơn, ảnh hưởng của từng biến dự báo lên biến mục tiêu cũng sẽ nhỏ hơn, giúp giảm thiểu sự tương quan giữa các biến dự báo và do đó giảm thiểu hiện tượng quá khớp.

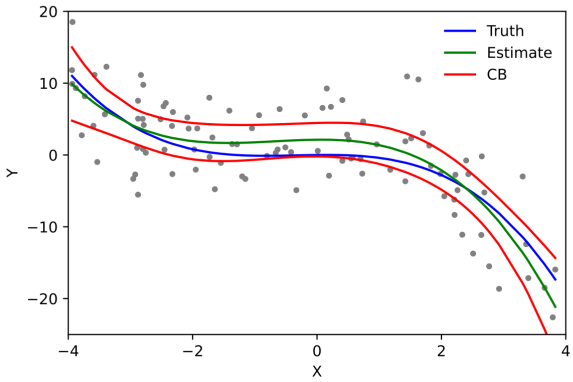
2.3.3 Polynomial Regression

Trong trường hợp các điểm không phân bố dưới dạng đường thẳng, thuật toán hồi quy tuyến tính *Linear Regression* trở nên không phù hợp, vì vậy cần phải sử dụng Hồi quy đa thức (*Polynomial Regression*) để tối ưu hơn.

2.3.3.1 Khái niệm

Hồi quy đa thức, một biến thể của hồi quy tuyến tính, là một phương pháp hồi quy mô hình hóa mối quan hệ giữa một biến độc lập với biến phụ thuộc dưới một đa thức bậc n, để tìm cách tốt nhất vẽ một đường qua các điểm dữ liệu sao cho tối ưu và phù hợp nhất.

Hồi quy đa thức đa biến (*Multivariate Polynomial Regression*), một biến thể mở rộng hơn, cho phép thể hiện mối quan hệ giữa biến phụ thuộc với nhiều biến độc lập. Nhờ đó có thể thể hiện các mối quan hệ phức tạp hơn khi có nhiều yếu tố cùng ảnh hưởng đến biến phụ thuộc.



Minh họa bài toán Polynomial Regression

2.3.3.2 Công thức

Công thức tổng quát cho Polynomial Regression có dạng:

$$y = w_0 + w_1 x + w_2 x^2 + \ldots + w_n x^n \tag{6}$$

Trong đó:

- y là giá trị dự đoán của biến phụ thuộc.
- x là biến dự báo
- $w_0, w_1, w_2, \ldots, w_n$ là các hệ số ước lượng.

Tuy Hồi quy đa thức có thể mô hình hóa được các mối quan hệ phức tạp hơn so với Linear Regression, ta cần lưu ý rằng việc tăng bậc của đa thức có thể dẫn đến *overfitting*. Do đó, việc lựa chọn bậc đa thức phù hợp là rất quan trọng.

3 Thực nghiệm

3.1 Giới thiệu tập dữ liệu

Tập dữ liệu nằm trong file *auto.csv*, bao gồm **205** bản ghi chứa thông tin của ô tô cùng với giá của chúng.

Mỗi bản ghi có các trường thuộc tính : *"symboling", "normalized-losses", "make", "fuel-type", "aspiration", "num-of-doors", "body-style", "drive-wheels", "engine-location", "wheel-base", "length", "width", "height", "curb-weight", "engine-type", "num-of-cylinders", "engine-size", "fuel-system", "bore", "stroke", "compression-ratio", "horsepower", "peak-rpm", "city-mpg", "highway-mpg", "price".*

3.2 Làm sạch và tích hợp dữ liệu

Bộ dữ liệu gốc có lượng điểm dữ liệu bị khuyết khá đáng kể.

```
miss_value = df.isnull().sum()
miss_value.loc[miss_value > 0]
```
[5] ✓ 0.0s

```
normalized-losses    37
num-of-doors          2
bore                  4
stroke                4
horsepower            2
peak-rpm              2
dtype: int64
```

Do đó, ta tiến hành thay thế các điểm dữ liệu khuyết bằng giá trị trung bình (đối với thuộc tính định lượng) và giá trị mode (giá trị xuất hiện nhiều lần nhất đối với thuộc tính định tính).

Tập dữ liệu sau khi làm sạch giá trị khuyết:

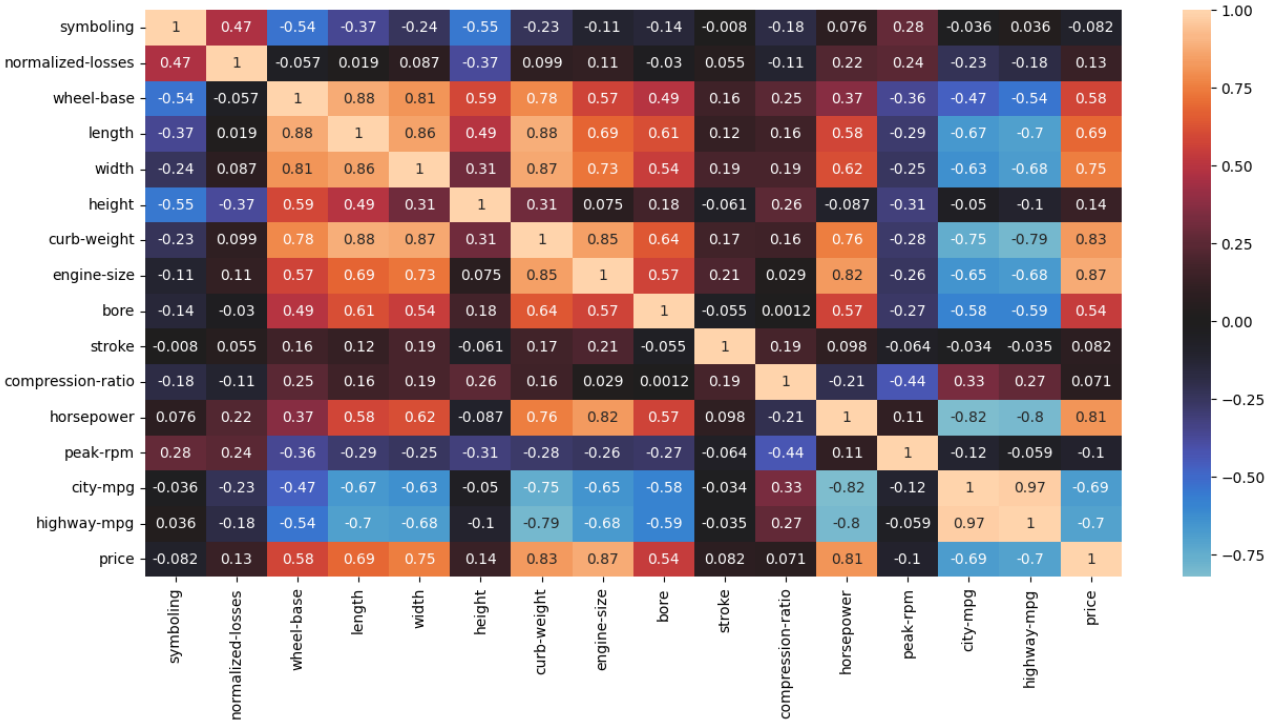
```
df.isnull().sum()
```
[10] ✓ 0.0s

```
symboling            0
normalized-losses    0
make                 0
fuel-type            0
aspiration           0
num-of-doors         0
body-style           0
drive-wheels         0
engine-location      0
wheel-base           0
length               0
width                0
height               0
curb-weight          0
engine-type          0
num-of-cylinders     0
engine-size          0
fuel-system          0
bore                 0
stroke               0
compression-ratio    0
horsepower           0
peak-rpm             0
city-mpg             0
highway-mpg          0
price                0
dtype: int64
```

3.3 Trực quan hóa dữ liệu và chọn biến độc lập

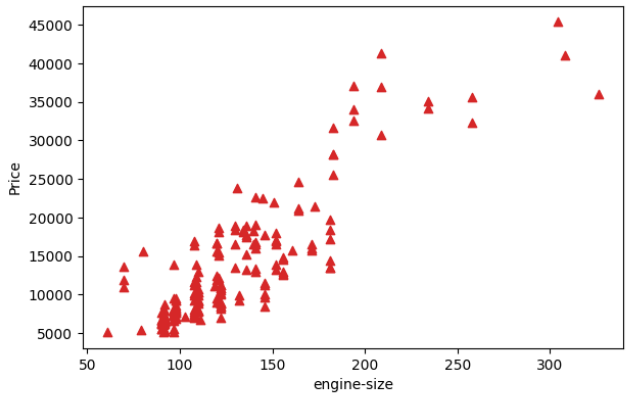
3.3.1 Thuộc tính định lượng

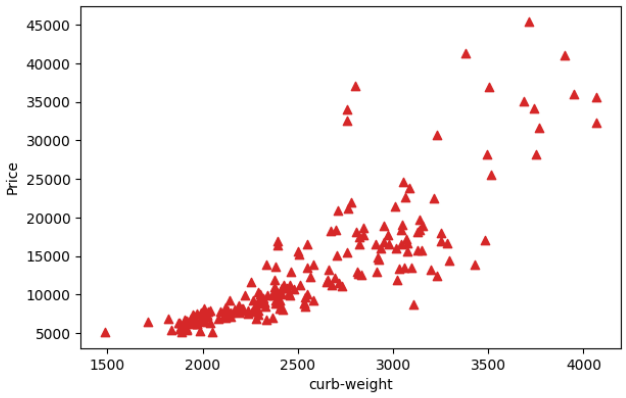
Sau khi tập dữ liệu đã được chuẩn bị sẵn sàng, vẽ **heat-map** của tập dữ liệu để nhận biết mức độ phụ thuộc của thuộc tính *"price"* vào các thuộc tính định lượng khác.

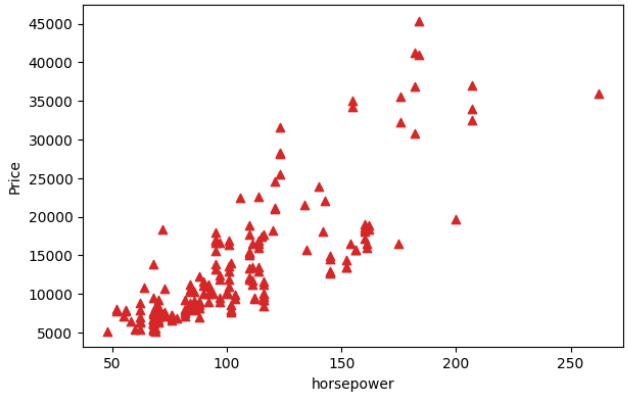


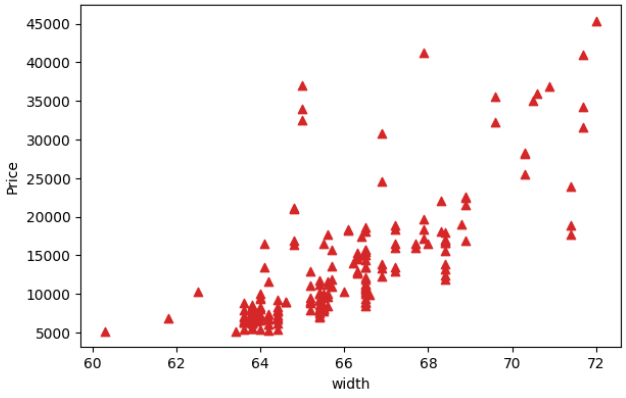
Dựa vào *heat-map* trên, chúng ta chọn ra được **7** thuộc tính mà *"price"* phụ thuộc khá cao: **"engine-size"**, **"curb-weight"**, **"horsepower"**, **"width"**, **"highway-mpg"**, **"city-mpg"** và **"length"**.

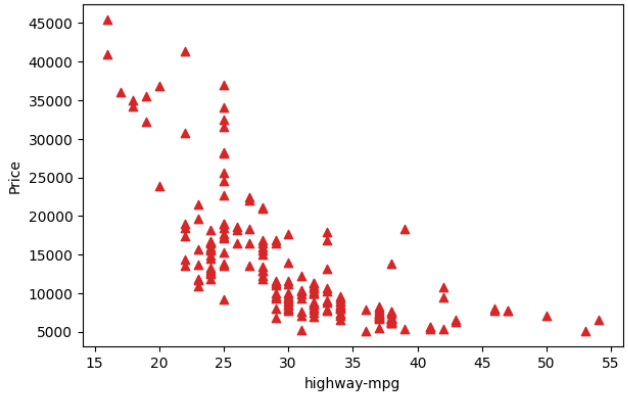
Chúng ta có thể dễ dàng nắm bắt được xu hướng đồng biến hay nghịch biến của *"price"* với từng thuộc tính thông qua hệ số tương quan (dương - đồng biến, và ngược lại) trong *heat-map*. Ngoài ra, ta cũng có thể minh họa bằng biểu đồ phân tán (*scatter plot*).

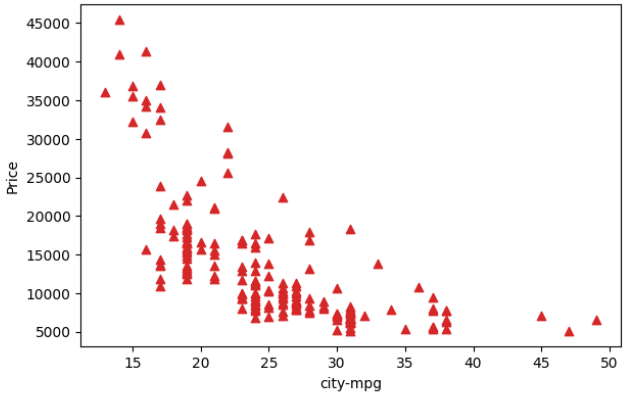


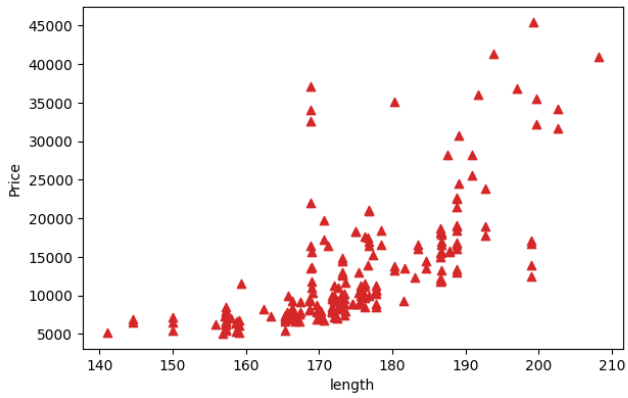






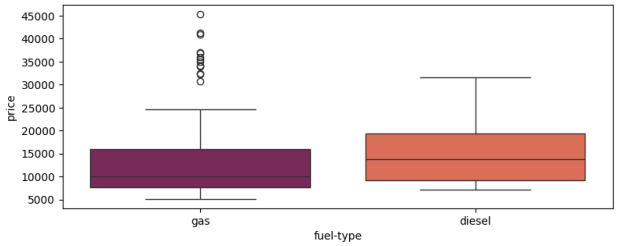


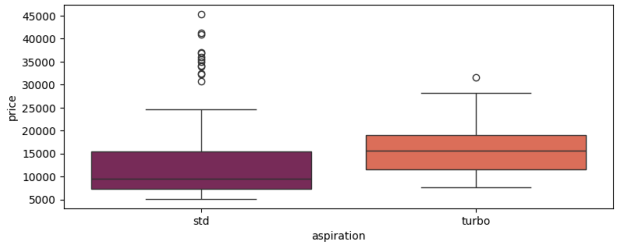


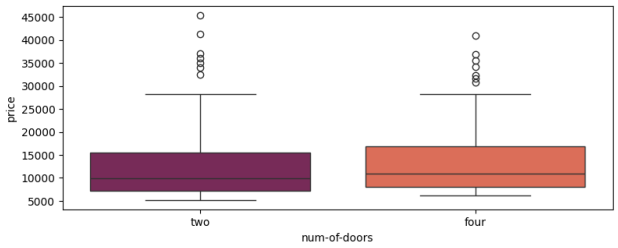


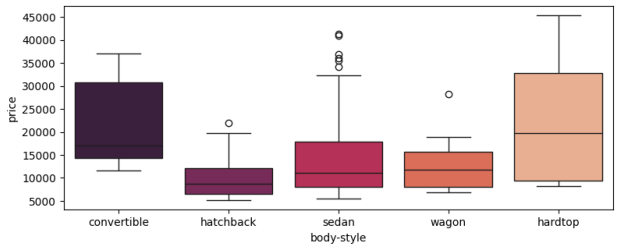
3.3.2 Thuộc tính định tính

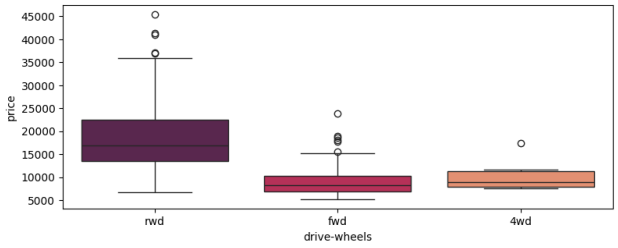
Với thuộc tính định tính, chúng ta vẽ **box-plot** để trực quan hóa mối tương quan giữa *"price"* với các thuộc tính.

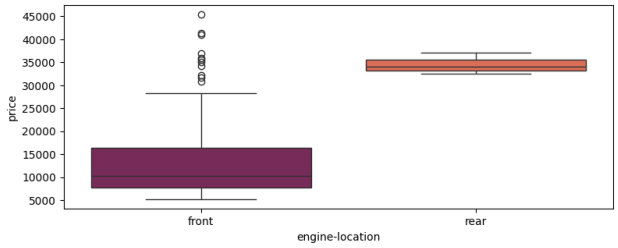


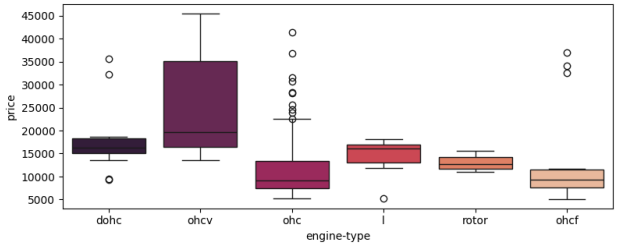


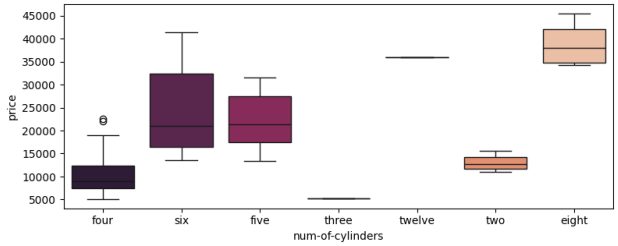


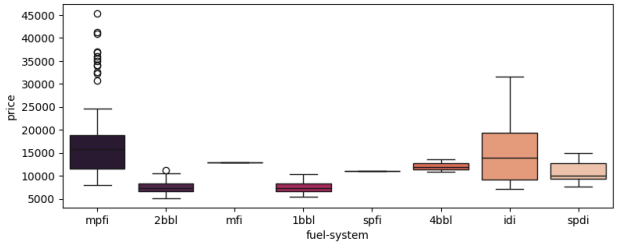


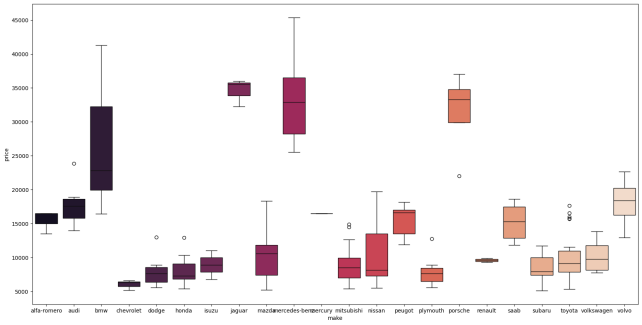












Chúng ta thấy rằng trong phân bố của *"price"* giữa các thuộc tính, thuộc tính **"engine-location"** với hai phân loại **"front"** và **"rear"**, là đủ khác biệt để xem **"engine-location** là yếu tố khả thi cho việc dự báo *"price"*. Tuy nhiên, phân bố của *"price"* đối với các thuộc tính còn lại đều có sự chồng chập đáng kể, vì vậy không thể là yếu tố khả thi để dự báo.

3.3.3 Kết quả lựa chọn biến độc lập

Sau quá trình chọn lọc, các thuộc tính thể hiện mức độ tương quan tốt đối với *"price"* được giữ lại bao gồm:

1. **Continuous numerical variables:**
   - "engine-size"
   - "curb-weight"
   - "horsepower"
   - "width"
   - "highway-mpg"
   - "city-mpg"
   - "length"
2. **Categorical Variables:** "engine-location" (Sẽ được chuyển đổi thành đại lượng định tính để tiến hành xây dựng mô hình học máy)

Việc cung cấp cho mô hình các biến ảnh hưởng có ý nghĩa đến biến mục tiêu của chúng ta sẽ cải thiện hiệu suất dự đoán của mô hình khi xây dựng các mô hình học máy.

3.4 Xây dựng và kiểm thử mô hình

Từ tập dữ liệu đã được chọn lọc:

	engine-size	curb-weight	horsepower	width	highway-mpg	city-mpg	length	engine-location-front	engine-location-rear
0	130	2548	111.0	64.1	27	21	168.8	True	False
1	130	2548	111.0	64.1	27	21	168.8	True	False
2	152	2823	154.0	65.5	26	19	171.2	True	False

Chúng ta tiếp tục đồng bộ hóa khoảng giá trị của từng thuộc tính:

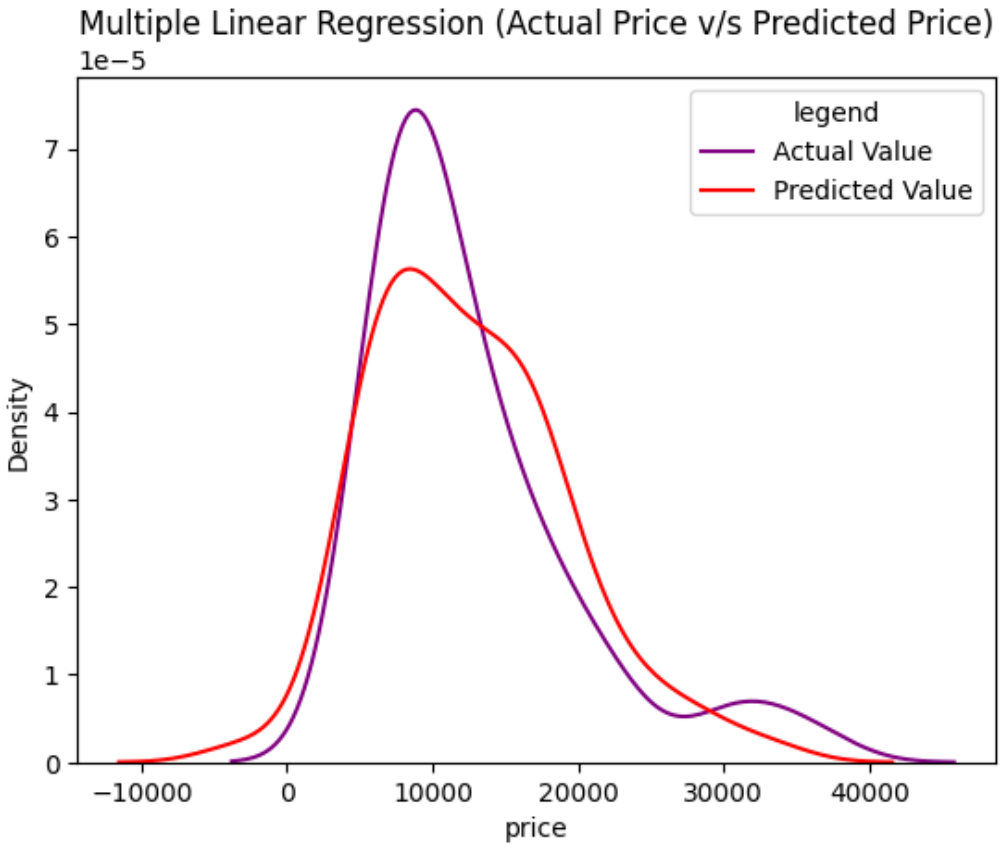
	engine-size	curb-weight	horsepower	width	highway-mpg	city-mpg	length	engine-location-front	engine-location-rear
0	0.075389	-0.014858	0.203984	-0.853460	-0.542288	-0.652249	-0.439409	0.123091	-0.123091
1	0.075389	-0.014858	0.203984	-0.853460	-0.542288	-0.652249	-0.439409	0.123091	-0.123091
2	0.606234	0.518080	1.357649	-0.185597	-0.689386	-0.964397	-0.244152	0.123091	-0.123091
3	-0.431327	-0.423766	-0.037480	0.148335	-0.100993	-0.184027	0.195176	0.123091	-0.123091
4	0.220165	0.520017	0.311302	0.243744	-1.277779	-1.120471	0.195176	0.123091	-0.123091

Sau đó, ta chia tập dữ liệu với tỉ lệ 6/4 cho tập huấn luyện (*train split*) và tập kiểm thử (*test split*):

```
shape of x_train (120, 9)
shape of y_train (120,)
shape of x_test (81, 9)
shape of y_test (81,)
```

Các mô hình sẽ được huấn luyện với *train split* để cho ra các giá trị ứng với các biến ở tập *test split*. Kết quả dự báo sẽ được so sánh với giá trị thực tế của *"price"* trong tập *test split*.

- Mô hình hồi quy tuyến tính - Linear Regression Model



Trung bình bình phương sai số: $MSE = 11810127.28950637$

Hệ số xác định: $R^2 = 0.87821798354801$

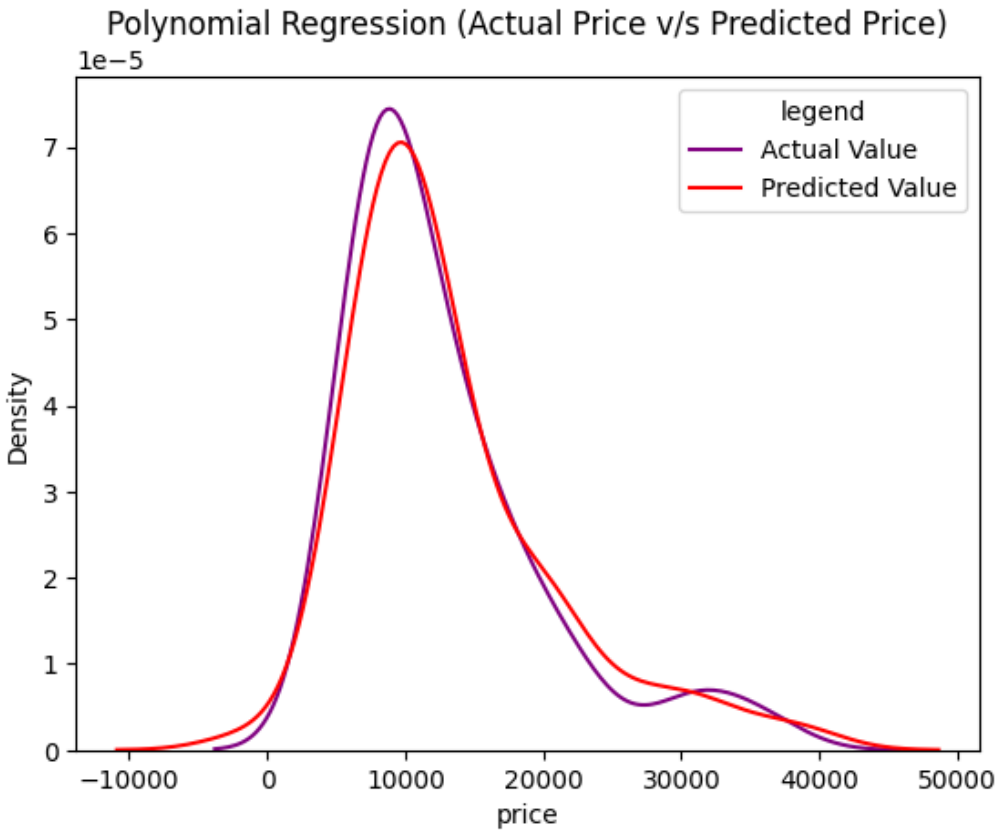
- Mô hình hồi quy Ridge - Ridge Regression Model

  (best alpha = 10)

Trung bình bình phương sai số: $MSE = 11497260.176787565$

Hệ số xác định: $R^2 = 0.8761679017052117$

- Mô hình hồi quy đa thức - Polynomial Regression Model



Trung bình bình phương sai số: $MSE = 13657800.514230987$

Hệ số xác định: $R^2 = 0.9496649792978403$

So sánh hiệu quả ba mô hình:

Model	Mean Squared Error	R-Squared
Multiple Linear Regression	1.181013e+07	0.878218
Ridge Regression	1.149726e+07	0.876168
Polynomial Regression	1.365780e+07	0.949665

Ba mô hình cho hiệu quả dự báo khá tốt ở tập kiểm thử. Trong đó, mô hình dự báo đa thức cho kết quả vượt trội hơn cả.

4 Kết luận

Mô hình được xây dựng với thuật toán *Hồi quy đa thức* mang đến hiệu quả tốt nhất bởi có nhiều nhân tố ảnh hưởng đến giá bán. Điều này cho thấy trong thực tế, các thuộc tính thường bị ảnh hưởng bởi nhiều nhân tố khác nhau, và thuật toán Hồi quy đa thức rất phù hợp cho những trường hợp này. Tuy nhiên, cần lựa chọn bậc của đa thức phù hợp để không dẫn đến tình trạng quá khớp (*overfitting*) cho mô hình dự báo.

Mô hình ước tính giá xe ô tô cũ là một công cụ hữu ích cho cả người bán, người mua và thị trường nói chung. Dự án này sẽ tập trung xây dựng và so sánh ba mô hình hồi quy tuyến tính để xác định phương pháp ước tính giá chính xác nhất. Bằng việc áp dụng mô hình này, thị trường xe ô tô cũ có thể trở nên minh bạch, hiệu quả và thuận tiện hơn cho mọi người.

Tham khảo

[1] IBM, *What is a data warehouse?*
https://www.ibm.com/topics/data-warehouse

[2] Astera (March 21st, 2024), *Data Warehouse Concepts: Kimball vs. Inmon Approach*
https://www.astera.com/type/blog/data-warehouse-concepts/

[3] Galaktikasoft (Dec 12th, 2017), *What is OLAP (Online Analytical Processing)?*
https://galaktika-soft.com/blog/overview-of-olap-technology.html

[4] iHOCLAPTRINH, *MachineLearning - Thuật toán Polynomial Regression*
https://www.ihoclaptrinh.com/machine-learning-polynomial-regression

[5] SaturnCloud, *Multivariate Polynomial Regression with Python*
https://saturncloud.io/blog/multivariate-polynomial-regression-with-python/
#step-3-create-the-feature-matrix-and-target-vector

[6] Lari Giba, *Ridge Regression Explained, Step by Step*
https://machinelearningcompass.com/machine_learning_models/ridge_regression/