HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN 1
_____ o0o _____

BÀI TẬP LỚN
NHẬP MÔN KHOA HỌC DỮ LIỆU

Tên đề tài: Thu thập, xử lý dữ liệu từ trang web oto.com.vn và áp dụng thuật toán K-means Clustering cho bài toán phân loại nhãn ô tô

LỚP : N03
Số thứ tự nhóm: 8

Nguyễn Đức Trung **MSSV: B22DCCN871**
Nguyễn Cao Duy **MSSV: B22DCCN150**
Vũ Thế Vinh **MSSV: B22DCCN907**

Giảng viên hướng dẫn: Ths. Hoài Thư

HÀ NỘI, 10/2025

MỤC LỤC

DANH MỤC HÌNH VẼ

DANH MỤC THUẬT NGỮ VÀ TỪ VIẾT TẮT

API	Giao diện lập trình ứng dụng (Application Programming Interface)
CSV	Định dạng tệp dữ liệu phân tách bằng dấu phẩy (Comma-Separated Values)
DBSCAN	Thuật toán phân cụm theo mật độ (Density-Based Spatial Clustering of Applications with Noise)
EDA	Phân tích dữ liệu khám phá (Exploratory Data Analysis)
GMM	Mô hình hỗn hợp Gaussian (Gaussian Mixture Model)
HTML	Ngôn ngữ đánh dấu siêu văn bản (HyperText Markup Language)
K-Means	Thuật toán phân cụm K-Means
ML	Học máy (Machine Learning)
WCSS	Tổng bình phương sai số trong cụm (Within-Cluster Sum of Squares)

Viết tắt	Tên tiếng Anh	Tên tiếng Việt
API	Application Programming Interface	Giao diện lập trình ứng dụng
CSV	Comma-Separated Values	Dữ liệu dạng bảng phân tách bằng dấu phẩy
EDA	Exploratory Data Analysis	Phân tích dữ liệu khám phá
K-Means	K-Means Clustering	Thuật toán phân cụm K-Means
WCSS	Within-Cluster Sum of Squares	Tổng phương sai trong cụm
DBSCAN	Density-Based Spatial Clustering of Applications with Noise	Thuật toán phân cụm theo mật độ
GMM	Gaussian Mixture Model	Mô hình hỗn hợp Gaussian
ML	Machine Learning	Học máy
HTML	HyperText Markup Language	Ngôn ngữ đánh dấu siêu văn bản

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

1.1 Đặt vấn đề

Trong bối cảnh phát triển mạnh mẽ của công nghệ thông tin và chuyển đổi số hiện nay, dữ liệu trở thành nguồn tài nguyên quan trọng, đóng vai trò then chốt trong việc ra quyết định ở hầu hết các lĩnh vực. Trong ngành công nghiệp ô tô, dữ liệu về giá cả, hãng xe, thông số kỹ thuật, tình trạng sử dụng và xu hướng thị trường được thu thập và phân tích nhằm hỗ trợ người tiêu dùng, nhà sản xuất cũng như các doanh nghiệp kinh doanh xe đưa ra các chiến lược phù hợp. Các trang web thương mại điện tử như *oto.com.vn* hiện nay đóng vai trò là nguồn dữ liệu phong phú, cập nhật và có tính thực tiễn cao, cung cấp hàng trăm nghìn bản ghi về xe mới và xe cũ từ nhiều phân khúc, thương hiệu và khu vực khác nhau.

Tuy nhiên, lượng dữ liệu khổng lồ này thường tồn tại dưới dạng phi cấu trúc hoặc bán cấu trúc, được hiển thị trực tiếp trên trang web mà không có API công khai để truy cập. Việc thu thập, xử lý và phân loại dữ liệu trở thành một thách thức lớn khi người dùng hoặc nhà nghiên cứu cần phân tích xu hướng giá, so sánh đặc điểm hoặc xác định nhóm sản phẩm tương đồng. Đặc biệt, trong bối cảnh dữ liệu ô tô ngày càng đa dạng, việc phân nhóm tự động theo các đặc trưng kỹ thuật hoặc giá bán sẽ giúp hỗ trợ hiệu quả cho việc định vị sản phẩm, đánh giá thị trường và ra quyết định mua bán.

Để giải quyết vấn đề này, hướng tiếp cận thu thập dữ liệu tự động (web crawling) kết hợp với các thuật toán học máy không giám sát như *K-means Clustering* tỏ ra đặc biệt hiệu quả. Việc áp dụng K-means cho phép hệ thống tự động phân chia các mẫu xe thành những nhóm có đặc điểm tương đồng mà không cần nhãn ban đầu, từ đó hỗ trợ các tác vụ như gợi ý mua xe, phân tích thị trường hoặc phát hiện các mẫu dữ liệu bất thường.

Chính vì vậy, việc nghiên cứu, thiết kế và thực hiện đề tài "Thu thập, xử lý dữ liệu từ trang web *oto.com.vn* và áp dụng thuật toán K-means Clustering cho bài toán phân loại nhãn ô tô" là cần thiết và mang tính ứng dụng thực tiễn cao. Đề tài không chỉ giúp người học nắm bắt quy trình khai thác và xử lý dữ liệu thực tế mà còn củng cố kiến thức về học máy, thuật toán phân cụm và kỹ năng lập trình xử lý dữ liệu trong Python.

1.2 Mục tiêu và phạm vi đề tài

Hiện nay, hầu hết các nền tảng rao bán ô tô tại Việt Nam như *oto.com.vn*, *bonbanh.com* hay *chotot.com* đều cung cấp thông tin xe theo từng bài đăng riêng lẻ,

khiến việc tổng hợp và phân tích dữ liệu trở nên khó khăn. Các công cụ hiện có chủ yếu tập trung vào hiển thị dữ liệu dạng bảng, không cung cấp tính năng phân nhóm hay phân tích dữ liệu nâng cao. Mặt khác, các nghiên cứu trong nước về khai thác dữ liệu thương mại điện tử ô tô vẫn còn hạn chế, chủ yếu dừng ở mức mô phỏng hoặc tập trung vào dữ liệu tĩnh.

Trên cơ sở đó, bài tập lớn này hướng đến việc xây dựng một hệ thống cơ bản có khả năng tự động thu thập dữ liệu ô tô từ trang web *oto.com.vn*, thực hiện tiền xử lý dữ liệu nhằm loại bỏ thông tin dư thừa, chuẩn hóa các thuộc tính (như giá, năm sản xuất, hãng xe, dòng xe, dung tích động cơ, hộp số, nhiên liệu, số km đã đi, v.v.), sau đó áp dụng thuật toán K-means để phân cụm các mẫu xe theo đặc điểm tương đồng.

Mục tiêu cụ thể của đề tài bao gồm: (1) Xây dựng công cụ thu thập dữ liệu tự động từ trang web *oto.com.vn*. (2) Thực hiện các bước làm sạch, mã hóa, và chuẩn hóa dữ liệu phục vụ cho phân tích. (3) Ứng dụng thuật toán K-means Clustering để phân nhóm dữ liệu ô tô dựa trên các đặc trưng định lượng và định tính. (4) Trực quan hóa kết quả phân cụm nhằm hỗ trợ người dùng nhận diện xu hướng và sự tương đồng giữa các nhóm xe.

Phạm vi của bài tập lớn giới hạn ở dữ liệu được thu thập từ trang *oto.com.vn* trong thời điểm thực hiện, với các thuộc tính cơ bản phục vụ cho việc phân cụm. Đề tài tập trung vào việc chứng minh khả năng áp dụng của K-means cho bài toán phân loại nhãn ô tô và không đi sâu vào khía cạnh thương mại, dự đoán giá hoặc đề xuất mua xe.

1.3 Định hướng giải pháp

Để hiện thực hóa các mục tiêu nêu trên, đề tài được triển khai dựa trên nền tảng công nghệ Python cùng các thư viện mã nguồn mở hỗ trợ mạnh mẽ cho quá trình thu thập, xử lý và phân tích dữ liệu. Cụ thể, quá trình thu thập dữ liệu được thực hiện bằng cách sử dụng các thư viện *Requests* và *BeautifulSoup* để gửi yêu cầu và trích xuất nội dung HTML từ các trang xe trên *oto.com.vn*. Trong trường hợp trang có cấu trúc động, thư viện *Selenium* có thể được sử dụng để mô phỏng hành vi người dùng và thu thập dữ liệu chính xác hơn.

Sau khi dữ liệu được thu thập, các bước tiền xử lý được thực hiện bằng thư viện *pandas* và *numpy*, bao gồm làm sạch dữ liệu (loại bỏ giá trị thiếu hoặc sai), chuyển đổi dữ liệu về dạng số, chuẩn hóa các thuộc tính nhằm đảm bảo hiệu quả của thuật toán phân cụm. Giai đoạn phân tích và phân cụm được triển khai bằng thư viện *scikit-learn*, trong đó thuật toán *K-means Clustering* được sử dụng để chia dữ liệu thành các nhóm có đặc điểm tương đồng về giá, năm sản xuất, hãng xe hoặc thông

số kỹ thuật.

Kết quả của quá trình phân cụm được trực quan hóa bằng các thư viện *matplotlib* và *seaborn*, giúp hiển thị mối quan hệ giữa các nhóm xe, từ đó hỗ trợ người dùng trong việc đánh giá xu hướng và nhận diện các phân khúc ô tô phổ biến.

Giải pháp đề xuất có ý nghĩa trong việc minh họa quy trình đầy đủ của một dự án khai phá dữ liệu thực tế – từ giai đoạn thu thập, xử lý đến phân tích và hiển thị kết quả. Đóng góp chính của bài tập lớn là xây dựng được một pipeline tự động hóa cơ bản cho bài toán phân loại nhãn ô tô dựa trên dữ liệu thực, đồng thời đánh giá được tính phù hợp của thuật toán K-means trong việc xử lý dữ liệu phi cấu trúc từ web thương mại điện tử.

1.4 Bố cục bài tập lớn

Phần còn lại của báo cáo bài tập lớn được tổ chức như sau.

Chương 2 trình bày về cơ sở lý thuyết và các khái niệm nền tảng liên quan đến bài toán, bao gồm giới thiệu về quy trình thu thập dữ liệu từ web, mô hình tiền xử lý dữ liệu, và lý thuyết về thuật toán K-means Clustering. Phần này cung cấp cơ sở học thuật cho các bước triển khai trong các chương tiếp theo.

Trong Chương 3, báo cáo tập trung mô tả chi tiết quy trình xây dựng hệ thống, bao gồm quá trình thu thập dữ liệu từ trang *oto.com.vn*, xử lý, chuẩn hóa và mã hóa dữ liệu, đồng thời mô tả cách áp dụng K-means để phân nhóm dữ liệu ô tô. Kết quả thực nghiệm và đánh giá hiệu quả thuật toán cũng được trình bày trong chương này.

Cuối cùng, Chương 4 trình bày kết luận của bài tập lớn, tóm tắt lại toàn bộ quy trình nghiên cứu, các kết quả đạt được, những hạn chế tồn tại và đề xuất hướng phát triển trong tương lai, bao gồm việc mở rộng mô hình sang các thuật toán phân cụm khác hoặc tích hợp thêm các yếu tố dữ liệu nâng cao như hành vi người dùng và dữ liệu hình ảnh xe.

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1 Tổng quan về khai thác dữ liệu và bài toán phân cụm

Khai thác dữ liệu (*Data Mining*) là quá trình trích xuất thông tin hữu ích từ một tập dữ liệu lớn, nhằm phát hiện ra các mẫu, mối quan hệ hoặc quy luật tiềm ẩn mà con người khó nhận biết trực tiếp. Trong kỷ nguyên dữ liệu hiện nay, khai thác dữ liệu đóng vai trò quan trọng trong nhiều lĩnh vực như thương mại điện tử, tài chính, y tế, và đặc biệt là phân tích thị trường.

Bài toán phân cụm (*Clustering*) là một trong những kỹ thuật cơ bản và quan trọng của khai thác dữ liệu. Mục tiêu của phân cụm là chia tập dữ liệu ban đầu thành các nhóm (*clusters*) sao cho các đối tượng trong cùng một nhóm có đặc điểm tương đồng, trong khi các nhóm khác biệt nhau rõ rệt. Khác với bài toán phân loại (classification), phân cụm thuộc nhóm học máy không giám sát (*unsupervised learning*), tức là dữ liệu không có nhãn trước, và mô hình phải tự khám phá cấu trúc tiềm ẩn trong dữ liệu.

Trong lĩnh vực thương mại điện tử ô tô, việc phân cụm có thể giúp nhóm các xe có cùng mức giá, hãng sản xuất hoặc đặc điểm kỹ thuật, từ đó hỗ trợ người dùng so sánh và đánh giá các phân khúc xe một cách trực quan hơn.

2.2 Cơ sở lý thuyết về thu thập dữ liệu từ web

2.2.1 Khái niệm và vai trò của Web Scraping

Web Scraping (thu thập dữ liệu từ web) là kỹ thuật tự động trích xuất thông tin từ các trang web. Khác với việc thu thập thủ công, Web Scraping sử dụng các chương trình tự động để gửi yêu cầu đến máy chủ, tải về mã HTML và bóc tách các phần tử cần thiết như văn bản, bảng, hình ảnh hoặc siêu dữ liệu.

Trong đề tài này, trang *oto.com.vn* được chọn làm nguồn dữ liệu vì chứa kho thông tin lớn về các mẫu xe, thông số kỹ thuật, giá bán, tình trạng xe và vị trí địa lý. Việc tự động hóa thu thập giúp tiết kiệm thời gian, đảm bảo tính nhất quán và có thể mở rộng quy mô thu thập dữ liệu.

2.2.2 Các công cụ và thư viện sử dụng

Để thu thập dữ liệu, đề tài sử dụng các thư viện phổ biến trong Python gồm:

- **Requests**: Dùng để gửi yêu cầu HTTP đến máy chủ và lấy về mã HTML của trang.
- **BeautifulSoup**: Dùng để phân tích cú pháp HTML, tìm kiếm và trích xuất thông tin từ các thẻ cụ thể (như `div`, `span`, `a`).

- **Selenium**: Dùng trong trường hợp trang web sử dụng JavaScript để hiển thị nội dung, cho phép mô phỏng hành vi người dùng và tải đầy đủ dữ liệu trước khi trích xuất.

Trong file thực nghiệm, quá trình thu thập dữ liệu được thực hiện bằng cách truy cập danh sách các xe trên *oto.com.vn*, lấy thông tin chi tiết như tên xe, hãng, giá, năm sản xuất, tình trạng, hộp số, nhiên liệu và số km đã đi. Dữ liệu sau đó được lưu dưới dạng bảng để dễ xử lý trong giai đoạn tiếp theo.

2.3 Tiền xử lý và làm sạch dữ liệu

2.3.1 Vai trò của tiền xử lý

Dữ liệu thu được từ web thường chứa nhiều giá trị thiếu, trùng lặp hoặc không đồng nhất về định dạng. Việc tiền xử lý giúp đảm bảo chất lượng dữ liệu trước khi đưa vào thuật toán học máy. Một tập dữ liệu sạch, chuẩn hóa giúp mô hình học chính xác hơn và kết quả phân tích có ý nghĩa hơn.

2.3.2 Các bước tiền xử lý chính

Trong đề tài, quy trình tiền xử lý được thực hiện bằng các thư viện *pandas* và *numpy*, bao gồm:

- **Loại bỏ dữ liệu trùng lặp hoặc thiếu**: Các bản ghi không đầy đủ về giá hoặc năm sản xuất được thay thế bằng các giá trị phù hợp như *median* hoặc *mode*.
- **Chuẩn hóa định dạng dữ liệu**: Giá được chuyển về dạng số nguyên, các chuỗi văn bản như “1 tỷ 200 triệu” được chuyển thành “1200000000”.
- **Chuẩn hóa dữ liệu định lượng**: Các giá trị như giá xe, năm sản xuất, số km được chuẩn hóa về cùng thang đo bằng *StandardScaler*, *MinMaxScaler* hoặc *RobustScaler*, nhằm tránh việc thuộc tính có giá trị lớn chi phối thuật toán.

Việc xử lý dữ liệu đúng cách giúp đảm bảo hiệu quả của thuật toán K-means, vốn rất nhạy cảm với sự chênh lệch về thang đo giữa các thuộc tính.

2.4 Thuật toán K-means Clustering

2.4.1 Nguyên lý hoạt động

K-means là một trong những thuật toán phân cụm phổ biến nhất, thuộc nhóm học máy không giám sát. Thuật toán do MacQueen đề xuất năm 1967, có mục tiêu chia tập dữ liệu thành k cụm sao cho tổng bình phương khoảng cách giữa các điểm dữ liệu và tâm cụm gần nhất là nhỏ nhất.

Thuật toán hoạt động qua các bước chính sau:

1. Chọn ngẫu nhiên k tâm cụm ban đầu.

2. Gán mỗi điểm dữ liệu vào cụm có tâm gần nhất (dựa trên khoảng cách Euclid).
3. Tính lại tâm cụm mới bằng trung bình cộng của các điểm trong cụm.
4. Lặp lại hai bước trên cho đến khi tâm cụm không thay đổi đáng kể hoặc đạt điều kiện dừng.

Mục tiêu tối ưu của thuật toán là hàm mất mát:

$$J = \sum_{i=1}^{k} \sum_{x_j \in C_i} \|x_j - \mu_i\|^2 \tag{2.1}$$

trong đó μ_i là tâm cụm thứ i, C_i là tập điểm thuộc cụm i.

2.4.2 Ưu và nhược điểm

Ưu điểm:

- Dễ hiểu, dễ triển khai và tính toán nhanh, đặc biệt hiệu quả với dữ liệu lớn.
- Thích hợp với dữ liệu có cấu trúc cầu hoặc dạng số.

Nhược điểm:

- Cần xác định trước số cụm k.
- Dễ bị ảnh hưởng bởi nhiễu và các giá trị ngoại lai.
- Kết quả phụ thuộc vào việc chọn tâm cụm ban đầu.

2.4.3 Phương pháp xác định số cụm tối ưu

Trong thực nghiệm, số cụm k được lựa chọn bằng phương pháp *Elbow Method* (khuỷu tay). Phương pháp này tính tổng bình phương khoảng cách nội cụm (Within-Cluster Sum of Squares – WCSS) cho nhiều giá trị k, sau đó chọn giá trị tại điểm "khuỷu tay" – nơi WCSS bắt đầu giảm chậm lại, biểu thị rằng việc tăng thêm cụm không cải thiện đáng kể chất lượng phân cụm.

2.5 Trực quan hóa dữ liệu và kết quả phân cụm

Để đánh giá và minh họa kết quả phân cụm, các thư viện *matplotlib* và *seaborn* được sử dụng nhằm biểu diễn dữ liệu dưới dạng biểu đồ hai chiều hoặc ba chiều, giúp người đọc dễ quan sát mối quan hệ giữa các thuộc tính (như giá – năm sản xuất – hãng xe). Các cụm được tô màu khác nhau thể hiện các nhóm xe có đặc điểm tương đồng.

Trực quan hóa không chỉ hỗ trợ đánh giá chất lượng mô hình mà còn giúp phát hiện các nhóm xe nổi bật, ví dụ như nhóm xe cũ giá thấp, nhóm xe sang cao cấp, hay nhóm xe mới cùng mức giá tầm trung. Điều này minh chứng tính thực tiễn của

mô hình trong việc hỗ trợ phân tích thị trường ô tô.

2.6 Tổng kết chương

Chương này đã trình bày các kiến thức nền tảng phục vụ cho việc triển khai đề tài, bao gồm lý thuyết về khai thác dữ liệu, thu thập dữ liệu web, tiền xử lý dữ liệu và thuật toán K-means. Các nội dung này tạo cơ sở khoa học cho quy trình thực nghiệm được trình bày trong Chương 3. Trên cơ sở các lý thuyết này, đề tài tiến hành triển khai toàn bộ pipeline thu thập, xử lý và phân cụm dữ liệu ô tô từ trang *oto.com.vn*, được mô tả chi tiết trong chương tiếp theo.

CHƯƠNG 3. THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ

3.1 Môi trường triển khai và công cụ

Hệ thống được triển khai bằng ngôn ngữ Python (phiên bản tương thích với các thư viện dùng trong notebook). Các thư viện chính sử dụng trong thực nghiệm gồm: `requests` và `BeautifulSoup` (`html5lib`) để thu thập và phân tích HTML; `pandas`, `numpy` để lưu trữ và xử lý dữ liệu dạng bảng; `scikit-learn` để thực hiện các bước chuẩn hóa và thuật toán phân cụm (K-means); `matplotlib` và `seaborn` để trực quan hóa dữ liệu và kết quả phân cụm. Notebook thực nghiệm tổ chức các bước theo thứ tự: thu thập dữ liệu (crawling), tổng hợp vào DataFrame, tiền xử lý và làm sạch, chuẩn hóa, lựa chọn số cụm bằng Elbow method, huấn luyện mô hình K-means, gán nhãn và trực quan hóa.

3.2 Thu thập dữ liệu (Web scraping)

Dữ liệu được thu thập từ trang *oto.com.vn* thông qua hai nguồn chính trong notebook: phần rao bán xe cũ (https://oto.com.vn/mua-ban-xe/pp) và phần rao bán xe mới (https://oto.com.vn/mua-ban-xe-moi/pp), với vòng lặp tuần tự qua nhiều trang (ví dụ: `for p in range(1, 100)`). Mỗi bài đăng xe được truy cập để trích xuất các thông tin chi tiết bằng cách phân tích cấu trúc HTML với BeautifulSoup. Các trường thông tin thu thập tiêu biểu gồm: tên xe, hãng, giá bán (chuỗi), năm sản xuất, số km đã đi, loại nhiên liệu, hộp số, số ghế, dung tích động cơ (nếu có), cùng các trường phụ khác được lấy từ danh sách ul.list-info li trong HTML. Dữ liệu từng bài đăng được lưu vào danh sách Python (`data, new_car_data`) và sau đó gom lại thành `pandas.DataFrame` (`all_car_df`) để xử lý tiếp.

Trong notebook đã áp dụng một số biện pháp thao tác HTTP cơ bản như sử dụng requests.Session() và cấu hình header `User-Agent` để giảm khả năng bị chặn; đồng thời có chèn thời gian chờ (sleep) để tránh gửi quá nhiều yêu cầu liên tiếp làm quá tải server.

3.2.1 Mô tả dữ liệu thu được

Sau khi thu thập, dữ liệu thô chứa nhiều cột thuộc tính dạng chuỗi (strings) và nhiều giá trị thiếu. Các cột quan trọng được sử dụng cho phân cụm gồm: `Giá` (giá bán), `Năm sản xuất`, `Km đã đi`, `Số ghế`, cùng một số thuộc tính kỹ thuật khác nếu đủ thông tin. Dữ liệu ban đầu có các đặc điểm sau: có giá trị thiếu ở một số cột, các giá trị giá và km có đơn vị/chuỗi khác nhau (ví dụ biểu diễn theo "tỷ", "triệu" hoặc có dấu phẩy, chữ), tồn tại các bản ghi trùng lặp và một số ngoại lệ (outliers) với giá rất lớn hoặc km rất lớn.

3.2.2 Lưu trữ dữ liệu

Dữ liệu sau khi thu thập được lưu dưới dạng DataFrame all_car_df và sao lưu thường xuyên trong notebook (có thể export CSV nếu cần). Việc lưu giữ bản sao cho phép lặp lại các bước tiền xử lý và so sánh các phương pháp chuẩn hóa khác nhau mà không phải crawl lại toàn bộ.

3.3 Các giả định và giới hạn dữ liệu

Quy phạm thu thập giới hạn vào dữ liệu có thể truy xuất được công khai trên trang tại thời điểm crawling; các bài đăng ẩn, dữ liệu thông qua API (nếu có) hoặc dữ liệu hình ảnh chưa được dùng trực tiếp cho phân cụm. Ngoài ra, do dữ liệu thực tế có nhiễu lớn, notebook đưa ra quyết định bảo tồn một số ngoại lệ (không loại bỏ toàn bộ outlier) để phản ánh đúng sự đa dạng của dữ liệu thị trường.

3.4 Ghi chú về đạo đức và tuân thủ

Việc thu thập dữ liệu thực hiện theo nguyên tắc tôn trọng robots.txt và quy định sử dụng của website; trong thực tế cần kiểm tra điều khoản sử dụng của *oto.com.vn* trước khi thu thập ở quy mô lớn và đảm bảo không vi phạm pháp lý hoặc quá tải dịch vụ.

3.5 Tiền xử lý dữ liệu và đặc trưng hoá

3.5.1 Làm sạch sơ bộ

Các bước làm sạch trong notebook bao gồm: loại bỏ bản ghi trùng lặp, chuẩn hóa encoding, và chuyển các chuỗi chứa thông tin số (ví dụ: giá, km, năm) về dạng số. Cụ thể, cột `Km đã đi` được tách và chuyển về dạng số (với xử lý các chuỗi có chứa "km" và dấu chấm giữa các số). Các cột dạng chuỗi có khoảng trắng hoặc ký tự vô nghĩa được strip/clean để giữ dữ liệu nhất quán.

3.5.2 Xử lý giá trị thiếu

Notebook kiểm tra từng cột để xác định các giá trị thiếu (NaN). Chiến lược được dùng là: nếu tỉ lệ thiếu ở một cột thấp và có thể suy đoán giá trị hợp lý thì điền bằng mode (giá trị xuất hiện nhiều nhất) hoặc một phép nội suy đơn giản; nếu thuộc tính quá quan trọng mà thiếu nhiều thì có thể loại bản ghi. Trong notebook, có đoạn mã dùng .fillna(`mode_value`) để điền các giá trị thiếu ở các cột định tính, đồng thời in ra tổng số giá trị thiếu sau xử lý để xác nhận.

3.5.3 Xử lý ngoại lai

Qua quan sát phân bố dữ liệu, notebook nhận thấy tồn tại nhiều ngoại lai (ví dụ một vài xe có giá cực lớn hoặc km bất thường). Thay vì loại bỏ toàn bộ ngoại lai, tác giả notebook đưa ra kết luận không loại bỏ hoàn toàn do những giá trị đó có thể phản ánh nhóm xe hạng sang/xe đặc thù, và sẽ dùng phương pháp chuẩn hóa phù

hợp (Robust Scaler) để giảm ảnh hưởng của ngoại lai khi phân cụm.

3.5.4 Chuẩn hóa thang đo (Scaling)

Do K-means rất nhạy cảm với thang đo các thuộc tính, notebook thử nghiệm ba phương pháp chuẩn hóa: `Standard Scaler`, `MinMax Scaler` và `Robust Scaler`. Kết quả minh họa bằng việc in 5 dòng đầu sau khi chuẩn hóa cho thấy: `Standard Scaler` đưa dữ liệu về mean = 0 và std = 1; `MinMax Scaler` nén dữ liệu vào khoảng [0,1]; `Robust Scaler` dùng median và IQR, ít bị ảnh hưởng bởi ngoại lai. Với dữ liệu ô tô có nhiều ngoại lai, notebook chọn `Robust Scaler` làm dữ liệu đầu vào cho K-means (biến lưu trong notebook là `df_scaled_robust`).

3.6 Phương pháp phân cụm và đánh giá

3.6.1 Thiết lập thuật toán K-means

Thuật toán K-means được thực thi bằng `sklearn.cluster.KMeans`. Trong thực nghiệm, các tham số chính được đặt như sau: `random_state=42` để tái lập kết quả, `n_init=10` (chạy nhiều lần với các khởi tạo ngẫu nhiên để tránh bị local minima), và `n_clusters=k` thay đổi theo quá trình tìm kiếm. Dữ liệu đưa vào là `df_scaled_robust` (ma trận các thuộc tính số đã chuẩn hóa).

3.6.2 Lựa chọn số cụm (Elbow method)

Notebook thực hiện Elbow method bằng cách tính WCSS (Within-Cluster Sum of Squares, trong sklearn là `kmeans.inertia_`) cho các giá trị k từ 1 đến 10. Giá trị WCSS được lưu vào danh sách `wcss` và vẽ đồ thị WCSS vs k. Điểm "khuỷu" nơi WCSS bắt đầu giảm chậm hơn được chọn làm số cụm tối ưu. Trong thực nghiệm của notebook, tác giả quan sát và chọn k=4 là hợp lý cho tập dữ liệu hiện tại. (Lưu ý: Elbow method mang tính trực quan - có thể kết hợp Silhouette score cho xác nhận nếu cần.)

3.6.3 Huấn luyện mô hình cuối và gán nhãn

Sau khi chọn `k_toi_uu` = 4, notebook khởi tạo model `KMeans(n_clusters = k_toi_uu, random_state = 42, n_init = 10)` và gọi `fit` trên `df_scaled_robust`. Nhãn thu được lưu vào mảng `labels` (giá trị thuộc tập 0,...,k1) và được gắn vào DataFrame gốc (sau khi chuyển đổi ngược nếu cần) vào cột `Nhãn` để phục vụ trực quan hóa và phân tích hậu xử lý.

3.6.4 Đánh giá chất lượng phân cụm

Notebook dùng hai cách chính để đánh giá: (1) giá trị nội bộ WCSS (inertia) và biểu đồ Elbow để chọn k; (2) trực quan hóa các mối quan hệ hai chiều giữa các thuộc tính quan trọng với màu sắc biểu diễn nhãn (ví dụ scatterplot giữa `Km đã`

đi và Giá theo nhãn, hoặc giữa Số ghế và Giá). Việc trực quan hóa cho phép đánh giá ý nghĩa phân cụm trong ngữ cảnh thực tế (ví dụ một cụm có thể tương ứng nhóm xe cũ giá thấp, cụm khác là xe mới/xe cao cấp giá cao). Notebook không tính Silhouette score trong bản hiện tại, nhưng có thể bổ sung để đánh giá cấu trúc phân cụm định lượng hơn.

3.7 Kết quả thực nghiệm và bình luận

3.7.1 Kết quả chính

Từ quá trình Elbow method, tác giả chọn k=4 làm số cụm tối ưu. Sau khi gán nhãn cho từng bản ghi, notebook trực quan hóa kết quả bằng các biểu đồ scatter (sử dụng seaborn.scatterplot). Hai biểu đồ tiêu biểu trong notebook là: (i) Giá theo Km đã đi với màu sắc biểu diễn nhãn phân cụm; (ii) Giá theo Số ghế với màu nhãn. Trên các đồ thị này, có thể quan sát phân bố nhóm rõ rệt - một số nhóm tập trung ở vùng giá thấp và km cao (các xe cũ/đã đi nhiều), trong khi một cụm khác tập trung ở vùng giá cao và km thấp (xe mới/xe sang).

3.7.2 Phân tích ý nghĩa các cụm

Dựa trên các thuộc tính chính (Giá, Km, Năm sản xuất, Số ghế), các cụm thu được có thể diễn giải ở mức ý nghĩa thực tiễn như sau: cụm A gồm các xe có giá thấp, km cao và năm sản xuất cũ - có thể là phân khúc xe đã qua sử dụng nhiều; cụm B gồm các xe giá trung bình với phân bố km đa dạng - phân khúc phổ thông; cụm C chứa các xe có giá cao, năm sản xuất mới và km thấp - phân khúc xe sang/xe mới; cụm D có thể là các xe đặc thù (ví dụ số ghế khác thường hoặc loại thân xe đặc thù). Đây là diễn giải khái quát dựa trên đồ thị trực quan; để khẳng định cần phân tích sâu hơn các giá trị trung tâm cụm (cluster centroids) và tỷ lệ thuộc từng hãng, loại xe trong cụm.

3.7.3 Nhận xét về chất lượng và hạn chế

Kết quả cho thấy K-means có khả năng tách các nhóm có khác biệt rõ rệt về quy mô (giá, km). Tuy nhiên, một số hạn chế hiển nhiên cần lưu ý: (1) K-means giả sử các cụm có dạng hình cầu trong không gian thuộc tính - điều này có thể không đúng với một số dạng phân bố dữ liệu ô tô; (2) ảnh hưởng của các biến định tính chưa được khai thác triệt để (mã hóa nhãn có thể làm mất thông tin quan hệ); (3) chưa dùng các tiêu chí đánh giá ngoài (external validation) do thiếu nhãn thực tế; (4) chưa tính Silhouette score hoặc Davies–Bouldin index để có thước đo cấu trúc phân cụm định lượng hơn; (5) kết quả bị phụ thuộc vào chất lượng tiền xử lý (cách tính giá, xóa/điền thiếu, xử lý ngoại lai).

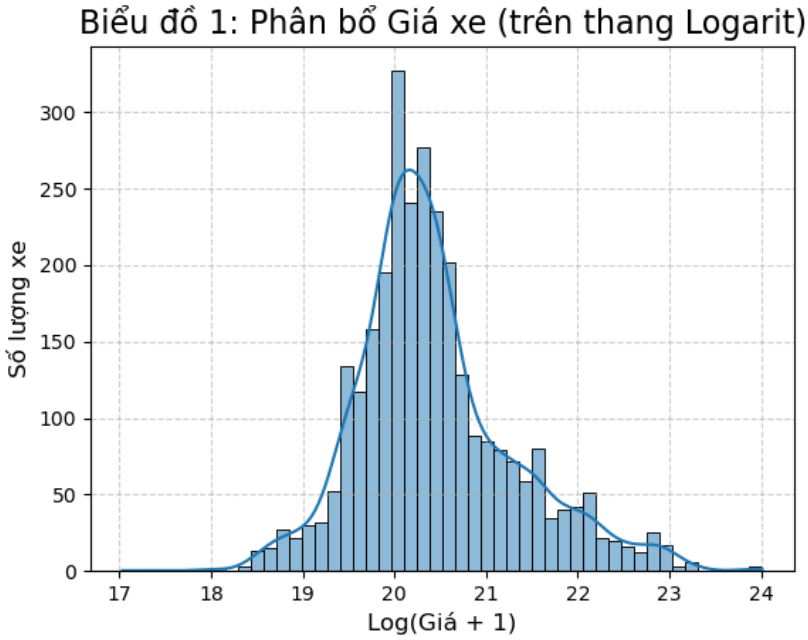
3.7.4 Các bước cải tiến khả dĩ

Từ kết quả hiện tại, các hướng cải tiến có thể thực hiện: (i) bổ sung chỉ số đánh giá nội bộ như Silhouette score để kiểm chứng lựa chọn k; (ii) thử các thuật toán phân cụm khác (ví dụ DBSCAN, Gaussian Mixture Models) để so sánh tính phù hợp khi dữ liệu có phân bố không tuyến tính hoặc có mật độ không đồng nhất; (iii) kết hợp các biến định tính bằng embedding hoặc feature hashing để giữ nhiều thông tin hơn; (iv) mở rộng dữ liệu bằng metadata (vùng miền, số lượt xem, thời gian đăng) và dữ liệu hình ảnh (trích xuất đặc trưng ảnh) để phân cụm đa chiều phong phú hơn.

3.8 Phân tích trực quan dữ liệu và kết quả phân cụm

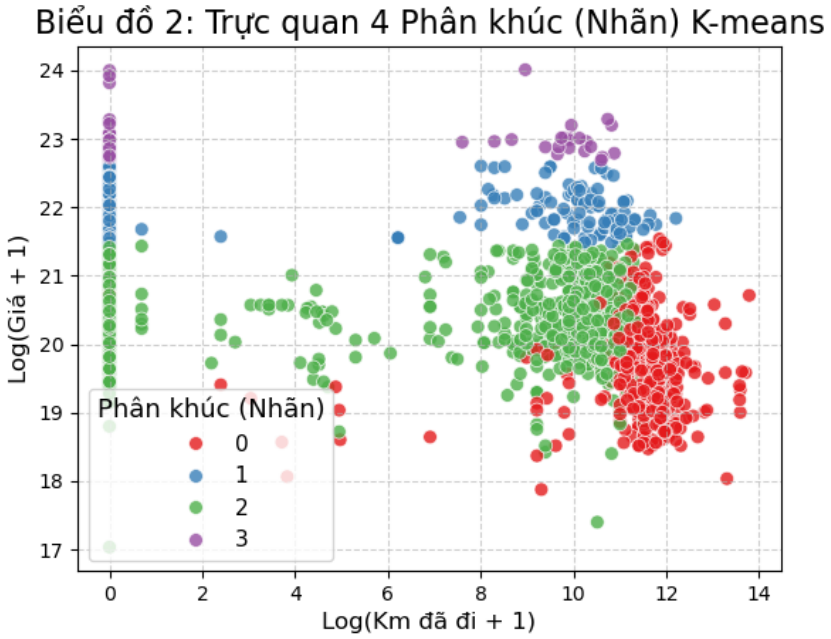
Phần này trình bày hệ thống các biểu đồ trực quan nhằm minh họa đặc trưng dữ liệu, mối quan hệ giữa các biến, cũng như kết quả phân cụm được thực hiện trong nghiên cứu.

3.8.1 Phân tích phân phối và tương quan



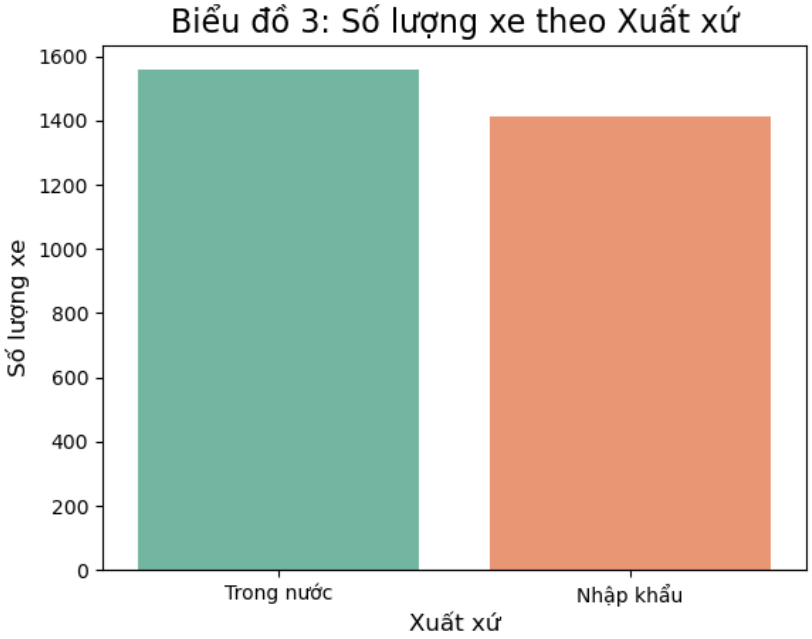
Hình 3.1: Phân bổ Giá xe (trên thang Logarit)

Biểu đồ 1 là một biểu đồ phân bổ (histogram) kết hợp với ước lượng mật độ nhân (KDE), minh họa phân phối của giá xe sau khi áp dụng phép biến đổi logarit (`log1p`). Phép biến đổi này giúp chuẩn hóa dữ liệu giá vốn bị lệch phải, làm phân phối tiệm cận chuẩn, tăng độ ổn định cho mô hình học máy.



Hình 3.2: Giá xe theo Km đã đi và Nhãn phân cụm

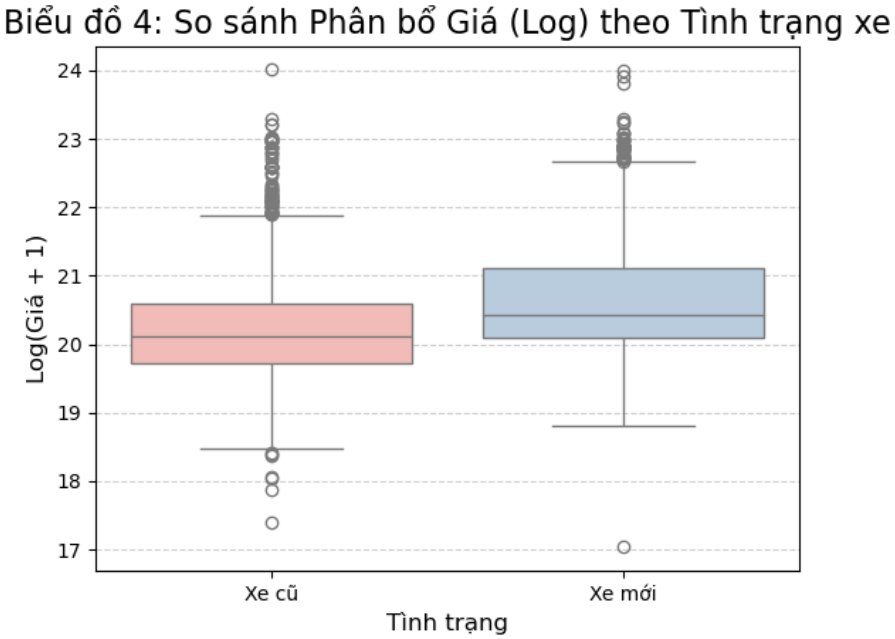
Biểu đồ 2 trực quan hóa kết quả thuật toán K-means ($K = 4$) trên hai trục 'Giá' và 'Km đã đi'. Kết quả cho thấy phân tách rõ rệt: Nhãn 0 ('Xe Cũ') có giá thấp và số km cao, trong khi các Nhãn 1, 2, và 3 tương ứng với các phân khúc xe sang, xe mới/lướt và siêu xe.



Hình 3.3: Phân bổ số lượng xe theo Xuất xứ

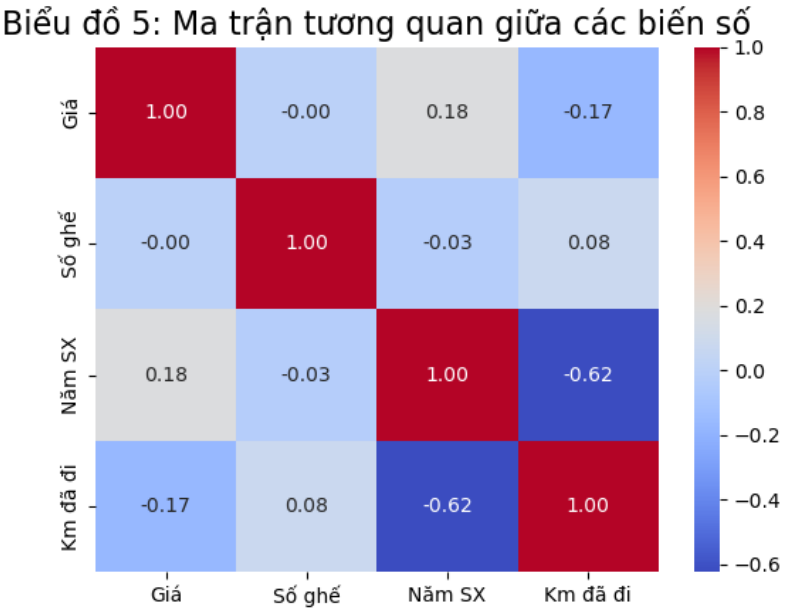
Biểu đồ 3 (cột ngang) trình bày tần suất xe theo 'Xuất xứ'. Xe 'Trong nước' chiếm tỷ trọng áp đảo và được chọn làm giá trị mode để điền khuyết cho các giá trị

thiếu.



Hình 3.4: Phân bổ Log(Giá) theo Tình trạng

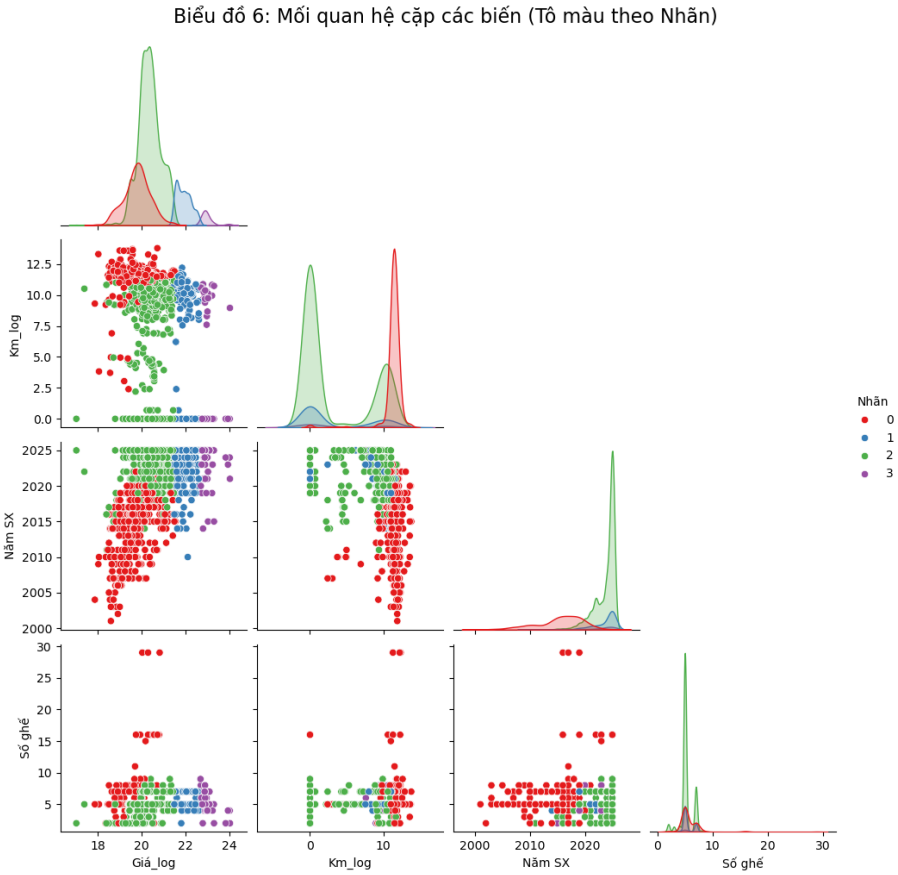
Biểu đồ 4 (boxplot) so sánh phân vị giá logarit giữa hai nhóm ’Xe mới’ và ’Xe cũ’. ’Xe mới’ có median cao hơn đáng kể, trong khi ’Xe cũ’ có IQR rộng hơn, thể hiện độ phân tán lớn về giá.



Hình 3.5: Ma trận tương quan Heatmap

Biểu đồ 5 minh họa ma trận tương quan Pearson. Kết quả cho thấy tương quan

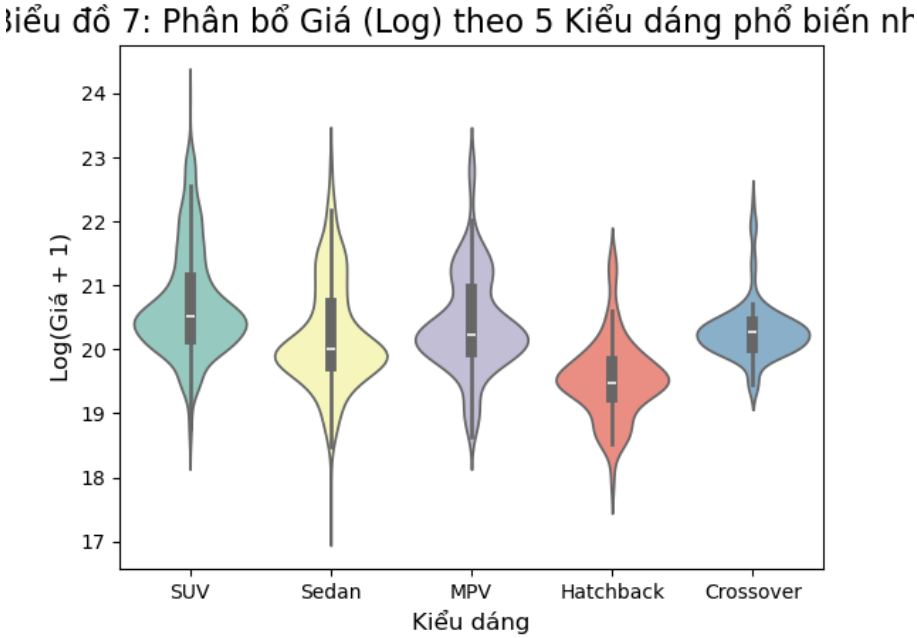
dương giữa ’Giá’ và ’Năm SX’ (r = 0.18) và tương quan âm mạnh giữa ’Năm SX’ và ’Km đã đi’ (r = -0.62), phù hợp với quy luật khấu hao xe.



Hình 3.6: Biểu đồ quan hệ cặp giữa các biến theo nhãn phân cụm

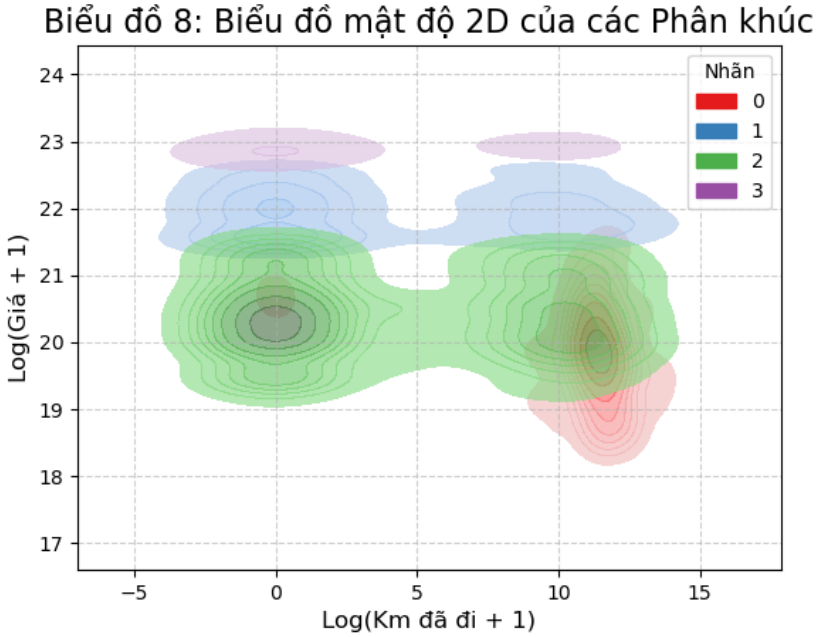
Biểu đồ 6 cho thấy sự phân tách rõ giữa bốn nhóm: Nhãn 0 chiếm ưu thế ở vùng giá thấp–km cao–năm cũ; Nhãn 1 tập trung giá trung bình–km thấp–năm mới; Nhãn 2 ở vùng trung gian; Nhãn 3 có giá rất cao và số ghế biến thiên lớn. Các phân bố biên cho thấy quy luật: năm càng mới → giá càng cao, km càng thấp.

3.8.2 Phân tích theo đặc trưng loại xe



Hình 3.7: Phân bổ Log(Giá) theo Kiểu dáng

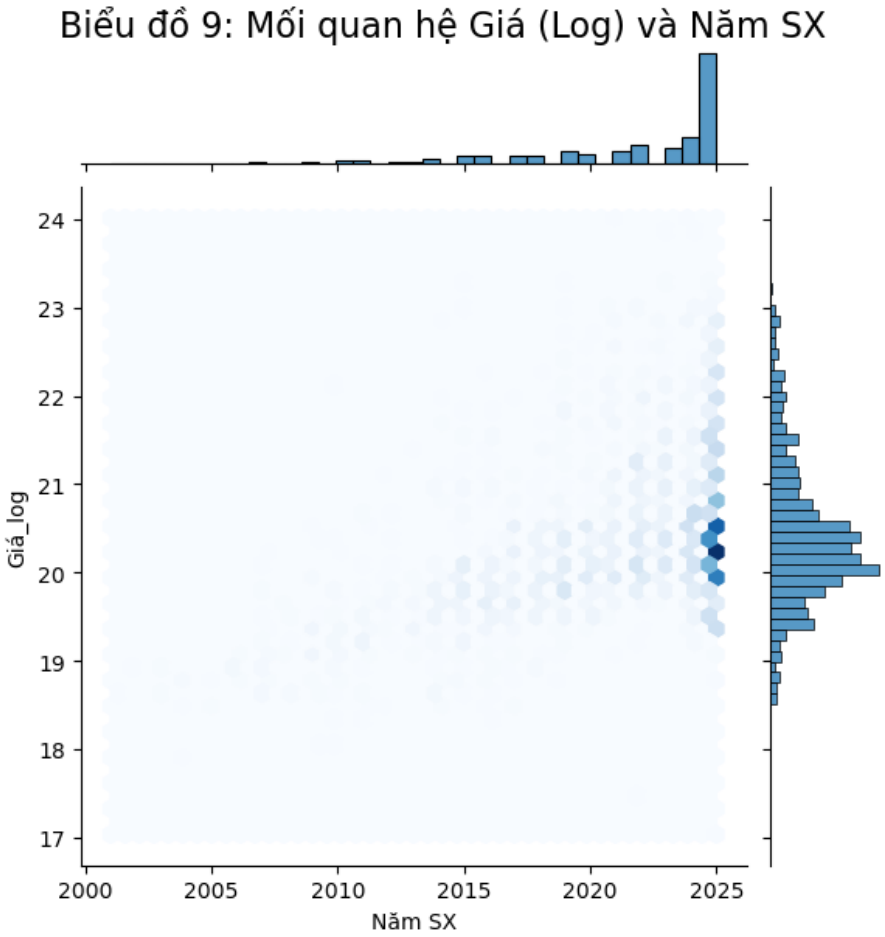
Biểu đồ 7 (violin plot) cho thấy 'SUV' và 'Sedan' có mật độ tập trung cao nhất ở mức giá trung bình. 'Hatchback' tập trung ở vùng giá thấp, còn 'Crossover' có mật độ ổn định.



Hình 3.8: Biểu đồ mật độ 2D (KDE) theo Cụm

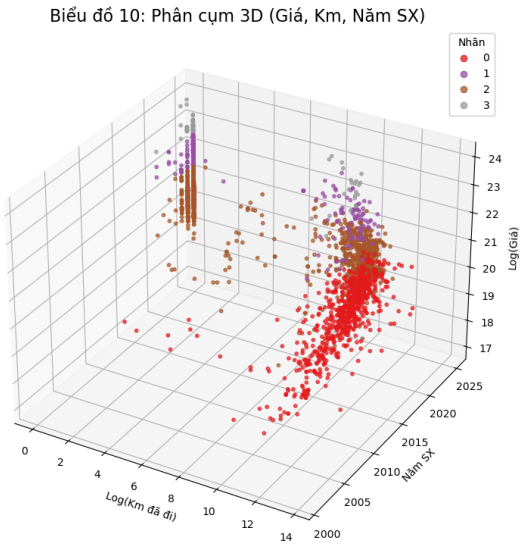
Biểu đồ 8 sử dụng KDE 2D cho thấy Nhãn 0 (Xe cũ) tập trung ở vùng km

cao–giá thấp, còn các nhãn khác tập trung ở năm SX mới, giá cao hơn.



Hình 3.9: Jointplot (Hexbin) giữa Năm SX và Log(Giá)

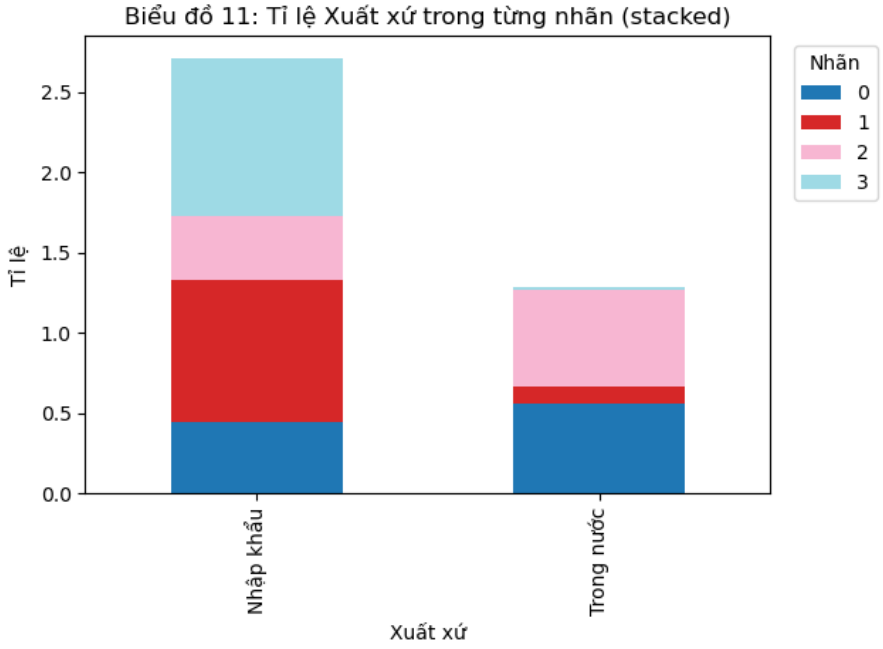
Biểu đồ 9 (jointplot) sử dụng phương pháp 'hexbin' (lưới lục giác) để trực quan hóa mật độ điểm dữ liệu tại giao điểm của 'Năm SX' và 'Giá log'. Vùng tập trung dày đặc nhất nằm ở các xe đời rất mới (sau 2020) và ở mức giá logarit trung bình-thấp. Các biểu đồ phân bổ lề (marginal histograms) ở trên và bên phải tái khẳng định rằng phần lớn xe trong tập dữ liệu được sản xuất sau năm 2020.



Hình 3.10: Biểu đồ phân tán 3D của các cụm

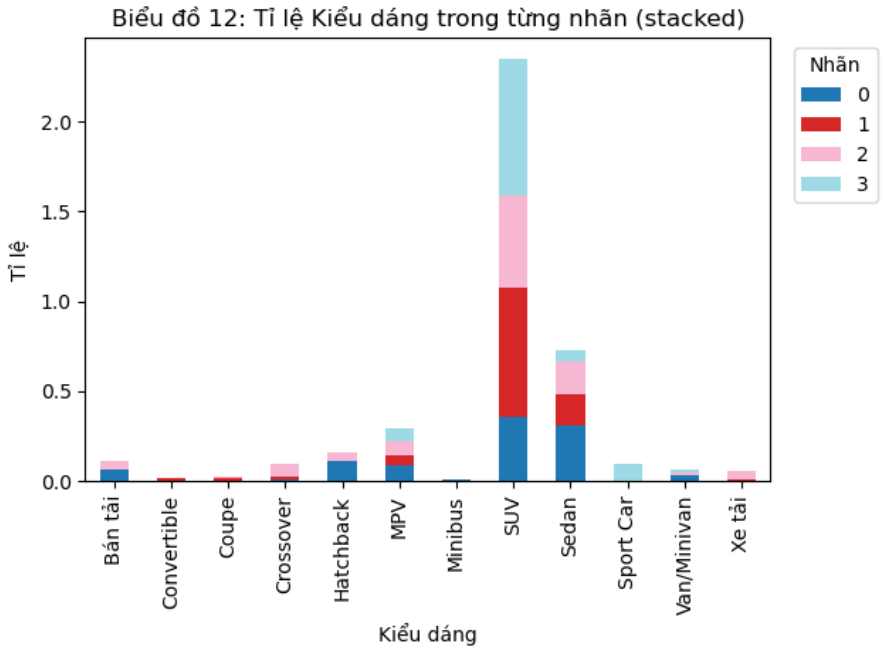
Biểu đồ 10 mở rộng phân tích phân cụm sang không gian 3 chiều, thêm trục ’Km đã đi’ vào các trục ’Năm SX’ và ’Giá log’. Trực quan hóa này cung cấp bằng chứng mạnh mẽ nhất về sự tách biệt của Nhãn 0 (Xe cũ), vốn nằm riêng biệt trên trục ’Km đã đi’ với giá trị cao. Ba cụm còn lại chủ yếu nằm trên một mặt phẳng có ’Km đã đi’ thấp, chỉ khác nhau về ’Giá log’ và ’Năm SX’.

3.8.3 Phân tích theo đặc trưng danh mục



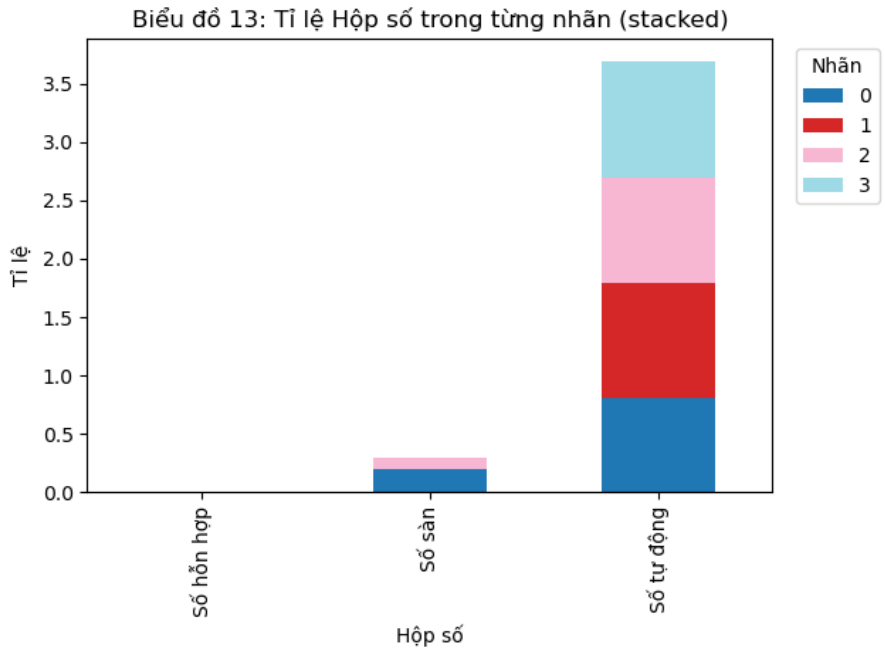
Hình 3.11: Phân bổ Xuất xứ trong từng Nhãn

Biểu đồ 11 (biểu đồ cột chồng tỷ lệ) phân tích thành phần 'Xuất xứ' trong 4 cụm K-means. Kết quả cho thấy mối tương quan rõ rệt giữa xuất xứ và phân khúc thị trường: các cụm giá thấp và trung bình (Nhãn 0 - 'Xe Cũ' và Nhãn 2 - 'Xe Phổ thông Mới') bị chi phối bởi xe 'Trong nước'. Ngược lại, các cụm giá cao (Nhãn 1 - 'Xe Sang' và Nhãn 3 - 'Xe Siêu Sang') bị thống trị bởi xe 'Nhập khẩu'.



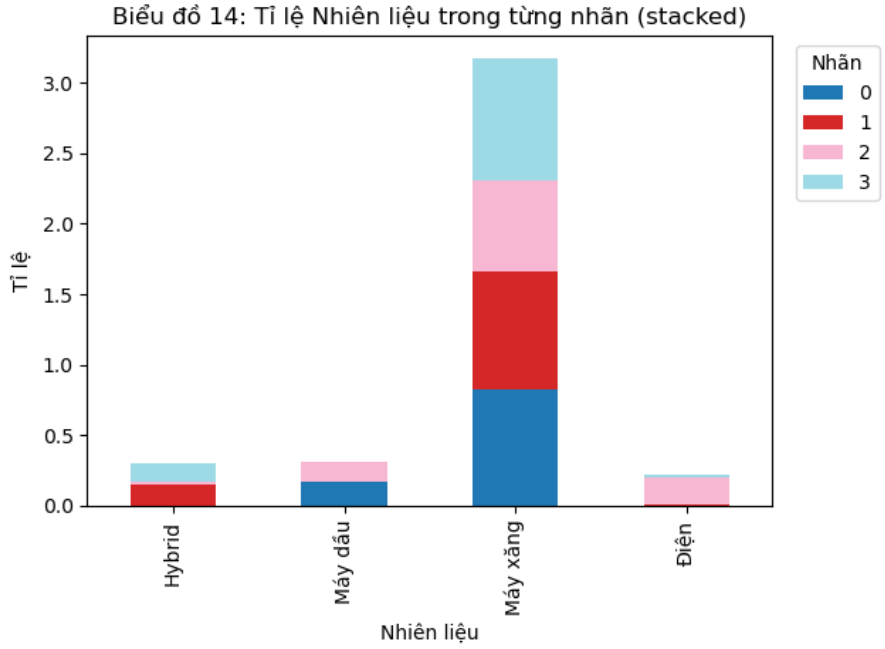
Hình 3.12: Phân bổ Kiểu dáng trong từng Nhãn

Biểu đồ 12 phân tích tỷ lệ 'Kiểu dáng' trong từng cụm. Mặc dù 'SUV' và 'Sedan' là hai kiểu dáng thống trị ở tất cả các phân khúc, Nhãn 0 ('Xe Cũ') thể hiện sự đa dạng cao nhất về kiểu dáng (bao gồm tỷ lệ đáng kể của 'Van/Minivan', 'Crossover', 'MPV'). Các phân khúc xe mới/lướt (1, 2, 3) cho thấy sự tập trung thị trường mạnh mẽ vào 'SUV', đặc biệt là ở phân khúc 'Xe Siêu Sang' (Nhãn 3).



Hình 3.13: Phân bổ Hộp số trong từng Nhãn

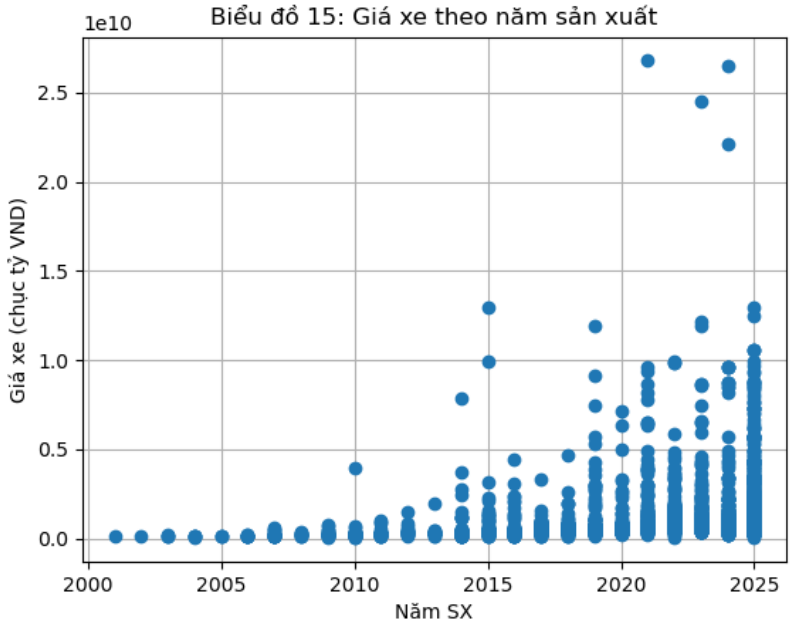
Biểu đồ 13 minh họa sự khác biệt rõ rệt về 'Hộp số' giữa các cụm. Các phân khúc xe mới/lướt (Nhãn 1, 2, 3) gần như hoàn toàn (xấp xỉ 100) sử dụng 'Số tự động'. 'Số sàn' chỉ xuất hiện với tỷ lệ đáng kể ở phân khúc 'Xe cũ' (Nhãn 0), phản ánh xu hướng thị trường hiện đại loại bỏ hộp số sàn, đặc biệt là ở các phân khúc xe đời mới và cao cấp.



Hình 3.14: Phân bổ Nhiên liệu trong từng Nhãn

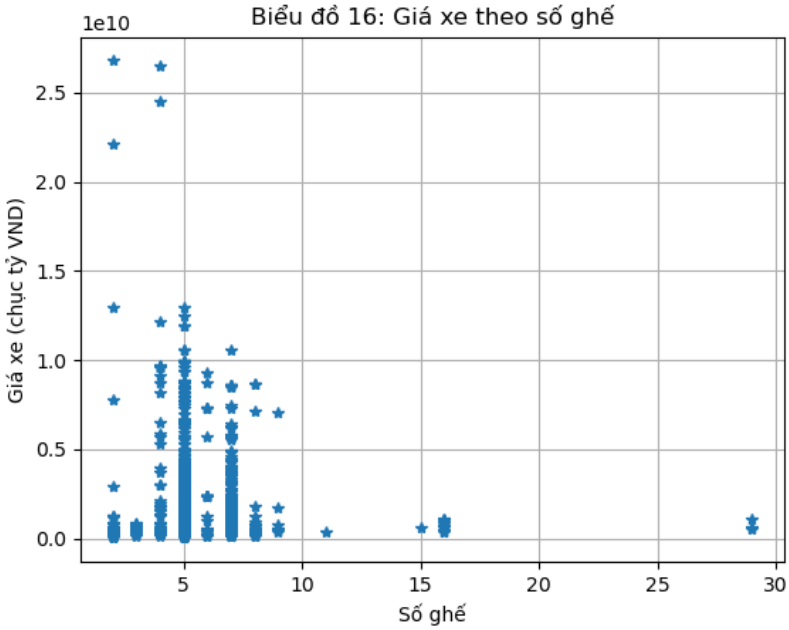
Biểu đồ 14 phân tích loại 'Nhiên liệu' sử dụng trong từng cụm. 'Máy xăng' là loại nhiên liệu chủ đạo trong cả bốn phân khúc. 'Máy dầu' xuất hiện với tỷ trọng đáng kể nhất ở các phân khúc 'Xe cũ' (Nhãn 0) và 'Xe phổ thông mới' (Nhãn 2). Xe 'Hybrid' và 'Điện' có tỷ lệ rất nhỏ, cho thấy sự thâm nhập ban đầu vào các phân khúc xe mới (chủ yếu là Nhãn 1 và 2).

3.8.4 Phân tích mối quan hệ giữa các biến định lượng



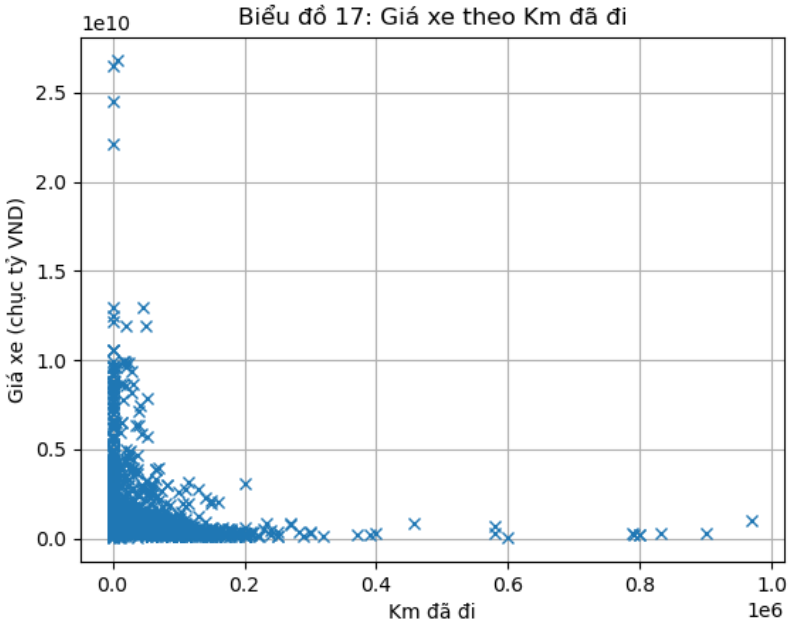
Hình 3.15: Tương quan giữa Giá (log) và Năm SX

Biểu đồ 15 là một biểu đồ phân tán minh họa mối quan hệ giữa 'Giá (log)' và 'Năm SX'. Kết quả cho thấy một mối tương quan dương rõ rệt: xe được sản xuất càng gần đây (Năm SX lớn hơn) thì giá trị logarit của giá càng cao. Phần lớn các quan sát trong tập dữ liệu tập trung dày đặc trong giai đoạn từ 2018 đến 2024.



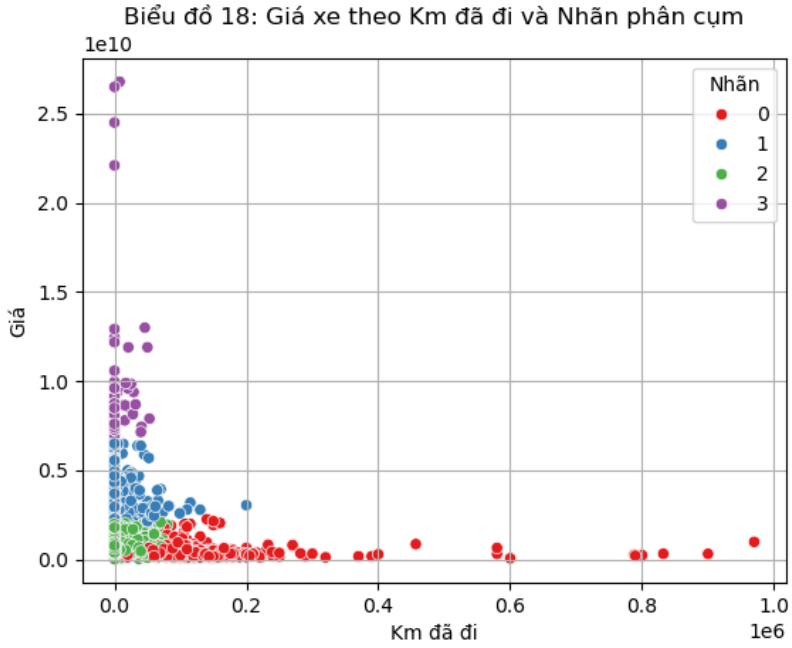
Hình 3.16: Phân bổ Giá (log) theo Số ghế

Biểu đồ 16 là một biểu đồ hộp (boxplot) so sánh phân vị của 'Giá (log)' theo 'Số ghế'. Dữ liệu cho thấy xe 5 chỗ và 7 chỗ là hai hạng mục phổ biến nhất, cả hai đều có độ phân tán giá (khoảng tứ phân vị) rất lớn. Đáng chú ý, xe 2 chỗ (thường là xe thể thao/coupe) có giá trị trung vị cao và phân bố lệch về phía giá trị cao. xe càng sang càng ít ghế



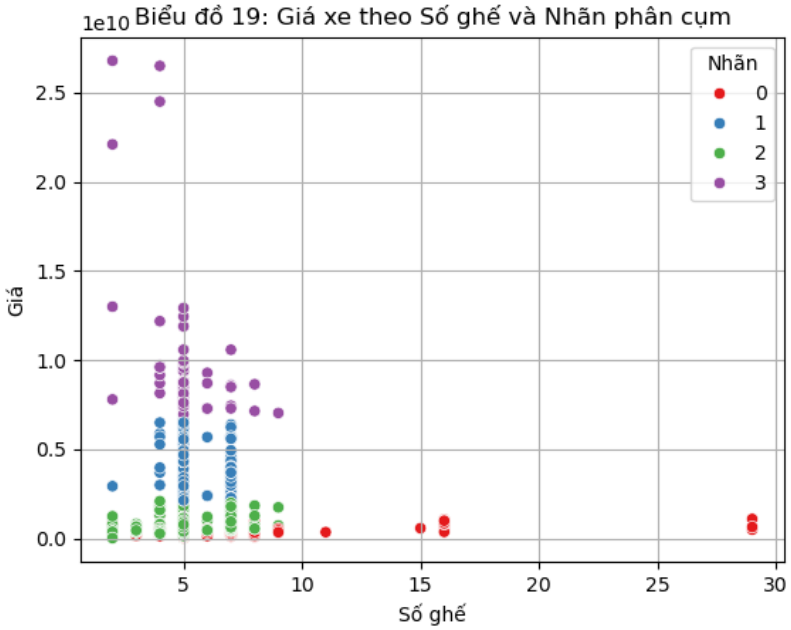
Hình 3.17: Tương quan giữa Giá (log) và Km đã đi

Biểu đồ 17 (biểu đồ phân tán) thể hiện mối tương quan nghịch biến rõ rệt giữa 'Giá (log)' và 'Km đã đi', phù hợp với lý thuyết khấu hao tài sản. Phần lớn xe mới/lướt (số km thấp, < 50,000 km) tập trung ở bên trái biểu đồ với độ phân tán giá rất cao. Khi số km tăng lên, giá trị logarit của giá có xu hướng giảm đều và độ phân tán giá hẹp lại.



Hình 3.18: Giá (log) vs Km đã đi (phân theo Nhãn)

Biểu đồ 18 trực quan hóa 4 cụm K-means trên hai trục 'Giá (log)' và 'Km đã đi'. Kết quả phân cụm là cực kỳ rõ rệt: Nhãn 0 ('Xe Cũ') được tách biệt hoàn toàn theo trục tung (giá < 4 tỷ). Ba cụm còn lại (xe mới/lướt) đều có số km thấp nhưng được phân tầng hoàn hảo theo trục tung (Giá), tương ứng với Nhãn 2 ('Xe Phổ thông Mới'), Nhãn 1 ('Xe Sang'), và Nhãn 3 ('Xe Siêu Sang') theo thứ tự giá tăng dần.



Hình 3.19: Phân bổ Giá (log) theo Số ghế (phân theo Nhãn)

Biểu đồ 19 (biểu đồ hộp đa chiều) phân tích 'Giá (log)' theo 'Số ghế', được chia nhỏ theo 4 cụm. Biểu đồ này xác nhận sự phân tầng giá trị: với cùng một số ghế (ví dụ: xe 5 chỗ), giá trị trung vị tăng dần một cách rõ rệt theo thứ tự Nhãn 0 ('Xe Cũ') < Nhãn 2 ('Xe Phổ thông Mới') < Nhãn 1 ('Xe Sang') < Nhãn 3 ('Xe Siêu Sang'). Điều này củng cố tính chính xác của mô hình phân cụm trong việc xác định các phân khúc thị trường.

Tổng hợp các biểu đồ trên cho thấy mô hình K-means không chỉ phản ánh đúng đặc trưng định lượng (Giá, Năm SX, Km) mà còn khớp hợp lý với các biến định tính (Xuất xứ, Kiểu dáng, Hộp số, Nhiên liệu), từ đó cho phép diễn giải các phân khúc thị trường một cách logic và nhất quán.

3.9 Kết chương

Chương này mô tả chi tiết quy trình triển khai hệ thống thực nghiệm trong notebook: từ thu thập dữ liệu từ *oto.com.vn*, làm sạch và chuẩn hóa dữ liệu đến áp dụng K-means clustering và trực quan hóa kết quả. Tác giả đã lựa chọn `Robust Scaler` nhằm giảm ảnh hưởng của ngoại lai và dùng Elbow method để lựa chọn k=4. Kết quả phân cụm cho thấy khả năng phân biệt các phân khúc xe theo các đặc trưng kinh tế và kỹ thuật, nhưng vẫn còn tồn tại các hạn chế về mặt đánh giá và khai thác biến định tính. Chương tiếp theo sẽ trình bày chi tiết các kết quả thực nghiệm (bảng/đồ thị), phân tích số liệu thống kê của từng cụm và đề xuất các cải tiến cụ thể dựa trên các hạn chế đã nêu.

CHƯƠNG 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

4.1 Kết luận

Phần kết luận trình bày tóm tắt những đóng góp chính của bài tập lớn, rút ra các bài học kinh nghiệm và nêu bật giá trị thực tiễn của giải pháp đã triển khai. Mục tiêu của chương là so sánh kết quả đạt được với các công trình tương tự, làm rõ những điểm mới của công việc và chỉ ra những hạn chế còn tồn tại để định hướng nghiên cứu trong tương lai.

4.1.1 Tổng kết các kết quả chính

Nghiên cứu đã xây dựng một quy trình (pipeline) hoàn chỉnh để thu thập dữ liệu từ trang web *oto.com.vn*, tiền xử lý dữ liệu thô, biểu diễn các thuộc tính số và phân loại (phân cụm) các mẫu xe bằng thuật toán K-means. Đóng góp chính của đề tài bao gồm: (i) một pipeline thu thập dữ liệu thực tế có khả năng mở rộng; (ii) chiến lược tiền xử lý biến chuỗi giá/đơn vị về dạng số phù hợp cho phân tích; (iii) lựa chọn và so sánh các phương pháp chuẩn hóa phù hợp với dữ liệu có ngoại lai; và (iv) minh họa tính khả thi của K-means trong phân chia các phân khúc ô tô thực tế (với số cụm thử nghiệm hợp lý là $k = 4$).

4.1.2 Đóng góp chi tiết

a, Pipeline thu thập và chuẩn hóa dữ liệu từ *oto.com.vn*

(i) Dẫn dắt. Các dữ liệu rao bán trên nền tảng thương mại điện tử thường ở dạng phi cấu trúc hoặc bán cấu trúc và được hiển thị động trên trang web, gây khó khăn cho việc thu thập và phân tích hàng loạt.

(ii) Giải pháp. Bài tập lớn đã thiết kế một pipeline thu thập tự động sử dụng `requests` kết hợp `BeautifulSoup` (và `Selenium` khi cần) để trích xuất các trường thông tin chủ chốt từ các trang tin; tiếp đó, pipeline thực hiện loạt bước chuẩn hóa (chuẩn hoá chuỗi giá, chuyển đơn vị, loại bỏ ký tự không cần thiết) và lưu trữ dưới dạng `pandas.DataFrame`. Quy trình cũng bao gồm cơ chế lưu phiên bản (export CSV) để đảm bảo khả năng lặp lại.

(iii) Kết quả đạt được. Pipeline cho phép thu thập hàng nghìn bản ghi có cấu trúc thống nhất, giảm đáng kể thời gian tiền xử lý thủ công và tạo cơ sở dữ liệu đủ điều kiện cho các bước phân tích tiếp theo.

b, Chiến lược tiền xử lý và chuẩn hóa phù hợp với dữ liệu có ngoại lai

(i) Dẫn dắt. Dữ liệu giá và quãng đường (km) thường có phân phối lệch phải và nhiều ngoại lai, làm giảm hiệu quả các thuật toán phân cụm phụ thuộc khoảng cách.

(ii) Giải pháp. Trong pipeline đã thử nghiệm các phương pháp chuẩn hóa khác nhau (StandardScaler, MinMaxScaler và RobustScaler) và áp dụng phép biến đổi log (log1p) cho biến Giá khi phân tích. Cuối cùng lựa chọn `RobustScaler` cho dữ liệu đưa vào K-means để giảm ảnh hưởng của ngoại lai.

(iii) Kết quả đạt được. Việc kết hợp biến đổi log và RobustScaler giúp các phép đo khoảng cách phản ánh tốt hơn cấu trúc dữ liệu thực tế, làm cho K-means tạo ra các cụm có ý nghĩa thực tiễn (ví dụ phân biệt xe cũ-km cao vs. xe mới/xe sang).

c, Ứng dụng K-means cho phân loại nhãn ô tô và phân tích phân khúc thị trường

(i) Dẫn dắt. Việc tự động phân nhóm các mẫu xe theo đặc trưng giúp hỗ trợ phân tích thị trường, gợi ý người mua và phân tích cạnh tranh.

(ii) Giải pháp. Áp dụng Elbow method để lựa chọn số cụm tối ưu sơ bộ, sau đó huấn luyện K-means với các tham số được chuẩn hóa (ví dụ `random_state` để đảm bảo tính tái lập, `n_init` để giảm khả năng dính local minima). Kết quả phân cụm được kiểm tra bằng trực quan hóa đa dạng (scatter, pairplot, KDE, boxplot) và so sánh các phân bố biến định tính giữa các cụm.

(iii) Kết quả đạt được. Mô hình phân cụm phân tách rõ rệt các phân khúc thị trường về giá, năm sản xuất và km, cho phép diễn giải các cụm như: *Xe Cũ*, *Xe Phổ thông Mới*, *Xe Sang*, *Xe Siêu Sang*. Kết quả này cho thấy K-means là lựa chọn hiệu quả để khám phá phân khúc trong dữ liệu rao bán ô tô.

4.1.3 Bài học kinh nghiệm

Quá trình làm bài tập lớn cho thấy tầm quan trọng của việc chuẩn hóa dữ liệu hợp lý trước khi áp dụng thuật toán khoảng cách, cũng như sự cần thiết của quá trình kiểm thử các phương pháp khác nhau (scaler, cách mã hóa biến phân loại). Bên cạnh đó, việc lưu trữ và phiên bản hoá dữ liệu thô giúp tăng tính khoa học và khả năng tái lập khi cần chạy lại các thí nghiệm với tham số khác.

4.2 Hướng phát triển

Phần này nêu các công việc cần triển khai để hoàn thiện bài toán hiện tại và một số hướng mở rộng nghiên cứu.

4.2.1 Hoàn thiện các chức năng đã triển khai

1. **Bổ sung các chỉ số đánh giá nội bộ:** Tích hợp Silhouette score và Davies–Bouldin index để định lượng hơn việc chọn k. Các chỉ số này nên được trình bày bằng bảng so sánh (khi thay đổi k từ 2 đến 10) để đưa ra căn cứ chọn k hợp lý thay vì chỉ dựa vào quan sát đồ thị Elbow.
2. **Bảng tóm tắt centroid và kích thước cụm:** Xuất bảng centroid (đã được đảo ngược chuẩn hóa về đơn vị gốc) và số lượng bản ghi mỗi cụm để người đọc dễ so sánh và diễn giải (ví dụ giá trung bình, km trung bình theo từng cụm).
3. **Đạt tính tái lập và tự động hoá:** Đóng gói pipeline thành script/Notebook có kiểm soát tham số (config file), ghi rõ `random_state` và phiên bản thư viện để tăng khả năng tái lập kết quả.
4. **Tăng cường xử lý biến phân loại:** Thử nghiệm các cách biểu diễn biến phân loại nâng cao (one-hot, target encoding, embedding) và đánh giá ảnh hưởng của chúng lên kết quả phân cụm.

4.2.2 Mở rộng và nghiên cứu hướng mới

1. **So sánh đa thuật toán phân cụm:** Thực hiện so sánh hệ thống giữa K-means, DBSCAN, Gaussian Mixture Models (GMM) và các phương pháp phân cụm phân cấp (hierarchical clustering) để tìm phương pháp phù hợp nhất với cấu trúc dữ liệu ô tô (các cụm không nhất thiết có dạng hình cầu).
2. **Kết hợp đặc trưng hình ảnh:** Thu thập ảnh xe (nếu có) và trích xuất đặc trưng bằng mô hình CNN tiền huấn luyện (ví dụ ResNet) để đưa đặc trưng ảnh vào phân cụm đa phương diện, giúp phân biệt rõ hơn phân khúc xe sang/xe bình dân khi thông số kỹ thuật không đủ phân biệt.
3. **Xây dựng hệ gợi ý (recommendation):** Dựa trên cụm phân khúc, phát triển module gợi ý xe tương tự cho người dùng (content-based recommendation) hoặc module phân tích thứ hạng giá.
4. **Thử nghiệm với dữ liệu tăng cường và kiểm định chéo:** Mở rộng tập dữ liệu theo thời gian, chạy kiểm định chéo trên các bảng thời gian khác nhau để kiểm chứng tính ổn định của mô hình phân cụm qua thời gian.
5. **Tích hợp yếu tố kinh tế và metadata:** Bổ sung thông tin như tỉnh/TP đăng bán, lượt xem tin, ngày đăng để phân tích thêm các yếu tố ảnh hưởng đến giá

và cấu trúc phân khúc.

4.3 Kết chương

Chương này đã hệ thống hóa những đóng góp chính của đề tài, trình bày các giải pháp then chốt đã thực hiện và chỉ ra hướng phát triển tiếp theo để hoàn thiện nghiên cứu. Mục tiêu dài hạn là nâng cao độ tin cậy và khả năng ứng dụng của pipeline — cả về mặt kỹ thuật (cải tiến thuật toán, bổ sung dữ liệu, nâng cao khả năng tái lập) và mặt thực tiễn (tích hợp module gợi ý, báo cáo phân khúc cho người dùng và doanh nghiệp).