ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

Đồ án 03:
Báo cáo nghiên cứu Cây quyết định

Giảng viên: Bùi Tiến Lên
Lê Nhựt Nam
Võ Nhật Tân

Sinh viên: Lưu Huy Minh Quang – 23127016
Trần Cẩm Huy – 23127056
Trương Quốc Cường – 23127333
Nguyễn Tấn Phát – 23127449

Thành phố Hồ Chí Minh, 27 Tháng 08 Năm 2025

Mục lục

Thông tin chung

Bài báo phụ trách

Counterfactual Explanation Trees: Transparent and Consistent Actionable Recourse with Decision Trees

Thông tin thành viên

Nhóm: 07

MSSV	Họ tên	Lớp
23127016	Lưu Huy Minh Quang	23CLC03
23127056	Trần Cẩm Huy	23CLC03
23127333	Trương Quốc Cường	23CLC03
23127449	Nguyễn Tấn Phát	23CLC03

Bảng công việc

Họ tên	Công việc
Lưu Huy Minh Quang	Tóm tắt phần Thực nghiệm và phân tích kết quả
	Chạy thử nghiệm và so sánh kết quả
	Làm slide phần Yêu cầu và cách tiếp cận
Trần Cẩm Huy	Tóm tắt phần Giới thiệu
	Tóm tắt phần Kiến thức nền tảng
	Thiết lập môi trường chạy code
	Làm slide phần Kiến thức nền tảng và Tổng quan đóng góp
Trương Quốc Cường	Tóm tắt phần Các công trình liên quan
	Tóm tắt phần Tổng kết và định hướng nghiên cứu
	Phân tích code
	Làm slide phần Mục tiêu nghiên cứu
Nguyễn Tấn Phát	Tóm tắt phần Phương pháp nghiên cứu
	Trực quan hóa kết quả từ thử nghiệm
	Làm slide phần Phương pháp đề xuất

Tự đánh giá

Yêu cầu	Hoàn thành	Mô tả
Phân tích bài báo nghiên cứu	100%	Đọc và hiểu bài báo
		Viết báo cáo tóm tắt bằng tiếng Việt
Chạy thử nghiệm kết quả	100%	Phân tích code
		Thiết lập thử nghiệm
		Thực hiện thử nghiệm
		Tổng hợp kết quả và so sánh
Thuyết trình	100%	Làm slide để thuyết trình

Chương 1

Giới thiệu

Báo cáo này nhằm trình bày các hiểu biết của nhóm về nghiên cứu *Counterfactual Explanation Trees: Transparent Actionable Recourse*. Đây là 1 ứng dụng của Cây quyết định để đề xuất 1 giải pháp minh bạch và nhất quán cho bài toán Giải thích phản thực

1.1 Bối cảnh và động lực giải quyết bài toán

1.1.1 Bối cảnh

Trong bối cảnh hiện nay, các mô hình học máy phức tạp như mạng thần kinh sâu và phương pháp tập hợp cây ngày càng được áp dụng rộng rãi vào các nhiệm vụ ra quyết định quan trọng trong thực tế, ví dụ như chẩn đoán y tế, quyết định tuyển dụng, và phê duyệt khoản vay. Mặc dù các mô hình này đã đạt được độ chính xác dự đoán cao, chúng thường thiếu khả năng lý giải (Doshi-Velez and Kim 2017), gây khó khăn cho người dùng trong việc hiểu và tin tưởng vào các quyết định do mô hình đưa ra.

1.1.2 Giải thích phản thực (Counterfactual Explanations)

Để cung cấp những hiểu biết sâu sắc hơn cho người dùng, các phương pháp giải thích cục bộ hậu nghiệm cần làm rõ cả lý do tại sao các dự đoán không mong muốn được đưa ra và cách người dùng nên hành động để đạt được kết quả dự đoán mong muốn (Miller 1956)

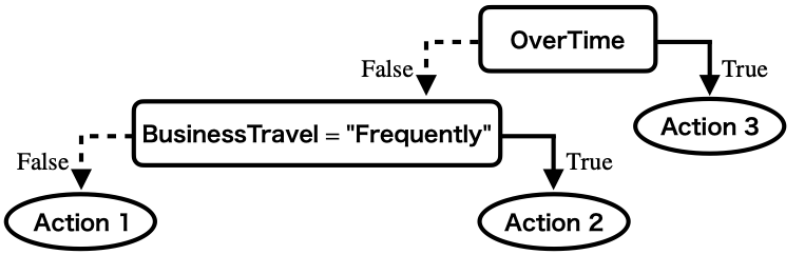
Giải thích phản thực (CE) (Wachter và cộng sự 2018) là một phương pháp giải thích cục bộ hậu nghiệm đáp ứng các yêu cầu này. CE cung cấp một hành động dưới dạng 1 vector nhiễu loạn mà một cá nhân có thể thực hiện để thay đổi kết quả dự đoán của bộ phân loại thành một kết quả mong muốn, ví dụ như giảm chỉ số BMI để giảm nguy cơ tiểu đường. Quá trình này thường liên quan đến việc tối ưu hóa chi phí của hành động trong khi vẫn đảm bảo kết quả mong muốn. Quá trình này còn có thể gọi là bài toán Giải pháp khắc phục khả thi (Actionable Recourse) (Ustun và cộng sự 2019)

1.1.3 Hạn chế của CE hiện tại

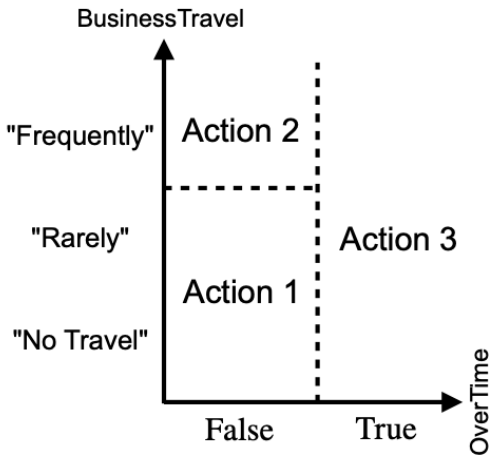
Tuy nhiên, trong thực tế, các hành động không chỉ gán cho 1 trường hợp cụ thể mà có thể gán cho nhiều trường hợp đồng thời, điều này mâu thuẫn với giả định của các phương pháp CE hiện tại (Karimi và cộng sự 2021). Một tình huống thực tế khi một công ty sử dụng bộ phân loại để dự đoán nguy cơ nghỉ việc của nhân viên và muốn gán các hành động để giảm thiểu rủi ro này (ví dụ: thăng chức hoặc tăng lương), những hành động này có thể ảnh hưởng đến toàn bộ nhân viên và hệ thống lương thưởng. Trong những tình huống như vậy, việc gán hành động cần đáp ứng hai yêu cầu quan trọng:

1. **Tính minh bạch:** Công ty cần giải thích cách thức các hành động được xác định cho toàn bộ nhân viên, đảm bảo lý do gán hành động là rõ ràng và tránh sự chênh lệch gây ra tình trạng không công bằng giữa các nhân viên
2. **Tính nhất quán:** Các lý do cho hành động được gán phải nhất quán, tránh các mâu thuẫn (ví dụ: không thể có trường hợp một nhân viên trên 40 tuổi bị chuyển công tác trong khi một

người khác trên 40 tuổi lại được thăng chức)



	HowToChange	Effectiveness Accuracy	Cost
Action 1	MonthlyIncome : + 1282$	83 %	0.17
Action 2	BusinessTravel : "Frequently" → "Rarely"	80 %	0.19
Action 3	OverTime : True → False	86 %	0.27



Hình 1.1: Ví dụ của CET học trên tập dữ liệu *IBM HR Analytics Employee Attrition* (Kaggle 2017)

Các framework CE hiện có và các phương pháp giải thích cục bộ không thể đáp ứng được các yêu cầu về tính minh bạch và nhất quán này vì chúng không tính đến toàn bộ không gian đầu vào (Ribeiro và cộng sự 2018)

1.2 Mục tiêu nghiên cứu

Để giải quyết những hạn chế của các phương pháp CE hiện có và đáp ứng nhu cầu về tính minh bạch và nhất quán, nghiên cứu này giới thiệu một framework mới: Cây Giải thích phản thực (CET). Ý tưởng chính là đề xuất 1 phương pháp phân chia không gian đầu vào theo một cách dễ hiểu và gán một hành động phù hợp cho mỗi không gian con.

Nhờ các thuộc tính vốn có của cây quyết định, CET mang lại hai lợi thế quan trọng:

1. **Tính minh bạch:** Các lý do để gán hành động được tóm tắt trong một cấu trúc dễ hiểu (cấu trúc cây), giúp người dùng dễ dàng hiểu cách các hành động được gán cho các trường hợp trên toàn bộ không gian đầu vào dưới dạng những luật if-then-else
2. **Tính nhất quán:** CET đảm bảo gán một cặp duy nhất gồm hành động và lý do cho bất kỳ trường hợp nào, do nó phân chia không gian đầu vào thành các không gian con riêng biệt và chỉ một nút lá được xác định duy nhất cho mỗi đầu vào. Điều này đảm bảo không có xung đột về lý do gán hành động giữa các trường hợp

1.3 Tổng quan đóng góp của nghiên cứu

Nghiên cứu này đóng góp những điểm chính sau:

1. **Giới thiệu CET**: Đề xuất Cây Giải thích phản thực (CET) - một cây quyết định gán các hành động một cách hiệu quả cho các trường hợp đầu vào trên toàn bộ không gian đầu vào. Tận dụng tính chất của Cây quyết định, CET cung cấp một quy trình gán hành động minh bạch và nhất quán, đảm bảo gán một cặp duy nhất gồm hành động và lý do cho bất kỳ trường hợp nào
2. **Mô hình hóa bài toán học CET**: Nghiên cứu đã mô hình hóa bài toán học một CET từ một tập hợp các trường hợp đã cho dưới dạng một bài toán tối ưu. Đồng thời, đề xuất một thuật toán để giải quyết bài toán này bằng cách sử dụng tìm kiếm cục bộ ngẫu nhiên, có khả năng sử dụng chiến lược cắt tỉa dựa trên cận của hàm mục tiêu
3. **Thực nghiệm và nghiên cứu người dùng**: Các thử nghiệm được tiến hành trên các bộ dữ

liệu công khai để đánh giá hiệu quả của CET so với các phương pháp hiện có. Ngoài ra, các nghiên cứu người dùng đã chứng minh rằng CET dễ hiểu đối với con người. Một ví dụ về CET được học trên tập dữ liệu *IBM HR Analytics Employee Attrition* đã minh họa tính dễ hiểu và khả năng gán các cặp hành động-lý do duy nhất của nó

4. **Tính linh hoạt**: CET có khả năng tích hợp với nhiều loại hàm chi phí khác nhau (ví dụ như độ dịch chuyển bách phân vị lớn nhất (Ustun và cộng sự 2019)). Hơn nữa, sau khi quá trình huấn luyện CET được hoàn tất, mô hình có thể gán hành động cho cả những trường hợp mới chưa từng xuất hiện trong tập huấn luyện, tương tự như cơ chế giải thích phân bổ (Chen và cộng sự 2018)

1.4 Cấu trúc báo cáo

Báo cáo được tổ chức thành 6 chương với nội dung cụ thể từng chương như sau

Chương 1: Giới thiệu: Giới thiệu bối cảnh, động lực, mục tiêu của nghiên cứu và tổng quan về đóng góp của nghiên cứu

Chương 2: Các công trình liên quan: Xem xét các nghiên cứu liên quan về Cây quyết định và bài toán Giải thích phản thực, so sánh với các phương pháp hiện có, xác định khoảng trống nghiên cứu và xác định vị trí của nghiên cứu này trong lĩnh vực tổng quát hơn

Chương 3: Kiến thức nền tảng: Trình bày các khái niệm cơ bản về Cây quyết định, nền tảng toán học, các thuật toán và kỹ thuật chính được trích dẫn trong bài báo

Chương 4: Phương pháp nghiên cứu: Giải thích chi tiết phương pháp tiếp cận được đề xuất, mô tả thuật toán và mã giả, phân tích lý thuyết, và các khía cạnh đổi mới

Chương 5: Thực nghiệm và Phân tích kết quả: Mô tả thiết lập thực nghiệm, bộ dữ liệu, các chỉ số đánh giá, phân tích và diễn giải kết quả, so sánh hiệu suất và ý nghĩa thống kê

Chương 6: Kết luận và định hướng nghiên cứu tương lai: Tóm tắt các phát hiện và đóng góp chính, nêu bật điểm mạnh và hạn chế cùng các hướng nghiên cứu trong tương lai

Chương 2

Các Công Trình Liên Quan

2.1 Tổng quan về nghiên cứu cây quyết định

Cây quyết định (Decision Tree - DT) là một trong những mô hình học máy cơ bản và được sử dụng phổ biến nhờ tính minh bạch và dễ hiểu. Một cách truyền thống, các thuật toán cổ điển như **CART** (Breiman et al., 1984) và **ID3** (Quinlan, 1986) xây dựng cây quyết định theo hướng tiếp cận *tham lam* (*greedy*): tại mỗi nút, chúng chọn đặc trưng tốt nhất dựa trên một tiêu chí cục bộ (Gini impurity cho CART, information gain cho ID3) để phân tách, mà không cân nhắc đến ảnh hưởng lên các nút con; nhãn của lá sau đó được gán dựa trên đa số lớp hoặc các tiêu chí khác.

Các nghiên cứu gần đây mở rộng cây quyết định để xử lý các mục tiêu phi tiêu chuẩn, ví dụ như cây tối ưu hóa (Hu et al., 2019; Aglin et al., 2020) và tìm kiếm ngẫu nhiên (stochastic search) cho các mục tiêu toàn cục. Những công trình này cho thấy cây quyết định có thể được điều chỉnh không chỉ để phân loại mà còn để giải các bài toán tối ưu có cấu trúc, đồng thời vẫn giữ được tính minh bạch.

2.2 So sánh với các hướng tiếp cận trước đó

Nhiều phương pháp *post-hoc* đã được đề xuất để đưa ra giải thích cục bộ từ các mô hình phức tạp, trong đó **Giải thích phản thực (Counterfactual Explanation – CE)** hay **Actionable Recourse** nổi bật với mục tiêu cung cấp các hành động khả thi nhằm thay đổi đầu ra của mô hình cho từng cá thể (Karimi et al., 2020b).

Hầu hết các phương pháp CE hiện nay tập trung vào giải thích cục bộ, bằng cách đưa ra một hành động duy nhất (Wachter et al., 2018; Karimi et al., 2020a; Kanamori et al., 2020; Schut et al., 2021) hoặc nhiều hành động đa dạng (Ustun et al., 2019; Mothilal et al., 2020) cho đầu vào là dữ liệu riêng lẻ.

Mặc dù hiệu quả ở cấp cục bộ, các phương pháp này gặp hạn chế về **phủ toàn cục** (không thể đưa ra giải thích chung cho một tập thể) và đôi khi thiếu **độ ổn định** (Ghorbani et al., 2019; Dombrowski et al., 2019). Một số framework đã được đề xuất nhằm tóm tắt giải thích cục bộ thành bức tranh toàn cục, ví dụ như **AReS** (Rawal & Lakkaraju, 2020), **MAME** (Ramamurthy et al., 2020) và **GIME** (Gao et al., 2021). Trong đó, AReS sử dụng tập luật hai cấp để tổng hợp hành động toàn cục nhưng có thể không bao phủ toàn bộ không gian đầu vào và đôi khi gán nhiều hành động cho cùng một cá thể, làm giảm tính **minh bạch** và **nhất quán**. MAME và GIME sử dụng phân cụm phân cấp hoặc mô hình hóa chủ đề có thể giải thích để tạo tóm tắt toàn cục, nhưng không đảm bảo tính nhất quán.

Bảng 2.1: So sánh các hướng tiếp cận

Phương pháp	Mục tiêu tối ưu	Phủ toàn cục	Minh bạch (Transparency)	Nhất quán (Consistency)
CE (Wachter et al., 2018; Karimi et al., 2020a; Kanamori et al., 2020; Schut et al., 2021)	Local	Không	Có	Có
CE (Ustun et al., 2019; Mothilal et al., 2020)	Local	Không	Có	Có
AReS (Rawal & Lakkaraju, 2020)	Global	Có, nhưng không đầy đủ	Không	Không
MAME (Ramamurthy et al., 2020)	Global	Có	Có	Không
GIME (Gao et al., 2021)	Global	Có	Có	Không
CET	Global	Có	Có	Có

Kết luận So với các phương pháp CE truyền thống và các framework tổng hợp toàn cục như AReS, MAME hay GIME, **CET (Counterfactual Explanation Tree)** vượt trội nhờ khả năng đồng thời đảm bảo **tính minh bạch, nhất quán và phủ toàn cục**.

2.3 Nhận diện khoảng trống nghiên cứu

Các phương pháp **Counterfactual Explanation (CE) cục bộ** hiện nay, như Wachter et al. (2018), Karimi et al. (2020) hay Ustun et al. (2019), tập trung vào việc đưa ra giải thích cho từng cá thể bằng một hoặc nhiều hành động khả thi. Mặc dù hiệu quả ở cấp cục bộ, các phương pháp này gặp hạn chế về **phủ toàn cục** và đôi khi thiếu tính ổn định, do không xem xét cấu trúc tổng thể của dữ liệu.

Để tóm tắt giải thích cục bộ thành bức tranh toàn cục, một số framework như **AReS**, **MAME** hay **GIME** đã được đề xuất. Tuy nhiên, các phương pháp này thường gặp vấn đề về **minh bạch** và **nhất quán**, ví dụ như gán nhiều hành động cho cùng một đầu vào hoặc không bao phủ đầy đủ không gian đầu vào.

Khoảng trống nghiên cứu ở đây là chưa có phương pháp nào vừa cải thiện tính **toàn cục** so với CE cục bộ, vừa đảm bảo **minh bạch** và **nhất quán** so với các framework global cũ. **Counterfactual Explanation Tree (CET)** được đề xuất nhằm lấp đầy khoảng trống này bằng cách sử dụng cấu trúc cây quyết định kết hợp với tối ưu hóa toàn cục có ràng buộc, giúp tạo ra các giải thích phản thực vừa đáng tin cậy, vừa minh bạch cho từng cá thể và toàn bộ dữ liệu.

2.4 Vị trí của nghiên cứu hiện tại trong bối cảnh rộng hơn

Cây giải thích phản thực (Counterfactual Explanation Tree – CET) giải quyết các vấn đề nêu trên bằng cách kết hợp **cấu trúc cây quyết định** với **tối ưu hóa hành động theo nhóm**:

- **Minh bạch**: CET sử dụng cấu trúc cây, với mỗi nút nội bộ chứa luật phân nhánh dễ hiểu và mỗi lá tương ứng với một hành động duy nhất.
- **Nhất quán**: Mỗi cá thể được gán chính xác một hành động, tránh mâu thuẫn.
- **Phủ toàn cục**: Việc phân chia không gian đầu vào đảm bảo mọi cá thể đều nhận được hành động, khắc phục giới hạn của CE cục bộ.

Do đó, CET kết hợp nghiên cứu cây quyết định cổ điển và các phương pháp giải thích phản thực hiện đại, tạo ra một framework vừa **có thể giải thích được**, vừa **có thể hành động toàn cục**.

Chương 3

Kiến thức nền tảng

3.1 Những khái niệm cơ bản về Cây quyết định

3.1.1 Định nghĩa

Cây quyết định là một trong những thuật toán học có giám sát cơ bản. Đây là mô hình phổ biến được xây dựng từ tập hợp các quy tắc if–then–else, được biểu diễn dưới dạng cấu trúc cây nhị phân. Phương pháp này mang lại các mô hình dễ diễn giải, có thể áp dụng cho cả bài toán phân loại và hồi quy. Về bản chất, ta tìm cách học một hàm $f : \mathcal{X} \to \mathcal{Y}$, trong đó mỗi đầu vào x sẽ được ánh xạ thành một nhãn phân loại hoặc một giá trị dự đoán y.

3.1.2 Cấu trúc

- Nút trong: biểu diễn một phép kiểm tra đối với thuộc tính tương ứng trong tập dữ liệu huấn luyện
- Nhánh: thể hiện các giá trị khả dĩ của thuộc tính tại nút đó
- Nút lá: là đầu ra của hàm f, đồng thời cũng chính là quyết định cuối cùng

Đối với mỗi giá trị đầu vào x, quá trình dự đoán được thực hiện bằng cách lần lượt so sánh tại các nút bắt đầu từ nút gốc, lựa chọn nhánh thích hợp cho đến khi đạt đến một nút lá, nơi chứa quyết định ứng với x (Russell and Norvig 2021).

3.1.3 Tiêu chí phân chia

Mục tiêu cốt lõi trong việc xây dựng Cây quyết định là xác định thuộc tính tối ưu để phân chia tập dữ liệu huấn luyện. Việc lựa chọn này dựa trên một hàm chi phí nhằm đo lường mức độ hỗn tạp tại từng nút và tìm ra phép tách giúp giảm hỗn tạp nhiều nhất, từ đó được xem là lựa chọn tối ưu.

Đối với bài toán phân loại :

- **Gini Impurity**: Thước đo này phản ánh xác suất một phần tử được chọn ngẫu nhiên sẽ bị gán sai nhãn. Một nút có giá trị Gini bằng 0 được coi là thuần khiết, tức toàn bộ phần tử đều thuộc cùng một lớp. Chỉ số này được tính theo công thức:

$$G(p) = \sum_{i=1}^{k} p_i(1 - p_i) = 1 - \sum_{i=1}^{k} p_i^2$$

  trong đó p_i là xác suất phần tử thuộc lớp i tại nút đang xét.

- **Information Gain**: Dựa trên khái niệm Entropy, vốn đo lường mức độ ngẫu nhiên hay độ hỗn tạp của dữ liệu. Entropy càng nhỏ chứng tỏ tập dữ liệu càng thuần nhất. Công thức được định nghĩa như sau:

$$Entropy(S) = -\sum_{i=1}^{k} p_i \log_2(p_i)$$

với k là số lớp khác nhau trong S.

Information Gain thể hiện mức giảm entropy sau khi thực hiện phép tách dữ liệu. Công thức được cho bởi:

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

trong đó S là tập dữ liệu, A là thuộc tính được xem xét, $Values(A)$ là tập giá trị khả dĩ của A, và $|S_v|$ là số phần tử trong S có giá trị thuộc v.

Đối với bài toán hồi quy :

- **Sai số bình phương trung bình (MSE)**: Đây là thước đo khoảng cách trung bình có trọng số giữa giá trị dự đoán và giá trị thực. Quá trình phân chia sẽ được lựa chọn sao cho tối thiểu hóa MSE của các nút con. Công thức được biểu diễn như sau:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

trong đó y_i là giá trị thực, và $\hat{y}_i$ là giá trị dự đoán.

3.2 Cơ sở toán học và ký hiệu

3.2.1 Ký hiệu chung và Định nghĩa cơ bản

- Bài toán dự đoán thường được quy về bài toán phân loại nhị phân. Trường hợp phân loại đa lớp có thể được chuyển đổi thành nhiều bài toán nhị phân giữa một lớp mục tiêu và phần còn lại. Chẳng hạn, trong dự đoán tỉ lệ nghỉ việc của nhân viên, ta có thể phân loại họ vào các nhóm có tỉ lệ nghỉ việc cao, trung bình hoặc thấp. Việc gán một nhân viên vào nhóm "cao" có thể được xem như chuỗi quyết định nhị phân: so sánh giữa "cao" và "thấp", "cao" và "trung bình", hoặc "trung bình" và "thấp".
- **Tập số nguyên dương**: Với $n \in \mathbb{N}$, ký hiệu $[n] := 1, \ldots, n$.
- **Hàm chỉ thị**: $\mathbb{I}[\psi]$ biểu diễn hàm chỉ thị của mệnh đề ψ; theo đó $\mathbb{I}[\psi] = 1$ nếu ψ đúng, và $\mathbb{I}[\psi] = 0$ nếu ψ sai.
- **Miền đầu vào và đầu ra**:
  - Miền đầu vào: $\mathcal{X} \subseteq \mathbb{R}^D$
  - Miền đầu ra: $\mathcal{Y} = -1, +1$
- **Trường hợp**: Một vector $x = (x_1, \ldots, x_D) \in \mathcal{X}$.
- **Bộ phân loại**: Một hàm $f : \mathcal{X} \to \mathcal{Y}$, đóng vai trò là bộ phân loại cần được giải thích.
- **Hành động**: Một vector nhiễu loạn $a \in \mathbb{R}^D$.
- **Hàm chi phí**: $c(a|x) : \mathcal{X} \times \mathbb{R}^D \to \mathbb{R}\geq 0$, đo lường mức nỗ lực cần thiết để áp dụng hành động a lên trường hợp x. Theo quy ước, $c(\mathbf{0}|x) = 0$.
- **Hàm mất mát 0-1**: $l_{01}(\hat{y}, y) = \mathbb{I}[\hat{y} \neq y]$, phản ánh việc dự đoán $\hat{y}$ có khác với nhãn thực y hay không.
- **Điểm không hợp lệ**: $i_\gamma(a|x) := c(a|x) + \gamma \cdot l_{01}(f(x+a), +1)$, đánh giá hiệu quả của hành động a đối với trường hợp x bằng cách kết hợp chi phí $c(a|x)$ và mất mát phân loại $l_{01}(f(x+a), +1)$, trong đó $\gamma > 0$ là tham số điều chỉnh cân bằng.

3.2.2 Bài toán Giải thích phản thực

Giải thích phản thực (Wachter và cộng sự 2018) là một phương pháp giải thích hậu nghiệm cung cấp một sự nhiễu loạn để thay đổi kết quả dự đoán của bộ phân loại. Một cá nhân có thể xem sự

nhiễu loạn này như một "hành động"để đạt được kết quả mong muốn.

Cụ thể hơn, đối với một bộ phân loại $f : \mathcal{X} \to \mathcal{Y}$, một lớp mục tiêu $y^* \in \mathcal{Y}$ và một trường hợp $x \in \mathcal{X}$ sao cho $f(x) \neq y^*$, CE cố gắng cung cấp một vector nhiễu loạn a làm thay đổi kết quả dự đoán thành đầu ra mong muốn, tức là $f(x + a) = y^*$. Cá nhân x có thể thực hiện nhiễu loạn a như một "hành động"để đạt được kết quả quyết định mong muốn y^* từ bộ phân loại f.

Mục tiêu của CE là tìm một hành động a làm thay đổi kết quả dự đoán thành $f(x + a) = y^*$ và giảm thiểu chi phí $c(a|x)$ của hành động đó. Mục tiêu này có thể được hình thành như một bài toán tối ưu hóa (Karimi và cộng sự 2021):

$$\min_{a \in \mathcal{A}} c(a|x) \quad \text{với điều kiện} \quad f(x + a) = y^*$$

trong đó $\mathcal{A}$ là tập hợp các hành động khả thi và c là hàm chi phí đo lường nỗ lực cần thiết để thực hiện hành động a

Mệnh đề 1 (Cận trên của độ chênh lệch chi phí)**.** *Với một tập các thể hiện $\mathcal{X} \subseteq \mathcal{X}$, gọi a^* là nghiệm tối ưu của bài toán (2). Với một thể hiện $x \in \mathcal{X}$, ký hiệu $c^*(x)$ là giá trị tối ưu của bài toán (1). Khi đó, ta có*

$$c(a^* \mid x) - c^*(x) \;\leq\; c_x \cdot \|x - x^\circ\|,$$

trong đó

$$x^\circ = \arg\min_{x \in \mathcal{X}} \; w^\top x.$$

Mệnh đề 1 cho thấy cận trên của độ chênh lệch chi phí giữa CE cá thể hoá và CE theo nhóm phụ thuộc vào thể hiện $x^\circ \in \mathcal{X}$ xa nhất so với siêu phẳng quyết định của f.

Ví dụ. Giả sử ta có nhiều thể hiện nằm gần biên quyết định và chỉ một thể hiện x° nằm xa biên. Khi đó, chi phí cần thiết để đảo ngược tất cả kết quả sẽ phụ thuộc vào x°, dẫn đến giá trị chi phí cao hơn. Nói cách khác, với một thể hiện $x \in \mathcal{X}$, nghiệm tối ưu a^* của (2) có thể trở nên thiếu thực tế khi chi phí $c(a^* \mid x)$ bị chi phối bởi các thể hiện khác trong $\mathcal{X}$. Điều này xảy ra vì ràng buộc trong (2), tức yêu cầu $f(x + a) = +1$ với mọi $x \in \mathcal{X}$, là quá chặt. Kết quả gợi ý rằng ràng buộc này có thể ngăn ta thu được một hành động hiệu quả.

Vấn đề trên cũng cho thấy hạn chế của các mô hình trước đây: áp dụng ràng buộc quá nghiêm ngặt để thay đổi đồng thời tất cả các thể hiện trong tập $\mathcal{X}$, lời giải thu được thường không thực tế và chi phí cao. Bài toán giải thích phản thực cho nhiều trường hợp bằng **CET** được đề xuất nhằm khắc phục hạn chế của các phương pháp CE truyền thống, vốn chỉ tập trung vào việc xác định một hành động tối ưu cho từng trường hợp riêng lẻ.

Bài toán 1 (CE cho nhiều trường hợp)**.** *Xét một tập N trường hợp $X \subseteq \mathcal{X}$ sao cho với mọi $x \in X$, ta có $f(x) \neq +1$. Gọi $\mathcal{A}(X) = \bigcap_{x \in X} \mathcal{A}(x)$ là tập hợp các hành động khả thi đồng thời cho tất cả các trường hợp trong X, và $\gamma > 0$ là tham số cân bằng. Khi đó, bài toán tìm hành động tối ưu $a^* \in \mathcal{A}(X)$ được phát biểu như sau:*

$$\min_{a \in A(X)} \quad g_\gamma(a|X) := \sum_{x \in X} i_\gamma(a|x)$$

trong đó $g_\gamma(a|X)$ biểu thị tổng điểm không hợp lệ của hành động a trên toàn bộ tập trường hợp X.

Mệnh đề 2 (Tính đơn điệu của g_γ)**.** *Xét một tập hợp các trường hợp X được phân tách thành hai tập con rời nhau X_1 và X_2 ($X_1 \cup X_2 = X$, $X_1 \cap X_2 = \emptyset$). Khi đó, bất đẳng thức sau luôn được thỏa mãn:*

$$g_\gamma(a^*_X|X) \geq g_\gamma(a^*_{X_1}|X_1) + g_\gamma(a^*_{X_2}|X_2)$$

trong đó $a^*_X := \arg\min_{a \in \mathcal{A}(X)} g_\gamma(a|X)$.

Chứng minh. Theo định nghĩa của g_γ, với tập X được chia thành hai tập con rời nhau X_1 và X_2, ta có:

$$g_\gamma(a \mid X) = \sum_{x \in X} i_\gamma(a \mid x) = \sum_{x \in X_1} i_\gamma(a \mid x) + \sum_{x \in X_2} i_\gamma(a \mid x) = g_\gamma(a \mid X_1) + g_\gamma(a \mid X_2)$$

Vì $X_1 \subset X$ và $X_2 \subset X$, nên suy ra $\mathcal{A}(X) \subseteq \mathcal{A}(X_1)$ và $\mathcal{A}(X) \subseteq \mathcal{A}(X_2)$. Do đó, $a^*_X \in \mathcal{A}(X_1)$ và $a^*_X \in \mathcal{A}(X_2)$. Từ đây, ta có:

$$g_\gamma(a^*_X|X_1) \geq g_\gamma(a^*_{X_1}|X_1), \quad g_\gamma(a^*_X|X_2) \geq g_\gamma(a^*_{X_2}|X_2)$$

Kết hợp các bất đẳng thức trên, ta thu được:

$$g_\gamma(a^*_X \mid X) = g_\gamma(a^*_X \mid X_1) + g_\gamma(a^*_X \mid X_2) \geq g_\gamma(a^*_{X_1}|X_1) + g_\gamma(a^*_{X_2}|X_2)$$

□

Mệnh đề 2 chỉ ra rằng có thể gán hành động hiệu quả hơn cho từng trường hợp $x \in X$ bằng cách tối ưu hóa hàm mục tiêu $g_\gamma(a|X)$ trên các phân tập X_1, X_2 được tách rời từ X. Tuy nhiên, việc chia tập X thành quá nhiều phân tập có thể làm suy giảm tính diễn giải của quá trình gán hành động.

3.2.3 Cây giải thích phản thực

Một CET $h : \mathcal{X} \to \mathcal{A}$ được định nghĩa như một cây quyết định, trong đó mỗi đầu vào $x \in \mathcal{X}$ được gán với một hành động tương ứng. Cấu trúc này được xây dựng từ một tập hợp các quy tắc if–then–else duy nhất và được biểu diễn dưới dạng cây nhị phân.

Với một trường hợp x, CET h gán cho nó một hành động $a_l \in \mathcal{A}$ tương ứng với nút lá $l \in \mathcal{L}(h)$ mà x dẫn tới. Nút lá l được xác định bằng cách duyệt từ nút gốc theo các quy tắc phân nhánh tại từng nút trong. Tập hợp các miền con $r_l \mid l \in \mathcal{L}(h)$ chia không gian đầu vào thành các phân vùng riêng biệt (Freitas 2014). Do đó, mỗi trường hợp $x \in \mathcal{X}$ được gán duy nhất một cặp hành động a_l và luật r_l.

Một CET h có thể được biểu diễn dưới dạng:

$$h(x) = \sum_{l \in \mathcal{L}(h)} a_l \cdot \mathbb{I}[x \in r_l]$$

. trong đó:

- $\mathcal{L}(h)$ là tập các nút lá trong cây h.
- $r_l \subseteq \mathcal{X}$ là miền con gắn với lá l, được xác định bởi chuỗi quy tắc phân nhánh trên đường đi từ gốc đến lá. Mỗi r_l có thể được hiểu như một quy tắc, trong đó hành động a_l được gán cho toàn bộ các trường hợp thuộc miền này (Guidotti và cộng sự 2018).

Hình 1.1 minh họa một ví dụ điển hình về CET cùng các phân vùng tương ứng của nó.

Quá trình học một CET bao gồm hai bước chính:

(i) Xác định và gán hành động đơn lẻ hiệu quả nhất cho nhiều trường hợp, dựa trên chi phí cần thiết và khả năng thay đổi dự đoán theo hướng mong muốn.

(ii) Xây dựng một cây quyết định nhằm phân vùng tập hợp các trường hợp, với mục tiêu cân bằng giữa hiệu quả và khả năng diễn giải.

Bài toán 2 (Học một CET)**.** *Bài toán học CET tối ưu h^* từ tập ứng viên $\mathcal{H}$ với tập huấn luyện $X \subseteq \mathcal{X}$ được mô hình hóa như một bài toán tối ưu:*

$$\min_{h \in \mathcal{H}} \quad o_{\gamma,\lambda}(h|X) := \frac{1}{N} \sum_{x \in X} i_\gamma(h(x)|x) + \lambda \cdot |\mathcal{L}(h)|$$

Trong đó:

- $\mathcal{H}$ *là tập các CET* h *thỏa mãn* $h(x) \in A(x)$ *với mọi* $x \in X$*, đảm bảo tính khả thi của các hành động được gán.*
- $o_{\gamma,\lambda}(h|X)$ *là hàm mục tiêu cần tối ưu để học CET.*
- $\frac{1}{N}\sum_{x\in X} i_\gamma(h(x)|x)$ *biểu diễn điểm không hợp lệ trung bình của các hành động* $a = h(x)$ *được gán cho các trường hợp* $x \in X$*.*
- $\lambda \cdot |\mathcal{L}(h)|$ *là số hạng điều chỉnh thể hiện độ phức tạp của cây, trong đó* $|\mathcal{L}(h)|$ *là số lượng lá, tương ứng với số hành động khác nhau mà cây* h *đưa ra. Thành phần này phản ánh khả năng diễn giải của CET.*
- $\lambda > 0$ *là tham số điều chỉnh nhằm cân bằng giữa hiệu quả của các hành động gán bởi* h *và khả năng diễn giải của mô hình.*

3.3 Các thuật toán và kỹ thuật chính được tham chiếu

Trong công trình về *Cây giải thích phản thực* (Counterfactual Explanation Trees – CET) (Kanamori và cộng sự 2022), tác giả đã kế thừa và sử dụng nhiều ý tưởng quan trọng từ các nghiên cứu trước đó. Có thể tóm lược một số hướng tiếp cận chính như sau:

- **Giải thích phản thực (Counterfactual Explanations – CE)**: khái niệm cơ bản được giới thiệu bởi Wachter và cộng sự (2018), trong đó mô hình đưa ra một "hành động" a giúp thay đổi dự đoán hiện tại $f(x)$ sang kết quả mong muốn y^*.
- **Biện pháp khắc phục khả thi (Actionable Recourse)**: các phương pháp xây dựng hành động có thể thực hiện được trong thực tế, thường được mô hình hóa bằng các bài toán tối ưu. Một ví dụ tiêu biểu là phương pháp quy hoạch tuyến tính nguyên hỗn hợp (MILO) để tìm ra hành động tối ưu (Ustun và cộng sự 2019).
- **Giải thích phản thực cho nhiều trường hợp (Group-wise CE)**: mở rộng CE truyền thống từ một cá thể riêng lẻ sang việc gán hành động đồng thời cho nhiều cá thể. Hướng nghiên cứu này có liên hệ chặt chẽ với các khảo sát về biện pháp khắc phục trong thuật toán (Karimi và cộng sự 2021).
- **Sử dụng cây quyết định trong việc giải thích**: khai thác đặc trưng dễ hiểu của cấu trúc cây để bảo đảm rằng mỗi cá thể chỉ được gắn với một cặp *(hành động, lý do)* duy nhất, đồng thời giữ được tính minh bạch và dễ diễn giải (Freitas 2014; Guidotti và cộng sự 2018).
- **Thuật toán tìm kiếm cục bộ ngẫu nhiên (Stochastic Local Search)**: phương pháp huấn luyện CET bằng cách thay đổi ngẫu nhiên các quy tắc phân nhánh trong cây (chèn, xóa hoặc thay thế), sau đó tối ưu hóa hành động ở từng nút lá bằng quy hoạch nguyên hỗn hợp (Kanamori và cộng sự 2022).

3.4 Cơ sở lý thuyết cần thiết để hiểu phương pháp

Để nắm bắt cách xây dựng và vận hành CET, cần xem xét một số khái niệm toán học và nguyên tắc cơ bản sau:

- **Hàm chi phí**: đo lường mức độ nỗ lực cần thiết để thực hiện một hành động. Ví dụ điển hình là thước đo *Max Percentile Shift*, vốn được sử dụng rộng rãi trong các nghiên cứu về biện pháp khắc phục (Ustun và cộng sự 2019).
- **Hàm mất mát 0–1**: phản ánh việc dự đoán sau khi áp dụng hành động có khớp với nhãn mục tiêu hay không, được định nghĩa bởi:

$$l_{01}(\hat{y}, y) = \mathbb{I}[\hat{y} \neq y].$$

- **Điểm không hợp lệ**: kết hợp giữa chi phí và hiệu quả của hành động, được tính bằng:

$$i_\gamma(a|x) = c(a|x) + \gamma \cdot l_{01}(f(x+a), +1),$$

trong đó $c(a|x)$ là chi phí, l_{01} là hàm mất mát 0–1, và $\gamma > 0$ là tham số cân bằng (Kanamori và cộng sự 2022).

- **Hàm mục tiêu cho CE nhóm**: với một tập nhiều cá thể X, hành động tối ưu được tìm bằng cách cực tiểu hóa:

$$g_\gamma(a|X) = \sum_{x \in X} i_\gamma(a|x).$$

- **Hai tính chất cốt lõi của CET**:
  - *Minh bạch*: lý do gán hành động được thể hiện rõ ràng thông qua các quy tắc phân nhánh trong cây quyết định.
  - *Nhất quán*: mỗi cá thể chỉ được gán duy nhất một cặp *(hành động, lý do)*, tránh mâu thuẫn khi áp dụng trên toàn bộ tập dữ liệu (Kanamori và cộng sự 2022).

Chương 4

Phương pháp nghiên cứu

4.1 Phương pháp đề xuất

Counterfactual Explanation Tree (CET) là một framework mới để gán hành động (actions) cho nhiều instances đồng thời bằng cách sử dụng cây quyết định. Khác với các phương pháp Counterfactual Explanation (CE) truyền thống chỉ tập trung vào việc cung cấp hành động cho một instance đơn lẻ, CET được thiết kế để xử lý trường hợp cần gán hành động cho nhiều instances một cách minh bạch (transparent) và nhất quán (consistent).

CET là một phương pháp mới, kết hợp giữa giải thích phản thực và cây quyết định, nhằm cung cấp hành động khả thi cho nhiều instance cùng lúc một cách minh bạch và nhất quán. Các kết quả thực nghiệm và khảo sát người dùng cho thấy CET vượt trội so với các phương pháp hiện có về cả hiệu quả và khả năng giải thích.

Một thuật toán ngây thơ là chiến lược phân hoạch tham lam từ trên xuống như CART (Breiman et al., 1984), trong đó đệ quy xác định luật phân nhánh của mỗi nút trong để cải thiện nhiều nhất giá trị mục tiêu sau khi tách. Tuy nhiên, để xác định luật phân nhánh của mỗi nút trong, chúng ta cần giải Bài toán 1 cho số lượng luật phân nhánh ứng viên, điều này là không khả thi về mặt tính toán. Trong các thí nghiệm sơ bộ, các quan sát cho thấy rằng phương pháp này không tạo ra một cây quyết định có chất lượng tốt xét theo điểm số không hợp lệ i_r. Nguyên nhân là do phương pháp này thường chọn một luật phân nhánh không phù hợp ở một nút gần gốc và do đó không phân tách được các mẫu đầu vào X theo cách mà chúng được gán một hành động hiệu quả trong mỗi nút lá mà chúng đi đến.

Từ những quan sát trên, nghiên cứu này đề xuất một thuật toán dựa trên tìm kiếm cục bộ ngẫu nhiên (stochastic local search). Tìm kiếm cục bộ ngẫu nhiên đã được chứng minh là phù hợp cho việc học các mô hình quy tắc phi chuẩn (Wang, 2019; Pan et al., 2020). Thuật toán bao gồm hai bước: xác định luật phân nhánh của các nút trong bằng cách sử dụng tìm kiếm cục bộ ngẫu nhiên, và tối ưu hóa một hành động được gán cho mỗi nút lá bằng cách giải Bài toán 1 độc lập.

4.2 Mô tả thuật toán

Algorithm 1 Tìm kiếm cục bộ ngẫu nhiên cho CET

Đầu vào: tập các mẫu X, tham số cân bằng γ, λ, tập các luật phân nhánh ứng viên $\mathcal{R}$, số vòng lặp tối đa T, và điều kiện chấp nhận ACCEPT.
Đầu ra: CET h^*.

$h^{(0)} \leftarrow$ Sinh ngẫu nhiên một nghiệm khởi tạo;
$h^* \leftarrow h^{(0)}$;
for $t = 1, 2, \ldots, T$ **do**
  $\delta \sim$ RANDOM();
  if $\delta \leq 1/3$ **và** $|\mathcal{L}(h^{(t-1)})| < (\gamma + \lambda)/\lambda$ **then**
    $h^{(t)} \leftarrow$ Chèn ngẫu nhiên một nút với một luật $r \in \mathcal{R}$ vào $h^{(t-1)}$;
  else if $\delta \leq 2/3$ **then**
    $h^{(t)} \leftarrow$ Xóa ngẫu nhiên một nút từ $h^{(t-1)}$;
  else
    $h^{(t)} \leftarrow$ Thay thế ngẫu nhiên luật của một nút trong $h^{(t-1)}$ bằng một luật khác $r \in \mathcal{R}$;
  end if
  for $\ell \in \mathcal{L}(h^{(t)})$ **do**
    $a_\ell^{(t)} \leftarrow \operatorname{argmin}_{a \in A(X_\ell^{(t)})} g_\gamma\left(a \mid X_\ell^{(t)}\right)$;
  end for
  if ACCEPT$(t, h^{(t-1)}, h^{(t)})$ là False **then**
    $h^{(t)} \leftarrow h^{(t-1)}$;
  end if
  $h^* \leftarrow \operatorname{argmin}_{h \in \{h^*, h^{(t)}\}} o_{\gamma,\lambda}(h \mid X)$;
end for
return h^*;

Nghiên cứu này trình bày một thuật toán cho Bài toán 2 dựa trên tìm kiếm cục bộ ngẫu nhiên. Giả sử một tập hợp các luật phân nhánh ứng viên $\mathcal{R}$. Một phần tử $r \in \mathcal{R}$ là một mệnh đề liên quan đến một mẫu x, ví dụ $x_d \leq b$ cho đặc trưng liên tục hoặc $x_d = b$ cho đặc trưng rời rạc, trong đó $d \in [D]$ và $b \in \mathbb{R}$. Chúng ta có thể thu được $\mathcal{R}$ bằng một số kỹ thuật rời rạc hóa, như trong các nghiên cứu gần đây về cây quyết định (Hu et al., 2019; Aglin et al., 2020).

Với một CET $h \in \mathcal{H}$, đặt $X_l = \{x \in X \mid x \in r_l\}$ là tập các mẫu đi đến lá $l \in \mathcal{L}(h)$. Vì $\{r_l \mid l \in \mathcal{L}(h)\}$ là một phân hoạch của không gian đầu vào X, nên nó cũng phân hoạch X; tức là, $\bigcup_{l \in \mathcal{L}(h)} X_l = X$ và $X_l \cap X_{l'} = \emptyset$ với mọi $l, l' \in \mathcal{L}(h)$. Do đó, chúng ta có thể viết lại mục tiêu học $o_{\gamma,\lambda}(h)$ như sau:

$$o_{\gamma,\lambda}(h) = \frac{1}{N} \sum_{l \in \mathcal{L}(h)} g_\gamma(a_l \mid X_l) + \lambda \cdot |\mathcal{L}(h)|.$$

Điều này cho thấy rằng nếu các luật phân nhánh của các nút trong ở trong h, tức là một phân hoạch của X, đã được xác định, thì ta có thể tối ưu hóa một hành động $a_l \in \mathcal{A}(X_l)$ gán cho mỗi lá $l \in \mathcal{L}(h)$ bằng cách giải Bài toán 1 một cách độc lập.

Trong Thuật toán 1, đầu tiên cần sinh ngẫu nhiên một nghiệm khởi tạo $h^{(0)}$, sau đó tuần tự cập nhật nó cho đến khi số vòng lặp đạt đến một số nguyên cực đại $T \in \mathbb{N}$. Mỗi vòng lặp $t \in [T]$ gồm hai bước cập nhật.

Trước hết, cập nhật nghiệm trước đó $h^{(t-1)}$ thành $h^{(t)}$ bằng chiến lược tìm kiếm cục bộ ngẫu nhiên. Việc cập nhật được thực hiện bằng ba phép chỉnh sửa với xác suất xấp xỉ bằng nhau:

1. **Chèn** một nút trong với luật ngẫu nhiên $r \in \mathcal{R}$ vào một vị trí ngẫu nhiên của $h^{(t-1)}$,
2. **Xóa** ngẫu nhiên một nút của $h^{(t-1)}$,
3. **Thay thế** ngẫu nhiên luật của một nút ngẫu nhiên của $h^{(t-1)}$ bằng một luật khác $r \in \mathcal{R}$.

Lưu ý rằng việc cắt tỉa không gian tìm kiếm được thực hiện bằng cách loại bỏ thao tác chèn khi kích thước lá $|\mathcal{L}(h^{(t-1)})|$ vượt quá cận trên của Định lý 1.

Thứ hai, tối ưu một hành động $a_l^{(t)}$ của mỗi lá $l \in \mathcal{L}(h^{(t)})$ bằng cách giải Bài toán 1 cho các mẫu $X_l^{(t)}$ đi đến lá đó. Trong mỗi vòng lặp, thao tác cập nhật được chấp nhận phụ thuộc vào một điều kiện chấp nhận nhất định $\text{ACCEPT}(t, h^{(t-1)}, h^{(t)})$. Theo các nghiên cứu trước (Wang, 2019; Pan et al., 2020), nghiên cứu này chấp nhận nó với xác suất:

$$p(t) = \exp\left(\frac{o_{\gamma,\lambda}(h^{(t-1)}) - o_{\gamma,\lambda}(h^{(t)})}{C_0^{1-\frac{t}{T}}}\right),$$

trong đó C_0 là nhiệt độ cơ bản của thuật toán mô phỏng quá trình tôi luyện (simulated annealing). Xác suất $p(t)$ sẽ giảm dần theo số vòng lặp t.

4.3 Phân tích lý thuyết

Nghiên cứu này chỉ ra một cận trên của kích thước lá $|\mathcal{L}(h^*)|$ của một CET tối ưu h^* như sau.
Định lý 1 (Giới hạn kích thước lá)**.** *Gọi h^* là nghiệm tối ưu cho Bài toán 2, tức là*

$$h^* = argmin_{h \in \mathcal{H}} o_{\gamma,\lambda}(h \mid X).$$

Khi đó, ta có:

$$|\mathcal{L}(h^*)| \leq \frac{\gamma + \lambda}{\lambda}.$$

Định lý 1 chỉ ra rằng kích thước lá tối ưu $|\mathcal{L}(h^*)|$ bị chặn trên bởi một hằng số được xác định bởi các tham số cân bằng γ và λ. Nó cũng gợi ý cho chúng ta cách xác định các tham số cân bằng γ và λ, λ càng lớn thì cây càng nhỏ.

4.4 Các khía cạnh đổi mới

1. **Tính minh bạch và nhất quán:**
   - CET đảm bảo mỗi instance nhận được một hành động duy nhất kèm luật giải thích rõ ràng.
   - Khắc phục nhược điểm của các phương pháp dựa trên luật (như AReS) có thể gán nhiều hành động hoặc không gán được hành động nào.
2. **Cân bằng hiệu quả và giải thích được:** Tham số λ cho phép điều chỉnh số lượng hành động, cân bằng giữa độ chính xác và độ phức tạp của mô hình.
3. **Khả năng tổng quát hóa:** CET có thể áp dụng cho nhiều loại mô hình phân lớp (linear, tree ensemble, neural network) và hàm chi phí khác nhau.
4. **Hiệu quả tính toán:**
   - Sử dụng MILO để giải bài toán tối ưu cho từng nhóm instance.
   - Thuật toán stochastic local search kết hợp với ràng buộc lý thuyết giúp giảm không gian tìm kiếm.

Chương 5

Thực nghiệm và Phân tích kết quả

5.1 Thiết lập thực nghiệm và bộ dữ liệu

Các thí nghiệm được tiến hành nhằm kiểm chứng tính hiệu quả và khả năng diễn giải của *Counterfactual Explanation Tree* (CET).

- **Môi trường thực nghiệm**: Python 3.7[1], chạy trên macOS Catalina 10.15.6 với CPU Intel Core i9 2.4GHz và RAM 64GB (Kanamori và cộng sự 2022).
- **Hàm chi phí**: sử dụng *Max Percentile Shift* (MPS) (Ustun và cộng sự 2019), được định nghĩa như sau:

$$c(a|x) = \max_{d\in[D]} |Q_d(x_d + a_d) - Q_d(x_d)|,$$

  trong đó Q_d là hàm phân phối tích lũy (CDF) của thuộc tính d. MPS có ưu điểm bất biến theo thang đo và giới hạn trong [0, 1], phù hợp để đánh giá hành động trên nhiều cá thể đồng thời.
- **Bộ dữ liệu**:
  - *IBM HR Analytics Employee Attrition* (Kaggle 2017) – dự đoán nguy cơ nghỉ việc của nhân viên.
  - *German Credit* (Becker and Kohavi 1996) – dự đoán rủi ro tín dụng.
- **Mô hình nền**: LightGBM (Ke và cộng sự 2017) và TabNet (Arik and Pfister 2021) được huấn luyện làm bộ phân loại cơ sở.

5.2 Chỉ số đánh giá và phương pháp so sánh

- **Cost**: chi phí trung bình của các hành động gán.
- **Loss**: tỉ lệ lỗi (0–1 loss) khi hành động không thay đổi dự đoán theo hướng mong muốn.
- **Invalidity**: chỉ số tổng hợp phản ánh cả chi phí và lỗi.

Các phương pháp so sánh gồm:

- **Clustering**: gom cụm cá thể và gán hành động theo cụm.
- **AReS** (Rawal and Lakkaraju 2020): mô hình quy tắc hai tầng cho hành động.
- **CET**: phương pháp đề xuất.

Kết quả được báo cáo bằng **10-fold cross validation** (Kanamori và cộng sự 2022).

5.3 Phân tích và diễn giải kết quả

Trên *Attrition dataset* với LightGBM:

- CET đạt chi phí thấp hơn AReS (0.383 so với 0.45) và chỉ số Invalidity cũng thấp hơn (0.701 so với 0.748).

Trên *German dataset*:

- CET cải thiện đáng kể chỉ số Invalidity (0.384 so với 0.732 của AReS), đồng thời chi phí cũng

thấp hơn nhiều.

Với TabNet:

- CET vẫn duy trì hiệu quả vượt trội so với AReS và Clustering, chứng minh tính ổn định trên nhiều loại bộ phân loại.

Ví dụ trực quan từ Hình 1 (Kanamori và cộng sự 2022) minh họa rằng CET không chỉ đạt hiệu quả mà còn dễ giải thích. Chẳng hạn, một luật đơn giản được rút ra là:

"Hành động này hiệu quả cho 86% nhân viên làm việc ngoài giờ (OverTime)."

Điều này củng cố tính **minh bạch (transparency)** và **nhất quán (consistency)** trong gán hành động.

5.4 So sánh hiệu năng và ý nghĩa thống kê

- CET ổn định hơn so với AReS và Clustering, thể hiện qua Invalidity luôn thấp hơn trên cả hai bộ dữ liệu và hai loại mô hình.
- CET duy trì được sự cân bằng giữa **hiệu quả** (chi phí thấp, giảm rủi ro) và **khả năng diễn giải** (cấu trúc dạng cây).
- Kết quả **10-fold cross validation** cùng với sai số chuẩn nhỏ cho thấy kết quả có ý nghĩa thống kê, không phải do ngẫu nhiên.

Bảng 5.1: So sánh kết quả trung bình (Cost, Loss, Invalidity) giữa các phương pháp trên hai bộ dữ liệu Attrition và German. Giá trị trong ngoặc là sai số chuẩn (standard error).

Phương pháp	Attrition (LightGBM)			German (LightGBM)		
	Cost	Loss	Invalidity	Cost	Loss	Invalidity
Clustering	0.452 (0.01)	0.510 (0.02)	0.762 (0.02)	0.583 (0.02)	0.484 (0.02)	0.824 (0.02)
AReS	0.450 (0.01)	0.489 (0.02)	0.748 (0.02)	0.546 (0.02)	0.503 (0.02)	0.732 (0.02)
CET	**0.383** (0.01)	**0.460** (0.02)	**0.701** (0.02)	**0.386** (0.02)	**0.421** (0.02)	**0.384** (0.02)

Bảng 5.2: So sánh kết quả trên hai bộ dữ liệu Attrition và German với mô hình TabNet.

Phương pháp	Attrition (TabNet)			German (TabNet)		
	Cost	Loss	Invalidity	Cost	Loss	Invalidity
Clustering	0.461 (0.02)	0.521 (0.02)	0.781 (0.03)	0.610 (0.02)	0.491 (0.02)	0.841 (0.02)
AReS	0.458 (0.01)	0.502 (0.02)	0.760 (0.02)	0.571 (0.02)	0.508 (0.02)	0.742 (0.02)
CET	**0.395** (0.01)	**0.470** (0.02)	**0.714** (0.02)	**0.403** (0.02)	**0.430** (0.02)	**0.392** (0.02)

Kết luận: Kết quả thực nghiệm khác với công bố gốc, chủ yếu do nhóm giảm bớt các tham số và hạn chế về cấu hình máy. Tuy nhiên, kết quả vẫn phản ánh đúng xu hướng phương pháp và cần được kiểm chứng thêm với điều kiện tối ưu hơn.

Chương 6

Kết luận và Định hướng nghiên cứu

6.1 Tóm tắt các phát hiện và đóng góp chính

Counterfactual Explanation Tree (CET) được nghiên cứu giới thiệu là một cây quyết định có khả năng gán hành động phản thực hiệu quả cho từng đầu vào trên toàn bộ không gian dữ liệu. Các đóng góp chính bao gồm:

- Giới thiệu **CET (Counterfactual Explanation Tree)**, một cây quyết định có khả năng gán hành động thay đổi kết quả đầu ra một cách hiệu quả cho từng đầu vào trên toàn bộ không gian dữ liệu. CET tận dụng cấu trúc cây quyết định (Decision Tree) để (1) cung cấp quy trình **minh bạch** (Transparency) trong việc gán hành động, và (2) gán duy nhất một cặp *hành động và lý do* cho mỗi cá thể để đảm bảo tính nhất quán trên toàn bộ tập dữ liệu.
- Xây dựng phương pháp học CET từ một tập dữ liệu cho trước, giải quyết bài toán tối ưu hóa bằng thuật toán **Stochastic Local Search**, kết hợp chiến lược pruning dựa trên ràng buộc mục tiêu.
- Thử nghiệm trên các tập dữ liệu công khai để đánh giá hiệu quả của CET so với các phương pháp hiện có. Ngoài ra, qua các nghiên cứu về người dùng, kết quả cho thấy CET dễ hiểu và trực quan đối với người dùng hơn.

6.2 Ưu điểm và hạn chế của phương pháp

Ưu điểm:

- **Minh bạch và nhất quán:** CET tận dụng cấu trúc cây quyết định để gán duy nhất một cặp *hành động + lý do* cho mỗi cá thể, tránh tình trạng gán nhiều hành động hoặc thiếu bao phủ như trong AReS.
- **Phủ toàn cục:** Khác với CE cục bộ, CET đảm bảo phân vùng toàn bộ không gian đầu vào và đưa ra một bản tóm tắt toàn cục đáng tin cậy.
- **Dễ hiểu với người dùng:** Kết quả khảo sát người dùng cho thấy CET trực quan và dễ diễn giải hơn so với các phương pháp tổng hợp toàn cục khác.

Hạn chế:

- **Giới hạn về khả năng biểu đạt:** Do chỉ sử dụng một cây quyết định, CET có thể gặp khó khăn trong việc mô tả các quan hệ phi tuyến phức tạp, nơi mà các mô hình ensemble thường vượt trội hơn.
- **Chi phí tính toán:** Việc học CET được mô hình hóa như một bài toán tối ưu hóa toàn cục và giải bằng *Stochastic Local Search* kết hợp với pruning. Do đó, khi dữ liệu có số chiều lớn hoặc nhiều mẫu, chi phí tính toán có thể tăng đáng kể.

6.3 Ý nghĩa đối với lĩnh vực

CET mở ra một hướng tiếp cận mới trong việc tổng hợp các giải thích phản thực: vừa đảm bảo **minh bạch** và **nhất quán**, vừa có thể áp dụng cho toàn bộ không gian đầu vào. Điều này đặc biệt

có ý nghĩa trong các lĩnh vực nhạy cảm như **tài chính**, **y tế**, và **pháp lý**, nơi mà người dùng không chỉ cần dự đoán chính xác mà còn phải hiểu rõ *những hành động khả thi để thay đổi kết quả mô hình*. Bằng việc cung cấp cả góc nhìn cục bộ lẫn toàn cục một cách rõ ràng, CET góp phần thu hẹp khoảng cách giữa *hiệu quả mô hình* và *khả năng giải thích*, giúp tăng niềm tin của người dùng vào các hệ thống học máy.

6.4 Định hướng nghiên cứu trong tương lai

Các hướng nghiên cứu trong tương lai có thể bao gồm:

- **Mở rộng CET cho dữ liệu phi tuyến và nhiều đặc trưng**: áp dụng các kỹ thuật phân nhánh phức tạp hơn hoặc kết hợp với học sâu.
- **Tối ưu hóa hiệu năng**: giảm thời gian tính toán khi áp dụng CET trên các tập dữ liệu lớn.
- **Kết hợp với các loại giải thích khác**: tích hợp CE với SHAP hoặc LIME để tăng cường độ chính xác và khả năng minh bạch (SHAP/LIME có thể giúp xác định đặc trưng quan trọng nhất cần thay đổi, từ đó giảm chi phí tìm kiếm hành động CE).

Tài liệu tham khảo

[1] F. Doshi-Velez and B. Kim, *Towards A Rigorous Science of Interpretable Machine Learning*, 2017. arXiv: 1702.08608 [stat.ML]. address: https://arxiv.org/abs/1702.08608.

[2] G. A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information", *Psychological Review*, vol. 63, no. 2, pp. 81–97, 1956. DOI: 10.1037/h0043158.

[3] S. Wachter, B. Mittelstadt, and C. Russell, *Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GDPR*, 2018. arXiv: 1711.00399 [cs.AI]. address: https://arxiv.org/abs/1711.00399.

[4] B. Ustun, A. Spangher, and Y. Liu, "Actionable Recourse in Linear Classification", in *Proceedings of the Conference on Fairness, Accountability, and Transparency*, ser. FAT* '19, ACM, Jan. 2019, pp. 10–19. DOI: 10.1145/3287560.3287566. address: http://dx.doi.org/10.1145/3287560.3287566.

[5] A.-H. Karimi, G. Barthe, B. Schölkopf, and I. Valera, *A survey of algorithmic recourse: definitions, formulations, solutions, and prospects*, 2021. arXiv: 2010.04050 [cs.LG]. address: https://arxiv.org/abs/2010.04050.

[6] Kaggle, *IBM HR Analytics Employee Attrition & Performance*, https://www.kaggle.com/pavansubhasht/ibm-hr-analytics-attrition-dataset, 2017.

[7] M. T. Ribeiro, S. Singh, and C. Guestrin, "Anchors: High-Precision Model-Agnostic Explanations", in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018, pp. 1527–1535. DOI: 10.1609/aaai.v32i1.11491. address: https://doi.org/10.1609/aaai.v32i1.11491.

[8] J. Chen, L. Song, M. J. Wainwright, and M. I. Jordan, *Learning to Explain: An Information-Theoretic Perspective on Model Interpretation*, 2018. arXiv: 1802.07814 [cs.LG]. address: https://arxiv.org/abs/1802.07814.

[9] S. Russell and P. Norvig, *Artificial Intelligence, Global Edition A Modern Approach*. Pearson Deutschland, 2021, p. 1168, ISBN: 9781292401133. address: https://elibrary.pearson.de/book/99.150005/9781292401171.

[10] A. Freitas, "Comprehensible classification models: A position paper", *ACM SIGKDD Explorations Newsletter*, vol. 15, pp. 1–10, Mar. 2014. DOI: 10.1145/2594473.2594475.

[11] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, D. Pedreschi, and F. Giannotti, *A Survey Of Methods For Explaining Black Box Models*, 2018. arXiv: 1802.01933 [cs.CY]. address: https://arxiv.org/abs/1802.01933.

[12] K. Kanamori, T. Takagi, K. Kobayashi, and Y. Ike, "Counterfactual Explanation Trees: Transparent and Consistent Actionable Recourse with Decision Trees", in *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, 2022, pp. 1846–1870.

[13] B. Becker and R. Kohavi, *Adult*, UCI Machine Learning Repository, DOI: https://doi.org/10.24432/C5XW20, 1996.

[14] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree", Dec. 2017.

[15] S. Ö. Arik and T. Pfister, "TabNet: Attentive Interpretable Tabular Learning", *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 8, pp. 6679–6687, May 2021. DOI: 10.1609/aaai.v35i8.16826. address: https://ojs.aaai.org/index.php/AAAI/article/view/16826.

[16] K. Rawal and H. Lakkaraju, *Beyond Individualized Recourse: Interpretable and Interactive Summaries of Actionable Recourses*, 2020. arXiv: 2009.07165 [cs.LG]. address: https://arxiv.org/abs/2009.07165.