BỘ XÂY DỰNG
TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TP. HỒ CHÍ MINH

Viện công nghệ thông tin và Điện, điện tử

------o0o-----

UTH
UNIVERSITY
OF TRANSPORT
HOCHIMINH CITY

Final Report

Lecturer	**:**	PhD. Nguyễn Thị Khánh Tiên
Course	**:**	Machine Learning
Course Code	**:**	010112210102
Group	**:**	08
Link Repo	**:**	Link

TP. HỒ CHÍ MINH, 2025

Đóng góp của các thành viên			
STT	Tên thành viên	MSSV	Nội dung công việc
1	Lê Văn An	056205001827	- Code bài Practice 2: Predicting Product Sales - Code bài Practice 3: Clustering Customers - Viết báo cáo chương 2 - Viết báo cáo chương 3 - Viết báo cáo chương 5.3
2	Phạm Gia Bảo	093205005065	- Code bài 1 - Viết báo cáo chương 1 - Viết báo cáo chương 5.1
3	Dương Hưng	086205003910	- Code bài 4 - Viết báo cáo chương 4 - Viết báo cáo chương 5.2
4	Lê Minh Duy Khang	060205001162	- Code bài 1 - Viết báo cáo chương 5.2
5	Nguyễn Việt Danh	054205000575	- Code bài 4 - Viết báo cáo chương 4 - Viết báo cáo chương 5.1

Lời cảm ơn

Nhóm xin chân thành cảm ơn giảng viên PhD. Nguyễn Thị Khánh Tiên đã tận tình hướng dẫn trong suốt học phần Machine Learning.

Những kiến thức, tài liệu và góp ý của cô đã giúp nhóm hoàn thành các bài thực hành cũng như báo cáo cuối kỳ một cách hiệu quả hơn. Nhóm cũng cảm ơn các thành viên đã phối hợp và hỗ trợ nhau trong quá trình thực hiện đồ án.Mặc dù còn nhiều hạn chế, nhóm đã cố gắng hoàn thiện bài báo cáo với tinh thần nghiêm túc và trách nhiệm.

Nhóm kính mong nhận được ý kiến đóng góp của cô để bài làm được hoàn thiện hơn.

Tóm tắt

Báo cáo tổng hợp kết quả thực hiện các bài thực hành trong học phần Machine Learning, bao gồm: phân loại email spam bằng các mô hình học máy cổ điển (Logistic Regression; dự đoán doanh số sản phẩm bằng các mô hình hồi quy và kỹ thuật xử lý đặc trưng trong thương mại điện tử; phân cụm khách hàng dựa trên dữ liệu nhân khẩu học và hành vi mua sắm; và xây dựng mô hình MLP để dự đoán giá nhà với dữ liệu chuẩn hoá.

Các bài thực hành tập trung vào quy trình chuẩn của học máy: tiền xử lý dữ liệu, khám phá dữ liệu, xây dựng mô hình, đánh giá bằng các chỉ số phù hợp và so sánh hiệu quả các thuật toán. Kết quả giúp nhóm hiểu rõ hơn về cách áp dụng các phương pháp học máy vào những bài toán thực tế.

Mục lục

Chương 1. Phân loại email spam bằng mô hình học máy

1.1. Mô tả bài toán

Bài toán phân loại email spam đặt mục tiêu xây dựng một mô hình có khả năng nhận diện liệu một thông điệp văn bản được gắn nhãn spam hay ham (không phải spam). Bản chất của bài toán được mô hình hóa dưới dạng một bài toán phân loại nhị phân, trong đó tập dữ liệu đầu vào bao gồm các câu văn trong trường Message và nhãn tương ứng trong trường Category. Việc phân loại chính xác đóng vai trò quan trọng trong hệ thống lọc thư rác vì nó giúp giảm thiểu các thông điệp không mong muốn, đồng thời góp phần bảo vệ người dùng khỏi nội dung độc hại hoặc quảng cáo gây phiền nhiễu.

Dữ liệu đầu vào trong bài thực nghiệm được nạp từ tệp spam.csv, chứa hai trường chính: Category và Message. Trường Category bao gồm hai giá trị spam và ham, còn Message chứa nội dung văn bản cần phân loại.

1.2. Khám phá và tiền xử lý dữ liệu

Sau khi tải dữ liệu, quá trình khám phá dữ liệu được thực hiện thông qua mô tả thống kê theo từng nhóm nhãn để đánh giá phân bố giữa hai lớp. Việc sử dụng nhóm lệnh df.groupby('Category').describe() cho phép nhận diện sự chênh lệch về số lượng giữa hai loại thông điệp, từ đó đánh giá mức độ mất cân bằng lớp.

	Message			
	count	unique	top	freq
Category				
ham	4825	4516	Sorry, I'll call later	30
spam	747	641	Please call our customer service representativ...	4

Tiền xử lý dữ liệu được thực hiện thông qua bước chuyển đổi nhãn văn bản thành giá trị nhị phân, trong đó nhãn spam được mã hóa thành 1 và nhãn ham thành 0. Việc mã hóa này nhằm đảm bảo dữ liệu phù hợp với các thuật toán học máy vốn yêu cầu đầu ra dạng số.

	Category	Message	spam
0	ham	Go until jurong point, crazy.. Available only ...	0
1	ham	Ok lar... Joking wif u oni...	0
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	1
3	ham	U dun say so early hor... U c already then say...	0
4	ham	Nah I don't think he goes to usf, he lives aro...	0
...	...	...	...
5567	spam	This is the 2nd time we have tried 2 contact u...	1
5568	ham	Will ü b going to esplanade fr home?	0
5569	ham	Pity, * was in mood for that. So...any other s...	0
5570	ham	The guy did some bitching but I acted like i'd...	0
5571	ham	Rofl. Its true to its name	0

5572 rows × 3 columns

Tiếp theo, tập dữ liệu được chia thành hai phần theo tỷ lệ 80% cho huấn luyện và 20% cho kiểm tra. Việc chia tách này nhằm đánh giá khả năng tổng quát hóa của mô hình và tránh hiện tượng overfitting.

```
x_train, x_test, y_train, y_test = train_test_split(df.Message, df.spam, test_size = 0.2)
```

1.3. Trích xuất đặc trưng

Vì dữ liệu đầu vào là văn bản thô nên cần chuyển đổi thành các đặc trưng có thể xử lý bằng mô hình học máy. Trong notebook, quá trình này được triển khai thông qua hàm tokenize, bao gồm các thao tác chuẩn hóa văn bản như chuyển toàn bộ ký tự về dạng chữ thường, loại bỏ ký tự đặc biệt bằng các biểu thức chính quy và tách văn bản thành các từ đơn.

Quá trình trích xuất đặc trưng không sử dụng các kỹ thuật vector hoá phức tạp như TF-IDF mà dựa trên cách tiếp cận dựa vào xác suất và tần suất xuất hiện từ trong quá trình ước lượng của bộ phân loại Naive Bayes tùy chỉnh. Sau khi tiền xử lý, mô hình sẽ học phân phối xác suất của các từ tương ứng với mỗi lớp bằng cách đếm tần suất xuất hiện trong tập huấn luyện và áp dụng làm cơ sở suy luận.

1.4. Xây dựng mô hình

Mô hình sử dụng trong thực nghiệm là một phiên bản cài đặt thủ công của bộ phân loại Naive Bayes. Cấu trúc mô hình bao gồm ba bước: thu thập tần suất xuất hiện từ trong từng lớp, ước lượng xác suất có điều kiện dựa trên công thức Bayes và áp dụng quy tắc quyết định tối đa hóa xác suất hậu nghiệm để dự đoán nhãn cho văn bản mới.

Hàm fit tiến hành duyệt qua từng câu trong tập huấn luyện, áp dụng bộ tách từ tự xây dựng và cập nhật các biến đếm. Các xác suất sau đó được ước lượng dựa trên tổng tần suất xuất hiện, đồng thời áp dụng xử lý Laplace smoothing nhằm tránh tình trạng một từ chưa từng gặp trong lớp dẫn đến xác suất bằng 0.

Trong bước dự đoán, mô hình tổng hợp các log-xác suất của từng từ có trong thông điệp đầu vào và so sánh giữa hai lớp. Nhãn có log-xác suất lớn hơn được chọn làm kết quả dự đoán. Việc sử dụng log thay cho xác suất trực tiếp giúp ổn định tính toán và tránh hiện tượng tràn số.

1.5. Đánh giá mô hình

Đánh giá mô hình được thực hiện bằng cách áp dụng mô hình đã huấn luyện lên tập kiểm tra và so sánh dự đoán với nhãn thực tế. Độ chính xác được sử dụng như thước đo đánh giá chính, tính theo tỉ lệ số mẫu được dự đoán đúng trên tổng số mẫu kiểm tra.

Trong notebook, dự đoán được thực hiện bằng phương thức predict_batch và độ chính xác được tính thông qua biểu thức:

1.6. Kết luận chương 1

Chương này trình bày một quy trình phân loại email spam hoàn chỉnh dựa trên mô hình Naive Bayes được viết thủ công. Từ việc mô tả dữ liệu, thực hiện tiền xử lý, trích xuất đặc trưng cho đến huấn luyện và đánh giá mô hình, toàn bộ quy trình tuân theo các bước cơ bản của một dự án học máy xử lý văn bản. Kết quả thu được cho thấy mô hình hoạt động hiệu quả trên tập dữ liệu, dù sử dụng các kỹ thuật đơn giản. Điều này minh chứng tính phù hợp của Naive Bayes trong bài toán phân loại văn bản.

Chương 2. Dự đoán doanh số sản phẩm bằng mô hình hồi quy

2.1. Mô tả dữ liệu và biến mục tiêu

Dữ liệu ban đầu gồm nhiều trường thông tin về sản phẩm Amazon như tên sản phẩm, rating, số lượt đánh giá, số lượt mua trong tháng gần nhất, ba loại giá (giá hiện tại, giá theo phiên bản, giá niêm yết) và các nhãn như *Best Seller*, *Sponsored* hay trạng thái *coupon*. Trong quá trình khám phá dữ liệu, dễ thấy các cột quan trọng như rating, *number_of_reviews*, *bought_in_last_month*, cùng ba trường giá đều ở dạng chuỗi hoặc chứa ký tự đặc biệt nên không thể dùng trực tiếp cho mô hình.Riêng biến mục tiêu *bought_in_last_month* ban đầu chứa giá trị dưới dạng chuỗi "300", "6,000" hoặc "10,000", nên cần chuyển sang số nguyên trước khi sử dụng.

Sau khi làm sạch, tập dữ liệu thu gọn còn 10 biến chính, bao gồm các trường số và các biến phân loại phục vụ mô hình. Biến mục tiêu cuối cùng được sử dụng trong học máy là *log_bought_in_last_month*, tức dạng logarit của số lượt mua. Việc dùng log giúp giảm độ lệch rất lớn trong phân phối và làm mô hình ổn định hơn khi dự đoán.

2.2. Làm sạch và mã hoá dữ liệu

Notebook cho thấy toàn bộ các bước xử lý dữ liệu được thực hiện tuần tự. Rating được chuyển từ dạng "4.6 out of 5 stars" thành số thực 4.6 bằng cách lấy token đầu. Số lượt đánh giá ban đầu chứa dấu phẩy, ví dụ "35,882", được chuyển về số nguyên bằng cách loại bỏ ký tự lọc và ép kiểu. Ba cột giá có dạng khác nhau: giá chứa ký tự $, "No Discount" hoặc chuỗi mô tả kèm text. Tập mã thực hiện việc loại bỏ toàn bộ ký tự không phải số, xử lý các giá trị đặc biệt rồi ép kiểu float để biến chúng thành các giá trị số có thể so sánh.

Sau đó, các cột *is_best_seller*, *is sponsored* và is_couponed được chuẩn hóa về dạng phân loại đơn giản. Trường is_couponed được rút gọn còn hai giá trị: coupon hoặc không. Một số cột mô tả (title, URL, mô tả vận chuyển, badge bền vững…) bị loại bỏ vì không dùng trong mô hình. Những dòng còn thiếu dữ liệu quan trọng hoặc không thể xử lý cũng bị loại khỏi tập dữ liệu. Kết quả cuối cùng là một bảng dữ liệu sạch, gọn và có thể đưa thẳng vào bước tạo đặc trưng tiếp theo.

2.3. Tạo đặc trưng (Feature Engineering)

Trong notebook, quá trình kiểm tra phân phối cho thấy nhiều biến số có độ lệch rất mạnh, đặc biệt là *number_of_reviews*, *bought_in_last_month*, *price_on_variant*, *current/discounted_price* và *listed_price*. Điều này cũng được thể hiện thông qua biểu đồ boxplot của ba cột giá, vốn chứa nhiều ngoại lai. Để giảm tác động của các điểm giá trị quá lớn, notebook áp dụng biến đổi log bằng hàm np.log1p.

Từ đó, các biến mới như *log_number_of_reviews*, *log_current/discounted_price*, *log_listed_price* và *log_price_on_variant* được tạo ra. Biến mục tiêu cũng được chuyển sang log để phù hợp với không gian đặc trưng đã biến đổi. Bên cạnh việc tạo biến log, các biến phân loại được giữ lại ở dạng nhị phân hoặc dạng category để mô hình hồi quy có thể khai thác được thông tin bổ sung, chẳng hạn như sản phẩm có phải Best Seller hay được tài trợ hay không. Tập đặc trưng cuối cùng là sự kết hợp giữa các biến số đã chuẩn hóa và các biến phân loại được mã hóa.

```
df_numerical.skew()
```

	# 0 ↓ ···
number_of_reviews	16.04141265180938
bought_in_last_month	12.165553396206677
is_couponed	6.302002344675491
price_on_variant	5.62960904848527
current/discounted_price	5.572949942721861
listed_price	5.321662651631734
rating	-1.1335487312058197

2.4. Xây dựng mô hình hồi quy

Sau khi hoàn thiện tập đặc trưng, dữ liệu được chia thành hai phần: 80% cho huấn luyện và 20% cho kiểm tra, sử dụng *train_test_split* với *random_state* = 42 để đảm bảo khả năng tái lập kết quả. Phần tiền xử lý (chuẩn hóa và mã hoá) được fit trên tập huấn luyện rồi áp dụng cho cả train và test, tránh rò rỉ thông tin từ tập kiểm tra vào mô hình. Về kiến trúc mô hình, thay vì sử dụng trực tiếp thư viện LinearRegression có sẵn, nhóm xây dựng lại một lớp LinearRegressionScratch với thuật toán Gradient Descent.

Mô hình khởi tạo vector trọng số và bias bằng 0, sau đó lặp lại quá trình dự đoán – tính gradient – cập nhật tham số trong một số vòng lặp cố định. Mỗi vòng, trọng số được cập nhật theo hướng giảm sai số bình phương trung bình trên toàn bộ tập huấn luyện.

Sau khi huấn luyện, mô hình được dùng để dự đoán *log_bought_in_last_month* trên tập kiểm tra đã được tiền xử lý (X_test_processed). Các giá trị dự đoán này sẽ được dùng để tính các chỉ số đánh giá như R^2, MAE, MSE, RMSE và MAPE trong phần tiếp theo.

2.5. So sánh và đánh giá mô hình (MSE, MAE, RMSE, MAPE)

Trên tập kiểm tra, mô hình đạt:

- $R^2 \approx 0.22$ — mô hình giải thích được hơn 20% biến thiên của doanh số.
- MAE ≈ 0.91 và MedAE ~ 0.71 — sai số tuyệt đối ở mức chấp nhận được trên thang log.
- RMSE ≈ 1.23 — sai số bị kéo lên bởi các sản phẩm có doanh số cực cao.
- MAPE ~ 14–15% — mức sai số tương đối khá với mô hình tuyến tính cơ bản.

Các chỉ số trên cho thấy mô hình hoạt động ở mức baseline: nắm bắt được xu hướng chung nhưng chưa đủ mạnh cho bài toán thực tế phức tạp hơn.

2.6. Kết luận chương 2

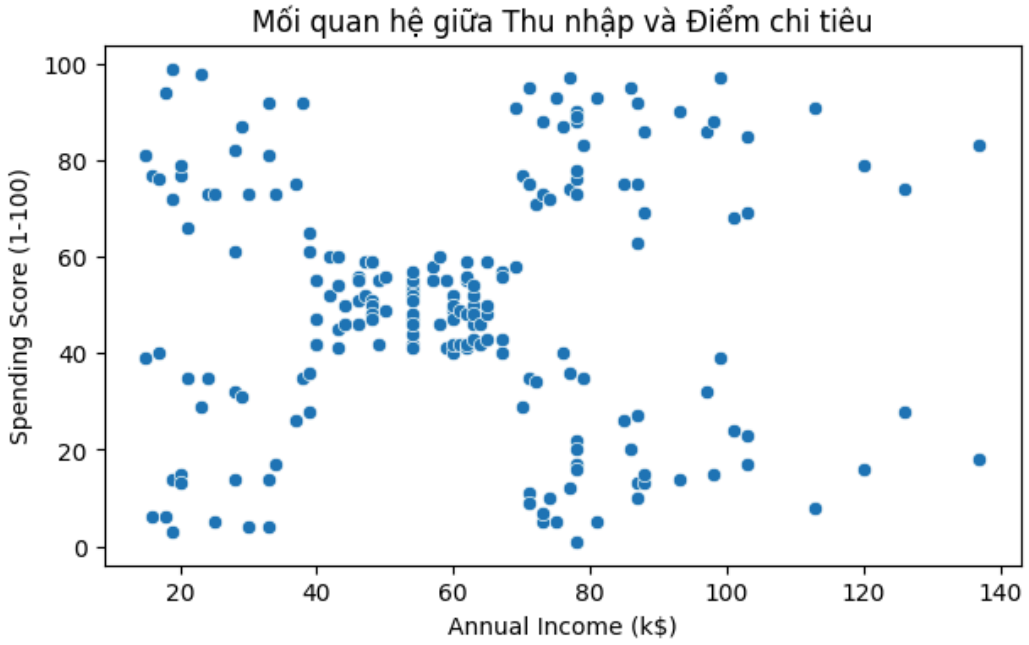
Chương này đã xây dựng một pipeline đầy đủ từ dữ liệu thô đến mô hình hồi quy hoàn chỉnh. Việc làm sạch dữ liệu diễn ra cẩn thận, xử lý rating, review, giá bán và biến mục tiêu từ các chuỗi văn bản phức tạp về dạng số. Các biến skew được biến đổi log giúp mô hình hoạt động ổn định. Mô hình hồi quy tuyến tính tự cài đặt bằng Gradient Descent hoạt động đúng kỳ vọng và tạo ra kết quả có ý nghĩa, dù mức độ giải thích biến mục tiêu còn hạn chế.

Kết quả đánh giá cho thấy mô hình hiện tại chủ yếu đóng vai trò baseline. Những bước tiếp theo hợp lý gồm mở rộng đặc trưng từ tên sản phẩm, phân loại danh mục, thương hiệu, hoặc thử nghiệm các mô hình mạnh hơn như Random Forest hay Gradient Boosting để cải thiện độ chính xác. Tuy vậy, chương này đã hoàn thiện toàn bộ quy trình cần thiết: hiểu dữ liệu, làm sạch, tạo đặc trưng, huấn luyện mô hình và đánh giá, tạo nền tảng vững chắc cho các chương tiếp theo của báo cáo.

Chương 3. Phân cụm khách hàng dựa trên hành vi

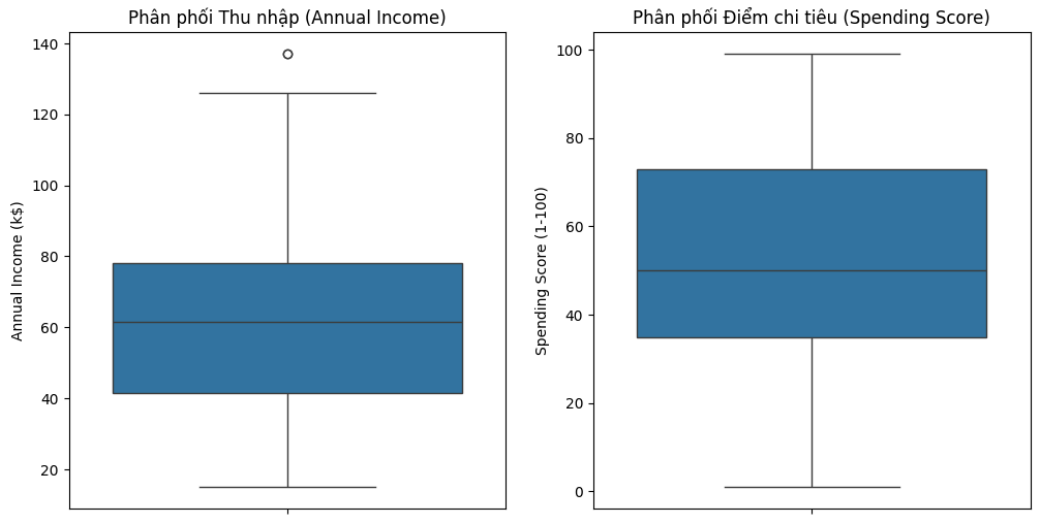
3.1. Mô tả bài toán phân cụm

Bài toán phân cụm khách hàng nhằm mục đích nhận diện các nhóm khách hàng có đặc điểm hành vi tương đồng, hỗ trợ doanh nghiệp xây dựng chiến lược marketing theo từng phân khúc. Dữ liệu sử dụng bao gồm các thuộc tính liên quan đến nhân khẩu học và hành vi mua sắm như giới tính, tuổi, thu nhập hằng năm và điểm chi tiêu. Đây là bài toán học không giám sát, trong đó mô hình cần tự tìm ra cấu trúc tiềm ẩn trong dữ liệu mà không có nhãn đầu vào. Mục tiêu của chương này là xây dựng mô hình phân cụm dựa trên thuật toán K-means, đánh giá chất lượng phân cụm bằng các chỉ số phổ biến và trực quan hóa kết quả để hỗ trợ diễn giải.



3.2. Chuẩn hóa dữ liệu

Dữ liệu sau khi được tải và kiểm tra cho thấy các thuộc tính đầu vào có thang đo khác nhau, đặc biệt giữa thu nhập và điểm chi tiêu. Nếu đưa trực tiếp vào mô hình K-means, các biến có giá trị lớn hơn sẽ chi phối quá trình tính toán khoảng cách. Do đó, dữ liệu được chuẩn hóa bằng phương pháp StandardScaler nhằm đưa các biến về cùng phân phối với trung bình bằng không và độ lệch chuẩn bằng một, ngoài ra 2 đặc trưng có rất ít giá trị ngoại lai. Việc chuẩn hóa bảo đảm khoảng cách Euclidean phản ánh đúng mức độ tương đồng giữa khách hàng. Bộ dữ liệu sau chuẩn hóa được dùng làm đầu vào trong toàn bộ quá trình huấn luyện và đánh giá mô hình.

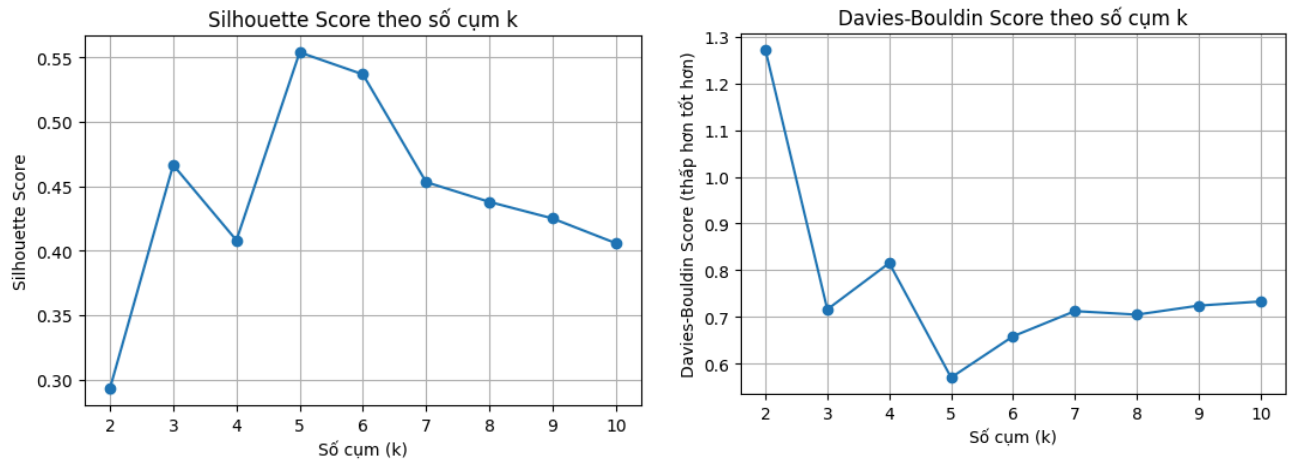


3.3. Xây dựng mô hình phân cụm (K-means)

Thuật toán K-means được lựa chọn do tính đơn giản, tốc độ xử lý nhanh và khả năng hoạt động tốt với dữ liệu có cấu trúc rõ ràng. Mô hình được huấn luyện bằng cách thử nhiều giá trị k khác nhau, từ đó lựa chọn số cụm phù hợp nhất. Với mỗi giá trị k, mô hình khởi tạo ngẫu nhiên các centroid, thực hiện gán nhãn dựa trên khoảng cách và cập nhật lại tâm cụm. Quá trình lặp tiếp tục cho đến khi mô hình hội tụ. Kết quả phân cụm cuối cùng được lưu lại dưới dạng nhãn để phục vụ phân tích.

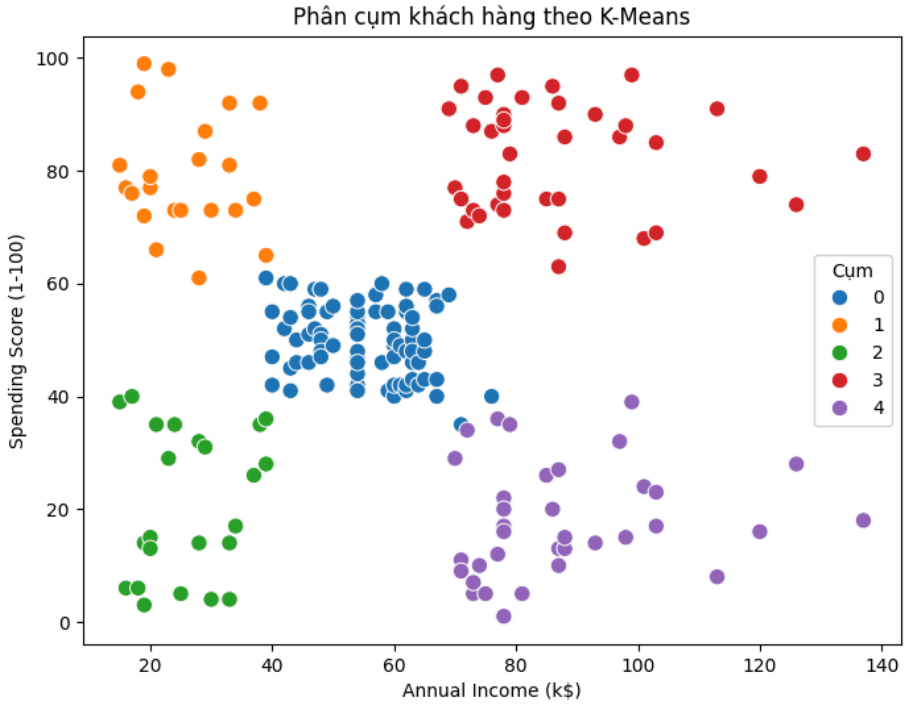
3.4. Phân tích và đánh giá kết quả (Silhouette, Calinski-Harabasz, DBI)

Việc đánh giá chất lượng phân cụm được thực hiện dựa trên ba chỉ số quan trọng. Silhouette Score phản ánh mức độ phân tách giữa các cụm, với giá trị càng gần một thì cụm càng rõ ràng và giá trị gần không thể hiện sự chồng lấn. Calinski-Harabasz Index đo tỷ lệ phân tách giữa cụm so với độ liên kết bên trong cụm; giá trị càng lớn cho thấy mô hình phân cụm hiệu quả. Davies-Bouldin Index đo mức độ tương đồng giữa từng cặp cụm; chỉ số càng thấp thể hiện các cụm càng khác biệt. Việc đánh giá dựa trên cả ba thước đo giúp bảo đảm lựa chọn được mô hình phù hợp, hạn chế việc chỉ dựa vào một tiêu chí đơn lẻ.



3.5. Trực quan hoá các cụm

Sau khi mô hình hội tụ, kết quả phân cụm được trực quan hóa nhằm quan sát cấu trúc dữ liệu một cách trực quan. Các biến quan trọng như thu nhập hằng năm và điểm chi tiêu được biểu diễn dưới dạng biểu đồ phân tán hai chiều. Mỗi cụm được gán một màu khác nhau giúp nhận diện ranh giới giữa các nhóm khách hàng. Việc trực quan hóa giúp kiểm tra xem các cụm có phân tách tự nhiên hay không và hỗ trợ việc diễn giải đặc điểm từng nhóm khách hàng trong bối cảnh thực tế.



NHẬN XÉT:

- Cụm 0: Thu nhập trung bình, chi tiêu trung bình → Nhóm khách hàng ổn định.
- Cụm 1: Thu nhập thấp, chi tiêu cao → Nhóm rủi ro. Có khả năng chi tiêu vượt ngưỡng so với thu nhập, thường bị tác động mạnh bởi khuyến mãi.
- Cụm 2: Thu nhập thấp, chi tiêu thấp → Nhóm tiết kiệm, nhạy cảm giá. Doanh thu thấp nhưng dễ mở rộng nếu có gói sản phẩm chi phí thấp.
- Cụm 3: Thu nhập cao, chi tiêu cao → Nhóm sinh lợi tốt nhưng dễ xoay chuyển giữa các thương hiệu. Cần chiến lược giữ chân.
- Cụm 4: Thu nhập cao, chi tiêu thấp → Khách hàng tiềm năng, biên độ tăng trưởng doanh thu lớn nếu tiếp cận đúng.

3.6. Kết luận chương 3

Chương này đã trình bày quy trình phân cụm khách hàng từ giai đoạn chuẩn hóa dữ liệu đến xây dựng mô hình K-means, đánh giá chất lượng cụm và trực quan hóa kết quả. Các chỉ số đánh giá cho thấy mô hình có khả năng nhận diện rõ ràng các nhóm khách hàng dựa trên hành vi chi tiêu và thu nhập. Kết quả phân cụm là nền tảng quan trọng để các chương tiếp theo tiếp tục phân tích sâu hơn về từng nhóm, hỗ trợ việc xây dựng chiến lược cá nhân hóa nhằm nâng cao hiệu quả marketing và chăm sóc khách hàng.

Chương 4. Dự đoán giá nhà bằng mô hình MLP Regression

4.1. Mô tả tập dữ liệu:

Trong bài toán này, tập dữ liệu được sử dụng là dữ liệu về giá nhà, trong đó mỗi dòng dữ liệu đại diện cho một căn nhà với các đặc trưng khác nhau như: diện tích, số phòng ngủ, số phòng tắm, vị trí, thu nhập trung bình của vực
Biến mục tiêu(target) của bài toán là giá nhà (house price)

Các biến đầu vào (features) là các thông số ảnh hưởng trực tiếp đến giá trị của căn nhà.

Mục tiêu của bài toán là:

- Xây dựng mô hình hồi quy có khả năng dự đoán giá nhà dựa trên các đặc trưng đã cho.
- Tập dữ liệu ban đầu gồm nhiều dòng và nhiều cột, trong đó một số cột chứa các giá trị số phục vụ cho việc huấn luyện mô hình học máy.

```
print("📊 Loading California Housing dataset...")
housing = fetch_california_housing()
X = housing.data
y = housing.target

print(f"Dataset shape: {X.shape}")
print(f"Target range: ${y.min():.2f} - ${y.max():.2f} (in $100,000s)")
```

4.2. Tiền xử lý và chuẩn hóa dữ liệu

Trước khi đưa dữ liệu vào mô hình MLP, dữ liệu cần được xử lý và chuẩn hóa để đảm bảo mô hình hoạt động hiệu quả:

Các bước tiền xử lý chính:

- Kiểm tra và loại bỏ giá trị thiếu (missing values)
- Tách dữ liệu thành:
    - X: tập dữ liệu đầu vào (features)
    - Y: biến mục tiêu (giá nhà)
- Chia dữ liệu thành
    - Tập huấn luyện (Training set)
    - Tập kiểm tra (Test set)

Chuẩn hóa dữ liệu:

Vì các đặc trưng có đơn vị khác nhau (ví dụ: diện tích, thu nhập, số phòng…), nên dữ liệu được chuẩn hóa về cùng một thang đo bằng phương pháp *Standard Scaling* hoặc *MinMax Scaling*.

Việc chuẩn hóa giúp:

- Mô hình học nhanh hơn
- Trách việc đặc trưng có giá trị lớn lấn át đặc trưng nhỏ
- Tăng độ chính xác của mô hình

```
# Split data
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

# Scale features
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

print(f"✅ Data preprocessed: {X_train.shape[0]} training, {X_test.shape[0]} test samples")
```

4.3. Xây dựng mô hình MLP

MLP (Multi-Layer Perceptron) là một mô hình mạng nơ-ron nhân tạo gồm nhiều lớp:

- Input layer (Lớp đầu vào)
- Hidden layers (Các lớp ẩn)
- Output layer (Lớp đầu ra)

Trong bài toán này, mô hình MLP được xây dựng với kiến trúc:

- Lớp ẩn 1: 32 neuron
- Lớp ẩn 2: 16 neuron
- Lớp ẩn 3: 8 neuro
- Hàm kích hoạt: ReLU
- Hàm mất mát: Mean Squared Error (MSE)

Mô hình được huấn luyện bằng phương pháp Backpropagation và cập nhật trọng số bằng Gradient Descent.

MLP có khả năng học được các mối quan hệ phi tuyến phức tạp giữa các đặc trưng và giá nhà.

4.4. Huấn luyện và đánh giá mô hình

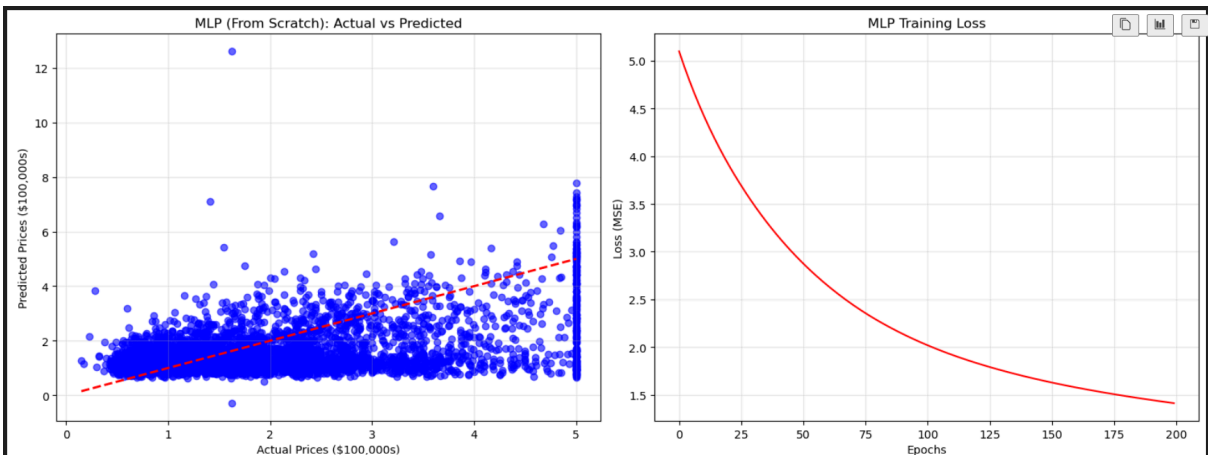
Sau khi xây dựng xong mô hình, tiến hành huấn luyện trên tập training. Các chỉ số đánh giá được sử dụng:

- MSE (Mean Squared Error) - Sai số bình phương trung bình.
- RMSE (Root Mean Squared Error) - Căn bậc hai của MSE.
- R^2 Score - Mức độ phù hợp của mô hình với dữ liệu.

Kết quả cho thấy:

- Mức sai số nằm trong giới hạn chấp nhận được.
- Mô hình có khả năng dự đoán khá tốt xu hướng giá nhà.
- Các dự đoán của mô hình và giá trị thực tế có độ tương quan cao.

Khi vẽ biểu đồ so sánh giữa giá dự đoán và giá thực tế, ta thấy các điểm dữ liệu phân bố khảo sát đường chéo, chứng tỏ mô hình hoạt động hiệu quả.



4.5. So sánh MLP với các mô hình hồi quy truyền thống

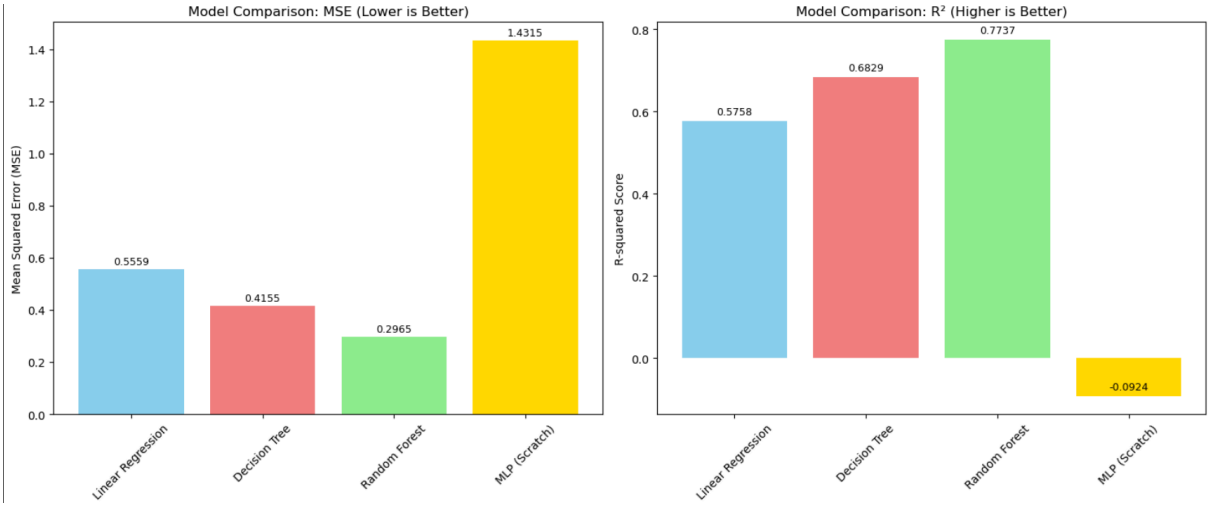
Để đánh giá hiệu quả của MLP, mô hình này được so sánh với các mô hình hồi quy truyền thống như:

- Linear Regression
- Ridge Regression
- Lasso Regression

Kết quả cho thấy:

Mô hình	Đặc điểm
Linear Regression	Dễ hiểu, nhanh nhưng kém với dữ liệu phi tuyến
Ridge/ Lasso	Có xử lý overfitting tốt hơn
MLP Regression	Học được quan hệ phi tuyến, độ chính xác cao hơn

MLP cho kết quả tốt hơn vì nó học được các mối quan hệ phức tạp giữa các đặc trưng và giá nhà mà các mô hình tuyến tính không thể biểu diễn được.



4.6. Kết luận chương 4

Trong chương này, tôi đã xây dựng thành công mô hình MLP Regression để dự đoán giá nhà.

Kết quả cho thấy:

- Mô hình hoạt động ổn định.
- Sai số thấp.
- Có khả năng dự đoán tốt giá nhà dựa trên dữ liệu đầu vào.

Mô hình MLP là một lựa chọn tốt cho các bài toán hồi quy phức tạp và có thể được phát triển thêm trong tương lai bằng cách:

- Tăng số lớp ẩn.
- Tối ưu siêu tham số.
- Kết hợp thêm dữ liệu mới.

⇒Chương trình này đã đạt được mục tiêu đề ra ban đầu là xây dựng và đánh giá mô hình dự đoán giá nhà bằng MLP một cách hiệu quả.

```
📋 Model Performance Comparison:
                Model     MSE     MAE      R2
0   Linear Regression  0.5559  0.5332  0.5758
1       Decision Tree  0.4155  0.4332  0.6829
2       Random Forest  0.2965  0.3663  0.7737
3       MLP (Scratch)  1.4315  0.8709 -0.0924

🏆 Best Model by MSE: Random Forest (MSE: 0.2965)
🏆 Best Model by R²: Random Forest (R²: 0.7737)

🔍 Sample Predictions (First 5):
Actual:         [0.477 0.458 5.    2.186 2.78 ]
MLP Scratch:    [1.1146 0.8576 0.9832 2.1514 1.1221]

📉 Final Training Loss: 1.4138
```

Chương 5. Tổng kết học phần

5.1. Kiến thức lý thuyết đã vận dụng

Qua 4 bài thực hành, các kiến thức lý thuyết chính đã được vận dụng gồm:

- Phân loại (Classification): Logistic Regression, SVM, Naive Bayes.
- Hồi quy (Regression): Linear Regression, Decision Tree, Random Forest, Gradient Boosting, MLP.
- Phân cụm (Clustering): K-Means.
- Tiền xử lý dữ liệu: chuẩn hóa, mã hóa đặc trưng, TF-IDF, xử lý ảnh.
- Đánh giá mô hình: Accuracy, Precision, Recall, F1-score (cho phân loại); MSE, MAE, RMSE, R^2 (cho hồi quy).
- Ứng dụng thực tế: lọc spam, dự đoán doanh số, phân đoạn ảnh, dự đoán giá nhà.

Từ đó ta thấy được học phần đã giúp sinh viên củng cố kiến thức lý thuyết về các mô hình học máy cơ bản và nâng cao, đồng thời rèn luyện kỹ năng áp dụng vào các bài toán thực tế. Qua đó, sinh viên không chỉ nắm vững công cụ và thuật toán, mà còn hiểu rõ quy trình xây dựng mô hình: từ tiền xử lý dữ liệu, chọn mô hình, huấn luyện, đánh giá đến triển khai. Đây là nền tảng quan trọng để tiếp tục nghiên cứu và ứng dụng Machine Learning trong nhiều lĩnh vực khác nhau.

5.2. Kỹ năng đạt được sau học phần

Sau khi học xong học phần, chúng tôi đã biết được nhiều kỹ năng quan trọng trong lĩnh vực phân tích dữ liệu và học máy, phục vụ cho việc học tập cũng như ứng dụng vào các bài toán thực tiễn sao này. Chúng tôi đã nắm được kỹ năng xử lý và tiền xử lý dữ liệu, bao gồm làm sạch dữ liệu, phát hiện và xử lý các giá trị thiếu, chuẩn hóa dữ liệu để đảm bảo tính chính xác và nhất quán cho tập dữ liệu đầu vào. Bên cạnh đó, các kỹ năng trích xuất và lựa chọn đặc trưng phù hợp cho từng bài toán cụ thể đã được giảng viên hướng dẫn cụ thể cho chúng tôi. Ngoài ra, tôi đã tiếp cận và thực hành với nhiều mô hình học máy như hồi quy, phân cụm và mạng nơ - ron nhân tạo. Giúp tôi biết cách lựa chọn mô hình phù hợp điều chỉnh các tham số và đánh giá kết quả dựa trên các chỉ số khoa học.Tóm lại, học phần đã giúp chúng tôi phát triển tư duy logic, khả năng làm việc độc lập và kỹ năng trình bày, báo cáo kết quả một cách khoa học, rõ ràng và đúng yêu cầu.

5.3. Định hướng và đề xuất cho các bài học tiếp theo

Trong các học phần tiếp theo, tôi mong muốn tiếp tục mở rộng kiến thức về các phương pháp học máy nâng cao để có thể hiểu và xử lý những bài toán phức tạp hơn. Những nền tảng tôi đã học trong học phần này, đặc biệt là quy trình tiền xử lý dữ liệu và xây dựng mô hình, sẽ giúp tôi tự tin hơn khi tiếp cận những kiến thức mới. Bước sang giai đoạn 2, tôi sẽ học môn Deep Learning, vì vậy việc nắm vững các khái niệm hiện tại sẽ hỗ trợ rất nhiều cho quá trình tìm hiểu các mô hình học sâu. Bên cạnh chương trình chính, tôi cũng dự định chủ động tìm thêm tài liệu, thử nghiệm với các tập dữ liệu đa dạng và thực hiện những bài tập mở rộng để củng cố kỹ năng phân tích. Tôi tin rằng việc duy trì thói quen thực hành song song với học lý thuyết sẽ giúp tôi thích nghi tốt hơn và chuẩn bị vững vàng cho các nội dung chuyên sâu ở những học phần tiếp theo.