TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

ĐỒ ÁN TỐT NGHIỆP

Thực nghiệm và so sánh các thuật toán lựa chọn đặc trưng có giám sát và không giám sát trong đánh giá điểm tín dụng cá nhân

TRẦN VĂN TRÍ
tri.tv173410@sis.hust.edu.vn

Ngành Khoa học máy tính
Chuyên ngành Khoa học máy tính

Giảng viên hướng dẫn: TS. Nguyễn Khánh Phương

Chữ ký của GVHD

Bộ môn: Khoa học máy tính
Viện: Công nghệ thông tin và truyền thông

HÀ NỘI, 2/2022

ĐỀ TÀI TỐT NGHIỆP

Biểu mẫu của Đề tài/khóa luận tốt nghiệp theo qui định của viện, tuy nhiên cần đảm bảo giáo viên giao đề tài ký và ghi rõ họ và tên.

Trường hợp có 2 giáo viên hướng dẫn thì sẽ cùng ký tên.

Giáo viên hướng dẫn
Ký và ghi rõ họ tên

Lời cảm ơn

Trải qua thời gian dài khó khăn, vất vả nhưng đáng giá tại trường Đại học Bách Khoa Hà Nội, tôi rất vui khi đã hoàn thành chương trình học tập của mình. Lời đầu tiên tôi xin cảm ơn các thầy cô đã chỉ dạy, mang đến cho tôi những kiến thức cần thiết trong suốt thời gian học tập tại trường. Tôi xin trân trọng cảm ơn TS. Nguyễn Khánh Phương và TS. Bùi Quốc Trung đã hướng dẫn tôi trong quá trình tìm hiểu, nghiên cứu, thực hành và hoàn thiện đồ án tốt nghiệp này. Tôi cũng xin cảm ơn tất cả bạn bè và gia đình đã luôn ủng hộ, giúp đỡ tôi, tạo điều kiện tốt nhất để tôi hoàn thành chương trình học tại Đại học Bách Khoa Hà Nội. Xin gửi lời cảm ơn đặc biết đến Phương Lê, người con gái luôn ở bên anh và cùng anh vượt qua khó khăn trong thời gian qua.

Tóm tắt nội dung đồ án

Ứng dụng học máy đang ngày càng trở nên phổ biến, xuất hiện trong nhiều lĩnh vực đời sống. Lựa chọn đặc trưng là một bước quan trọng trong lĩnh vực học máy, với mục đích là loại bỏ các đặc trưng không liên quan và đặc trưng dư thừa. Các đặc trưng này không những không có ý nghĩa với bản chất việc học mà còn ảnh hưởng đến hiệu năng học tập của mô hình. Lựa chọn đặc trưng làm giảm kích thước dữ liệu đồng thời tăng tốc, tăng hiệu năng quá trình học tập của mô hình. Tuy nhiên bài toàn lựa chọn đặc trưng là NP-khó nên đòi hỏi phải có các phương pháp phù hợp để có được kết quả tối ưu.

Ngày càng nhiều phương pháp lựa chọn đặc trưng được giới thiệu, phát triển và áp dụng. Các phương pháp, thuật toán lựa chọn đặc trưng hiện nay đều có các ưu điểm riêng, áp dụng với các bài toán cụ thể, tuy nhiên chúng đều gặp phải các vấn đề về thời gian tính toán cũng như tối ưu cục bộ nên hiệu năng của mô hình phân loại với dữ liệu là kết quả của các thuật toán lựa chọn đặc trưng là không ổn định. Hơn nữa để chọn được một phương pháp lựa chọn đặc trưng phù hợp với mục đích sử dụng và đem lại kết quả tin cậy là vấn đề rất khó. Các phương pháp lựa chọn đặc trưng dựa trên sự tương tự của dữ liệu và dựa trên lý thuyết thông tin là các phương pháp dễ dàng triển khai và chi phí tính toán thấp đang được áp dụng rộng rãi.

Trong đồ án này tôi thực hiện nghiên cứu, tái tạo, thực nghiệm và so sánh các thuật toán lựa chọn đặc trưng sử dụng điểm số dựa trên sự tương tự của dữ liệu (5 thuật toán) và dựa trên lý thuyết thông tin (5 thuật toán). Sau đó sử dụng mô hình phân loại để đánh giá chi phí, hiệu năng của các thuật toán này. Việc thực nghiệm được thực hiện trên tám bộ dữ liệu thực tế trong đánh điểm giá tín dụng cá nhân, lĩnh vực tài chính – ngân hàng. Kết quả cho thấy các thuật toán sử dụng điểm số dựa trên sự tương tự dễ dàng triển khai và thời gian tính toán thấp. Các thuật toán sử dụng điểm số dựa trên lý thuyết thông tin cho hiệu năng ổn định và chi phí bộ nhớ thấp hơn.

Sinh viên thực hiện
Ký và ghi rõ họ tên

MỤC LỤC

DANH MỤC HÌNH VẼ

DANH MỤC BẢNG

DANH SÁCH TỪ VIẾT TẮT VÀ THUẬT NGỮ

Từ viết tắt	Từ gốc	Nghĩa
FS	Feature Selection	Lựa chọn đặc trưng
FE	Feature Extraction	Trích chọn đặc trưng
Fintech	Financial Technology	Công nghệ tài chính
GA	Genetic Algorithm	Giải thuật di truyền
SFS	Sequential Forward Search	Tìm kiếm tuần tự tiến
SBS	Sequential Backward Search	Tìm kiếm tuần tự lùi
SBE	Sequential Backward Elimination	Loại bỏ tuần tự lùi
SFS	Sequential Feature Selection	Lựa chọn đặc trưng tuần tự
SBFS	Sequential Backward Floating Selection	Lựa chọn tuần tự tiến động
SFFS	Sequential Forward Floating Selection	Lựa chọn tuần tự lùi động
SVM	Suport Vector Machine	Máy véc tơ hỗ trợ
PSO	Particle Swarm Optimization	Tối ưu bầy đàn
EC	Evolutionary Computing	Kĩ thuật tính toán tiến hóa
ACO	Ant Colony Optimization	Tối ưu đàn kiến
LASSO	LASSO	Phương pháp hồi quy LASSO
Decision trees	Decision trees	Thuật toán cây quyết định
SPEC	Spectral Feasture Selection	Thuật toán lựa chọn đặc trưng SPEC
MIM	Mutual Information Maximization	Điểm số cực đại thông tin tương hỗ

MIFS	Mutual Information Feature Selection	Điểm số lựa chọn đặc trưng thông tin tương hỗ
MRMR	Minimum Redundancy Maximum Relevance	Điểm số cực tiểu dư thừa cực đại liên quan
CIFE	Conditional Infomax Feature Extraction	Điểm số cực đại thông có điều kiện
JMI	Joint Mutual Information	Điểm số thông tin tương hỗ chung
AUC	Area Under the Curve	Chỉ số AUC
ROC	Receiver Operating Characteristics	Đường cong biểu diễn xác suất ROC
AUROC	Area Under the Receiver Operating Characteristics	Chỉ số AUC
TP	True Positive	Số mẫu dương tính (nhãn 1) thật
FP	False Positive	Số mẫu dương tính giả
TN	True Negative	Số mẫu âm tính (nhãn 0) thật
FN	False Negative	Số mẫu âm tính giả

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

Chương này trước tiên trình bày khái niệm và vị trí của lựa chọn đăc trưng trong bài toán học máy nói chung và quá trình tiền xử lí dữ liệu nói riêng. Sau đó là phạm vi, đóng góp và cấu trúc của đồ án.

1.1 Quá trình tiền xử lí dữ liệu

Tiền xử lí dữ liệu là giai đoạn rất quan trọng trong việc giải quyết bất kỳ vấn đề nào trong lĩnh vưc học máy. Hầu hết các bộ dữ liệu được sử dụng trong các vấn đề liên quán đến học máy cần được xử lý, làm sạch và biến đổi trước khi một thuật toán học máy có thể được huấn luyện trên những bộ dữ liệu này.

Quá trình tiền xử lí dữ liệu bao gồm việc làm sạch dữ liệu, biến đổi dữ liệu và giảm kích thước dữ liệu. Dữ liệu thu thập được trong nhiều tình huống rất lớn, dư thừa không cần thiết. Dữ liệu dư thừa gây nhiễu trong việc học dẫn tới mô hình bị overfit (học tủ) cũng như không học được cái bản chất, do đó việc làm giảm kích thước của dữ liệu là cần thiết. Giảm kích thước dữ liệu được thực hiện bằng cách giảm số lượng bản ghi hoặc giảm số lượng đặc trưng hoặc đồng thời.

Để giảm số lượng đặc trưng, có hai phương pháp đó là Lựa chọn đặc trưng FS (Feature Selection) và Trích chọn đặc trưng FE (Feature Extraction). Trích chọn đặc trưng là suy ra các đặc trưng mới từ tập các đặc trưng ban đầu, trong khi lựa chọn đặc trưng là chọn lựa một số đặc trưng từ toàn bộ tập đặc trưng. Kết quả của lựa chọn đặc trưng luôn dễ dàng diễn giải hơn, do đó đây là phương pháp được quan tâm và ứng dụng nhiều hơn.

1.2 Lựa chọn đặc trưng

Lựa chọn đặc trưng là quá trình loại bỏ các đặc trưng không liên quan, đặc trưng dư thừa ra khỏi tập dữ liệu ban đầu, chỉ giữ lại và không thay đổi giá trị của các đặc trưng liên quan. Kết quả của lựa chọn đặc trưng là tập hợp con gồm các đặc trưng (sau đây gọi tắt là tập con đặc trưng) được lựa chọn.

1.3 Phạm vi của đồ án

Đồ án này tập trung tìm hiểu, cài đặt, thực nghiệm và so sánh các thuật toán lựa chọn đặc trưng dựa trên các tiêu chí về thời gian tính toán, chi phí tài nguyên bộ nhớ và hiệu năng. Cụ thể là các thuật toán lựa chọn đặc trưng sử dụng điểm số để đánh giá đặc trưng hoặc tập con đặc trưng. Các điểm số thuộc hai nhóm là dựa trên sự tương tự của dữ liệu và dựa trên lý thuyết thông tin. Các thuật toán sử dụng điểm số dựa trên sự tương tự của dữ liệu được thực nghiệm gồm các thuật toán sử dụng điểm số Laplacian, Fisher, SPEC, Trace Ratio và ReliefF. Các thuật toán sử dụng điểm số dựa trên lý thuyết thông tin gồm các thuật toán sử dụng điểm số MIM, MIFS, MRMR, CIFE và JMI. Các thuật toán được thực nghiệm trên tám bộ dữ liệu thực tế trong bài toán đánh giá điểm tín dụng cá nhân thuộc lĩnh vực tài chính – ngân hàng.

1.4 Đóng góp của đồ án

Đồ án đã giới thiệu các thuật toán lựa chọn đặc trưng sử dụng điểm số dựa trên sự tương tự của dữ liệu và dựa trên lý thuyết thông tin, diễn giải thông qua các ví dụ đơn giản.

Thực hiện cài đặt lại và thực nghiệm các thuật toán này trên các bộ dữ liệu trong bài toán chấm điểm tín dụng cá nhân thuộc lĩnh vực tài chính – ngân hàng. Sau đó đưa ra các phân tích và nhận xét về các thuật toán dựa trên các tiêu chí về thời gian tính toán, chi phí bộ nhớ và hiệu năng của mô hình phân loại với dữ liệu là kết quả của các thuật toán lựa chọn đặc trưng.

1.5 Cấu trúc của đồ án

Đồ án gồm các nội dung chính sau:

Chương 1 trình bày về khái niệm và vị trí của lựa chọn đặc trưng, phạm vi và đóng góp của đồ án. Chương 2 trình bày các kiến thức về lựa chọn đặc trưng, các hướng tiếp cận, các hướng nghiên cứu lựa chọn đặc trưng. Chương 3 giới thiệu và minh họa các thuật toán lựa chọn đặc trưng dựa trên sự tương tự của dữ liệu, bao gồm các công thức và ví dụ minh họa. Chương 4 giới thiệu và minh họa các thuật toán lựa chọn đặc trưng dựa trên lý thuyết thông tin, bao gồm các công thức và ví dụ minh họa. Chương 5 trình bày quá trình thực nghiệm các thuật toán với dữ liệu công nghệ tài chính, đưa ra kết quả thực nghiệm và phân tích các kết quả. Cuối cùng, chương 6 đưa ra kết luận và các hướng có thể phát triển của đồ án trong tương lai. Các kết quả chi tiết được trình bày trong phần Phụ lục.

CHƯƠNG 2. TỔNG QUAN VỀ LỰA CHỌN ĐẶC TRƯNG

Chương này sẽ trình bày chi tiến các kiến thức được sử dụng trong đồ án. Bao gồm các kiến thức về lựa chọn đặc trưng, các hướng tiếp cận bài toán lựa chọn đặc trưng và các hướng nghiên cứu về lựa chọn đặc trưng

2.1 Giới thiệu bài toán lựa chọn đặc trưng

2.1.1 Lựa chọn đặc trưng là gì?

Đối với bài toán lựa chọn đặc trưng, các đặc trưng được phân loại thành: đặc trưng liên quan (Relevant features), đặc trưng không liên quan (Irrelevant features) và đặc trưng dư thừa (Redundant features) [1]

- **Đặc trưng liên quan l**à các đặc trưng quan trọng, có liên quan đến nhãn và bài toán phân loại, có ý nghĩa đối với mô hình học tập.
- **Đặc trưng không liên quan** là các đặc trưng có thể không liên quan đến nhãn và đến mục đích phân loại, không cải thiện độ chính xác cho mô hình học tập. Việc đưa các đặc trưng này vào làm giảm tốc độ học tập, thậm chí giảm độ chính xác của mô hình.
- **Đặc trưng dư thừa** là các đặc trưng mà có thể suy ra từ một hoặc nhiều đặc trưng khác, về cơ bản chúng đem lại thông tin tương tự như các đặc trưng khác. Các đặc trưng này là dư thừa, không cần thiết.

Lựa chọn đặc trưng là quá trình loại bỏ các đặc trưng không liên quan, đặc trưng dư thừa ra khỏi tập dữ liệu ban đầu, chỉ giữ lại và không thay đổi giá trị của các đặc trưng liên quan. Kết quả của lựa chọn đặc trưng là tập hợp con gồm các đặc trưng (sau đây gọi tắt là tập con đặc trưng) được lựa chọn.

2.1.2 Bài toán lựa chọn đặc trưng

Cho tập dữ liệu gồm n bản ghi và d đặc trưng được mô tả bởi ma trận $X^{n\times d}$ gồm n hàng và d cột. $f_1, f_2, \dots f_d$ là các vecto cột biểu diễn các đặc trưng. Yêu cầu lựa chọn ra k đặc trưng liên quan ($k < d$) mà không làm thay đổi các đặc trưng đó.

$$X = (f_1 \quad f_2 \quad \dots \quad f_i \quad \dots \quad f_d)$$

Lựa chọn đặc trưng

$$(f_1 \quad f_2 \quad \dots \quad f_k)$$

Hình 2.1 Bài toán lựa chọn đặc trưng

2.1.3 Mục đích lựa chọn đặc trưng

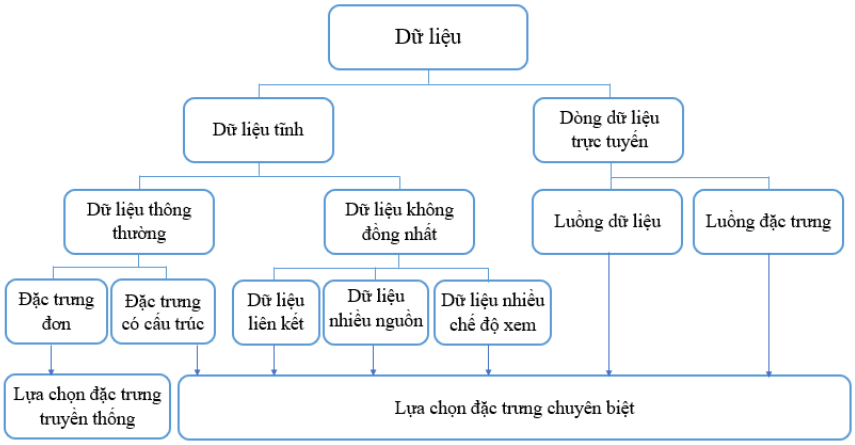
Theo S. L. Ladha và cộng sự [2], lựa chọn đặc trưng nhằm mục đích:

- Làm giảm kích thước dữ liệu, giúp tăng tốc độ học tập của mô hình;
- Tiết kiệm chi phí tài nguyên và chi phí tính toán;
- Loại bỏ đặc trưng không liên quan, giúp cải thiện độ chính xác của mô hình;
- Cải thiện chất lượng dữ liệu;

- Tiết kiệm tài nguyên trong vòng thu thập dữ liệu tiếp theo hoặc trong quá trình sử dụng;
- Hiểu biết về dữ liệu để có được kiến thức về quá trình tạo ra dữ liệu hoặc trực quan hóa dữ liệu;

2.1.4 Lựa chọn đặc trưng với các dữ liệu khác nhau

Lựa chọn đặc trưng phụ thuộc rất nhiều vào dữ liệu. Các dạng dữ liệu khác nhau có thể cần đến các phương pháp lựa chọn đặc trưng khác nhau. Hình vẽ sau đây phân loại các dạng dữ liệu phổ biến hiện nay [3]:



Hình 2.2 Lựa chọn đặc trưng dựa trên dữ liệu khác nhau

Có thể tiếp tục phân loại bài toán lựa chọn đặc trưng dựa trên thông tin về dữ liệu. Nếu dựa trên việc biết nhãn lớp ta có: lựa chọn đặc trưng có giám sát, không giám sát và bán giám sát. Nếu dựa trên số lượng nhãn lớp ta có: lựa chọn đặc trưng lớp nhị phân và lựa chọn đặc trưng đa lớp…

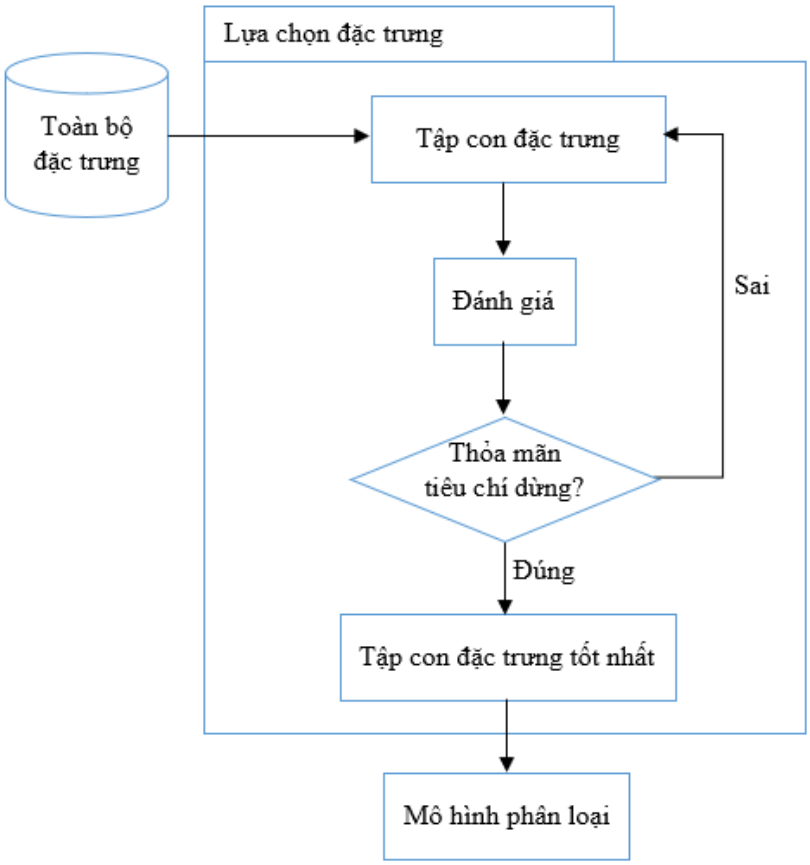
Đồ án này chỉ tập trung vào lựa chọn đặc trưng truyền thống áp dụng với dữ liệu thông thường có các đặc trưng đơn lẻ, áp dụng cho cả bài toán có giám sát và không giám sát.

2.2 Mô hình chung của của bài toán lựa chọn đặc trưng

2.2.1 Mô hình chung

Các thuật toán lựa chọn đặc trưng thường bao gồm bốn thành phần chính: Chiến lược tìm kiếm (Search strategy), hướng tìm kiếm (Search direction), tiêu chí đánh giá (Evaluation criterion) và tiêu chí dừng (Stop criterion) [4].

Hình vẽ sau thể hiện mô hình chung của các phương pháp lựa chọn đặc trưng:



Hình 2.3 Mô hình chung của các phương pháp lựa chọn đặc trưng

2.2.2 Chiến lược tìm kiếm

Không gian tìm kiếm của tập dữ liệu ban đầu là rất lớn. Với tập dữ liệu có n đặc trưng, ta có 2^n tập con. Việc khảo sát hết tất cả các tập con đặc trưng này dẫn đến chi phí vô cùng thâm chí là không thể. Do đó chúng ta cần xác định chiến lược tìm kiếm để chọn được tập đặc trưng tối ưu một cách hiệu quả nhất.

Các chiến lược tìm kiếm có thể là: tìm kiếm vét cạn (Exhausted Search), tìm kiếm tuần tự (Sequential Search) hoặc tìm kiếm ngẫu nhiên (Random Search)

2.2.2.1. Tìm kiếm vét cạn (Exhausted Search)

Tìm kiếm vét cạn là phương pháp tìm kiếm xem xét tất cả các giải pháp có thể có, từ đó chọn ra giải pháp tốt nhất. Tìm kiếm vét đảm bảo luôn thu được giải pháp tốt nhất nhưng chi phí tìm kiếm rất lớn. Trong trường hợp tối nhất, việc tìm kiếm vét cạn có thể không thực hiện được vì không gian tìm kiếm quá lớn. Như đã nói ở trên, với tập hợp gồm n đặc trưng thì không gian tìm kiếm có thể có là 2^n.

2.2.2.2. Tìm kiếm tuần tự (Sequential Search)

Tìm kiếm tuần tự là phương pháp tìm kiếm "tự nhiên" và đơn giản nhất, bằng cách tuần tự xem xét từng phần tử đến khi tìm được phần tử cần tìm. Đối với lựa chọn đặc trưng, tìm kiếm tuần tự là phương pháp đánh giá tuần tự từng đặc trưng của tập dữ liệu để thêm vào hoặc loại bỏ khỏi tập con đặc trưng đang xét.

Có hai phương pháp tìm kiếm tuần tự hay được sử dụng là:

- **Tìm kiếm tuần tự tiến:** (Sequential Forward Search SFS) Chiến lược tìm kiếm này thực hiện tuần tự thêm các đặc trưng vào tập con đặc trưng đang xét đến khi đạt đủ số lượng đặc trưng mong muốn.
- **Tìm kiếm tuần tự lùi** (Sequential Backward Search SFS) Chiến lược tìm kiếm này thực hiện tuần tự việc loại bỏ các đặc trưng ra khỏi tập con đặc trưng đang xét đến khi đạt được số lượng đặc trưng mong muốn

Tìm kiếm tuần tự làm cho không gian tìm kiếm nhỏ hơn rất nhiều, ta chỉ cần xét mỗi đặc trưng một lần (hoặc nhiều lần nếu có nhiều bước lặp). Tuy nhiên việc thêm mỗi phần tử vào theo một hướng như vậy có thể dẫn đến tối ưu cục bộ.

2.2.2.3. Tìm kiếm ngẫu nhiên (Random Search)

Trong bài toán lựa chọn đặc trưng, tìm kiếm ngẫu nhiên bắt đầu với một tập con gồm các đặc trưng ngẫu nhiên. Có thể tiếp tục bằng cách sử dụng tìm kiếm tuần tự cổ điển. Cũng có thể thêm vào các đặc trưng một cách ngẫu nhiên hoặc tạo ra các tập con đặc trưng hoàn toàn ngẫu nhiên. Sự ngẫu nhiên của thuật toán tìm kiếm ngẫu nhiên có thể giúp tránh được tối ưu cục bộ, trong khi đó vẫn duy trì được không gian tìm kiếm không quá lớn. Tuy nhiên sự ngẫu nhiên này cũng có thể dẫn đến hiệu năng không ổn định của thuật toán.

2.2.3 Hướng tìm kiếm

Khi đã xác định được chiến lược tìm kiếm, chúng ta sẽ xác định hướng tìm kiếm. Hướng tìm kiếm ở đây có thể là hướng tiến (forward) hoặc hướng lùi (backward) hoặc kết hợp linh hoạt cả hai hướng [5].

2.2.3.1. Hướng tiến (forward)

Lựa chọn hướng tiến có thể bắt đầu với tập con đặc trưng rỗng, sau đó liên tiếp thêm các đặc trưng vào tập con đăc trưng đang xét đến khi thỏa mãn tiêu chí dừng. Tiêu chí dừng sẽ được trình bày trong phần 2.2.5

***2.2.3.2.** Hướng lùi (backward)*

Lựa chọn hướng lùi có thể bắt đầu với tập con đặc trưng gồm đầy đủ các đặc trưng của tập dữ liệu ban đầu, sau đó liên tiếp loại bỏ dần các đặc trưng khỏi tập đang xét đến khi thỏa mãn tiêu chí dừng

2.2.3.3. Hướng hai chiều (bidirectional)

Hướng hai chiều bao gồm cả hướng tiến và hướng lùi. Tại mỗi thời điểm, có thể thực hiện thêm vào k đặc trưng đồng thời loại bỏ m đặc trưng khỏi tập con đặc trưng đang xét.

2.2.3.4. Hướng ngẫu nhiên (random)

Bao gồm tất cả các hướng đã nêu trên, tại mỗi bước có thể thực hiện ngẫu nhiên theo một trong các hướng.

2.2.4 Tiêu chí đánh giá

Các phương pháp lựa chọn đặc trưng tìm kiếm tập con đăc trưng mà mô tả tốt nhất biến mục tiêu. Khi xem xét một đặc trưng hoặc một tập con đặc trưng nào đó, cần có tiêu chí cụ thể để đánh giá chúng. Dựa vào tiêu chí đánh giá ta biết được một đặc trưng hoặc một tập con đặc trưng là quan trọng hay không quan trọng, dư thừa

hay không dư thừa. Kết quả tập con đặc trưng tối ưu thu được phụ thuộc rất nhiều vào tiêu chí đánh giá được sử dụng, do đó cần chọn được tiêu chí đánh giá phù hợp với bài toán đang xét.

Các tiêu chí đánh giá được phân thành hai nhóm dựa trên sự phụ thuộc của chúng vào các thuật toán phân loại mà cuối cùng sẽ được sử áp dụng lên tập con đặc trưng đã chọn (Liu và Yu, 2005 [6]).

- **Tiêu chí độc lập:** Là các tiêu chí đánh giá độc lập với thuật toán học máy sử dụng để phân loại. Các tiêu chí này có thể là công thức dựa trên khoảng cách, thống kê, xác suất, nhằm đánh mức độ liên quan và quan trọng của đặc trưng cũng như tập con đặc trưng. Các phương pháp sử dụng tiêu chí độc lập luôn cho kết quả tốt và thời gian cũng như chi phí thực hiện thấp.
- **Tiêu chí phụ thuộc:** Là các tiêu chí đánh giá phụ thuộc vào thuật toán học máy sử dụng để phân loại. Các tiêu chí này có thể là độ chính xác, hiệu suất của mô hình phân loại. Tiêu chí phụ thuộc thường đem lại hiệu quả cao hơn nhưng chi phí tính toán cũng cao hơn so với sử dụng tiêu chí độc lập.

2.2.5 Tiêu chí dừng

Tiêu chí dừng cho biết khi nào thì các thuật toán lựa chọn đặc trưng dừng lại. Tiêu chí dừng phụ thuộc vào tiêu chí đánh giá được sử dụng và nhu cầu về chi phí cũng như hiệu suất mong muốn.

Đối với các tiêu chí độc lập, tiêu chí dừng có thể là danh sách một số lượng các đặc trưng được sắp xếp theo thứ tự hoặc thuật toán có thể dừng khi trải qua đủ số bước lặp cho trước.

Đối với tiêu chí đánh giá phụ thuộc, thuật toán có thể ngừng khi không thể cải thiện được hiệu suất của mô hình dự đoán hiện tại.

Một số tiêu chí dừng thường được sử dụng là:

- Thuật toán dừng khi tất cả các đặc trưng được đánh giá
- Thuật toán dừng khi kết quả thay đổi không đáng kể
- Thuật toán dừng khi đã thực hiện đủ số bước lặp hoặc thêm/bớt đủ số lượng đặc trưng mong muốn
- Việc thực hiện thêm bước lặp cũng không thể tạo ra tập con đặc trưng tốt hơn

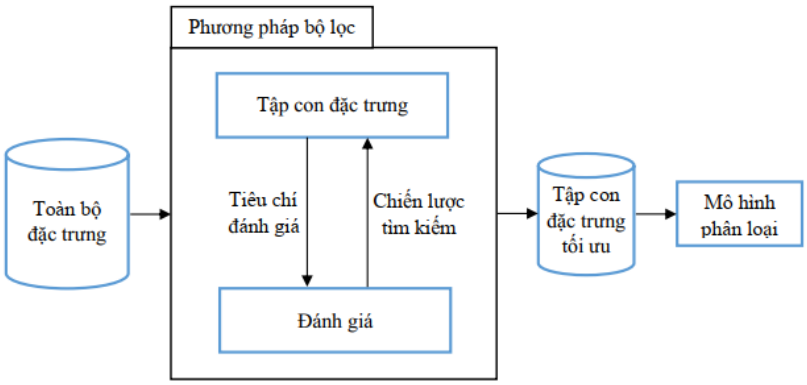
2.3 Các phương pháp tiếp cận bài toán lựa chọn đặc trưng

Lựa chọn các chiến lược tìm kiếm, hướng tìm kiếm và tiêu chí đánh giá khác nhau dẫn đến các phương pháp lựa chọn đặc trưng được khác nhau. Có rất nhiều phương pháp lựa chọn đặc trưng đã được đưa ra và thực nghiệm, nhưng có thể phân loại chúng vào bốm nhóm phương pháp chính: phương pháp bộ lọc (Filter methods), phương pháp bao bọc (Wrapper methods), phương pháp nhúng (Embedded methods). Sau đây sẽ là phần giới thiệu, ưu nhược điểm và ví dụ về các phương pháp này.

2.3.1 Phương pháp bộ lọc (Filter methods)

Phương pháp bộ lọc là nhóm các phương pháp sử dụng tiêu chí đánh giá độc lập. Các tiêu chí được sử dụng để tính điểm số cho các đặc trưng một cách độc lập hoặc có sự phụ thuộc. Sau đó, các đặc trưng được chọn một cách tham lam hoặc thêm vào tập con đặc trưng (hoặc xóa bỏ) và trải qua nhiều bước lặp.

Hình vẽ sau thể hiện mô hình lựa chọn đặc trưng theo phương pháp bộ lọc:



Hình 2.4 Mô hình lựa chọn đặc trưng theo phương pháp bộ lọc

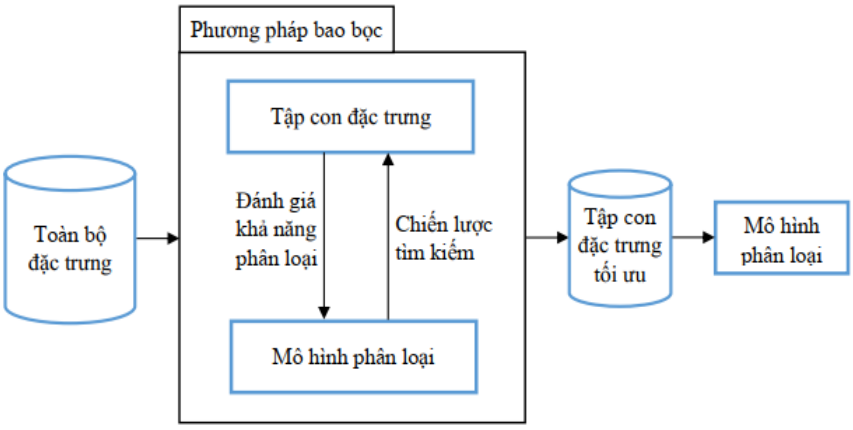
Phương pháp bộ lọc thường là đơn biến và xét các đặc trưng một cách độc lập hoặc liên quan đến biến mục tiêu. Một số ưu, nhược điểm của phương pháp bộ lọc [7]:

- Ưu điểm: Nhanh, dễ mở rộng, không phụ thuộc vào mô hình phân loại;
- Nhược điểm: Có thể bỏ qua sự phụ thuộc của các đặc trưng, bỏ qua sự tương tác với mô hình phân loại;
- Ví dụ: Thuật toán lựa chọn đặc trưng sử dụng điểm số Laplacian, Relief, …

2.3.2 Phương pháp trình bao bọc (Wrapper methods)

Phương pháp bao bọc là các phương pháp sử dụng các tiêu chí đánh giá phụ thuộc. Trong khi phương pháp bộ lọc tìm kiếm tập con đặc trưng tối ưu độc lập với bước học tập của mô hình, thì phương pháp trình bao bọc sử dụng hiệu suất của mô hình để tìm kiếm tập con đặc trưng tối ưu.

Hình vẽ sau thể hiện mô hình lựa chọn đặc trưng theo phương pháp trình bao bọc:



Hình 2.5 Mô hình lựa chọn đặc trưng theo phương pháp trình bao bọc

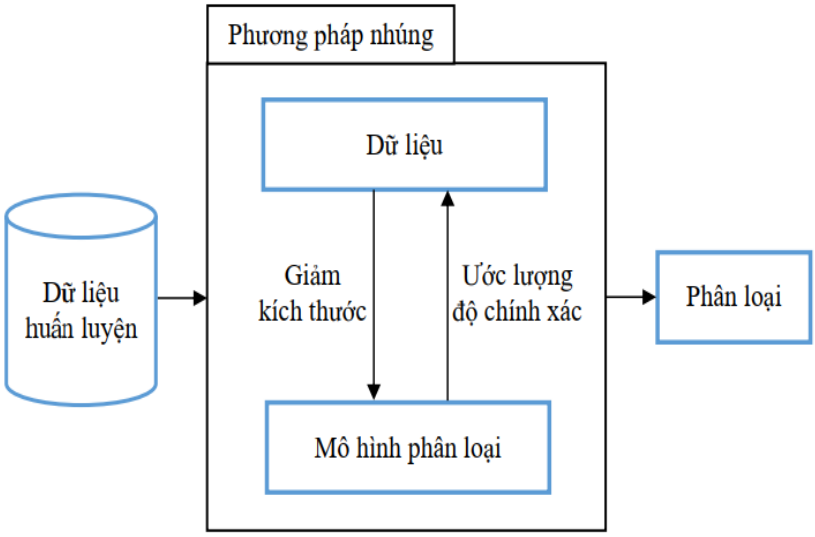
Một thuật toán tìm kiếm được bao bọc xung quanh mô hình phân loại, có thể là: tìm kiếm tuần tự (Sequential Search), tìm kiếm ngẫu nhiên (Random Search), tìm kiếm kinh nghiệm (Heuristic Search) hoặc kỹ thuật tính toán tiến hóa (EC).

- Ưu điểm: Phù hợp với mô hình phân loại, đem lại hiệu năng cao hơn phương pháp bộ lọc;
- Nhược điểm: Chi phí tính toán, tài nguyên lớn;
- Ví dụ: GA (thuật toán tiến hóa), SFS (Lựa chọn đặc trưng tuần tự)…

2.3.3 Phương pháp nhúng (Embedded methods)

Đối với hai phương pháp đã giới thiệu ở trên, phương pháp bộ lọc và phương pháp trình bao bọc, quá trình lựa chọn đặc trưng được thực hiện trước và cho ra kết quả là đầu vào của mô hình phân loại. Với phương pháp nhúng, việc lựa chọn đặc trưng được tích hợp vào cấu trúc bộ phân loại, tức là việc lựa chọn đặc trưng diễn ra tự nhiên như là một phần trong quá trình học tập của mô hình. Sử dụng toàn bộ đặc trưng làm đầu vào để xây dựng mô hình, sau đó đánh giá mô hình để suy ra mức độ liên quan của các đặc trưng với mô hình học. Phương pháp nhúng cũng sử dụng một mô hình học tập cụ thể, nhưng ít tốn kém hơn về mặt tính toán so với phương pháp trình bao bọc.

Hình vẽ sau thể hiện mô hình lựa chọn đặc trưng theo phương pháp nhúng:



Hình 2.6 Mô hình lựa chọn đặc trưng theo phương pháp nhúng

- Ưu điểm: Đem lại hiệu năng cao hơn phương pháp bộ lọc, phù hợp với mô hình phân loại, chi phí tính toán thấp hơn phương pháp bao bọc.
- Nhược điểm: Triển khai phức tạp, đặc biệt là với dữ liệu lớn.
- Ví dụ: LASSO, decision trees, Bayes...

2.3.4 So sánh các phương pháp

Các phương pháp lựa chọn đặc trưng có thể so sánh theo các mục đích khác nhau, không có phương pháp nào là hoàn hảo. Với mục đích nhanh và đơn giản, phương pháp bộ lọc là lựa chọn tốt nhất. Để cải thiện hệu suất, phương pháp trình bao bọc

nên được ưu tiên hơn vì chúng thích hợp hơn với các nhiệm vụ phân loại. Đôi khi phương pháp nhúng sẽ phù hợp với các mục đích phức tạp hơn. Ràng buộc về thời gian tính toán nếu không đòi hỏi, nên áp dụng chiến lược tìm kiếm vét cạn để đảm bảo thu được lời giải tối ưu toàn cục.

Bảng sau đây tóm tắt và so sánh các phương pháp lựa chọn đặc trưng đã nêu:

	Tiêu chí đánh giá	**Chiến lược tìm kiếm**	**Ưu điểm**	**Nhược điểm**
Phương pháp bộ loc	Khoảng cách, độ lợi thông tin, ước lượng	Tìm kiếm tuần tự, trọng số đặc trưng	Thời gian tính toán, chi phí tài nguyên thấp, Dễ triển khai và mở rộng.	Hiệu quả không cao, dễ gặp tối ưu cục bộ.
Phương pháp trình bao bọc	Hiệu năng của mô hình phân loại	Tìm kiếm vét cạn, tìm kiếm kinh nghiệm	Đạt được các giải pháp tối ưu hơn, phù hợp với bài toán phân loại, xem xét sự phụ thuộc của các đặc trưng.	Khó triển khai, thời gian tính toán , chi phí tài nguyên cao.
Phương pháp nhúng	Hiệu năng phân loại của mô hình phân loại	Tìm kiếm hướng dẫn bởi quá trình học tập	Thời gian tính toán thấp, hiệu quả cao, phù hợp với bài toán phân loại, xem xét sự phụ thuộc của các đặc trưng	Khó triển khai và mở rộng

Bảng 2.1 So sánh các phương pháp lựa chọn đặc trưng

2.4 Các hướng nghiên cứu lựa chọn đặc trưng

Với mỗi phương pháp lựa chọn đặc trưng, có các hướng nghiên cứu khác nhau. Phần này trình bày các hướng nghiên cứu phổ biến hiện nay ở mỗi phương pháp.

2.4.1 Phương pháp bộ lọc

2.4.1.1. Hướng nghiên cứu dựa trên thống kê

Khoa học thống kê sớm phát triển, là giải pháp tính toán cho hầu hết các vấn đề trước khi học máy phát triển như hiện nay. Các phương pháp dựa trên thống kê là các phương pháp lựa chọn đặc trưng được giới thiệu và phát triển đầu tiên. Chúng sử dụng các phương pháp thống kê để tìm ra các đặc trưng liên quan mạnh. Phương pháp dựa trên thống kê có thể hiệu quả với một số bài toán, nhưng nhìn chung với dữ liệu ngày càng lớn, hiệu quả của chúng không cao. Các thuật toán, điểm số (score) dựa trên thống kê như:

- T-Score (David và Sampson, 1986 [8])
- Chi-Square Score (Liu và Setiono, 1995 [9])
- CFS (Hall và Smith, 1999 [10])

2.4.1.2. Hướng nghiên cứu dựa trên sự tương tự của dữ liệu

Hay còn gọi là hướng nghiên cứu dựa trên khoảng cách, sử dụng thước đo khoảng cách để đánh giá độ quan trọng của đặc trưng. Ý tưởng chung là các dữ liệu có cùng giá trị nhãn lớp thì tương tự và ngược lại. Một số độ đo dựa trên khoảng cách để đánh giá độ quan trọng của đặc trưng nổi bật là thuật toán Laplacian, Relief... Các phương pháp dựa trên độ đo khoảng cách cũng thuộc phương pháp bộ lọc cho nên cũng được dùng để kết hợp với các phương pháp khác.

Có rất nhiều thuật toán, phương pháp đã được giới thiệu theo hướng này, có thể kể đến như:

- Thuật toán Laplacian (He và cộng sự, 2005 [11])
- Thuật toán ReliefF (Kononenko và cộng sự, 2003 [12])
- Thuật toán Fisher (Duda và cộng sự, 2012 [13])

2.4.1.3. Hướng nghiên cứu dựa trên lý thuyết thông tin (Information

Phần lớn các phương pháp lựa chọn đăc trưng hiện này là theo hướng dựa trên lý thuyết thông tin. Các thuật toán dự trên lý thuyết thông tin sử dụng khái niệm entropy làm cơ sở để đánh giá các đặc trưng. Nếu một đặc trưng không đem lại thông tin gì cho việc phân loại, thì đó là đặc trưng không liên quan, có thể loại bỏ. Nếu một đặc trưng đem lại thông tin tương tự như các đặc trưng khác, thì đặc trưng đó là dư thừa, không cần thiết.

Một số thuận toán dựa trên lý thuyết thông tin có thể kể đến như:

- Cực đại thông tin tương hỗ MIM (Lewis và cộng sự, 1992 [14])
- Cực tiểu dư thừa, cực đại liên quan MRMR (Peng và cộng sự, 2005 [15])
- Cực đại thông tin tương hỗ có điều kiện CMIM (Vidal, Ullman, 2003 [16])
- Thuật toán thông tin tương hỗ chung JMI (Meyer và cộng sự, 2008 [17])

2.4.1.4. Hướng nghiên cứu dựa trên sự tương quan

Tương quan là một phép đo thống kê về mối quan hệ của hai biến số. Các đặc trưng có độ tương quan cao thì có ảnh hưởng giống nhau đến nhãn phân loại. Do đó, khi hai đặc trưng có độ tương quan cao, một trong hai đặc trưng là dư thừa, có thể loại bỏ.

Một số thước đo tương quan đã và đang được áp dụng: Hệ số tương quan Pearson (PCC), kiểm thử chi bình phương …

2.4.2 Phương pháp trình bao bọc

2.4.2.5. Hướng nghiên cứu dựa trên tìm kiếm tuần tự

Các phương pháp áp dụng chiến lược tìm kiếm rất hiếm gặp vì chi phí quá lớn. Chiến lược tìm kiếm tuần tự không thực hiện vét cạn nhưng lại hiệu quả và được áp dụng rộng rãi, đặc biệt là cho phương pháp trình bao bọc. Ví dụ như thuật toán tìm kiếm tham lam được áp dụng trong lựa chọn tiến tuần tự SFS [18], loại bỏ lùi tuần tự SBE [19]. Tuy nhiên cả hai phương pháp đều có nhược điểm là đặc trưng đã chọn hoặc đã bị loại bỏ thì sẽ không được xét đến nữa. Do đó hai phương pháp là SBFS (Sequential Backward Floating Selection) và SFFS (Sequential Forward Floating Selection) đã được Pudil và cộng sự giới thiệu năm 1994 [20], hai phương pháp được kiểm chứng và kết luận là cho kết quả tốt hơn SBE và SFS.

2.4.2.6. Hướng nghiên cứu dựa trên thuật toán tiến hóa

Nhiều kỹ thuật tìm kiếm được áp dụng để lựa chọn đặc trưng như tìm kiếm tuần tự, tìm kiếm tham lam và tìm kiếm kinh nghiệm nhưng hầu hết các phương pháp đều vướng phải vấn đề tối ưu cục bộ hoặc chi phí tính toán cao. Gần đây kỹ thuật tính toán tiến hóa (EC) là một phương pháp hiệu quả được áp dụng vào giải quyết các vấn đề lựa chọn đặc trưng, bao gồm các thuật toán tính toán tiến hóa như GA, PSO, ACO. Các thống kê cho thấy các thuật toán tiến hóa đang nhận được sự quan tâm lớn từ các nhà nghiên cứu lựa chọn đặc trưng [21].

So với các phương pháp tìm kiếm truyền thống, các kỹ thuật EC không cần đưa ra bất kỳ giả định nào về không gian tìm kiếm. Ưu điểm lớn nhất của các kỹ thuật tính toán tiến hóa là chúng có thể tạo ra nhiều giải pháp trong một lần chạy với cơ chế dựa trên quần thể. Tuy nhiên các kỹ thuật EC có một hạn chế lớn là đòi hỏi chi phí tính toán cao vì chúng đòi hỏi có một lượng lớn các tiêu chí đánh giá. Kỹ thuật tính toán tiến hóa (EC) còn gặp vấn đề về tính ổn định do chúng dựa trên tính đột biến, dẫn tới kết quả khác nhau với mỗi lần chạy.

2.4.3 Phương pháp nhúng

2.4.3.7. Hướng nghiên cứu dựa trên học thưa

Các phương pháp lựa chọn đặc trưng dựa trên học thưa có mục tiêu là giảm giá trị của hàm lỗi (Loss function) bằng các điều kiện chính quy thưa. Những điều kiện chính quy thưa làm cho các hệ số đặc trưng nhỏ dần hoặc bằng 0 và sau đó các đặc trưng tương ứng sẽ bị loại bỏ [3].

Hướng nghiên cứu dưa trên học thưa gần đây phổ biến hơn rất nhiều do hiệu quả tốt và khả năng diễn đạt kết quả cao, ví dụ như hồi quy logistic thưa để lựa chọn đặc trưng [22], được sử dụng với hàng triệu đặc trưng.

Điểm mạnh của phương pháp này là nó nhúng phương pháp lựa chọn đặc trưng vào một thuật toán học điển hình (chẳng hạn như SVM).Do đó, nó thường có thể dẫn đến hiệu suất rất tốt với các thuật toán học tập cơ bản. Tuy nhiên phương pháp này vẫn còn một số hạn chế như các đặc trưng đã chọn sẽ không phù hợp cho các thuật toán phân loại khác và chi phí tính toán cao vì thường dẫn tới các phép toán ma trận phức tạp trong hầu hết các trường hợp.

CHƯƠNG 3. CÁC THUẬT TOÁN LỰA CHỌN ĐẶC TRƯNG SỬ DỤNG ĐIỂM SỐ DỰA TRÊN SỰ TƯƠNG TỰ CỦA DỮ LIỆU

Chương này sẽ trình bày về các thuật toán lựa chọn đặc trưng sử dụng điểm số dựa trên sự tương tự của dữ liệu (gọi tắt là các thuật toán dựa trên sự tương tự). Đối với các bài toán lựa chọn đặc có giám sát, sự tương tự được thể hiện ở nhãn tức là, các bản ghi có dữ liệu tương tự nhau thì nên có cùng nhãn hay thuộc cùng một lớp. Đối với các bài toán lựa chọn đặc trưng không giám sát, đa số các điểm sốsử dụng độ đo về khoảng cách để đo sự tương tự của dữ liệu

Cho bộ dữ liệu gồm n bản ghi và d đặc trưng được mô tả bởi ma trận $X \epsilon R^{n \times d}$ gồm n bản ghi và d đặc trưng $f_1, f_2, \ldots f_d$ biểu diễn bằng các vecto cột. Bảng sau mô tả các kí hiệu được sử dụng trong Chương 3 và Chương 4:

Kí hiệu	Ý nghĩa, mô tả
n	Số bản ghi của bộ dữ liệu
d	Số đặc trưng của bộ dữ liệu
k	Số đặc trưng đã chọn
c	Số lớp
S	Tập con chứa k đặc trưng đã chọn
$x_1, x_2, \ldots x_n$	n vector hàng biểu diễn n bản ghi
$f_1, f_2, \ldots f_d$	d vecto cột biểu diễn d đặc trưng
$X^{n \times d}$	Ma trận dữ liệu ban đầu
X_S	Ma trận dữ liệu chỉ gồm các đặc trưng trong tập S
y	Vecto cột biểu diễn nhãn
A^T	Ma trận chuyển vị của A
I	Ma trận đơn vị
1	Vector cột với tất cả thành phần bằng 1
$\tilde{f}_l$	Vector biến đổi từ f_i
n_l	Số phần tử của lớp l
$\|x_i - x_j\|$	Khoảng cách Euclid
(u, v)	Góc giữa hai vector u, v
$tr(A)$	Trace ratio của ma trận A

Bảng 3.1 Kí hiệu sử dụng trong các điểm số dựa trên sự tương tự

Để cho việc trình bày các thuật toán trở nên dễ hiểu và trực quan, hãy bắt đầu với một ví dụ nhỏ. Cho bộ dữ liệu Điểm thi về điểm thi của học sinh gồm 3 đặc trưng và 10 bản ghi tương ứng với 10 học sinh.

Chi tiết về bộ dữ liệu được thể hiện trong bảng:

x_i	f_1	f_2	f_3	y
1	7,4	6,5	8	0
2	9	9	5	0
3	8,4	8	7,4	0
4	8,8	6	7	0
5	9,6	9	8,4	1
6	8,4	7,6	10	1
7	9,2	7	10	1
8	8	9,2	9	1
9	9	8	7,8	1
10	10	9,2	8	1

Bảng 3.2 Bộ dữ liệu Điểm thi

Nhãn y phân loại học sinh Đạt (nhãn 1) hay Không đạt (nhãn 0), nhãn 1 chiếm 60%.

3.1 Thuật toán lựa chọn đặc trưng sử dụng điểm số Laplacian

Điểm số Laplacian [11] (He và cộng sự, 2005) dựa trên việc xây dựng ma trận Laplacian có nhiều ứng dụng trong nhiều bài toán học máy khác.

Điểm số Laplacian áp dụng cho lựa chọn đặc trưng không giám sát, đem lại hiệu quả cao. Để tính được điểm số Laplacian cho các đặc trưng, thức hiện các bước:

Bước 1: Xây dựng ma trận tương đồng S được định nghĩa $S(i,j) = e^{-\frac{\left\|x_i - x_j\right\|^2}{t}}$ nếu x_j thuộc k láng giềng gần của x_i, ngược lại thì $S(i,j) = 0$

Bước 2: Sau đó, ta tính ma trận đường chéo D bằng cách $D(i,i) = \sum_{j=1}^{n} S(i;j)$ và ma trận Laplacian $L = D - S$

Bước 3: Tính các vector $\widetilde{f_i} = f_i - \frac{f_i^T . D.1}{1^T . D.1} . 1$

Bước 4: Tính toán điểm số Laplacian cho mỗi đặc trưng f_i theo công thức:

$$laplacian\_score(f_i) = \frac{\widetilde{f_i}^T . L . \widetilde{f_i}}{\widetilde{f_i}^T . D . \widetilde{f_i}} \qquad PT\ 3.1$$

Thuật toán lựa chọn đặc trưng sử dụng điểm số Laplacian (sau đây gọi tắt là thuật toán Laplacian) thực hiện tính điểm $laplacian\_score$ cho mỗi đặc trưng một cách độc lập và riêng biệt, sau đó lựa chọn tham lam k đặc trưng có điểm số Laplacian **nhỏ nhất**.

Với bộ dữ liệu Điểm thi, bảng sau thể hiện điểm số Laplacian của các đặc trưng và các đặc trưng được sắp xếp theo điểm số **tăng dần** (độ quan trọng giảm dần).

Tham số	f_1	f_2	f_3	Đặc trưng quan trọng
$k = 2, t = 1$	0,291	0,140	0,061	f_3, f_2, f_1
$k = 3, t = 1$	0,347	0,187	0,089	f_3, f_2, f_1
$k = 4, t = 1$	0,381	0,204	0,116	f_3, f_2, f_1

Bảng 3.3 Điểm số Laplacian các đặc trưng bộ dữ liệu Điểm thi

Thuật toán Laplacian có ưu điểm là việc cài đặt dễ dàng, tính toán nhanh chóng, đã được chứng minh hiệu năng cao và được áp dụng phổ biến.

3.2 Thuật toán lựa chọn đặc trưng sử dụng điểm số Fisher

Khác với điểm số Laplacian, điểm số Fisher (Duda và cộng sự, 2012 [13]) áp dụng cho lựa chọn đặc trưng có giám sát. Nó coi các đặc trưng liên quan là các đặc trưng mà các giá trị đặc trưng của các bản ghi thuộc cùng lớp thì tương tự, trong khi đó, với các bản ghi không cùng lớp thì không tương tự. Điểm số Fisher cho mỗi đặc trưng được tính theo công thức:

$$fisher\_score(f_i) = \frac{\sum_{j=1}^{c} n_j(\mu_{ij} - \mu_i)^2}{\sum_{j=1}^{c} n_j \sigma_{ij}^2} \qquad PT\ 3.2$$

Trong đó, n_j là số bản ghi thuộc lớp j, μ_i là giá trị trung bình của đặc trưng thứ i. μ_{ij} và σ_{ij}^2 là giá trị trung bình và phương sai của đặc trưng thứ i nhưng cùng thuộc lớp j.

Điểm số Fisher có thể coi là biến thể của điểm số Laplacian, bằng cách xây dựng ma trân tương đồng $S(i,j) = \frac{1}{n_l}$ nếu x_j và x_i cùng thuộc lớp l, ngược lại thì $S(i,j) = 0$. Trong trường hợp này, mối quan hệ giữa điểm số Fisher và điểm số Laplacian là: $fisher\_score(f_i) = \frac{1}{1 - laplacian\_score(f_i)}$. Trong đồ án này, tôi sử dụng phương pháp xây dựng ma trân tương đồng dựa trên nhãn lớp sau đó tính điểm $fisher\_score$ cho từng đặc trưng giống với điểm Laplacian.

Tương tự như thuật toán Laplacian, thuật toán lựa chọn đặc trưng sử dụng điểm số Fisher (sau đây gọi tắt là thuật toán Fisher) cũng thực hiện tính điểm $fisher\_score$ cho từng đặc trưng một cách riêng biệt, nhưng sau đó chọn tham lam k đặc trưng có điểm số Fisher **lớn nhất**.

Với bộ dữ liệu Điểm thi, bảng kết quả sau thể hiện điểm số Fisher của các đặc trưng và thứ tự các đặc trưng được sắp xếp theo điểm số **giảm dần** (độ quan trọng giảm dần).

f_1	f_2	f_3	Thứ tự đặc trưng quan trọng
0,225	0,219	0,999	f_3, f_1, f_2

Bảng 3.4 Điểm số Fisher các đặc trưng bộ dữ liệu Điểm thi

3.3 Thuật toán lựa chọn đặc trưng sử dụng điểm số SPEC

Điểm số SPEC (Zhao và Liu, 2007 [23]) là một sự mở rộng của điểm số Laplacian, áp dụng cho cả bài toán có giám sát và không giám sát. Với bài toán không giám sát, sự tương tự của dữ liệu được tính bằng nhân RBF (RBF kernel). Còn với bài toán có giám sát, bắt đầu bằng việc xây dựng ma trân tương đồng $S(i,j) = \frac{1}{n_l}$ nếu x_j và x_i cùng thuộc lớp l, ngược lại thì $S(i,j) = 0$. Ma trận đường chéo D được xây dựng tương tự thuật toán Laplacian: $D(i,i) = \sum_{j=1}^{n} S(i;j)$, và ma trận Laplacian L được tính bằng $L = D^{-\frac{1}{2}}(D-S)D^{-\frac{1}{2}}$ với $D^{-\frac{1}{2}}(i,j) = [D(i,j)]^{-\frac{1}{2}}$

Đặc trưng liên quan có thể đánh giá bằng 3 tiêu chí đánh giá khác nhau:

$$SPEC\_score1(f_i) = \sum_{j=1}^{n} \alpha_j^2 \gamma(\lambda_j) \qquad PT\ 3.3$$

$$SPEC\_score2(f_i) = \frac{\sum_{j=2}^{n} \alpha_j^2 \gamma(\lambda_j)}{\sum_{j=2}^{n} \alpha_j^2} \qquad PT\ 3.4$$

$$SPEC\_score3(f_i) = \sum_{j=1}^{m} \alpha_j^2 \big(\gamma(2) - \gamma(\lambda_j)\big) \qquad PT\ 3.5$$

Trong đó, λ_j và v_j là cặp giá trị riêng và vecto riêng tương ứng thứ j của ma trận L và $\alpha_j = \cos(v_j, f_i)$. Còn $\gamma(.)$ là một hàm số đồng biến để giảm nhiễu dữ liệu. Nếu dữ liệu đã được xử lí nhiễu, có thể chọn $\gamma(x) = x$.

Các thuật toán lựa chọn đặc trưng sử dụng điểm số SPEC (sau đây gọi tắt là thuật toán SPEC) đánh giá các đặc trưng bằng cách sử dụng một trong ba điểm số trên, hoặc sử dụng đồng thời. Sau đó lựa chọn tham lam k đặc trưng có điểm số $SPEC\_score$ **lớn nhất**. Tiêu chí thứ hai $SPEC\_score2$ đã được chứng minh là cho kết tốt hơn và ý tưởng tương tự như điểm số Laplacian nếu $\gamma(x) = x$.

Trong đồ án này, tôi sử dụng điểm số $SPEC\_score2$ với $\gamma(x) = x$ để đánh giá các đặc trưng, áp dụng với các lớp bài toán có giám sát.

Với bộ dữ liệu Điểm thi, bảng kết quả sau thể hiện điểm số SPEC của các đặc trưng và thứ tự các đặc trưng được sắp xếp theo điểm số **giảm dần** (độ quan trọng giảm dần).

$\gamma(x)$	f_1	f_2	f_3	Đặc trưng quan trọng
$\gamma(x) = x$	0,150	0,054	0,024	f_1, f_2, f_3

Bảng 3.5 Điểm số SPEC các đặc trưng bộ dữ liệu Điểm thi

3.4 Thuật toán lựa chọn đặc trưng sử dụng điểm số Trace Ratio

Điểm số Trace Ratio (Nie và cộng sự, 2008 [24]) lựa chọn trực tiếp tập con đặc trưng tối ưu toàn cục dựa trên chuẩn vết của ma trận $tr(A) = \sum_{i=1}^{n} A(i,i)$

Trace Ratio xây dựng hai ma trận tương đồng S_w và S_b để tính sự tương tự của dữ liệu trong cùng lớp và giữa các lớp với nhau. Để phản ánh mối quan hệ cùng lớp (within-class) trong dữ liệu, $S_w(i,j)$ sẽ nhận gì trị lớn nếu x_i và x_j thuộc cùng một lớp (nếu dùng nhãn lớp) hoặc gần nhau (nếu dùng khoảng cách) và nhận giá trị rất nhỏ nếu ngược lại. Để phản ảnh mối quan hệ giữa các lớp (beetween-class), $S_b(i,j)$ sẽ nhận gì trị lớn nếu x_i và x_j khác lớp (nếu dùng nhãn lớp) hoặc cách xa nhau (nếu dùng khoảng cách) và nhận giá trị rất nhỏ nếu ngược lại. Việc xây dựng hai ma trận tương đồng này phụ thuộc vào lớp bài toán có giám sát hay không giám sát. Cụ thể, trong đồ án này, S_w và S_b được định nghĩa như sau:

$S_w(i,j) = e^{-\frac{\left\|x_i - x_j\right\|^2}{t}}$ nếu x_j thuộc k láng giềng gần của x_i, ngược lại $S_w(i,j) = 0$.

$S_b(i,j) = \frac{1}{1^T.D_w.1} D_w.1.1^T.D_w$ với D_w là ma trận đường chéo của ma trận S_w

Từ đó ta xây dựng hai ma trận Laplacian tương ứng là L_w và L_b. Sau đó ta định nghĩa ma trận $B = X\,L_b X^T \in R^{d\times d}$ và $E = X.L_w.X^T \in R^{d\times d}$.

Một chỉnh hợp chập m của d phần tử $\{1,2,3 \dots d\}$ biểu diễn một tập con gồm m đặc trưng có lấy thứ tự gọi là I. Ta xây dựng ma trận lựa chọn $W \epsilon R^{m\times d}$ gồm các vecto cột w_i có phần tử thứ i bằng 1 và tất cả phần tử còn lại đều bằng 0. Từ đó ta có $W_I = \left[w_{I(1)}, w_{I(2)}, \dots w_{I(m)}\right]$.

Điểm số cho các đặc trưng được tính theo công thức:

$$Trace\_ratio(f_i) = w_i^T.(B - \lambda E).w_i \qquad \textit{PT 3.6}$$

Điểm số cho tập con đặc trưng đang xét I được tính bằng:

$$\lambda = \frac{tr(W_I^T.B.W_I)}{tr(W_I^T.E.W_I)} \qquad \textit{PT 3.7}$$

Thuật toán lựa chọn đặc trưng sử dụng điểm số Trace Ratio (sau đây gọi tắt là thuật toán Trace Ratio) được thực hiện theo các bước sau:

1. Khởi tạo tập con đặc trưng $I = \{1,2,3,4, \dots m\}$ và tính giá trị λ
2. Tính điểm $Trace\_ratio$ cho từng đặc trưng f_i
3. Sắp xếp các đặc trưng theo điểm số với thứ tự giảm dần
4. Chọn ra m đặc trưng có điểm số lớn nhất để cập nhật tập I và λ
5. Lặp lại các bước 2 – 4 đến khi đạt tiêu chí dừng $\lambda - \lambda_{old} < \varepsilon$ cho trước

Với bộ dữ liệu Điểm thi, tiêu chí dừng $\varepsilon = 10^{-3}$, bảng kết quả sau thể hiện điểm số Trace Ratio của các đặc trưng tại các bước lặp và thứ tự các đặc trưng được sắp xếp theo điểm số **giảm dần** (độ quan trọng giảm dần).

Bước lặp	f_1	f_2	f_3	λ	Đặc trưng quan trọng
Lặp lần 1	4,523	-1,330	-3,193	0,536	f_1, f_2, f_3
Lặp lần 2	4,523	-1,330	-3,193	0,536	f_1, f_2, f_3

Bảng 3.6 Điểm số TraceRatio các đặc trưng bộ dữ liệu Điểm thi

3.5 Thuật toán lựa chọn đặc trưng sử dụng điểm số ReliefF

Điểm số ReliefF (Kononenko và cộng sự, 2003 [12]) lựa chọn các đặc trưng mà phân loại các bản ghi thuộc các lớp khác nhau. Giả sử l bản ghi được chọn ngẫu nhiên từ n bản ghi ban đầu, và dữ liệu gồm c lớp. Khi đó, điểm số ReliefF của đặc trưng f_i được tính:

$$reliefF\_score(f_i) = \frac{1}{c}\sum_{j=1}^{l}\left(\frac{1}{m_j}\sum_{x_r \in NH(j)}|X(j,i)-X(r,i)| + \sum_{y\neq y_j}\frac{1}{h_{jy}}\frac{p(y)}{1-p(y)}\sum_{x_r \in NM(j,y)}|X(j,i)-X(r,i)|\right) \quad \textit{PT 3.8}$$

Trong đó, $NH(j)$ và $NM(j,y)$ là tập k bản ghi gần với x_j nhất nằm cùng lớp và cùng thuộc lớp y, m_j và h_{jy} là số phần tử của các tập này. $p(y)$ là xác suất tiền định của lớp y.

Thuật toán lựa chọn đặc trưng sử dụng điểm số ReliefF (sau đây gọi tắt là thuật toán ReliefF) thực hiện tính điểm $reliefF\_score$ cho từng đặc trưng một cách độc lập, sau đó lựa chọn tham lam k đặc trưng có điểm số **lớn nhất**.

Với bộ dữ liệu Điểm thi, bảng kết quả sau thể hiện điểm số ReliefF của các đặc trưng và thứ tự các đặc trưng được sắp xếp theo điểm số **giảm dần** (độ quan trọng giảm dần).

Tham số	f_1	f_2	f_3	Đặc trưng quan trọng
$k=3, l=10$	0,467	0,467	7,733	f_3, f_2, f_1
$k=4, l=10$	0,65	1,9	9,2	f_3, f_2, f_1
$k=5, l=10$	0,16	1,14	6,8	f_3, f_2, f_1

Bảng 3.7 Điểm số ReliefF các đặc trưng bộ dữ liệu Điểm thi

Kết quả tổng hợp các thuật toán lựa chọn đặc trưng dựa trên sự tương tự với bộ dữ liệu Điểm thi:

Thuật toán	Thứ tự đặc trưng (độ quan trọng giảm dần)
Laplacian	f_3, f_2, f_1
Fisher	f_3, f_1, f_2
SPEC	f_1, f_2, f_3
Trace Ratio	f_1, f_2, f_3
ReliefF	f_3, f_2, f_1

Bảng 3.8 Kết quả các thuật toán dựa trên sự tương tự với bộ dữ liệu Điểm thi

CHƯƠNG 4. CÁC THUẬT TOÁN LỰA CHỌN ĐẶC TRƯNG SỬ DỤNG ĐIỂM SỐ DỰA TRÊN LÝ THUYẾT THÔNG TIN

Chương này sẽ trình bày các thuật toán lựa chọn đặc trưng sử dụng điểm số dựa trên lý thuyết thông tin. Các điểm số này sử dụng Entropy để làm thước đo cho sự liên quan giữa đặc trưng với nhãn và các đặc trưng với nhau. Để minh họa các thuật toán hiệu quả hơn, tôi tiếp tục sử dụng bộ dữ liệu Điểm thi đã giới thiệu và sử dụng trong Chương 3.

4.1 Các khái niệm

Trước tiên hãy bắt đầu với một số khái niệm và công thức.

4.1.1 En-trô-py (Entropy)

Entropy của biến ngẫu nhiên rời rạc X được tính theo công thức:

$$H(X) = -\sum_{x_i \in X} P(x_i)\log\left(P(x_i)\right) \qquad PT\ 4.1$$

Trong đó, $P(x_i)$ là xác suất xảy ra $X = x_i$

Entropy có điều kiện của biến X khi biết biến Y là:

$$H(X|Y) = -\sum_{y_j \in Y} P(y_j) \sum_{x_i \in X} P(x_i|y_j)\log\left(P(x_i|y_j)\right) \qquad PT\ 4.2$$

Trong đó, $P(x_i|y_j)$ là xác suất xảy ra $X = x_i$ khi biết $Y = y_i$

4.1.2 Thông tin tương hỗ

Lượng thông tin tương hỗ giữa hai biến ngẫu nhiên rời rạc X và Y được tính theo công thức:

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= \sum_{x_i \in X} \sum_{y_j \in Y} P(x_i, y_j)\log \frac{P(x_i, y_j)}{P(x_i)P(y_j)} \end{aligned} \qquad PT\ 4.3$$

Trong đó, $P(x_i, y_j)$ là xác suất đồng thời xảy ra $X = x_i$ và $Y = y_i$. Lượng thông tin tương hỗ càng lớn thì mối quan hệ hay sự phụ thuộc giữa hai biến càng lớn.

4.1.3 Công thức chung

Các thuật toán lựa chọn đặc trưng sử dụng điểm số dựa trên lý thuyết thông tin (gọi tắt là các thuật toán dựa trên lý thuyết thông tin) đều tuân theo công thức dạng tuyến tính của Shannon [3]:

$$J_x(f_i) = I(f_i;Y) - \beta \sum_{f_j \epsilon S} I(f_i;f_j) + \gamma \sum_{f_j \epsilon S} I(f_j, f_i|Y) \qquad PT\ 4.4$$

Trong đó $J_x(f_i)$ là điểm số được tính cho đặc trưng f_i, β và γ là các tham số cho trước, nhận giá trị từ 0 đến 1, còn Y là không gian nhãn. Các thuật toán thường bắt đầu với tập con đặc trưng S rỗng. Tại mỗi bước lặp, tính điểm cho tất cả các đặc trưng, sau đó chọn ra đặc trưng có điểm số cao nhất để thêm vào tập con đặc trưng. Quá trình này sẽ dừng khi nhận được đủ số đặc trưng mong muốn.

Đại lượng $I(f_i|Y)$ thể hiện sự liên quan, tức là sự phụ thuộc giữa đặc trưng f_i và nhãn. Giá trị $I(f_i|Y)$ lớn cho thấy đặc trưng đó có khả năng phân loại nhãn, tức là một đặc trưng quan trọng, ngược lại cho thấy đặc trưng đó không có ảnh hưởng gì đến nhãn, tức là một đặc trưng không liên quan.

Đại lượng $\sum_{f_j \epsilon S} I(f_i|f_j)$ thể hiện sự dư thừa, tức là sự phụ thuộc giữa đặc trưng f_i và các đặc trưng f_j đã được thêm vào tập con đặc trưng đang xét. Giá trị này lớn cho thấy đặc trưng mà ta đang xét phụ thuộc rất nhiều vào các đặc trưng đã chọn.

Đại lượng $\sum_{f_j \epsilon S} I(f_j, f_i|Y)$ thể hiện sự dư thừa có điều kiện, tức là sự phụ thuộc giữa đặc trưng f_i khi biết nhãn và các đặc trưng f_j đã được thêm vào tập con đặc trưng đang xét. Giá trị này lớn cho thấy đặc trưng mà ta đang và các đặc trưng đã chọn đều ảnh hưởng rất lớn đến nhãn.

Việc lựa chọn các tham số β và γ phụ thuộc vào việc đánh giá độ quan trọng của các thành phần đó đối với sự liên quan của đặc trưng. Rất khó để chọn được các tham số β và γ sao cho tối ưu, nên chúng thường được chọn một cách đơn giản, để dễ dàng diễn giải, ví dụ bằng 0, bằng 1 hoặc bằng nghịch đảo của số phần tử tập con đặc trưng đang xét.

Các thuật toán dựa trên lý thuyết thông tin có ưu điểm là có tính đến sự phụ thuộc giữa các đặc trưng, có thể tránh được sự ảnh hưởng của các giá trị ngoại lai do đó thường cho kết quả tốt hơn các thuật toán dựa trên sự tương tự.

Nhược điểm của các thuật toán này là chỉ áp dụng được với dữ liệu đã biết nhãn tức bài toán lựa chọn đặc trưng có giám sát. Hơn nữa các dữ liệu phải là dữ liệu rời rạc, các dữ liệu liên tục cần phải được rời rạc hóa. Thời gian, chi phí tính toán có thể sẽ cao hơn so với các thuật lựa chọn đặc trưng dựa trên sự tương tự.

4.2 Thuật toán lựa chọn đặc trưng sử dụng điểm số MIM

Điểm số MIM (Mutual Information Maximization) được Lewis và cộng sự giới thiệu năm 1992 [14], sử dụng cả hai tham số β và γ đều bằng 0.

$$J_{MIM}(f_i) = I(f_i|Y) \qquad \textit{PT 4.5}$$

Thuật toán lựa chọn đặc trưng sử dụng điểm số MIM (gọi tắt là thuật toán MIM) tính điểm J_{MIM} cho tất cả đặc trưng, sau đó lựa chọn tham lam k đặc trưng có điểm số J_{MIM} **cao nhất**.

Ưu điểm của thuật toán MIM là đơn giản và dễ dàng triển khai và hiệu năng cao.

Nhược điểm của MIM là xét các đặc trưng một cách độc lập, bỏ qua sự liên quan, phụ thuộc giữa các đặc trưng với nhau.

Với bộ dữ liệu Điểm thi, điểm số MIM cho các đặc trưng được thể hiện trong bảng. Sau đó các đặc trưng được sắp xếp theo thứ tự điểm số **giảm dần**

Bước lặp	f_1	f_2	f_3	Đặc trưng quan trọng
Lặp lần 1	0,571	0,571	0,771	f_3, f_2, f_1

Bảng 4.1 Điểm số MIM các đặc trưng bộ dữ liệu Điểm thi

4.3 Thuật toán lựa chọn đặc trưng sử dụng điểm số MIFS

Trên thực tế, các đặc trưng tốt không chỉ liên quan mạnh với nhãn mà còn phải không liên quan đến các đặc trưng khác. Nói cách khác, nên cực tiểu hóa sự liên quan giữa các đặc trưng với nhau. Điểm số MIFS (Mutual Information Feature Selection) được Battiti và cộng sự giới thiệu năm 1994 [25] sử dụng γ bằng 0 và β nhận giá trị từ 0 đến 1.

$$J_{MIFS}(f_i) = I(f_i|Y) - \beta \sum_{f_j \epsilon S} (f_i|f_j) \qquad PT\ 4.6$$

Thuật toán lựa chọn đặc trưng sử dụng điểm số MIFS (gọi tắt là thuật toán MIFS) bắt đầu với tập con đặc trưng S rỗng. Tại mỗi bước lặp, thực hiện tính điểm J_{MIFS} cho các đặc trưng chưa có trong tập S, chọn ra đặc trưng có điểm số **cao nhất** để thêm vào S. Thuật toán sẽ dừng khi nhận được đủ số đặc trưng mong muốn.

Với bộ dữ liệu Điểm thi, (chọn tham số $\beta = 0{,}1$). Bảng kết quả sau thể hiện điểm số MIFS của các đặc trưng ở mỗi bước lặp, thứ tự đặc trưng theo điểm số **giảm dần** và đặc trưng được chọn thêm vào tập S.

Tập S	f_1	f_2	f_3	Thứ tự đặc trưng	Chọn
$S = \{\}$	0,571	0,571	0,771	f_3, f_1, f_2	f_3
$S = \{f_3\}$	0,319	0,339		f_2, f_1	f_2
$S = \{f_3, f_2\}$	0,087			f_1	f_1

Bảng 4.2 Điểm số MIFS các đặc trưng bộ dữ liệu Điểm thi

4.4 Thuật toán lựa chọn đặc trưng sử dụng điểm số MRMR

Điểm số MRMR (Minimum Redundancy Maximum Relevance) được Peng và cộng sự giới thiệu năm 2005 [15], sử dụng tham số β bằng nghịch đảo của số phần tử của tập con đặc trưng S đang xét, trong khi γ bằng 0.

$$J_{MRMR}(f_i) = I(f_i|Y) - \frac{1}{|S|} \sum_{f_j \epsilon S} (f_i|f_j) \qquad PT\ 4.7$$

Ý tưởng chính ở đây là một đặc trưng phụ thuộc vào một hoặc vài đặc trưng khác thì là một đặc trưng dư thừa, nhưng nếu nó phụ thuộc vào rất nhiều đặc trưng thì tính dư thừa sẽ kém hơn rất nhiều. Chọn $\beta = \frac{1}{|S|}$ làm cho càng nhiều đặc trưng được chọn, ảnh hưởng của sự phụ thuộc vào các đặc trưng đã chọn càng giảm. Nói cách khác, càng nhiều đặc trưng được chọn, đặc trưng đang xét càng khó trở thành đặc trưng dư thừa.

Thuật toán lựa chọn đặc trưng sử dụng điểm số MRMR (gọi tắt là thuật toán MRMR) tương tự như thuật toán MIFS ở mục 4.3: Bắt đầu với tập con đặc trưng S rỗng. Tại mỗi bước lặp, thực hiện tính điểm J_{MRMR} cho các đặc trưng chưa có trong tập S, chọn ra đặc trưng có điểm số **cao nhất** để thêm vào S. Thuật toán sẽ dừng khi nhận được đủ số đặc trưng mong muốn.

Với bộ dữ liệu Điểm thi, bảng kết quả sau thể hiện điểm số MRMR của các đặc trưng ở mỗi bước lặp, thứ tự đặc trưng theo điểm số **giảm dần** và đặc trưng được chọn thêm vào tập S

Tập S	f_1	f_2	f_3	Thứ tự đặc trưng	Chọn
$S = \{\}$	0,571	0,571	0,771	f_3, f_1, f_2	f_3
$S = \{f_3\}$	-1,951	-1,751		f_2, f_1	f_2
$S = \{f_3, f_2\}$	-1,851			f_1	f_1

Bảng 4.3 Điểm số MRMR các đặc trưng bộ dữ liệu Điểm thi

4.5 Thuật toán lựa chọn đặc trưng sử dụng điểm số CIFE

Điểm số CIFE (Conditional Infomax Feature Extraction) được Lin, Tang và cộng sự giới thiệu năm 2006 [26], sử dụng cả hai tham số β và γ đều bằng 1.

$$J_{CIFE}(f_i) = I(f_i|Y) - \sum_{f_j \epsilon S} (f_i|f_j) + \sum_{f_j \epsilon S} (f_j, f_i|Y) \quad PT\ 4.8$$

So với các thuật toán MIFS và MRMR, thuật toán CIFE có thêm thành phần thứ ba để cực đại hóa sự dư thừa có điều kiện. Thuật toán CIFE không chỉ quan tâm đến sự phụ thuộc giữa các đặc trưng, mà còn xem xét mối quan hệ đó khi biết nhãn. Sự phụ thuộc giữa đặc trưng đang xét khi biết nhãn và các đặc trưng đã chọn nên cực đại hóa.

Thuật toán lựa chọn đặc trưng sử dụng điểm số CIFE (gọi tắt là thuật toán CIFE) bắt đầu với tập con đặc trưng S rỗng. Tại mỗi bước lặp, thực hiện tính điểm J_{CIFE} cho các đặc trưng chưa có trong tập S, chọn ra đặc trưng có điểm số **cao nhất** để thêm vào S. Thuật toán sẽ dừng khi nhận được đủ số đặc trưng mong muốn.

Với bộ dữ liệu Điểm thi, bảng kết quả sau thể hiện điểm số CIFE của các đặc trưng ở mỗi bước lặp, thứ tự đặc trưng theo điểm số **giảm dần** và đặc trưng được chọn thêm vào tập S

Tập S	f_1	f_2	f_3	Thứ tự đặc trưng	Chọn
$S = \{\}$	0,571	0,571	0,771	f_3, f_1, f_2	f_3
$S = \{f_3\}$	0,199	0,199		f_1, f_2	f_1
$S = \{f_3, f_2\}$	0,029			f_2	f_2

Bảng 4.4 Điểm số CIFE các đặc trưng bộ dữ liệu Điểm thi

4.6 Thuật toán lựa chọn đặc trưng sử dụng điểm số JMI

Điểm số JMI (Joint Mutual Information) được Meyer và cộng sự giới thiệu năm 2008 [17], sử dụng cả hai tham số β và γ đều bằng nghịch đảo của số phần tử của tập con đặc trưng S đang xét.

$$J_{JMI}(f_i) = I(f_i|Y) - \frac{1}{|S|}\sum_{f_j \epsilon S} (f_i|f_j) + \frac{1}{|S|}\sum_{f_j \epsilon S} (f_j, f_i|Y) \quad PT\ 4.9$$

Điểm số JMI cũng có thêm thành phần thứ ba để cực đại hóa sự dư thừa có điều kiện, nhưng sử dụng tham số thích nghi $\frac{1}{|S|}$.

Thuật toán lựa chọn đặc trưng sử dụng điểm số JMI (gọi tắt là thuật toán JMI) bắt đầu với tập con đặc trưng S rỗng. Tại mỗi bước lặp, thực hiện tính điểm J_{JMI} cho các đặc trưng chưa có trong tập S, chọn ra đặc trưng có điểm số **cao nhất** để thêm vào S. Thuật toán sẽ dừng khi nhận được đủ số đặc trưng mong muốn.

Với bộ dữ liệu Điểm thi, bảng kết quả sau thể hiện điểm số của các đặc trưng ở mỗi bước lặp, thứ tự đặc trưng theo điểm số **giảm dần** và đặc trưng được chọn thêm vào tập S.

Tập S	f_1	f_2	f_3	Thứ tự đặc trưng	Chọn
$S = \{\}$	0,571	0,571	0,771	f_3, f_1, f_2	f_3
$S = \{f_3\}$	0,199	0,199		f_1, f_2	f_1
$S = \{f_3, f_2\}$	0,299			f_2	f_2

Bảng 4.5 Điểm số JMI các đặc trưng bộ dữ liệu Điểm thi

Nhận xét: Các thuật toán dựa trên lý thuyết thông tin đem lại hiệu năng cao và ổn định, khắc phục được hai vấn đề: Xem xét sự các đặc trưng dư thừa, tránh được các điểm dữ liệu ngoại lai. Tuy nhiên các thuật toán này cần nhiều chi phí tính toán và tài nguyên hơn.

Kết quả tổng hợp của các thuật toán lựa chọn đặc trưng dựa trên lý thuyết thông tin với bộ dữ liệu Điểm số. Kết quả cho biết thứ tự quan trọng của đặc trưng (thứ tự thêm vào tập S)

Thuật toán	Đặc trưng quan trọng
MIM	f_3, f_2, f_1
MIFS	f_3, f_2, f_1
MRMR	f_3, f_2, f_1
CIFE	f_3, f_1, f_2
JMI	f_3, f_1, f_2

Bảng 4.6 Kết quả các thuật toán dựa trên lý thuyết thông tin với bộ dữ liệu Điểm thi

CHƯƠNG 5. THỰC NGHIỆM

Chương này trình bày quá trình và kết quả thực nghiệm, bao gồm dữ liệu thực nghiệm, phương pháp thực nghiệm, phương pháp đánh giá, kết quả thực nghiệm, so sánh và phân tích các kết quả. Kết quả chi tiết được trình bày ở phần Phụ lục của đồ án này.

5.1 Dữ liệu thực nghiệm

Để có được kết quả thực nghiệm chính xác và khách quan, cần sử dụng các dữ liệu thực tế, chính xác. Các bộ dữ liệu được sử dụng để thực nghiệm là các bộ dữ liệu vừa và lớn thuộc lĩnh vực công nghệ - tài chính (fintech). Các bộ dữ liệu bao gồm các đặc trưng như thông tin cá nhân, gia đình, các thông tin về tài chính, tín dụng, các thông tin về lịch sử giao dịch tài chính, ghi nợ. Mỗi bản ghi có thể là một cá nhân, một tổ chức hoặc một giao dịch tài chính. Các bộ dữ liệu gồm 2 lớp, giá trị nhãn gồm nhãn 0 (tốt, không,..) và nhãn 1 (xấu, không,..). Nhãn có tỉ lệ không cân bằng, tỉ lệ nhãn 0 luôn chiếm đa số (chiếm từ 69 đến 99%).

Bảng tóm tắt thông tin chung của các bộ dữ liệu

STT	Tên bộ dữ liệu	Số đặc trưng	Số bản ghi	Nhãn	Ứng dụng
1	CreditScore	304	80000	Phân loại tín dụng xấu	Chấm điểm tín dụng cá nhân
2	FintechUser	29	27000	Phân loại khách hàng	Dự đoán khách hàng quay lại mua hàng
3	CreditCard	29	284807	Phân loại giao dịch	Phát giao dịch thẻ gian lận
4	HomeCredit	120	307513	Phân loại nhóm nợ xấu	Phân loại nhóm nợ xấu
5	HomeCredit2	300	307516		
6	CreditRisk	71	855000		
7	FinancialRisk	45	1048575		
8	VehicleLoan	39	233154		

Bảng 5.1 Thông tin cơ bản về các bộ dữ liệu

Sau đây là thông tin chi tiết về các bộ dữ liệu.

5.1.1 CreditScore

Bộ dữ liệu về tài chính cá nhân để chấm điểm tín dụng cá nhân bao gồm 304 đặc trưng. Có 80000 bản ghi tương ứng với dữ liệu của 80000 cá nhân. Các đặc trưng bao gồm thông tin cá nhân và các thông tin về tài chính như tài sản, các khoản nợ, các tài khoản, thẻ tín dụng,... Nhãn phân loại mức tín dụng của cá nhân: nhãn 0 ứng với tín dụng tốt, điểm tín dụng lớn hơn 500, nhãn 1 ứng với tín dụng xấu, có nguy cơ vỡ nợ. Tỉ lệ nhãn 0 chiếm đa số với 85%.

5.1.2 FintechUser

Bộ dữ liệu về tài chính cá nhân do một công ty thu thập để phục vụ cho việc bán hàng, bao gồm 29 đặc trưng và 27000 bản ghi tương ứng với 27000 cá nhân. Các đặc trưng bao gồm tuổi, sở hữu nhà, điểm tín dụng, phương thức thanh toán, sử dụng nền tảng web, android hay ios, có đang ghi nợ,… Nhãn phân cá nhân dựa trên việc quay lại mua hàng: nhãn 0 tương ứng với khách hàng không quay lại mua hàng, nhãn 1 tương ứng với khách hàng có trở lại mua hàng. Tỉ lệ nhãn 0 chiếm 68%, tức là có 32% khách hàng tiếp tục mua hàng sau khi đã mua sản phẩm của công ty.

5.1.3 CreditCard

Bộ dữ liệu về giao dịch thẻ tín dụng và phân loại giao dịch gian lận. Dữ liệu gồm 29 đặc trưng và 284807 bản ghi tương ứng với 284807 giao dịch bằng thẻ tín dụng diễn ra trong hai ngày, được ghi nhận tại Đức. Các đặc trưng bao gồm thời điểm giao dịch, thông tin giao dịch, số lượng giao dịch,… Nhãn phân loại giao dịch gian lận với nhãn 1 và giao dịch bình thường với nhãn 0. Tỉ lệ nhãn 0 chiếm đến 99,8%, tức là chỉ có 0,2% số giao dịch là gian lận.

5.1.4 HomeCredit

Bộ dữ liệu về các khoản nợ cá nhân và phân loại nhóm nợ. Dữ liệu gồm 120 đặc trưng và 307513 bản ghi tương ứng với 307513 khoản nợ cá nhân. Các đặc trưng bao gồm các thông tin về khoản nợ, thông tin cá nhân, gia đình, tài sản, công việc, các thông tin tín dụng mới và lịch sử tín dụng. Nhãn phân loại nhóm nợ tín dung của khoản nợ: dư nợ đủ chuẩn với nhãn 0 tức là trả nợ đúng hạn, dư nợ xấu với nhãn 1 tức trả nợ quá hạn một số ngày nhất định, trì hoãn trả nợ hoặc trốn nợ. Tỉ lệ nhãn 0 chiếm đến 92%, tức là có 8% số khoản nợ là nợ xấu.

5.1.5 HomeCredit2

Bộ dữ liệu về các khoản nợ cá nhân và phân loại nhóm nợ. Dữ liệu gồm 300 đặc trưng và 307516 bản ghi tương ứng với 307516 khoản nợ cá nhân. Các đặc trưng bao gồm các thông tin về khoản nợ (số lượng, phân loại, kì hạn, cách thức thanh toán,..), thông tin cá nhân (tuổi, giới tính, nghề nghiệp), gia đình (vợ chồng, con cái), tài sản (sở hữu ô tô, xe máy), các thông tin tín dụng mới (tạo tài khoản, thẻ mới, mua xe,…) và lịch sử tín dụng (các khoản nợ, số lượng tài khoản, thời gian giao dịch…). Nhãn phân loại nhóm nợ tín dung của khoản nợ: dư nợ đủ chuẩn với nhãn 0 tức là trả nợ đúng hạn, dư nợ xấu với nhãn 1 tức trả nợ quá hạn một số ngày nhất định, trì hoãn trả nợ hoặc trốn nợ. Tỉ lệ nhãn 0 chiếm đến 92%, tức là có 8% số khoản nợ là nợ xấu.

5.1.6 CreditRisk

Bộ dữ liệu về các khoản nợ cá nhân và phân loại nhóm nợ. Dữ liệu gồm 73 đặc trưng và 855969 bản ghi tương ứng với 855969 khoản nợ cá nhân. Các đặc trưng bao gồm các thông tin về khoản nợ (số lượng, phân loại, kì hạn, cách thức thanh toán,..), thông tin cá nhân, gia đình (vợ chồng, con cái), tài sản (sở hữu ô tô, xe máy), công việc, các thông tin tín dụng mới (tạo tài khoản, thẻ mới, mua xe,..) và các tài liệu, văn bản đã cung cấp. Nhãn phân loại nhóm nợ tín dung của khoản nợ:

dư nợ đủ chuẩn với nhãn 0 tức là trả nợ đúng hạn, dư nợ xấu với nhãn 1 tức trả nợ quá hạn một số ngày nhất định, trì hoãn trả nợ hoặc trốn nợ. Tỉ lệ nhãn 0 chiếm đến 95%, tức là có 5% số khoản nợ là nợ xấu.

5.1.7 FinancialRisk

Bộ dữ liệu về các khoản nợ cá nhân và phân loại nhóm nợ. Dữ liệu gồm 45 đặc trưng và 1048575 bản ghi tương ứng với 1048575 khoản nợ cá nhân. Các đặc trưng bao gồm các thông tin về khoản nợ (số lượng, phân loại, kì hạn, cách thức thanh toán,..), thông tin cá nhân, tài sản (sở hữu ô tô, xe máy), và và các tài liệu, văn bản đã cung cấp. Nhãn phân loại nhóm nợ tín dung của khoản nợ: dư nợ đủ chuẩn với nhãn 0 tức là trả nợ đúng hạn, dư nợ xấu với nhãn 1 tức trả nợ quá hạn một số ngày nhất định, trì hoãn trả nợ hoặc trốn nợ. Tỉ lệ nhãn 0 chiếm đến 80%, tức là có 20% số khoản nợ là nợ xấu.

5.1.8 VehicleLoan

Bộ dữ liệu về các khoản nợ mua xe cá nhân và phân loại nhóm nợ. Dữ liệu gồm 73 đặc trưng và 855969 bản ghi tương ứng với 855969 khoản nợ cá nhân. Các đặc trưng bao gồm các thông tin cá nhân (tuổi, lương, nghề nghiệp) thông tin về khoản nợ (số lượng, phân loại, kì hạn, cách thức thanh toán,..) và lịch sử tín dụng (số lượng tài khoản, các khoản nợ khác). Nhãn phân loại nhóm nợ tín dung của khoản nợ: dư nợ đủ chuẩn với nhãn 0 tức là trả nợ đúng hạn, dư nợ xấu với nhãn 1 tức trả nợ quá hạn một số ngày nhất định, trì hoãn trả nợ hoặc trốn nợ. Tỉ lệ nhãn 0 chiếm đến 78%, tức là có 22% số khoản nợ là nợ xấu.

5.2 Phương pháp thực nghiệm

5.2.1 Tái tạo các thuật toán lựa chọn đăc trưng

Việc thực hiện tái tạo các thuật toán lựa chọn đặc trưng được thực trên nền tảng Google Colab, sử dụng ngôn ngữ lập trình python (phiên bản 3.10) cùng các thư viện liên quan.

Hình 5.1 Google Colab cho phép thực thi Python trên trình duyệt

Google Colab cho phép chúng ta viết và thực thi Python ngay trên trình duyệt với các ưu điểm:

- Không yêu cầu cấu hình
- Sử dụng miễn phí GPU
- Chia sẻ dễ dàng

Colab phù hợp với cả sinh viên, các nhà nghiên cứu AI, giúp hoàn thành công việc dễ dàng hơn.

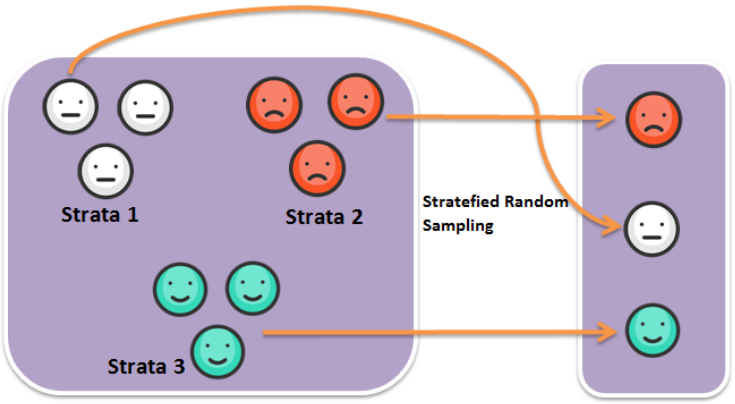
5.2.2 Lấy mẫu

Dữ liệu được đọc vào thông qua pandas, một thư viện của python cho phép đọc các tập tin định dạng csv và xử lí dưới dạng dataframe. Một ma trận X lưu trữ dữ liệu các bản ghi với toàn bộ đặc trưng không bao gồm id và nhãn, vecto y lưu trữ nhãn của các bản ghi.

Với chi phí tài nguyên giới hạn, các thuật toán không thể hoàn thành việc lựa chọn đặc trưng trên các bộ dữ liệu đã giới thiệu. Các bộ dữ liệu này có kích thước lớn nên với các thuật toán dựa trên sự tương tự, chúng tạo ra các ma trận rất lớn dẫn tới vượt quá giới hạn tài nguyên cho phép dẫn tới chương trình ngừng thực thi.

Cụ thể bộ dữ liệu CreditScore có kích thước 80000 bản ghi và 304 đặc trưng, thuật toán MIM có thể hoàn thành trong hơn 3 giờ (10954,5 giây) trong khi thuật toán Laplacian bị tràn bộ nhớ và ngừng thực thi khi thực đang thi được 8 giây.

Do đó, không mất tính tổng quát, tôi sử dụng phương pháp Lấy mẫu phân tổ (Stratified Sampling) để thu được các bộ dữ liệu với số lượng bản ghi nhỏ hơn mà vẫn giữ được toàn bộ đặc trưng và các đặc tính ban đầu.



Hình 5.2 Minh họa kỹ thuật lấy mẫu phân tổ. Nguồn: Internet

Stratified Sampling là phương pháp mà các đơn vị mẫu được chọn khi tổng thể chung đã được phân chia thành các tổ/lớp/nhóm theo tiêu chí liên quan trực tiếp đến mục đích nghiên cứu. Trong đồ án náy, tôi sử dụng kĩ thuật Stratified Sampling được tích hợp sẵn trong thư viện Pandas, cho phép lấy mẫu theo số lượng bản ghi mong muốn mà vẫn giữ được tỉ lệ giữa các lớp.

```
[ ] #total_samples = 80000
    n_samples = 20000;
    df = df.groupby('y', group_keys=False).
      apply(lambda x: x.sample(frac=n_samples/total_samples))
```

Hình 5.3 Lấy mẫu phân tổ trong Pandas

Áp dụng kỹ thuật lấy mẫu phân tổ với các bộ dữ liệu để thu được bộ dữ liệu nhỏ phù hợp với tất cả các thuật toán.

Kích thước thực nghiệm của cá bộ dữ liệu được thể hiện trong bảng:

Bộ dữ liệu	Số đặc trưng	Tổng số bản ghi	Số bản ghi thực nghiệm
CreditScore	304	80000	12500
FintechUser	29	27000	13000
CreditCard	29	284807	10000
HomeCredit	120	307513	11000
HomeCredit2	300	307516	11000
CreditRisk	71	855000	11500
FinancialRisk	45	1048575	14500
VehicleLoan	39	233154	13500

Bảng 5.2 Kích thước các bộ dữ liệu thực nghiệm

Các bộ dữ liệu sau khi rút gọn sẽ là dữ liệu chính thức cho quá trình thực nghiệm. Dữ liệu sau đó được tách id và nhãn để lưu vào các ma trận X và y.

5.2.3 Ngưỡng lựa chọn đặc trưng

Các thuật toán được cài đặt để lựa chọn số lượng đặc trưng nhất định, đó là các ngưỡng 10%, 15% và 25% số đặc trưng ban đầu.

Bộ dữ liệu	Số đặc trưng	Ngưỡng 10%	Ngưỡng 15%	Ngưỡng 25%
CreditScore	304	30	45	76
FintechUser	29	3	4	7
CreditCard	29	3	4	7
HomeCredit	120	12	18	30
HomeCredit2	300	30	45	75
CreditRisk	71	7	10	17
FinancialRisk	45	4	7	11
VehicleLoan	39	4	6	10

Bảng 5.3 Số đặc trưng lựa chọn theo các ngưỡng của các bộ dữ liệu

5.3 Phương pháp đánh giá

5.3.1 Xây dựng mô hình phân loại

Để đánh giá được kết quả lựa chọn đặc trưng, ta cần xây dựng một mô hình phân loại, mà sẽ sử dụng các đặc trưng đã lựa chọn được để phân loại.

Mô hình phân loại được sử dụng ở đây là CatBoostClassifier, một thuật toán phân loại tương tự như XGBoost, LightGBM,… Tôi chọn sử dụng CatBoost vì tính dễ sử dụng, hiệu quả và hoạt động tốt với các biến phân loại (categorical variable).

Các bước xây dựng mô hình phân loại:

- Dữ liệu đầu vào: Dữ liệu với các đặc trưng đã chọn
- Phân chia dữ liệu train/test: tỉ lệ dữ liệu test chiếm 20%
- Huấn luyện mô hình với tập train
- Dự đoán, phân loại dữ liệu test
- Tính toán các độ đo, ghi lại kết quả

Kích thước dữ liệu train, test (số bản ghi) được thể hiện trong bảng:

Bộ dữ liệu	Dữ liệu thực nghiệm	Dữ liệu train (80%)	Dữ liệu test (20%)
CreditScore	12500	10000	2500
FintechUser	13000	10400	2600
CreditCard	10000	8000	2000
HomeCredit	11000	8800	2200
HomeCredit2	11000	8800	2200
CreditRisk	11500	9200	2300
FinancialRisk	14500	11600	2900
VehicleLoan	13500	10800	2700

Bảng 5.4 Kích thước dữ liệu train và test

5.3.2 Đánh giá khả năng phân loại của mô hình

Sau đây là các phương pháp, độ đo để đánh giá một mô hình phân loại, từ đó có thể đánh giá được các thuật toán lựa chọn đặc trưng.

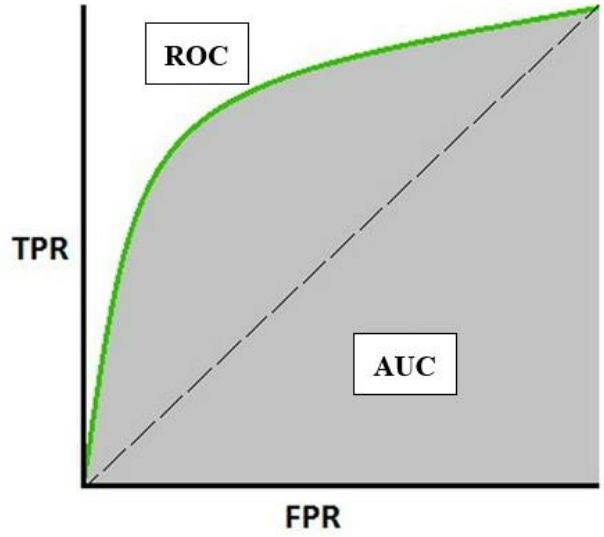
5.3.2.1. Độ chính xác (Accuracy)

Độ chính xác chỉ ra khả năng phân loại chính xác của mô hình. Độ chính xác được tính bằng tỉ lệ giữa số mẫu phân loại chính xác so với tổng số mẫu phân loại.

Độ chính xác sẽ không có ý nghĩa nếu như tỉ lệ giữa các lớp mất cân bằng. Đối với các bộ dữ liệu thuộc lĩnh vực tài chính công nghệ thì tỉ lệ lớp luôn không đều, tỉ lệ mẫu âm tính có thể lên tới 99%, khi đó mô hình mặc dù phân loại sai hoàn toàn các mẫu dương tính nhưng độ chính xác vẫn ở mức 99%. Do đó, độ chính xác chỉ nên là một trong các thước đo tham khảo.

5.3.2.2. Chỉ số AUC – ROC

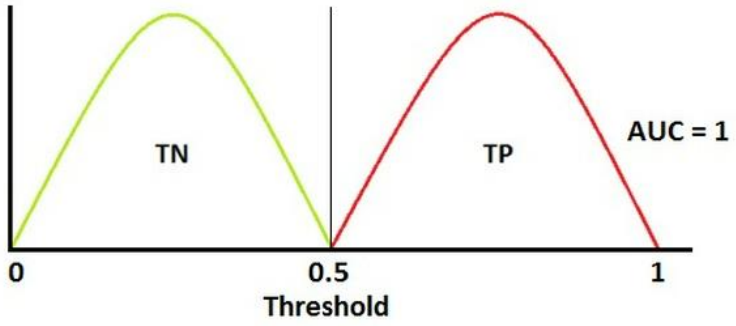
AUC là viết tắt của Area Under the Curve, còn ROC là Receiver Operating Characteristics. AUC-ROC là một phương pháp tính toán hiệu năng của một mô hình phân loại theo các ngưỡng phân loại khác nhau. ROC là một đường cong biểu diễn xác suất và AUC biểu diễn mức độ phân loại của mô hình. AUC-ROC còn được viết là AUROC (Area Under the Receiver Operating Characteristics). Ý nghĩa của AUROC có thể hiểu: Là xác suất rằng một mẫu dương tính được lấy ngẫu nhiên sẽ được xếp hạng cao hơn một mẫu âm tính được lấy ngẫu nhiên.



Hình 5.4 Đường cong ROC và chỉ số AUC

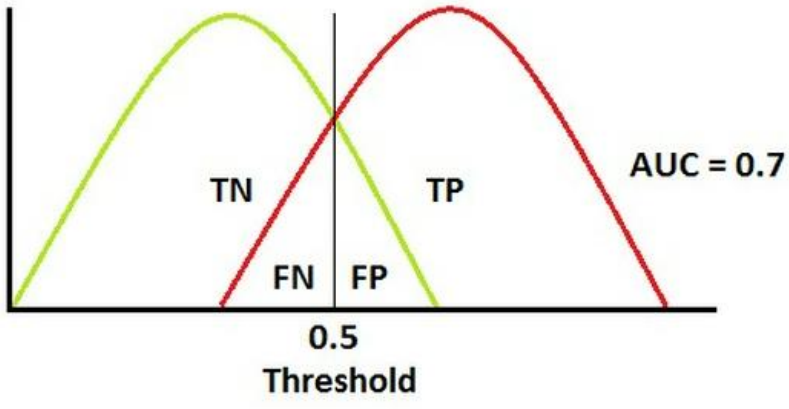
Chỉ số AUC có giá trị trong đoạn từ 0 tới 1. Giá trị của AUC càng cao thì mô hình phân loại càng chính xác.

- $AUC \approx 1$: Đây là trường hợp tốt nhất, mô hình phân loại hoàn toàn chính xác



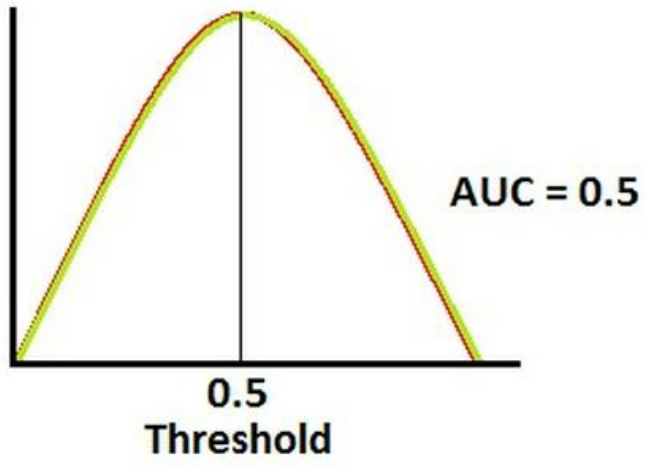
Hình 5.5 Minh họa trường hợp chỉ số AUC bằng 1

- $AUC \approx 0{,}7 \div 0{,}8$: Đây là trường hợp phổ biến, mô hình có hiệu năng phân loại tương đối và ổn định.



Hình 5.6 Minh họa trường hợp chỉ số AUC bằng 0.7

- $AUC \approx 0{,}5$: Đây là trường hợp tồi nhất, mô hình hoàn toàn không có khả năng phân loại.



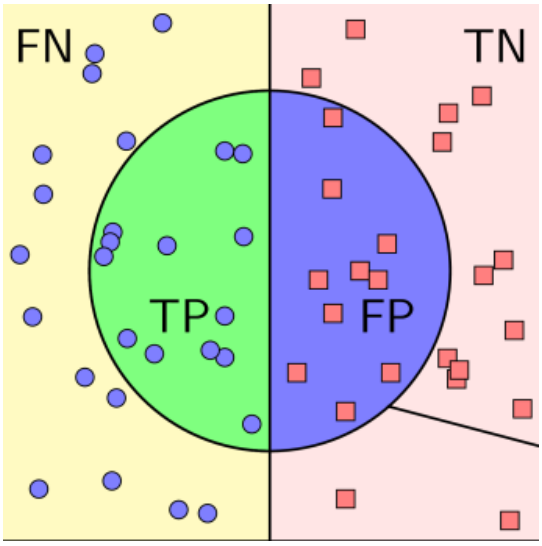
Hình 5.7 Minh họa trường hợp chỉ số AUC bằng 0.5

- $AUC \approx 0$: Trong trường hợp này, mô hình phân loại ngược hoàn toàn hai lớp với việc phân loại âm tính thành dương tính, dương tính thành âm tính. Ta chỉ cần đảo ngược đầu ra của mô hình là có thể sửa được điều này.

Chỉ số AUC phản ánh khả năng phân của mô hình một cách chính xác và hiệu quả hơn, kể cả trong trường hợp dữ liệu có tỉ lệ nhãn lớp mất cân bằng. AUC đặc biệt phù hợp với các dữ liệu về tài chính công nghệ, có thể coi là độ đo quan trọng nhất để đánh giá mô hình từ đó đánh giá các phương pháp lựa chọn đặc trưng.

5.3.2.3. Chỉ số F1-score

F1-score là một độ đo đánh giá khả năng phân loại chính xác của mô hình mà có thể hiệu quả với mô hình phân loại các dữ liệu mất cân bằng.



Hình 5.8 Ví dụ về kết quả phân loại của một mô hình phân loại

F1-score được tính dựa trên hai đại lượng Precesion và Recall. Precesion là tỉ lệ số mẫu dương tính mô hình dự đoán đúng trên tổng số mẫu mà mô hình dự đoán là dương tính: $Precesion = \frac{TP}{TP+FP}$. Precesion càng cao thì mô hình dự đoán các mẫu dương tính càng chính xác, tức là càng ít mẫu âm tính bị dự đoán nhầm là dương tính. Recall là tỉ lệ số mẫu dương tính mà mô hình dự đoán đúng trên tổng số mẫu dương tính thật: $Recall = \frac{TP}{TP+FN}$. Recall càng cao thì mô hình cũng dự đoán các

mẫu dương tính càng chính xác, nhưng càng ít mẫu dương tính bị nhầm thành âm tính.

Tuy nhiên, chỉ có Precesion hay chỉ có Recall thì không đánh giá được chất lượng mô hình.

- Khi $Precesion = 1$ tức là toàn bộ các mẫu mô hình dự đoán dương tính đều chính xác là dương tính nhưng chưa thể kết luận là mô hình phân loại tốt, vì có thể có rất nhiễu mẫu dương tính nhưng mô hình dự đoán là âm tính.
- Khi $Recall = 1$ tức là toàn bộ các mẫu dương tính thì mô hình đều dự đoán là dương tính tuy nhiên ta cũng không thể nói mô hình phân loại tốt, vì có thể có rất nhiễu mẫu âm tính nhưng mô hình dự đoán là dương tính

Đặc biệt là mô hình phân loại các dữ liệu mất cân bằng về nhãn, khi đó F1-score được sử dụng. F1-score là trung bình điều hòa (harmonic mean) của Precesion và Recall, được tính theo công thức:

$$\frac{2}{F1} = \frac{1}{Precesion} + \frac{1}{Recall} \qquad PT\ 5.1$$

F1-score nhận giá trị từ 0 đến 1, giá trị càng cao cho thấy mô hình càng tốt và ngược lại.

- $F1 \approx 1$: Đây là trường hợp tốt nhất, mô hình phân loại hoàn toàn chính xác
- $F1 \approx 0{,}5 \div 0{,}7$: Đây là trường hợp phổ biến, mô hình có hiệu năng phân loại tương đối tốt.
- $F1 \approx 0{,}3$: Đây là trường hợp mô hình có khả năng phân loại nhưng rất kém.
- $F1 \approx 0$: Đây là trường hợp tồi nhất, mô hình hoàn toàn không có khả năng phân loại.

F1 sử dụng trung bình điều hòa yêu cầu cả hai đại lượng Precesion và Recall đều phải cao để cho ra kết quả cao, từ đó có thể đánh giá đúng nhất về mô hình.

5.3.2.4. Thời gian tính toán

Với các bộ dữ liệu nhỏ, thời gian thực hiện của các thuật toán lựa chọn đặc trưng gần như không đáng kể (chỉ chưa đầy 1 giây cho đến vài giây). Trong đồ án này, quá trình thực nghiệm được thực hiện trên các bộ dữ liệu vừa và lớn, do đó thời gian thực hiện của các thuật toán là rất lớn. Do đó chi phí về thời gian cũng là một thước đo để đánh giá một thuật toán lựa chọn đặc trưng.

Thời gian thực hiện được tính từ lúc chương trình bắt đầu thực hiện tính toán (sau khi đọc dữ liệu) cho đến khi kết thúc quá trình lựa chọn đặc trưng

5.3.2.5. Chi phí bộ nhớ

Ngoài yêu cầu về thời gian tính toán, khi lựa chọn các thuật toán lựa chọn đặc trưng, yêu cầu về chi phí bộ nhớ cũng cần được xem xét, đặc biệt là trong trường hợp bộ nhớ vừa phải (không quá lớn), một số thuật toán có thể không thực hiện được.

Các bộ dữ liệu lớn có số lượng bản ghi là rất lớn (vài trăm nghìn bản ghi), việc tính toán trên các bộ dữ liệu này cần chi phí về tài nguyên rất lớn. Đặc biệt là các thuật

toán tạo ra các ma trận tương đồng, chi phí về bộ nhớ là rất lớn. Với chi phí tài nguyên giới hạn, cụ thể cấu hình được sử dụng:

- Vi xử lý: CPU Intel ® Xeon, tốc độ 2.30 Ghz, 2 nhân
- Hệ điều hành: Ubuntu 18.04.5 LTS 64bit
- Bộ nhớ RAM: 12,69 GB
- Bộ nhớ lưu trữ: 211,8 GB

Các thuật toán lựa chọn đặc trưng không thể hoàn thành trên tất cả các bộ dữ liệu, mà chỉ thực hiện được với dữ liệu rút gọn (Lấy mẫu phân tổ). Các thuật toán với từng bộ dữ liệu chỉ có thể hoàn thành với kích thước số lượng bản ghi nhỏ hơn ngưỡng nhất định. Tiến hành thực nghiệm và tìm ra ngưỡng (số bản ghi tối đa) của bộ dữ liệu mà thuật toán có thể hoàn thành với cấu hình giới hạn như trên, có thể chỉ ra chi phí bộ nhớ của các thuật toán.

5.4 Kết quả thực nghiệm và phân tích

Phần này trình bày các kết quả thực nghiệm các thuật toán theo nhóm dựa trên sự tương tự và dựa trên lý thuyết thông tin. Các kết quả được trình bày theo từng tiêu chí: thời gian tính toán, bộ nhớ và hiệu năng của mô hình phân loại.

Để so sánh các thuật toán chi tiết và trực quan hơn, tôi sử dụng biểu đồ so sánh với 03 bộ dữ liệu điển hình, cho các kết quả tốt nhất. Đó là các bộ dữ liệu đại diện cho các ứng dụng khác nhau: CreditScore (chấm điểm tín dụng cá nhân), FintechUser (dự đoán khách hàng mua hàng), CreditCard (phát hiện giao dịch gian lận)).

5.4.1 Thực nghiệm các thuật toán dựa trên sự tương tự

5.4.1.1. Thời gian tính toán của các thuật toán

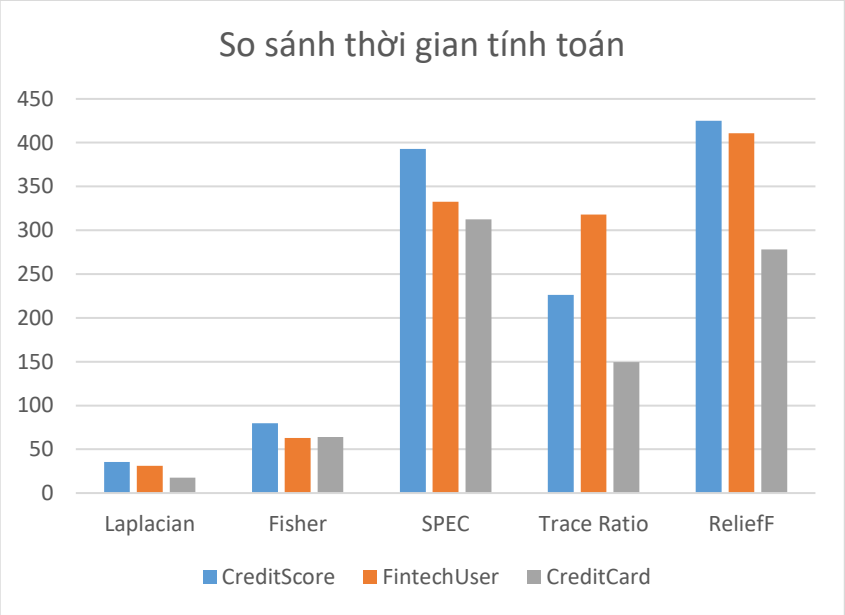
Thời gian thực hiện lựa chọn đặc trưng của các thuật toán dựa trên sự tương tự. Đơn vị: giây

Thuật toán / Bộ dữ liệu	Lapla-cian	Fisher	SPEC	Trace Ratio	ReliefF
CreditScore	**35,6**	79,8	393,0	226,3	425,0
FintechUser	**31,1**	63,1	332,6	317,9	410,9
CreditCard	**17,6**	63,9	312,6	149,7	278,2
HomeCredit	**23,9**	74,3	402,5	-*	256,6
HomeCredit2	**27,8**	69,1	506,3	-	283,7
CreditRisk	**25,4**	69,3	420,0	-	300,1
FinancialRisk	**22,5**	53,9	369,4	-	255,6
VehicleLoan	**34,3**	81,2	668,7	301,8	371,5

Bảng 5.5 Thời gian tính toán các thuật toán dựa trên sự tương tự

(*) Thuật toán Trace Ratio gặp lỗi và không thể hoàn thành lựa chọn đặc trưng.

So sánh chi tiết hơn thời gian thực hiện các thuật toán với các bộ dữ liệu điển hình (CreditScore, FintechUser và CreditCard) bằng biểu đồ trực quan:



Hình 5.9 Biểu đồ so sánh thời gian tính toán của các thuật toán dựa trên sự tương tự

Các thuật toán dựa trên sự tương tự thực hiện tính điểm cho các đặc trưng độc lập và tính trực tiếp không qua nhiều bước lặp nên nhìn chung đều có thời gian tính toán thấp, thấp hơn nhiều so với các thuật toán dựa trên lý thuyết thông tin.

Dựa vào biểu đồ và bảng kết quả ta thấy thời gian tính toán của thuật toán Laplacian thực sự ấn tượng, của thuật toán Fisher cũng rất nhỏ. Thuật toán SPEC và thuật toán ReliefF có thời gian tính toán lớn nhất. Các bộ dữ liệu phản ánh kết quả tương tự nhau. Với bộ dữ liệu CreditScore, thuật toán Laplacian chỉ mất 35,6 giây, thuật toán Fisher cần 79,8 giây và các thuật toán SPEC, Trace Ratio, ReliefF cần lần lượt 393,0 226,3 và 425,0 giây.

Tốc độ thực thi của các thuật toán dựa trên sự tương tự không phụ thuộc nhiều vào số đặc trưng, thời gian thực hiện các bộ dữ liệu CreditCard, FintechUser có ít đặc trưng (29 đặc trưng) cũng không thấp hơn nhiều so với thời gian thực hiện các bộ dữ liệu CreditScore, HomeCredit2 với hơn 300 đặc trưng. Thuật toán Trace Ratio không thể hoàn thành việc lựa chọn đặc trưng với các bộ dữ liệu HomeCredit, HomeCredit2, CreditRisk, FinancialRisk do đó không có số liệu.

5.4.1.2. Chi phí bộ nhớ của các thuật toán

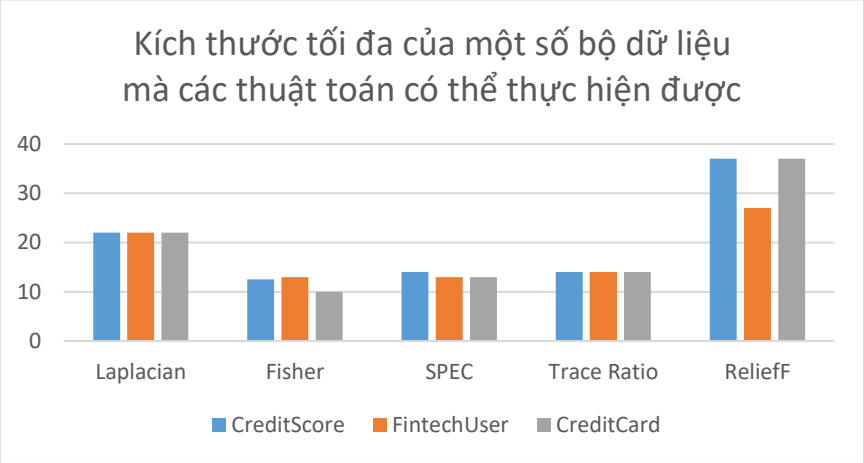
Chi phí bộ nhớ của các thuật toán được phản ánh bởi kích thước tối đa của bộ dữ liệu mà thuật toán có thể thực hiện được (cấu hình đã nêu). Đơn vị: nghìn bản ghi

Thuật toán / Bộ dữ liệu	Số bản ghi	Lapla-cian	Fisher	SPEC	Trace Ratio	ReliefF
CreditScore	80	22	12,5	14	14	**37**
FintechUser	27	22	13	13	14	**27**
CreditCard	284,8	22	10	13	14	**37**
HomeCredit	307,5	20	11	13	-*	**37**
HomeCredit2	307,5	21	11	13	-	**37**
CreditRisk	855	20	11,5	14,5	-	**37**
FinancialRisk	1048,6	22	17,5	14,5	-	**37**
VehicleLoan	233,1	20,5	13,5	14,5	14,5	**38**

Bảng 5.6 Kích thước tối đa các bộ dữ liệu mà các thuật toán dựa trên sự tương tự có thể thực hiện được

(*) Thuật toán Trace Ratio gặp lỗi và không thể hoàn thành lựa chọn đặc trưng.

Biểu đồ trực quan so sánh chi tiết hơn kích thước tối đa mà các thuật toán có thể thực hiện được với các bộ dữ liệu điển hình (CreditScore, FintechUser và CreditCard):



Hình 5.10 Biểu đồ so sánh chi phí bộ nhớ của các thuật toán dựa trên sự tương tự

Các thuật toán dựa trên sự tương tự yêu cầu chi phí tài nguyên cao hơn nhiều so với các thuật toán dựa trên lý thuyết thông tin. Các thuật toán chủ yếu dựa trên việc xây dựng ma trận tương đồng (kích thước $n \times n$), chi phí tài nguyên phụ thuộc nhiều vào số lượng bản ghi của bộ dữ liệu chứ không phụ thuộc vào số lượng đặc trưng. Do đó mà với các bộ dữ liệu vừa và lớn gồm nhiều bản ghi mặc dù ít đặc trưng, các thuật toán dựa trên sự tương tự không thể thực hiện được với tài nguyên giới hạn. Thuật toán Laplacian Score và thuật toán ReliefF yêu cầu chi phí về tài nguyên (RAM) thấp nhất. Thuật toán ReliefF có thể thực hiện với các bộ dữ liệu lớn hơn rất nhiều so với khả năng của các thuật toán Fisher, SPEC, Trace Ratio. Thuật toán Fisher có chi phí tài nguyên lớn nhất, gấp 2 lần thuật toán Laplacian và

gấp 3 lần thuật toán ReliefF. Thuật toán SPEC và Trace Ratio có chi phí tài nguyên tương đương nhau và thấp hơn thuật toán Fisher nhưng vẫn rất lớn.

5.4.1.3. Đánh giá khả năng phân loại của mô hình phân loại

Kết quả so sánh khả năng của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi các thuật toán lựa chọn đặc trưng dựa trên sự tương tự. Các kết quả dưới đây là kết quả thực nghiệm với ngưỡng lựa chọn 25% tổng số đặc trưng, kết quả với các ngưỡng 10%, 15% xem ở phần Phụ lục của đồ án này.

Sau đây là bảng kết quả các độ đo đánh giá mô hình phân loại với dữ liệu là đầu ra của các thuật toán lựa chọn đặc trưng dựa trên sự tương tự: độ chính xác (Accuracy), chỉ số AUC và chỉ số F1.

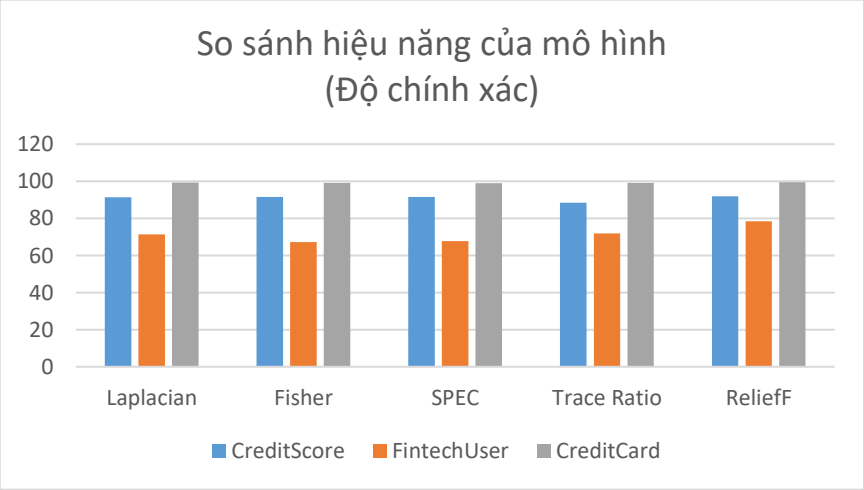
a) Độ chính xác (Accuracy)

Thuật toán / Bộ dữ liệu	Lapla-cian	Fisher	SPEC	Trace Ratio	ReliefF
CreditScore	91,4	91,6	91,5	88,4	**91,9**
FintechUser	71,4	67,3	67,7	71,8	**78,4**
CreditCard	99,3	99,1	99,0	99,2	**99,5**
HomeCredit	**91,8**	91,7	91,3	-*	91,4
HomeCredit2	91,5	91,5	91,5	-	**91,6**
CreditRisk	99,6	94,2	94,1	-	**99,7**
FinancialRisk	80,4	80,4	80,2	-	**80,5**
VehicleLoan	**78,3**	78,0	77,6	84,7	78,0

Bảng 5.7 Độ chính xác của mô hình phân loại

(*) Thuật toán Trace Ratio gặp lỗi và không thể hoàn thành lựa chọn đặc trưng.

Biểu đồ trực quan so sánh chi tiết hơn độ chính xác của mô hình phân loại với các bộ dữ liệu điển hình (CreditScore, FintechUser và CreditCard):



Hình 5.11 Biểu đồ so sánh độ chính xác của mô hình với một số bộ dữ liệu

Với từng thuật toán, độ chính xác của mô hình với các bộ dữ liệu khác nhau thì khác nhau rất nhiều. Ví dụ với thuật toán Laplacian, với bộ dữ liệu CreditCard và CreditRisk, độ chính xác đạt được trên 99%, nhưng với bộ dữ liệu FintechUser, độ chính xác chỉ đạt 71,4% và 78,3% với bộ dữ liệu VehicleLoan. Các thuật toán khác cũng có các kết quả tương tự. Giải thích cho điều này là do các bộ dữ liệu đều có nhãn mất cân bằng, nên dù đạt được độ chính xác cao nhưng không phản ánh được mô hình phân loại tốt. Bộ dữ liệu CreditCard có tỉ lệ nhãn 0 lên tới 99,8%, do đó, mô hình phân loại với chính xác luôn trên 99%.

Với từng bộ dữ liệu, độ chính xác của mô hình với các thuật toán là gần tương đương nhau, thuật toán Laplacian và thuật toán ReliefF cho kết quả tốt hơn với hầu hết các bộ dữ liệu.

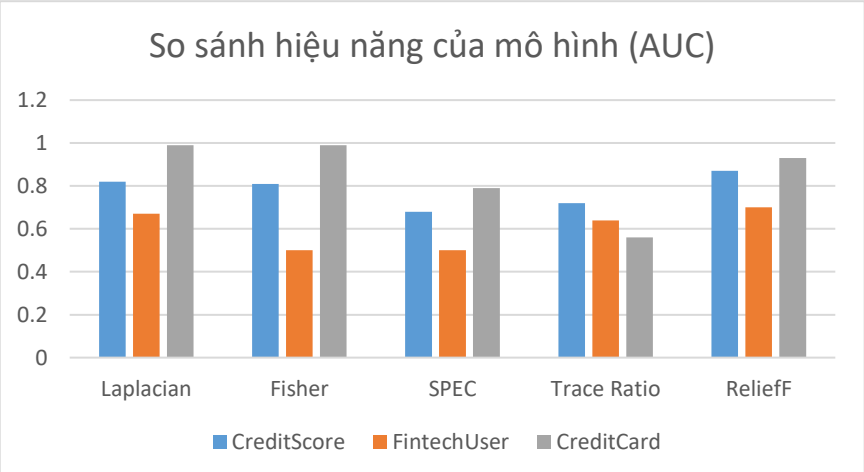
b) Chỉ số AUC

Thuật toán / Bộ dữ liệu	Lapla-cian	Fisher	SPEC	Trace Ratio	ReliefF
CreditScore	0,82	0,81	0,68	0,72	**0,87**
FintechUser	0,67	0,50	0,50	0,64	**0,70**
CreditCard	**0,99**	**0,99**	0,79	0,56	0,93
HomeCredit	0,46	0,51	0,51	-*	**0,54**
HomeCredit2	0,50	**0,52**	0,50	-	0,50
CreditRisk	**0,99**	0,50	0,50	-	**0,99**
FinancialRisk	0,50	0,50	0,50	-	**0,52**
VehicleLoan	0,50	0,50	0,50	**0,66**	0,50

Bảng 5.8 Chỉ số AUC của mô hình phân loại

(*) Thuật toán Trace Ratio gặp lỗi và không thể hoàn thành lựa chọn đặc trưng.

Biểu đồ trực quan so sánh chi tiết hơn hiệu năng của mô hình phân loại thông qua chỉ số AUC với các bộ dữ liệu điển hình (CreditScore, FintechUser và CreditCard):



Hình 5.12 Biểu đồ so sánh chỉ số AUC của mô hình với một số bộ dữ liệu

Với từng thuật toán, chỉ số AUC của mô hình với các bộ dữ liệu khác nhau thì khác nhau rất nhiều. Ví dụ với thuật toán Laplacian, bộ dữ liệu CreditCard và CreditRisk, AUC đạt được 0,99 nhưng với các bộ dữ liệu HomeCredit2, FinancialRisk, VehicleLoan, AUC rơi vào trường hợp tồi nhất, xấp xỉ 0,5. Các thuật toán khác cũng có các kết quả tương tự.

Với các bộ dữ liệu HomeCredit, Homecredit2, FinancialRisk và VehicleLoan, chỉ số AUC của mô hình với các thuật toán đều ở mức xấp xỉ 0,5. Mô hình phân loại không có khả năng phân loại với các bộ dữ liệu này. Với các bộ dữ liệu còn lại, chỉ số AUC của mô hình với các thuật toán Laplacian và ReliefF là cao nhất.

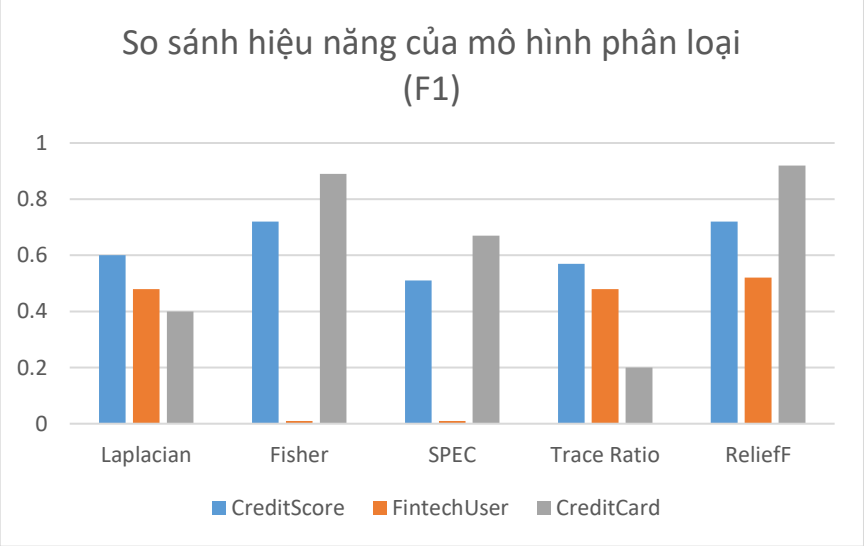
c) Chỉ số F1

Thuật toán / Bộ dữ liệu	Lapla-cian	Fisher	SPEC	Trace Ratio	ReliefF
CreditScore	0,60	**0,72**	0,51	0,57	**0,72**
FintechUser	0,48	0,00	0,00	0,48	**0,52**
CreditCard	0,40	0,89	0,67	0,20	**0,92**
HomeCredit	0,00	**0,52**	0,04	-*	0,01
HomeCredit2	0,00	0,04	0,00	-	0,00
CreditRisk	0,99	0,00	0,00	-	**0,99**
FinancialRisk	0,03	0,01	0,01	-	**0,10**
VehicleLoan	0,01	0,01	0,02	**0,47**	0,01

Bảng 5.9 Chỉ số F1 của mô hình phân loại

(*) Thuật toán Trace Ratio gặp lỗi và không thể hoàn thành lựa chọn đặc trưng.

Biểu đồ trực quan so sánh chi tiết hơn hiệu năng của mô hình phân loại thông qua chỉ số F1 với các bộ dữ liệu điển hình (CreditScore, FintechUser và CreditCard):



Hình 5.13 Biểu đồ so sánh chỉ số F1 của mô hình với một số bộ dữ liệu

Bảng kết quả và biểu đồ chỉ số F1 phản ánh kết quả tương tự chỉ số AUC.

Với từng thuật toán, chỉ số F1 của mô hình với các bộ dữ liệu khác nhau thì khác nhau rất nhiều. Ví dụ với thuật toán Laplacian, bộ dữ liệu CreditCard và CreditRisk, F1 đạt được 0,99 nhưng với các bộ dữ liệu HomeCredit2, FinancialRisk, VehicleLoan, chỉ số F1 rơi vào trường hợp tồi nhất, xấp xỉ 0. Các thuật toán khác cũng có các kết quả tương tự.

Với các bộ dữ liệu HomeCredit, Homecredit2, FinancialRisk và VehicleLoan, chỉ số F1 của mô hình với các thuật toán đều ở mức xấp xỉ 0. Mô hình phân loại không có khả năng phân loại với các bộ dữ liệu này. Với các bộ dữ liệu còn lại, chỉ số F1 của mô hình với các thuật toán Laplacian và ReliefF là cao nhất.

Thông qua bảng kết quả chỉ số AUC và chỉ số F1 của mô hình phân loại, ta thấy khả năng phân loại của mô hình kém hơn rất nhiều. Có nhiều bộ dữ liệu (HomeCredit, HomeCredit2, FinancialRisk) mà mô hình không có khả năng phân loại (AUC xấp xỉ 0,50 và F1 xấp xỉ 0). Thuật toán cho kết quả tốt nhất là thuật toán ReliefF và thuật toán Laplacian. Thuật toán SPEC và thuật toán Fisher cho kết quả rất tốt với một vài bộ dữ liệu, nhưng không ổn định.

5.4.1.4. Nhận xét về các thuật toán

Đặc điểm chung của các thuật toán dựa trên sự tương tự đó là hiệu năng tương đối, thời gian tính toán thấp trong khi tài nguyên cao. Nhóm các thuật toán này có thể áp dụng cho cả lớp bài toán có giám sát (thuật toán Fisher, ReliefF, SPEC) và không giám sát (thuật toán Laplacian, Trace Ratio). Thuật toán cho kết quả tốt nhất là thuật toán Laplacian và thuật toán ReliefF. Thuật toán Trace Ratio không thể thực hiện được với một số bộ dữ liệu nên cần được xem xét lại.

Sau đây là tổng hợp các đặc điểm nổi bật của các thuật toán dựa trên sự tương tự

Lớp bài toán	Thuật toán	Đặc điểm
Không giám sát	Laplacian Score	Dễ dàng triển khai, hiệu năng cao và ổn định, thời gian tính toán rất thấp, chi phí tài nguyên vừa phải.
	Trace Ratio	Khó triển khai, hiệu năng không ổn định, thời gian tính toán vừa phải, chi phí tài nguyên lớn.
Có giám sát	SPEC	Khó triển khai, hiệu năng không ổn định, thời gian tính toán vừa phải, chi phí tài nguyên lớn.
	Fisher Score	Dễ dàng triển khai, hiệu năng tốt, thời gian tính toán thấp, chi phí tài nguyên lớn.
	ReliefF	Dễ dàng triển khai, hiệu năng cao và ổn định, thời gian tính toán thấp, chi phí tài nguyên thấp.

Bảng 5.10 Đặc điểm của các thuật toán dựa trên sự tương tự

Như vậy trong nhóm các thuật toán dựa trên sự tương tự, với bài toán có giám sát, thuật toán nên được ưu tiên sử dụng là thuật toán ReliefF sau đó là thuật toán Fisher. Còn đối với bài toán không giám sát, thuật toán Laplacian nên được ưu tiên sử dụng.

5.4.2 Thực nghiệm các thuật toán dựa trên lý thuyết thông tin

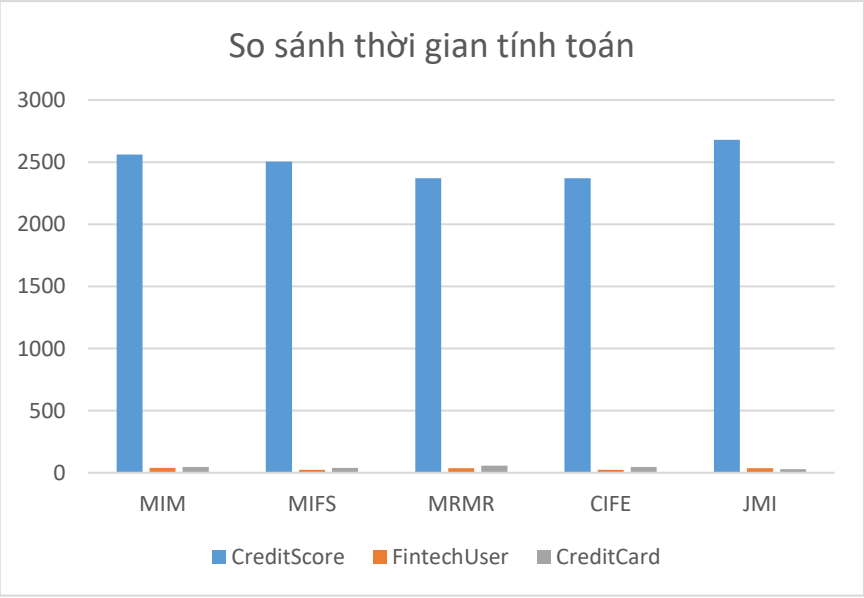
5.4.2.1. Thời gian tính toán của các thuật toán

Kết quả so sánh thời gian thực hiện lựa chọn đặc trưng của các thuật toán dựa trên sự tương tự. Đơn vị: giây

Thuật toán / Bộ dữ liệu	MIM	MIFS	MRMR	CIFE	JMI
CreditScore	2560,3	2503,7	2369,9	**2369,9**	2680,8
FintechUser	37,9	**23,2**	35,9	23,4	36,8
CreditCard	46,3	**38,7**	57,8	45,8	28,8
HomeCredit	**337,3**	382,3	364,5	376,9	348,7
HomeCredit2	**2215,6**	2342,6	2310,4	2331,7	2150,4
CreditRisk	**132,1**	158,3	148,9	155,3	144,9
FinancialRisk	**48,9**	67,9	67,7	68,9	62,5
VehicleLoan	**43,0**	67,4	69,6	69,1	44,7

Bảng 5.11 Thời gian tính toán các thuật toán dựa trên lý thuyết thông tin

Biểu đồ trực quan so sánh chi tiết hơn thời gian tính toán của các thuật toán với các bộ dữ liệu điển hình (CreditScore, FintechUser và CreditCard).



Hình 5.14 Biểu đồ so sánh thời gian tính toán của các thuật toán dựa trên lý thuyết thông tin

Các thuật toán lựa chọn đặc trưng dựa trên lý thuyết thông tin có thời gian tính toán tương đương nhau.

Thời gian tính toán của các thuật toán với các bộ dữ liệu khác nhau thì khác nhau và phụ thuộc nhiều vào số lượng đặc trưng. Các bộ dữ liệu CreditScore, HomeCredit2 với hơn 300 đặc trưng cần đến 2560,3 và 2215,6 giây để hoàn thành (thuật toán MIM). Trong khi đó, với các bộ dữ liệu có 30 – 45 đặc trưng như FintechUser, CreditCard, FinancialRisk, VehicleLoan chỉ cần 37 – 48 giây để hoàn thành, nhanh hơn các thuật toán dựa trên sự tương tự (Fisher, SPEC, ReliefF) và gần bằng thuật toán Laplacian.

5.4.2.2. Chi phí bộ nhớ của các thuật toán

Chi phí bộ nhớ của các thuật toán được phản ánh bởi kích thước tối đa của bộ dữ liệu mà thuật toán có thể thực hiện được (cấu hình đã nêu). Đơn vị: nghìn bản ghi

Thuật toán / Bộ dữ liệu	Số bản ghi	MIM	MIFS	MRMR	CIFE	JMI
CreditScore	80	80	80	80	80	80
FintechUser	27	27	27	27	27	27
CreditCard	284,8	284,8	284,8	284,8	284,8	284,8
HomeCredit	307,5	307,5	307,5	307,5	307,5	307,5
HomeCredit2	307,5	307,5	307,5	307,5	307,5	307,5
CreditRisk	855,9	855,9	855,9	855,9	855,9	855,9
FinancialRisk	1048	1048	1048	1048	1048	1048
VehicleLoan	233,1	233,1	233,1	233,1	233,1	233,1

Bảng 5.12 Kích thước tối đa các bộ dữ liệu mà các thuật toán dựa trên lý thuyết thông tin có thể thực hiện được

Các thuật toán dựa trên lý thuyết thông tin không thực hiện việc xây dựng các ma trận, mà thực hiện nhiều các phép tính đơn giản nên chi phí bộ nhớ rất thấp, thấp hơn rất nhiều so với các thuật toán dựa trên sự tương tự. Nếu như các thuật toán dựa trên sự tương tự chỉ có thể thực hiện được với các bộ dữ liệu vừa (khoảng 20 – 30 nghìn bản ghi) thì các thuật toán dựa trên lý thuyết thông tin có thể thực hiện với các bộ dữ liệu lớn (vài trăm nghìn bản ghi). Chi phí bộ nhớ của các thuật toán dựa trên lý thuyết thông tin có thể coi là tương đương nhau.

5.4.2.3. Đánh giá khả năng phân loại của mô hình phân loại

Kết quả so sánh khả năng của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi các thuật toán lựa chọn đặc trưng dựa trên lý thuyết thông tin. Các kết quả dưới đây là kết quả thực nghiệm với ngưỡng lựa chọn 25% tổng số đặc trưng, kết quả với các ngưỡng 10%, 15% xem ở phần Phu luc của đồ án này.

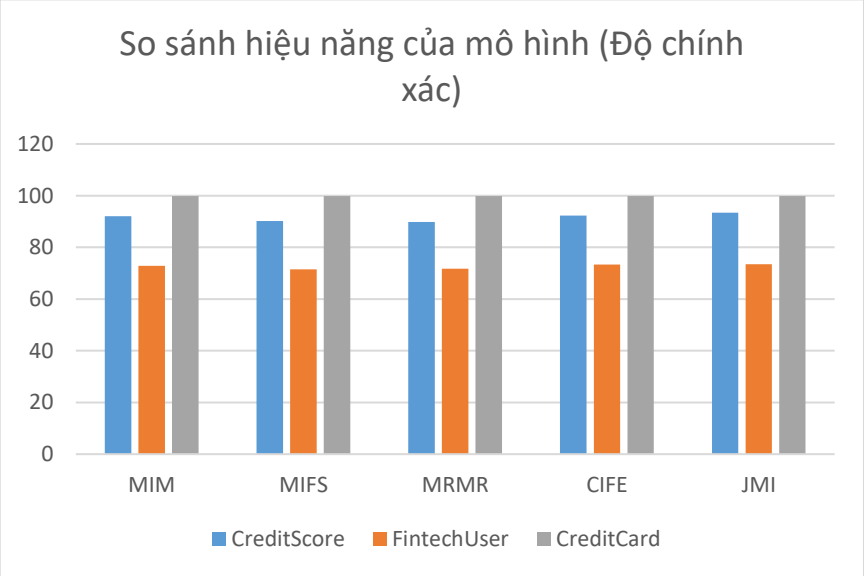
Sau đây là bảng kết quả các độ đo đánh giá mô hình phân loại với dữ liệu là đầu ra của các thuật toán lựa chọn đặc trưng dựa trên lý thuyết thông tin: độ chính xác (Accuracy), chỉ số AUC và chỉ số F1.

a) Độ chính xác

Thuật toán / Bộ dữ liệu	MIM	MIFS	MRMR	CIFE	JMI
CreditScore	92,1	90,2	89,8	92,3	**93,4**
FintechUser	72,8	71,5	71,7	73,4	**73,5**
CreditCard	99,8	99,9	99,8	99,9	**99,9**
HomeCredit	91,5	91,5	91,5	**91,6**	91,5
HomeCredit2	91,5	91,5	91,3	91,5	**91,7**
CreditRisk	99,8	97,1	97,3	94,0	**99,9**
FinancialRisk	**80,3**	79,8	**80,3**	**80,3**	80,1
VehicleLoan	**84,7**	77,9	77,8	84,4	**84,7**

Bảng 5.13 Độ chính xác của mô hình phân loại

Biểu đồ trực quan so sánh hiệu năng của mô hình phân loại thông qua độ chính xác với các bộ dữ liệu điển hình (CreditScore, FintechUser và CreditCard):



Hình 5.15 Biểu đồ so sánh độ chính xác của mô hình với một số bộ dữ liệu

Với từng thuật toán, độ chính xác của mô hình với các bộ dữ liệu khác nhau thì khác nhau rất nhiều. Ví dụ với thuật toán MIM, với bộ dữ liệu CreditCard và CreditRisk, độ chính xác đạt được trên 99%, nhưng với bộ dữ liệu FintechUser, độ chính xác chỉ đạt 72,8% và 80,3% với bộ dữ liệu FinancialRisk. Các thuật toán khác cũng có các kết quả tương tự. Giải thích cho điều này là do các bộ dữ liệu đều có nhãn mất cân bằng, nên dù đạt được độ chính xác cao nhưng không phản

ánh được mô hình phân loại tốt. Bộ dữ liệu CreditCard có tỉ lệ nhãn 0 lên tới 99,8%, do đó, mô hình phân loại với chính xác luôn trên 99%.

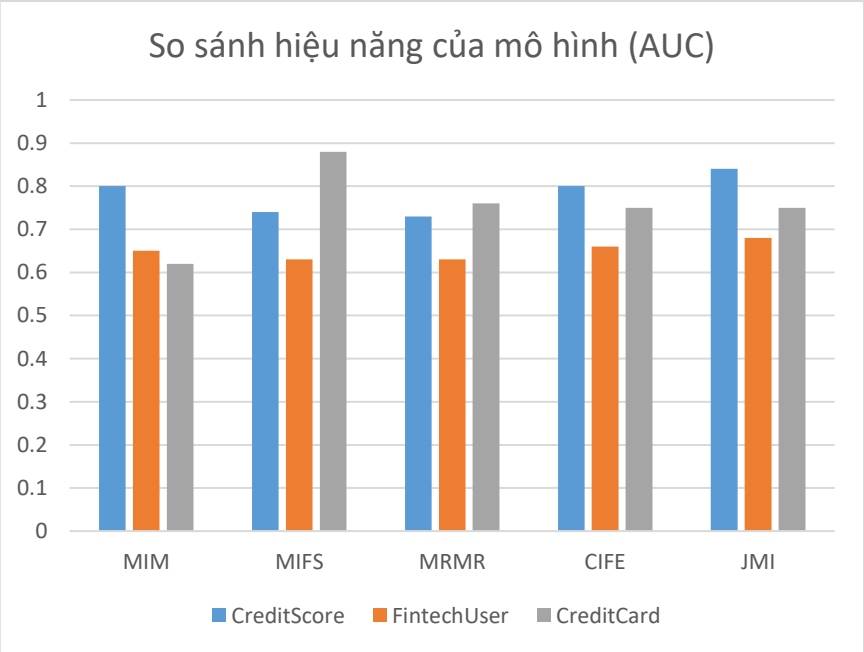
Với từng bộ dữ liệu, độ chính xác của mô hình với các thuật toán là gần tương đương nhau, các thuật toán cho kết quả tốt hơn là thuật toán MIM và thuật toán JMI.

b) Chỉ số AUC

Thuật toán / Bộ dữ liệu	MIM	MIFS	MRMR	CIFE	JMI
CreditScore	0,80	0,74	0,73	0,80	**0,84**
FintechUser	0,65	0,63	0,63	0,66	**0,68**
CreditCard	0,62	**0,88**	0,76	0,75	0,75
HomeCredit	**0,51**	**0,51**	0,50	**0,51**	0,50
HomeCredit2	0,50	0,50	**0,51**	0,50	**0,51**
CreditRisk	0,98	0,77	0,79	0,50	**0,99**
FinancialRisk	**0,54**	0,51	0,51	0,53	**0,54**
VehicleLoan	**0,66**	0,50	0,50	0,65	**0,66**

Bảng 5.14 Chỉ số AUC của mô hình phân loại

Biểu đồ trực quan so sánh chi tiết hơn hiệu năng của mô hình phân loại thông qua chỉ số AUC với các bộ dữ liệu điển hình (CreditScore, FintechUser và CreditCard):



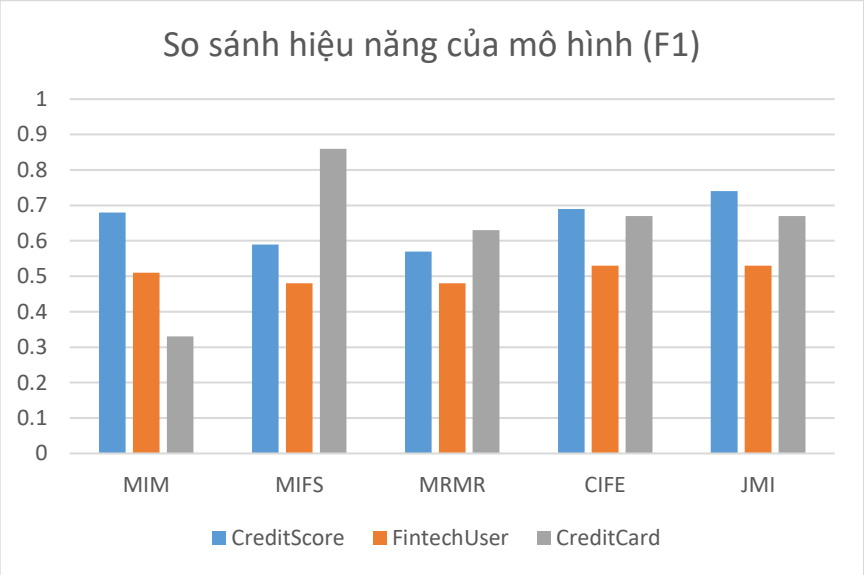
Hình 5.16 Biểu đồ so sánh chỉ số AUC của mô hình với một số bộ dữ liệu

c) Chỉ số F1

Thuật toán / Bộ dữ liệu	MIM	MIFS	MRMR	CIFE	JMI
CreditScore	0,68	0,59	0,57	0,69	**0,74**
FintechUser	0,51	0,48	0,48	**0,53**	**0,53**
CreditCard	0,33	**0,86**	0,63	0,67	0,67
HomeCredit	**0,03**	0,02	0,00	**0,03**	0,01
HomeCredit2	0,01	0,02	**0,04**	0,01	0,01
CreditRisk	**0,99**	0,69	0,72	0,01	**0,99**
FinancialRisk	0,16	0,03	**0,56**	0,13	0,17
VehicleLoan	**0,48**	0,00	0,01	0,46	**0,48**

Bảng 5.15 Chỉ số F1 của mô hình phân loại

Biểu đồ trực quan so sánh chi tiết hơn hiệu năng của mô hình phân loại thông qua chỉ số F1 với các bộ dữ liệu điển hình (CreditScore, FintechUser và CreditCard):



Hình 5.17 Biểu đồ so sánh chỉ số F1 của mô hình với một số bộ dữ liệu

Các thuật toán dựa trên lý thuyết thông tin đều cho hiệu năng cao và ổn định, do đó mà độ chính xác của mô hình phân loại với dữ liệu là đầu ra của các thuật toán lựa chọn đăc trưng đều phản ánh kết quả tốt và ổn định. Tuy nhiên, độ chính xác bị ảnh hưởng nhiều bởi tỉ lệ nhãn mất cân bằng của các bộ dữ liệu, nên không thể phản ảnh đúng về khả năng phân loại của mô hình.

Chỉ số AUC và chỉ số F1 cho thấy mô hình có khả năng phân loại kém hơn, có một số bộ dữ liệu (HomeCredit, HomeCredit2) mà mô hình không có khả năng phân loại (AUC xấp xỉ 0,50 và F1 xấp xỉ 0). Thuật toán có kết quả tốt nhất là thuật toán MIM và thuật toán JMI. Khả năng phân loại của mô hình với dữ liệu là đầu ra của các thuật toán này đều rất tốt và ổn định.

5.4.2.4. So sánh các thuật toán

Đặc điểm chung của các thuật toán dựa trên lý thuyết thông tin đó là hiệu năng cao, chi phí tài nguyên thấp trong khi thời gian tính toán cao. Nhóm các thuật toán này chỉ có thể áp dụng cho lớp bài toán có giám sát. Các thuật toán cho kết quả gần tương đương nhau, các thuật toán cho kết quả tốt hơn là thuật toán MIM và thuật toán JMI.

Sau đây là các đặc điểm nổi bật của các thuật toán dựa trên lý thuyết thông tin

Lớp bài toán	Thuật toán	Đặc điểm chung	Đặc điểm nổi bật
Có giám sát	MIM	Thời gian tính toán lớn, phụ thuộc vào số đặc trưng Chi phí tài nguyên thấp	Dễ dàng triển khai, hiệu năng cao, ổn định
	MIFS		Khó triển khai, hiệu năng cao
	MRMR		Khó triển khai, hiệu năng cao
	CIFE		Khó triển khai, hiệu năng cao
	JMI		Khó triển khai, hiệu năng cao và ổn định

Như vậy trong nhóm các thuật toán dựa trên lý thuyết thông tin, thuật toán nên được ưu tiên sử dụng là thuật toán MIM sau đó là thuật toán JMI. Các thuật toán còn lại có hiệu năng kém hơn không đáng kể, nên đều có thể cân nhắc sử dụng.

5.4.3 So sánh các thuật toán lựa chọn đặc trưng nổi bật

5.4.3.1. So sánh các thuật toán lựa chọn đặc trưng không giám sát

Với lớp bài toán không giám sát, có 2 thuật toán được thực nghiệm là thuật toán Laplacian và thuật toán Trace Ratio.

Bộ dữ liệu	Chi phí thời gian		Chi phí Bộ nhớ		Hiệu năng (AUC)	
	Lap-lacian	Trace Ratio	Lap-lacian	Trace Ratio	Lap-lacian	Trace Ratio
CreditScore	35,6	226,3	22	14	0,82	0,72
FintechUser	31,1	317,9	22	14	0,67	0,64
CreditCard	17,6	149,7	22	14	0,99	0,56
HomeCredit	23,9	-*	20	-	0,46	-
HomeCredit2	27,8	-	21	-	0,50	-
CreditRisk	25,4	-	20	-	0,99	-
FinancialRisk	22,5	-	22	-	0,50	-
VehicleLoan	34,3	301,8	20,5	14,5	0,50	0,66

Bảng 5.16 So sánh các thuật toán lựa chọn đặc trưng không giám sát

(*) Thuật toán Trace Ratio gặp lỗi, không thể hoàn thành lựa chọn đặc trưng

Thuật toán Laplacian cho hiệu năng cao hơn với thời gian tính toán chỉ bằng một phần mười của thuật toán Trace Ratio, còn chi phí tài nguyên cũng thấp hơn nên có thể thực hiện được các bộ dữ liệu lớn hơn. Thuật toán Trace Ratio không thể hoàn thành việc lựa chọn đặc trưng với một số bộ dữ liệu

5.4.3.2. So sánh các thuật toán lựa chọn đặc trưng có giám sát

Với lớp bài toán không giám sát, các thuật toán cho kết quả tốt nhất là các thuật toán Fisher, ReliefF, MIM và JMI.

a) So sánh thời gian tính toán

Bộ dữ liệu	Fisher	ReliefF	MIM	JMI
CreditScore	**79,8**	425,0	2560,3	2680,8
FintechUser	63,1	410,9	**37,9**	36,8
CreditCard	63,9	278,2	**46,3**	28,8
HomeCredit	**74,3**	256,6	337,3	348,7
HomeCredit2	**69,1**	283,7	2215,6	2150,4
CreditRisk	**69,3**	300,1	132,1	144,9
FinancialRisk	53,9	255,6	**48,9**	62,5
VehicleLoan	81,2	371,5	**43,0**	44,7

Bảng 5.17 So sánh thời gian tính toán các thuật toán lựa chọn đặc trưng có giám sát

Các thuật toán dựa trên sự tương tự được đánh giá là nhanh hơn các thuật toán dựa trên lý thuyết thông tin, nhưng với các bộ dữ liệu khác nhau, đánh giá này chưa thực sự chính xác.

Các thuật toán dựa trên lý thuyết thông tin (MIM và JMI) có thời gian tính toán phụ thuộc vào số lượng đặc trưng, nên với các bộ dữ liệu gồm ít đặc trưng (FintechUser và CreditCard với 29 đặc trưng, FinancialRisk với 45 đặc trưng, VehicleLoan với 39 đặc trưng) thì thời gian tính toán các thuật toán này là thấp hơn. Với các bộ dữ liệu với số lượng đặc trưng vừa phải (HomeCredit với 120 đặc trưng, CreditRisk với 71 đặc trưng) thời gian tính toán của các thuật toán MIM, JMI và ReliefF là tương đương. Với các bộ dữ liệu gồm nhiều đặc trưng (CreditScore và HomeCredit2 với hơn 300 đặc trưng) thì thời gian tính toán của các thuật toán MIM và JMI là cao hơn rất nhiều.

Như vậy với tiêu chí về thời gian, các thuật toán dựa trên lý thuyết thông tin sẽ phù hợp hơn với các bộ dữ liệu với số lượng đặc trưng nhỏ.

b) So sánh chi phí bộ nhớ (kích thước số bản ghi tối đa có thể thực hiện)

Bộ dữ liệu	Fisher	ReliefF	MIM	JMI
CreditScore	12,5	37	80	80
FintechUser	13	27	27	27
CreditCard	10	37	284,8	284,8
HomeCredit	11	37	307,5	307,5
HomeCredit2	11	37	307,5	307,5
CreditRisk	20	37	855,0	855,0
FinancialRisk	22	37	1048	1048
VehicleLoan	13,5	38	233,1	233,1

Bảng 5.18 So sánh chi phí bộ nhớ các thuật toán lựa chọn đặc trưng có giám sát

Các thuật toán dựa trên lý thuyết thông tin có thể thực hiện với các bộ dữ liệu gốc (rất lớn), do đó chi phí bộ nhớ thấp hơn rất nhiều.

c) So sánh hiệu năng (chỉ số AUC)

Bộ dữ liệu	Fisher	ReliefF	MIM	JMI
CreditScore	0,82	**0,87**	0,80	0,84
FintechUser	0,50	0,70	0,65	**0,68**
CreditCard	**0,99**	0,93	0,62	0,75
HomeCredit	0,51	**0,54**	0,51	0,50
HomeCredit2	0,52	0,50	0,50	0,51
CreditRisk	0,50	**0,99**	0,98	**0,99**
FinancialRisk	0,50	0,52	**0,54**	**0,54**
VehicleLoan	0,50	0,50	**0,66**	**0,66**

Bảng 5.19 So sánh hiệu năng các thuật toán lựa chọn đặc trưng có giám sát

Các thuật toán đều cho hiệu cao, tương đương. Các thuật toán MIM và JMI không cho kết quả tốt hơn nhưng ổn định hơn nhiều.

Thuật toán Fisher có ưu điểm về thời gian nhưng nhược điểm là hiệu năng không ổn định và chi phí tài nguyên cao. Thuật toán ReliefF có ưu điểm về hiệu năng cao,

nhược điểm là thời gian tính toán cao hơn các thuật toán dựa trên sự tương tự khác. Các thuật toán MIM và JMI có ưu điểm hiệu năng cao và ổn định, thời gian tính toán thấp với các bộ dữ liệu có số lượng đặc trưng nhỏ, nhược điểm là thời gian tính toán tăng lên rất nhiều khi số lượng đặc trưng tăng lên.

5.4.3.3. So sánh thuật toán Laplacian và thuật toán MIM

Thuật toán nổi bật trong nhóm các thuật toán dựa trên sự tương tự là thuật toán Laplacian với tốc độ thực thi nhanh hơn, chi phí bộ nhớ thấp, hiệu năng cao và ổn định. Trong nhóm các thuật toán dựa trên lý thuyết thông tin, thuật toán MIM nổi bật với sự đơn giản, hiệu năng cao và ổn định, chi phí bộ nhớ thấp hơn. Thực hiện so sánh hai thuật toán này để so sánh hai nhóm thuật toán.

Bộ dữ liệu	Chi phí thời gian		Chi phí tài nguyên		Hiệu năng (AUC)	
	Lap-lacian	MIM	Lap-lacian	MIM	Lap-lacian	MIM
CreditScore	**35,6**	2560,3	22	**80**	**0,82**	0,80
FintechUser	**31,1**	37,9	22	**27**	**0,67**	0,65
CreditCard	**17,6**	46,3	22	**284,8**	**0,99**	0,62
HomeCredit	**23,9**	337,3	20	**307,5**	0,46	**0,51**
HomeCredit2	**27,8**	2215,6	21	**307,5**	0,50	0,50
CreditRisk	**25,4**	132,1	20	**855,0**	**0,99**	0,98
FinancialRisk	**22,5**	48,9	22	**1048**	0,50	**0,54**
VehicleLoan	**34,3**	43,0	20,5	**233,1**	0,50	**0,66**

Bảng 5.20 So sánh thuật toán Laplacian và thuật toán MIM

Thuật toán Laplacian có chí thời gian thấp hơn trong mọi trường hợp, hiệu năng đạt được cao hơn. Trong khi đó, thuật toán MIM có chi phí tài nguyên thấp hơn rất nhiều, hiệu năng ổn định hơn. Các thuật toán đã phản ánh được những đặc điểm chung của hai nhóm thuật toán dựa trên sự tương tự và dựa trên lý thuyết thôn tin.

5.4.4 Phân tích tổng hợp

Các thuật toán lựa chọn đặc trưng được thực nghiệm là các thuật toán được giới thiệu cách đây hơn 10 năm, đều là các thuật toán được áp dụng rộng rãi và được cho là có hiệu năng cao. Tuy nhiên khi thực nghiệm các thuật toán này với các bộ dữ liệu công nghệ tài chính, các kết quả chưa thực sự ấn tượng.

Các bộ dữ liệu CreditScore, FintechUser và CreditCard cho kết quả tốt với hầu hết các thuật toán. Trong khi đó các bộ dữ liệu HomeCredit, CreditRisk và FinancialRisk cho kết quả tồi nhất với tất cả các thuật toán. Các bộ dữ liệu này có

thể cần đến các phương pháp lựa chọn đặc trưng thuộc nhóm phương pháp trình bao bọc để có thể cho kết quả chấp nhận được.

Với lớp bài toán không giám sát, thuật toán Laplacian vượt trội hơn hẳn thuật toán SPEC về tất cả các yếu tố hiệu năng, chi phí, triển khai.

Với lớp bài toán có giám sát, các thuật toán ReliefF, MIM và JMI cho hiệu năng cao và ổn định. Các thuật toán Fisher và ReliefF cho thời gian tính toán thấp. Các thuật toán thuộc nhóm thuật toán dựa trên lý thuyết thông tin đều cho chí phí tài nguyên thấp và tương đương nhau.

Các thuật toán SPEC, Trace Ratio không đem lại kết quả nổi bật, hiệu năng không ổn định, không dễ dàng triển khai, chi phí tài nguyên cao, thời gian tính toán vừa phải. Các thuật toán MIFS, MRMR, CIFE cho kết quả tương đối tốt, hiệu năng ổn định, nhưng không vượt trội.

Bảng đánh giá tất cả các thuật toán lựa chọn đặc trưng dựa theo các tiêu chí về hiệu năng, triển khai, thời gian tính toán, chi phí tài nguyên.

Lớp bài toán	Thuật toán	Hiệu năng cao, ổn định	Dễ dàng triển khai	Thời gian tính toán thấp	Chi phí bộ nhớ thấp
Không giám sát	Laplacian	x	x	x	
	Trace Ratio				
Có giám sát	Fisher		x	x	
	SPEC				
	ReliefF	x	x	x	
	MIM	x	x		x
	MIFS				x
	MRMR				x
	CIFE				x
	JMI	x			x

Bảng 5.21 So sánh tất cả các thuật toán lựa chọn đặc trưng

CHƯƠNG 6. KẾT LUẬN

6.1 Kết luận

Đồ án đã giới thiệu và tái tạo thành công các thuật toán lựa chọn đặc trưng dưa trên sự tương tự của dữ liệu và dựa trên lý thuyết thông tin. Sau đó thực nghiệm các thuật toán này trên các bộ dữ liệu thực tế vừa và lớn thuộc lĩnh vực công nghệ tài chính. Các kết quả thực nghiệm cho thấy các thuật toán có các ưu điểm và nhược điểm riêng, phù hợp với các mục đích khác nhau.

Với lớp bài toán không giám sát, các thuật toán có thể áp dụng là thuật toán Laplacian, SPEC, Trace Ratio. Các thuật toán áp dụng với lớp bài toán có giám sát là thuật toán Fisher, Trace Ratio, ReliefF và tất cả các thuật toán dựa trên lý thuyết thông tin.

Các thuật toán dựa trên sự tương tự dễ dàng cài đặt, thời gian tính toán nhanh và hiệu năng tương đối tốt nhưng đòi hỏi chi phí tài nguyên cao. Các thuật toán tốt nhất trong nhóm này là thuật toán Laplacian với hiệu năng cao, thời gian tính toán rất thấp, chi phí tài nguyên vừa phải và thuật toán ReliefF với hiệu năng cao và ổn định, chi phí tài nguyên thấp, thời gian tính toán vừa. Thuật toán Fisher có thể coi là tương tự thuật toán Laplacian, nhưng áp dụng được với các lớp bài toán có giám sát. Các thuật toán SPEC, Trace Ratio phức tạp hơn mà không đem lại hiệu năng cũng như chi phí tốt hơn.

Các thuật toán dựa trên lý thuyết thông cho hiệu năng cao hơn với chi phí tài nguyên thấp hơn nhưng đòi hỏi thời gian tính toán cao hơn và khó áp dụng hơn (chỉ áp dụng với lớp bài toán có giám sát). Các thuật toán trong nhóm này đều cho kết quả tốt không chênh lệch quá nhiều. Các thuật toán tốt nhất là thuật toán MIM và thuật toán JMI với hiệu năng cao hơn trong khi thời gian tính toán và tài nguyên thì tương tự. Tuy nhiên thuật toán MIM đơn giản và dễ hiểu hơn nhiều.

Các thuật toán phù hợp với các mục đích sử dụng được trình bày trong bảng:

Lớp bài toán	Yêu cầu	Ưu tiên sử dụng
Không giám sát	Dễ dàng triển khai	Laplacian
	Hiệu năng cao, ổn định	
	Thời gian tính toán thấp	
	Chi phí tài nguyên thấp	
Có giám sát	Dễ dàng triển khai	Fisher, MIM
	Hiệu năng cao, ổn định	ReliefF, MIM, JMI
	Thời gian tính toán thấp	Fisher, MIM và JMI (số lượng đặc trưng nhỏ)
	Chi phí tài nguyên thấp	MIM, MIFS, MRMR, CIFE, JMI, ReliefF

Bảng 6.1 Các thuật toán khuyên dùng theo yêu cầu sử dụng

Tùy theo kỳ vọng cũng như mục đích sử dụng thuật toán lựa chọn đặc trưng mà người dùng chọn phương pháp phù hợp. Với bài toán không giám sát, thuật toán Laplacian là phù hợp nhất. Với bài toán có giám sát, nếu mục đích đơn giản, nhanh cho kết quả thì thuật toán Fisher hoặc MIM (với các bộ dữ liệu có số đặc trưng nhỏ) là phù hợp. Nếu kỳ vọng hiệu năng cao hơn, ổn định hơn thì thuật toán ReliefF hoặc JMI sẽ phù hợp hơn.

6.2 Hướng phát triển của đề tài trong tương lai

Trong tương lai, đồ án có thể có các hướng phát triển sau:

- Thực nghiệm và so sánh các thuật toán chi tiết hơn (điều chỉnh các tham số, cách xây dựng ma trận,…)
- Thực nghiệm và so sánh các thuật toán khác dựa trên thống kê, sự tương quan, sự nhất quán,…
- Thực nghiệm và so sánh các thuật toán lựa chọn đặc trưng thuộc phương pháp trình bao bọc

TÀI LIỆU THAM KHẢO

[1] V. Kumar and S. Minz, "Feature Selection: A literature Review," *Smart Computing Review*, vol. 4, no. 3, pp. 211-229, 2014.

[2] T. Deepa and L. Ladha, "Feature Selection Methods And Algorithms," *International Journal on Computer Science and Engineering (IJCSE)*, vol. 3, no. 5, pp. 1787-1797, 2011.

[3] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang and H. Liu, "Feature Selection: A Data Perspective," pp. 1-73, 2016.

[4] W. Bouaguel, "On Feature Selection for Credit Scoring," 2015.

[5] C. Yun, D. Shin, H. Jo, J. Yang and S. Kim, "An Experimental Study on Feature Subset Seletion Methods," *In Proceedings of the 7th IEEE International Conference on Computer and Information Technology*, pp. 77-82, 2007.

[6] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 4, pp. 491-502, 2005.

[7] R. A. Musheer, C. K. Verma and N. Srivastava, "Dimension reduction methods for microarray data: a review," *AIMS Bioengineering*, vol. 4, no. 2, pp. 179-197, 2017.

[8] J. C. Davis and R. J. Sampson, "Statistics and Data Analysis in Geology," vol. 646.

[9] H. Liu and R. Setiono, "Feature selection and discretization of numeric attributes," *ICTAI*, pp. 388-391, 1995.

[10] M. A. Hall and L. A. Smith, "Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper," *FLAIRS*, pp. 235-239, 1999.

[11] Xiaofei He, Deng Cai and Partha Niyogi, "Laplacian score for feature selection," *NIPS*, pp. 507-514, 2005.

[12] M. R.-Š. a. I. Kononenko, "Theoretical and Empirical Analysis of ReliefF and RReliefF," in *Machine Learning*, The Netherlands, Kluwer Academic Publishers, 2003, pp. 23-69.

[13] R. O. Duda, P. E. Hart and D. G. Stork, Pattern Classification, 2012.

[14] D. D. Lewis, "Feature selection and feature extraction for text categorization," *Proceedings of the Workshop on Speech and Natural Language*, pp. 212-217, 1992.

[15] H. Peng, F. Long and C. Ding, "Feature selection based on mutual information criteria of maxdependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal,* vol. 27, no. 8, pp. 1226-1238, 2005.

[16] M. Vidal and S. Ullman, "Object recognition with informative features and linear classification," *ICCV,* pp. 281-288, 2003.

[17] P. E. Meyer, C. Schretter and G. Bontempi, "Information-theoretic feature selection in microarray data using variable complementarity," *IEEE J. Select,* vol. 2, no. 3, pp. 261-274, 2008.

[18] A. W. Whitney, "A direct method of nonparametric measurement selection," *EEE Transactions on Computers,* vol. C20, no. 9, pp. 1100-1103, 1971.

[19] T. Marill and D. Green, "On the effectiveness of receptors in recognition systems," *IEEE Transactions on Information Theory,* vol. 9, no. 1, pp. 11-17, 1963.

[20] P. Pudil, J. Novovicova and J. V. Kittler, "Floating search methods in Feature Selection," *Pattern Recognition Letters,* vol. 15, no. 11, pp. 1119-1125, 1994.

[21] B. Xue, M. Zhang and X. Yao, "A Survey on Evolutionary Computation Approaches to Feature Selection," *IEEE Transactions on Evolutionary Computation,* vol. 20, no. 4, pp. 606-626, 2016.

[22] M. Tan, I. Tsang and L. Wang, "Minimax sparse logistic regression for very high-dimensional feature selection," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 24, no. 10, pp. 1609-1622, 2013.

[23] Zheng Zhao and Huan Liu, "Spectral feature selection for supervised and unsupervised learning," *ICML,* vol. 07, pp. 1151-1157, 2007.

[24] F. Nie, S. Xiang, Y. Jia, C. Zhang and S. Yan, "Trace Ratio Criterion for Feature Seletion," *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence,* pp. 671-676, 2008.

[25] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Trans. Neural,* vol. 5, no. 4, pp. 537-550, 1994.

[26] D. Lin and X. Tang, "Conditional infomax learning: An integrated framework for feature extraction and fusion," *ECCV,* pp. 68-82, 2006.

[27] F. Min and W. Zhu, "Feature selection with test cost constraint," *International Journal of Approximate Reasoning,* vol. 55, no. 1, pp. 167-179, 2014.

PHỤ LỤC

A1. Kết quả thực nghiệm các thuật toán dựa trên sự tương tự

i. Kết quả thực nghiệm thuật toán Laplacian

Kết quả thực nghiệm thuật toán Laplacian Score với các bộ dữ liệu.

Bộ dữ liệu	Tổng số bản ghi (nghìn)	Số bản ghi tối đa có thể thực hiện (nghìn)	Thời gian tính toán (giây)
CreditScore	80	22	35,6
FintechUser	27	22	31,1
CreditCard	284,8	22	17,6
HomeCredit	307,5	20	23,9
HomeCredit2	307,5	21	27,8
CreditRisk	855	20	25,4
FinancialRisk	1048,5	21,5	22,5
VehicleLoan	233,1	20,5	34,3

Kết quả đánh giá khả năng phân loại thông qua các chỉ số (Độ chính xác, AUC, F1) của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi thuật toán Laplacian ở các ngưỡng 10%, 15% và 25% tổng số đặc trưng.

	Độ chính xác			AUC			F1		
Bộ dữ liệu	10%	15%	25%	10%	15%	25%	10%	15%	25%
CreditScore	87,4	87,9	91,4	0,77	0,80	0,82	0,43	0,46	0,60
FintechUser	71,2	71,4	71,4	0,66	0,67	0,67	0,47	0,48	0,48
CreditCard	99,3	99,3	99,3	0,50	0,99	0,99	0,00	0,40	0,40
HomeCredit	91,6	91,6	91,8	0,46	0,46	0,46	0,00	0,00	0,00
HomeCredit2	91,4	91,4	91,5	0,50	0,50	0,50	0,01	0,01	0,00
CreditRisk	99,2	99,2	99,6	0,97	0,98	0,99	0,96	0,98	0,99
FinancialRisk	80,1	80,4	80,4	0,50	0,50	0,50	0,01	0,00	0,03
VehicleLoan	78,3	78,3	78,3	0,50	0,50	0,50	0,01	0,01	0,01

ii. Kết quả thực nghiệm thuật toán Fisher Score

Kết quả thực nghiệm thuật toán Fisher Score với các bộ dữ liệu.

Bộ dữ liệu	Tổng số bản ghi (nghìn)	Số bản ghi tối đa có thể thực hiện (nghìn)	Thời gian tính toán (giây)
CreditScore	80	12,5	79,8
FintechUser	27	13	63,1
CreditCard	284,8	10	63,9
HomeCredit	307,5	11	74,3
HomeCredit2	307,5	11	69,1
CreditRisk	855	11,5	69,3
FinancialRisk	1048,5	17,5	53,9
VehicleLoan	233,1	13,5	81,2

Kết quả đánh giá khả năng phân loại thông qua các chỉ số (Độ chính xác, AUC, F1) của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi thuật toán Fisher ở các ngưỡng 10%, 15% và 25% tổng số đặc trưng.

	Độ chính xác			AUC			F1		
Bộ dữ liệu	10%	15%	25%	10%	15%	25%	10%	15%	25%
CreditScore	90,4	91,2	91,6	0,76	0,77	0,81	0,64	0,65	0,72
FintechUser	67,1	67,3	67,3	0,50	0,50	0,50	0,00	0,00	0,00
CreditCard	99,1	99,1	99,1	0,87	0,87	0,99	0,67	0,75	0,89
HomeCredit	91,3	91,4	91,7	0,51	0,50	0,51	0,04	0,01	0,52
HomeCredit2	91,5	91,5	91,5	0,52	0,52	0,52	0,09	0,07	0,04
CreditRisk	94,0	94,0	94,2	0,50	0,50	0,50	0,00	0,00	0,00
FinancialRisk	80,0	80,4	80,4	0,50	0,50	0,50	0,00	0,01	0,01
VehicleLoan	78,0	78,0	78,0	0,50	0,50	0,50	0,00	0,00	0,01

iii. Kết quả thực nghiệm thuật toán SPEC

Kết quả thực nghiệm thuật toán SPEC với các bộ dữ liệu.

Bộ dữ liệu	Tổng số bản ghi (nghìn)	Số bản ghi tối đa có thể thực hiện (nghìn)	Thời gian tính toán (giây)
CreditScore	80	14	91,5
FintechUser	27	13	67,7
CreditCard	284,8	13	99,0
HomeCredit	307,5	13	91,3
HomeCredit2	307,5	13	91,5
CreditRisk	855	14,5	94,1
FinancialRisk	1048,5	14,5	80,2
VehicleLoan	233,1	14,5	81,2

Kết quả đánh giá khả năng phân loại thông qua các chỉ số (Độ chính xác, AUC, F1) của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi thuật toán SPEC ở các ngưỡng 10%, 15% và 25% tổng số đặc trưng.

	Độ chính xác			AUC			F1		
Bộ dữ liệu	10%	15%	25%	10%	15%	25%	10%	15%	25%
CreditScore	88,5	88,8	91,5	0,66	0,66	0,68	0,46	0,47	0,51
FintechUser	67,5	67,5	67,7	0,50	0,50	0,50	0,00	0,00	0,00
CreditCard	99,0	99,0	99,0	0,50	0,57	0,79	0,00	0,25	0,67
HomeCredit	91,1	91,1	91,3	0,50	0,51	0,51	0,00	0,05	0,04
HomeCredit2	91,1	91,5	91,5	0,50	0,50	0,50	0,00	0,00	0,00
CreditRisk	94,0	94,2	94,1	0,50	0,50	0,50	0,00	0,00	0,00
FinancialRisk	80,0	80,3	80,2	0,50	0,50	0,50	0,01	0,01	0,01
VehicleLoan	77,1	77,4	77,6	0,50	0,50	0,50	0,00	0,00	0,02

iv. Kết quả thực nghiệm thuật toán Trace Ratio

Kết quả thực nghiệm thuật toán Fisher Score với các bộ dữ liệu.

Bộ dữ liệu	Tổng số bản ghi (nghìn)	Số bản ghi tối đa có thể thực hiện (nghìn)	Thời gian tính toán (giây)
CreditScore	80	14	226,3
FintechUser	27	14	317,9
CreditCard	284,8	14	149,7
HomeCredit	307,5	-	-
HomeCredit2	307,5	-	-
CreditRisk	855	-	-
FinancialRisk	1048,5	-	-
VehicleLoan	233,1	14,5	301,8

Kết quả đánh giá khả năng phân loại thông qua các chỉ số (Độ chính xác, AUC, F1) của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi thuật toán Trace Ratio ở các ngưỡng 10%, 15% và 25% tổng số đặc trưng.

	Độ chính xác			AUC			F1		
Bộ dữ liệu	10%	15%	25%	10%	15%	25%	10%	15%	25%
CreditScore	86,4	87,4	88,4	0,59	0,65	0,72	0,30	0,44	0,57
FintechUser	68,1	67,8	71,8	0,50	0,52	0,64	0,00	0,18	0,48
CreditCard	99,1	99,2	99,2	0,50	0,56	0,56	0,00	0,20	0,20
HomeCredit	-	-	-	-	-	-	-	-	-
HomeCredit2	-	-	-	-	-	-	-	-	-
CreditRisk	-	-	-	-	-	-	-	-	-
FinancialRisk	-	-	-	-	-	-	-	-	-
VehicleLoan	78,2	78,2	84,7	0,50	0,51	0,66	0,00	0,04	0,47

v. Kết quả thực nghiệm thuật toán ReliefF

Kết quả thực nghiệm thuật toán ReliefF với các bộ dữ liệu.

Bộ dữ liệu	Tổng số bản ghi (nghìn)	Số bản ghi tối đa có thể thực hiện (nghìn)	Thời gian tính toán (giây)
CreditScore	80	37	425,0
FintechUser	27	27	410,9
CreditCard	284,8	37	278,2
HomeCredit	307,5	37	256,6
HomeCredit2	307,5	37	283,7
CreditRisk	855	37	300,1
FinancialRisk	1048,5	37	255,6
VehicleLoan	233,1	38	371,5

Kết quả đánh giá khả năng phân loại thông qua các chỉ số (Độ chính xác, AUC, F1) của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi thuật toán ReliefF ở các ngưỡng 10%, 15% và 25% tổng số đặc trưng.

	Độ chính xác			AUC			F1		
Bộ dữ liệu	10%	15%	25%	10%	15%	25%	10%	15%	25%
CreditScore	91,4	91,6	91,9	0,84	0,85	0,87	0,65	0,69	0,72
FintechUser	67,5	70,2	78,4	0,57	0,65	0,70	0,12	0,41	0,52
CreditCard	99,1	99,4	99,5	0,74	0,92	0,93	0,25	0,83	0,92
HomeCredit	91,5	91,4	91,4	0,58	0,96	0,54	0,01	0,01	0,01
HomeCredit2	91,4	91,5	91,6	0,50	0,50	0,50	0,00	0,01	0,00
CreditRisk	94,1	94,5	99,7	0,50	0,50	0,99	0,00	0,04	0,99
FinancialRisk	80,1	80,4	80,5	0,50	0,50	0,52	0,02	0,03	0,10
VehicleLoan	78,0	78,0	78,0	0,50	0,50	0,50	0,01	0,00	0,01

A2. Kết quả thực nghiệm các thuật toán dựa trên lý thuyết thông tin

vi. Kết quả thực nghiệm thuật toán MIM

Kết quả thực nghiệm thuật toán MIM với các bộ dữ liệu.

Bộ dữ liệu	Tổng số bản ghi (nghìn)	Số bản ghi tối đa có thể thực hiện (nghìn)	Thời gian tính toán (giây)
CreditScore	80	80	2560,3
FintechUser	27	27	37,9
CreditCard	284,8	284,8	46,3
HomeCredit	307,5	307,5	337,3
HomeCredit2	307,5	307,5	2215,6
CreditRisk	855,0	855,0	132,1
FinancialRisk	1048,5	1048,5	48,9
VehicleLoan	233,1	233,1	43,0

Kết quả đánh giá khả năng phân loại thông qua các chỉ số (Độ chính xác, AUC, F1) của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi thuật toán MIM ở các ngưỡng 10%, 15% và 25% tổng số đặc trưng

	Độ chính xác (%)			AUC			F1		
Bộ dữ liệu	10%	15%	25%	10%	15%	25%	10%	15%	25%
CreditScore	89,8	91,2	92,1	0,73	0,77	0,80	0,57	0,63	0,68
FintechUser	72,1	73,5	72,8	0,64	0,66	0,65	0,48	0,51	0,51
CreditCard	99,8	99,8	99,8	0,50	0,62	0,62	0,00	0,33	0,33
HomeCredit	91,5	91,5	91,5	0,50	0,50	0,51	0,00	0,02	0,03
HomeCredit2	91	91,5	91,5	0,50	0,50	0,50	0,00	0,00	0,01
CreditRisk	98,0	99,0	99,8	0,83	0,93	0,98	0,77	0,92	0,99
FinancialRisk	80,3	80,4	80,3	0,51	0,54	0,54	0,05	0,16	0,16
VehicleLoan	84,4	84,5	84,7	0,65	0,65	0,66	0,46	0,47	0,48

vii. Kết quả thực nghiệm thuật toán MIFS

Kết quả thực nghiệm thuật toán MIFS với các bộ dữ liệu.

Bộ dữ liệu	Tổng số bản ghi (nghìn)	Số bản ghi tối đa có thể thực hiện (nghìn)	Thời gian tính toán (giây)
CreditScore	80	80	2503,7
FintechUser	27	27	23,2
CreditCard	284,8	284,8	38,7
HomeCredit	307,5	307,5	382,3
HomeCredit2	307,5	307,5	2342,6
CreditRisk	855,0	855,0	158,3
FinancialRisk	1048,5	1048,5	67,9
VehicleLoan	233,1	233,1	67,4

Kết quả đánh giá khả năng phân loại thông qua các chỉ số (Độ chính xác, AUC, F1) của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi thuật toán MIFS ở các ngưỡng 10%, 15% và 25% tổng số đặc trưng

	Độ chính xác			AUC			F1		
Bộ dữ liệu	10%	15%	25%	10%	15%	25%	10%	15%	25%
CreditScore	88,2	89,6	90,2	0,65	0,70	0,74	0,43	0,52	0,59
FintechUser	70,9	71,9	71,5	0,61	0,63	0,63	0,43	0,48	0,48
CreditCard	99,8	99,9	99,8	0,50	0,75	0,88	0,00	0,44	0,86
HomeCredit	91,6	91,6	91,5	0,50	0,50	0,51	0,00	0,00	0,02
HomeCredit2	91,5	91,5	91,5	0,50	0,50	0,50	0,00	0,00	0,02
CreditRisk	97,1	97,2	97,1	0,76	0,76	0,77	0,69	0,69	0,69
FinancialRisk	80,2	79,6	79,8	0,50	0,50	0,51	0,00	0,01	0,03
VehicleLoan	78,0	77,9	77,9	0,50	0,50	0,50	0,00	0,00	0,00

viii. Kết quả thực nghiệm thuật toán MRMR

Kết quả thực nghiệm thuật toán MRMR với các bộ dữ liệu.

Bộ dữ liệu	Tổng số bản ghi (nghìn)	Số bản ghi tối đa có thể thực hiện (nghìn)	Thời gian tính toán (giây)
CreditScore	80	80	2369,9
FintechUser	27	27	35,9
CreditCard	284,8	284,8	57,8
HomeCredit	307,5	307,5	364,5
HomeCredit2	307,5	307,5	2310,4
CreditRisk	855,0	855,0	148,9
FinancialRisk	1048,5	1048,5	67,7
VehicleLoan	233,1	233,1	69,6

Kết quả đánh giá khả năng phân loại thông qua các chỉ số (Độ chính xác, AUC, F1) của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi thuật toán MRMR ở các ngưỡng 10%, 15% và 25% tổng số đặc trưng:

	Độ chính xác			AUC			F1		
Bộ dữ liệu	10%	15%	25%	10%	15%	25%	10%	15%	25%
CreditScore	89,1	89,8	89,8	0,69	0,70	0,73	0,50	0,52	0,57
FintechUser	70,0	69,7	71,7	0,60	0,59	0,63	0,41	0,41	0,48
CreditCard	99,8	99,8	99,8	0,50	0,50	0,76	0,00	0,00	0,63
HomeCredit	91,4	91,5	91,5	0,50	0,50	0,50	0,00	0,00	0,00
HomeCredit2	91,5	91,5	91,3	0,51	0,51	0,51	0,04	0,05	0,04
CreditRisk	97,5	97,5	97,3	0,79	0,79	0,79	0,73	0,73	0,72
FinancialRisk	80,3	80,1	80,3	0,50	0,50	0,51	0,00	0,01	0,56
VehicleLoan	78,0	78,0	77,8	0,50	0,50	0,50	0,00	0,00	0,01

ix. Kết quả thực nghiệm thuật toán CIFE

Kết quả thực nghiệm thuật toán CIFE với các bộ dữ liệu:

Bộ dữ liệu	Tổng số bản ghi (nghìn)	Số bản ghi tối đa có thể thực hiện (nghìn)	Thời gian tính toán (giây)
CreditScore	80	80	2369,9
FintechUser	27	27	23,4
CreditCard	284,8	284,8	45,8
HomeCredit	307,5	307,5	376,9
HomeCredit2	307,5	307,5	2331,7
CreditRisk	855,0	855,0	155,3
FinancialRisk	1048,5	1048,5	68,9
VehicleLoan	233,1	233,1	69,1

Kết quả đánh giá khả năng phân loại thông qua các chỉ số (Độ chính xác, AUC, F1) của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi thuật toán CIFE ở các ngưỡng 10%, 15% và 25% tổng số đặc trưng:

	Độ chính xác			AUC			F1		
Bộ dữ liệu	10%	15%	25%	10%	15%	25%	10%	15%	25%
CreditScore	89,8	91,5	92,3	0,70	0,77	0,80	0,54	0,64	0,69
FintechUser	70,0	70,6	73,4	0,61	0,63	0,66	0,44	0,47	0,53
CreditCard	99,8	99,8	99,9	0,50	0,62	0,75	0,00	0,33	0,67
HomeCredit	91,6	91,6	91,6	0,51	0,51	0,51	0,02	0,02	0,03
HomeCredit2	91,6	91,6	91,5	0,50	0,50	0,50	0,02	0,02	0,01
CreditRisk	93,9	94,0	94,0	0,50	0,51	0,50	0,00	0,04	0,01
FinancialRisk	80,3	80,3	80,3	0,51	0,51	0,53	0,04	0,04	0,13
VehicleLoan	84,1	84,3	84,4	0,65	0,65	0,65	0,46	0,46	0,46

x. Kết quả thực nghiệm thuật toán JMI

Kết quả thực nghiệm thuật toán JMI với các bộ dữ liệu.

Bộ dữ liệu	Tổng số bản ghi (nghìn)	Số bản ghi tối đa có thể thực hiện (nghìn)	Thời gian tính toán (giây)
CreditScore	80	80	2680,8
FintechUser	27	27	36,8
CreditCard	284,8	284,8	28,8
HomeCredit	307,5	307,5	348,7
HomeCredit2	307,5	307,5	2150,4
CreditRisk	855,0	855,0	144,9
FinancialRisk	1048,5	1048,5	62,5
VehicleLoan	233,1	233,1	44,7

Kết quả đánh giá khả năng phân loại thông qua các chỉ số (Độ chính xác, AUC, F1) của mô hình phân loại với dữ liệu gồm các đặc trưng được chọn bởi thuật toán JMI ở các ngưỡng 10%, 15% và 25% tổng số đặc trưng:

	Độ chính xác			AUC			F1		
Bộ dữ liệu	10%	15%	25%	10%	15%	25%	10%	15%	25%
CreditScore	91,7	92,2	93,4	0,78	0,81	0,84	0,66	0,69	0,74
FintechUser	71,3	71,6	73,5	0,63	0,64	0,68	0,45	0,49	0,53
CreditCard	99,8	99,8	99,9	0,50	0,75	0,75	0,00	0,57	0,67
HomeCredit	91,2	91,4	91,5	0,50	0,50	0,50	0,00	0,02	0,01
HomeCredit2	91,2	91,7	91,7	0,50	0,51	0,51	0,01	0,01	0,01
CreditRisk	98,2	98,4	99,9	0,87	0,88	0,99	0,83	0,85	0,99
FinancialRisk	80,3	79,8	80,1	0,50	0,50	0,54	0,02	0,01	0,17
VehicleLoan	84,9	84,9	84,7	0,66	0,66	0,66	0,48	0,49	0,48