NHẬP MÔN TRÍ TUỆ NHÂN TẠO

HỌC MÁY

ThS. Vũ Hoài Thư

Nội dung

Giới thiệu

Tài liệu tham khảo

- N. Nilsson. Introduction to machine learning: http://ai.stanford.edu/people/nilsson/mlbook.html
- T. Mitchell. Machine learning. McGraw-Hill, 1997.
- E. Alpaydin. Introduction to machine learning. MIT Press, 2004.
- M. Mohri, A. Rostamizadeh, A. Talwalkar. Foundations of Machine Learning. MIT Press, 2012
- Vũ Hữu Tiệp. Machine Learning cơ bản https://machinelearningcoban.com/about/

Công cụ và dữ liệu

- Bộ công cụ Weka: http://www.cs.waikato.ac.nz/ ml/weka
- Kho dữ liệu mẫu UC Irvine: http://www.ics.uci.edu/ mlearn/ML/Repository.html

Một số ứng dụng của học máy (1/3)

- Những ứng dụng khó lập trình theo cách thông thường do không tồn tại hoặc khó giải thích kinh nghiệm, kỹ năng của con người
  - Nhận dạng chữ viết, âm thanh, hình ảnh
  - Lái xe tự động, thám hiểm sao Hoả
- Chương trình máy tính có khả năng thích nghi: lời giải thay đổi theo thời gian hoặc theo tình huống cụ thể
  - Chương trình trợ giúp cá nhân
  - Định tuyến mạng

Một số ứng dụng của học máy (2/3)

- Khai phá (phân tích) dữ liệu
  - Hồ sơ bệnh án $\rightarrow$ tri thức y học
  - Dữ liệu bán hàng $\rightarrow$ Quy luật kinh doanh

Một số ứng dụng của học máy (3/3)

Hầu hết các ứng dụng trí tuệ nhân tạo ngày nay có sử dụng học máy

Hình: Trí tuệ nhân tạo giải mã văn tự trên 2000 năm tuổi

Học máy là gì?

- Học máy (Machine Learning–ML) là một tập con của trí tuệ nhân tạo.
- Nó là một lĩnh vực nhỏ trong khoa học máy tính, có khả năng tự học hỏi dựa trên dữ liệu được đưa vào mà không cần phải được lập trình cụ thể
- Học máy là khả năng của chương trình máy tính sử dụng kinh nghiệm, quan sát, hoặc dữ liệu trong quá khứ để cải thiện các công việc trong tương lai thay vì chỉ thực hiện theo đúng các quy tắc đã được lập trình sẵn.

Ví dụ

- Học nhận dạng chữ:
  - Task (T): Nhận dạng chữ cái từ hình ảnh
  - Performance (P): Phần trăm chữ nhận dạng đúng
  - Experience (E): Ảnh số của chữ và chữ tương ứng
- Dịch máy:
  - Task (T): Dịch một câu tiếng Anh sang tiếng Việt
  - Performance (P): Độ đo dịch máy (ví dụ số câu đúng, số mệnh đề đúng,..)
  - Experience (E): Cặp câu tiếng Anh và tiếng Việt tương ứng

Vấn đề cần quan tâm (1/3)

- Kinh nghiệm hoặc dữ liệu cho học máy được cho dưới dạng nào?
- Lựa chọn biểu diễn cho hàm đích ra sao?

Vấn đề cần quan tâm (2/3)

Việc sử dụng những dạng kinh nghiệm và dạng biểu diễn khác nhau dẫn tới những dạng học máy khác nhau:

- Học có giám sát (supervised learning)
- Học không giám sát (un-supervised learning)
- Học bán giám sát (semi-supervised learning)
- Học tăng cường (reinforcement learning)

Vấn đề cần quan tâm (3/3)

Lựa chọn biểu diễn cho hàm đích ra sao?

- Sử dụng hàm: $y = w_1x_1 + w_2x_2 + \ldots + w_nx_n$
- Sử dụng các luật
- Sử dụng mạng nơron
- Sử dụng cây quyết định
- Sử dụng các mô hình xác suất

Một số khái niệm

- Mẫu hay ví dụ (samples): là đối tượng cần xử lý (ví dụ phân loại)
  - Ví dụ: Khi lọc thư rác thì mỗi thư là một mẫu
- Mẫu thường được mô tả bằng tập thuộc tính hay đặc trưng (features)
  - Ví dụ: trong chuẩn đoán bệnh, thuộc tính là triệu chứng của người bệnh, và các tham số khác như chiều cao, cân nặng,...
- Nhãn phân loại (label): Thể hiện loại của đối tượng mà ta cần dự đoán
  - Ví dụ: nhãn phân loại thư rác có thể là "rác" hoặc "bình thường"

Ví dụ

thuộc tính

nhãn

mẫu

Ngày	Trời	Nhiệt độ	Độ ẩm	Gió	Chơi tennis
D1	nắng	nóng	cao	yếu	không
D2	nắng	nóng	cao	mạnh	không
D3	u ám	nóng	cao	yếu	có
D4	mưa	trung bình	cao	yếu	có
D5	mưa	lạnh	bình thường	yếu	có
D6	mưa	lạnh	bình thường	mạnh	không
D7	u ám	lạnh	bình thường	mạnh	có
D8	nắng	trung bình	cao	yếu	không
D9	nắng	lạnh	bình thường	yếu	có
D10	mưa	trung bình	bình thường	yếu	có
D11	nắng	trung bình	bình thường	mạnh	có
D12	u ám	trung bình	cao	mạnh	có
D13	u ám	nóng	bình thường	yếu	có
D14	mưa	trung bình	cao	mạnh	không

Một số dạng học máy phổ biến

- Học có giám sát (supervised learning): Là dạng học máy trong đó cho trước tập dữ liệu huấn luyện dưới dạng các ví dụ cùng với giá trị đầu ra hay giá trị đích.
  - Phân loại (classification)
  - Hồi quy (regression)
- Học không giám sát (un-supervised learning): Là dạng học máy trong đó các ví dụ được cung cấp nhưng không có giá trị đầu ra hay giá trị đích.
  - Học luật kết hợp (association)
  - Phân cụm (clustering)
- Học bán giám sát (semi-supervised learning): Là dạng học máy trong đó chỉ một phần tập dữ liệu huấn luyện dưới dạng các ví dụ được cho cùng với giá trị đầu ra hay giá trị đích.
- Học tăng cường (reinforcement learning)

Phân loại (Classification)

Giá trị đích là các giá trị rời rạc

Dung tích động cơ	Loại xe	Phân khúc
3200	Sedan	Cao cấp
2500	Sedan	Cao cấp
2500	SUV	Trung bình
2000	Sedan	Trung bình
3500	SUV	Cao cấp
1800	Sedan	Trung bình

Hồi quy (Regression)

Giá trị đích là các giá trị liên tục

Dung tích động cơ	Tuổi của xe	Giá bán (triệu đồng)
3200	1	2500
2500	2	1600
2500	4	1300
2000	5	600
1800	1	915
1800	3	725

Ứng dụng: Dự đoán giá xe, giá vàng, chứng khoán,...

Học luật kết hợp (Association)

- Ví dụ: Phân tích giao dịch, mua bán (hoá đơn mua hàng)
- P(Y|X): Xác suất người mua hàng X còn mua hàng Y
- Ví dụ luật kết hợp
  - Người mua bánh mì thường mua bơ
  - Người mua lạc rang thường mua bia

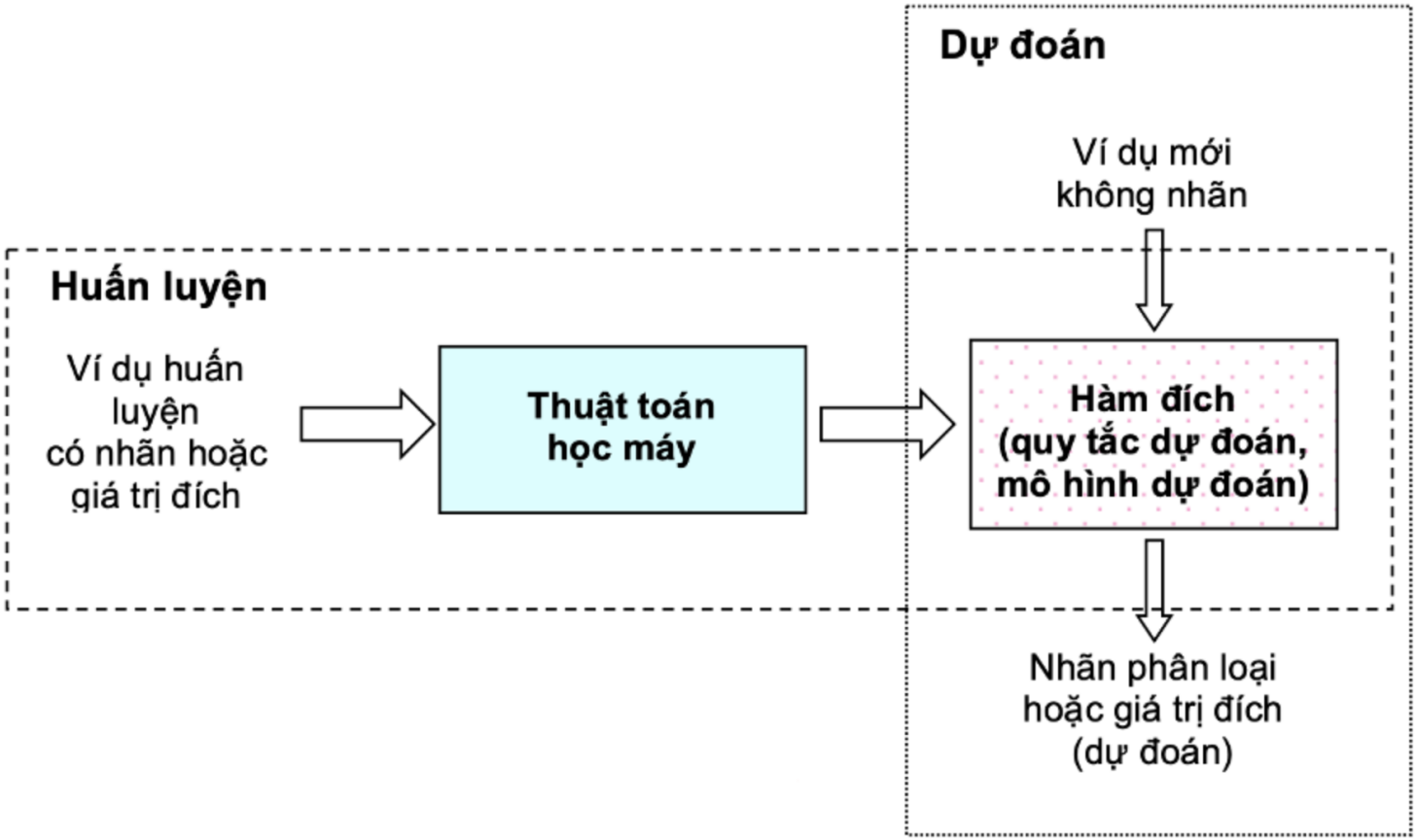
Phân cụm (Clustering)

- Nhóm những trường hợp tương tự với nhau
- Không có giá trị đầu ra
- Ứng dụng:
  - Phân cụm khách hàng, phân cụm sinh viên
  - Phân đoạn ảnh
  - Thiết kế vi mạch

Học tăng cường (Reinforcement learning)

- Kinh nghiệm không được cho trực tiếp dưới dạng đầu vào / đầu ra
- Hệ thống nhận được một giá trị thưởng (reward) là kết quả cho một chuỗi hành động nào đó
- Thuật toán cần học cách hành động để cực đại hóa giá trị thưởng

Hệ thống học máy điển hình



Học cây quyết định

Giới thiệu

- Học cây quyết định (Decision Tree) được sử dụng để học một hàm mục tiêu có giá trị rời rạc
- Hàm phân lớp được biểu diễn bởi một cây quyết định
- Một cây quyết định có thể biểu diễn (diễn giải) bằng 1 tập các luật IF-THEN
- Học cây quyết định có thể thực hiện ngay cả với các dữ liệu có chứa nhiễu/lỗi
- Là một trong các phương pháp học quy nạp (inductive learning) được dùng phổ biến nhất
- Được áp dụng thành công trong rất nhiều các bài toán ứng dụng thực tế

Dữ liệu huấn luyện

thuộc tính

nhãn

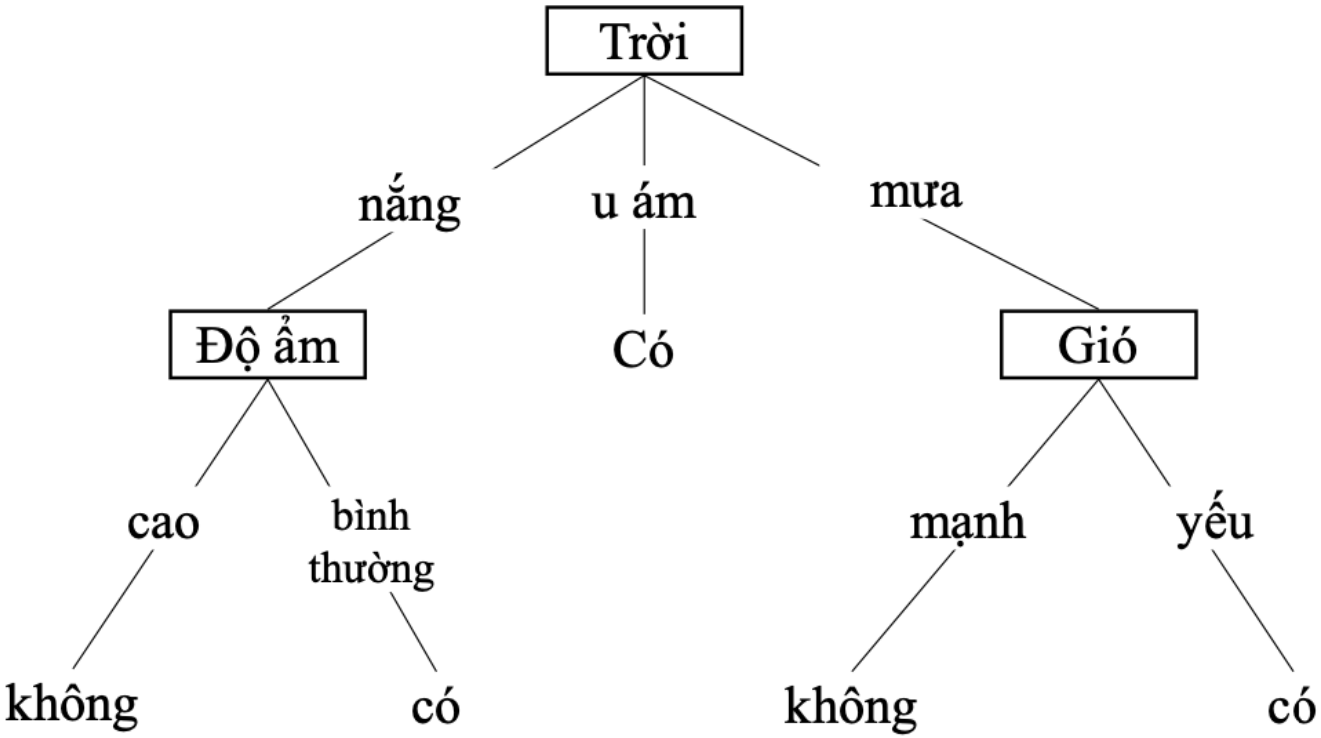
mẫu

Ngày	Trời	Nhiệt độ	Độ ẩm	Gió	Chơi tennis
D1	nắng	nóng	cao	yếu	không
D2	nắng	nóng	cao	mạnh	không
D3	u ám	nóng	cao	yếu	có
D4	mưa	trung bình	cao	yếu	có
D5	mưa	lạnh	bình thường	yếu	có
D6	mưa	lạnh	bình thường	mạnh	không
D7	u ám	lạnh	bình thường	mạnh	có
D8	nắng	trung bình	cao	yếu	không
D9	nắng	lạnh	bình thường	yếu	có
D10	mưa	trung bình	bình thường	yếu	có
D11	nắng	trung bình	bình thường	mạnh	có
D12	u ám	trung bình	cao	mạnh	có
D13	u ám	nóng	bình thường	yếu	có
D14	mưa	trung bình	cao	mạnh	không

Dữ liệu

- n mẫu huấn luyện, mỗi mẫu là một cặp $< \mathbf{x}, y >$
  - $\mathbf{x}$ là vector thuộc tính
  - y là nhãn phân loại, $y \in C$ (tập các nhãn)
- Ví dụ mẫu D4
  - $\mathbf{x} =$ (mưa, trung bình, cao, yếu)
  - $y =$ có

Ví dụ cây quyết định



Biểu diễn cây quyết định

- Mỗi nút trong (internal node) biểu diễn một thuộc tính cần kiểm tra giá trị đối với các ví dụ
- Mỗi nhánh (branch) từ một nút sẽ tương ứng với một giá trị có thể của thuộc tính gắn với nút đó
- Mỗi nút lá (leaf node) biểu diễn một phân lớp.
- Một cây quyết định học được sẽ được phân lớp đối với một ví dụ, bằng cách duyệt từ nút gốc tới nút lá

  $\rightarrow$ Nhãn lớp gắn với nút lá đó sẽ được gán cho ví dụ cần phân lớp.

Biểu diễn dưới dạng quy tắc

- Cây quyết định có thể biểu diễn tương đương dưới dạng các quy tắc logic
- Mỗi cây là tuyển của các quy tắc, mỗi quy tắc bao gồm các phép hội
- Ví dụ:

(Trời = nắng ∧ Độ ẩm = bình_thường)
∨ (Trời = u_ám)
∨ (Trời = mưa ∧ Gió = yếu)

Học cây quyết định

- Cây quyết định được học (xây dựng) từ dữ liệu huấn luyện
- Với mỗi bộ dữ liệu có thể xây dựng nhiều cây quyết định
  - Chọn cây nào?

Quá trình học là quá trình tìm kiếm cây quyết định phù hợp với dữ liệu huấn luyện

- Cho phép phân loại đúng dữ liệu huấn luyện

Thuật toán ID3 - Ý tưởng

- Thực hiện giải thuật tìm kiếm tham lam (greedy search) đối với không gian các cây quyết định có thể.
- Xây dựng (học) một cây quyết định theo chiến lược top-down, bắt đầu từ nút gốc.
- Ở mỗi nút, thuộc tính kiểm tra là thuộc tính có khả năng phân loại tốt nhất đối với các ví dụ học gắn với nút đó.
- Tạo mới một cây con (sub-tree) của nút hiện tại cho mỗi giá trị có thể của thuộc tính kiểm tra, và tập học sẽ được tách ra (thành các tập con) tương ứng với cây con vừa tạo
- Mỗi thuộc tính chỉ được phép xuất hiện tối đa 1 lần đối với bất kì một đường đi nào trong cây

Thuật toán ID3

- Xây dựng lần lượt các nút của cây bắt đầu từ gốc
- Thuật toán
  - **Khởi đầu:** nút hiện thời là nút gốc chứa toàn bộ tập dữ liệu huấn luyện
  - Tại nút hiện thời n, lựa chọn thuộc tính:
    - Chưa được sử dụng ở nút tổ tiên
    - Cho phép phân chia tập dữ liệu hiện thời thành các tập con một cách tốt nhất
    - Với mỗi giá trị thuộc tính được chọn thêm một nút con bên dưới
    - Chia các ví dụ ở nút hiện thời về các nút con theo giá trị thuộc tính được chọn
  - **Lặp** (đệ quy) cho tới khi:
    - Tất cả các thuộc tính đã được sử dụng ở các nút phía trên, hoặc
    - Tất cả ví dụ tại nút hiện thời có cùng nhãn phân loại
    - Nhãn của nút được lấy theo đa số nhãn của ví dụ tại nút hiện thời

Vấn đề: Lựa chọn thuộc tính tại mỗi nút như thế nào?

Tiêu chuẩn chọn thuộc tính của ID3

- Tại mỗi nút n
  - Tập (con) dữ liệu ứng với nút đó
  - Cần lựa chọn thuộc tính cho phép phân chia tập dữ liệu tốt nhất
- Tiêu chuẩn:
  - Dữ liệu sau khi phân chia càng đồng nhất càng tốt
  - Đo bằng độ tăng thông tin (Information Gain - IG)
  - Chọn thuộc tính có độ tăng thông tin lớn nhất
  - IG dựa trên entropy của tập (con) dữ liệu

Entropy

- Entropy là một đại lượng được sử dụng trong lĩnh vực Lý thuyết thông tin
- Được sử dụng để đánh giá mức độ hỗn tạp của một tập dữ liệu
- Trường hợp dữ liệu S có 2 loại nhãn:

  $Entropy(S) = -p_1 log_2 p_1 - p_2 log_2 p_2$

  Trong đó: p_1: tỉ lệ các mẫu trong tập S thuộc vào lớp 1, p_2: tỉ lệ các mẫu trong tập S thuộc vào lớp 2

Entropy

- Trường hợp tổng quát: Khi tập dữ liệu S có C loại nhãn (C phân lớp):

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

Trong đó p_i là tỉ lệ các ví dụ trong tập S thuộc vào lớp i

Entropy - Ví dụ

- S gồm 14 ví dụ, trong đó có 9 ví dụ thuộc lớp c_1 và 5 ví dụ thuộc lớp c_2
- Entropy của tập S với 2 lớp:

$$Entropy(S) = -\frac{9}{14}.log_2(\frac{9}{14}) - \frac{5}{14}.log_2(\frac{5}{14}) = 0.94$$

- Entropy=0, nếu tất cả các ví dụ cùng một lớp (c_1 hoặc c_2)
- Entropy=1, số lượng ví dụ thuộc về lớp c_1 bằng số lượng ví dụ thuộc về lớp c_2
- Entropy thuộc khoảng (0,1) nếu như số lượng các ví dụ thuộc về lớp c_1 khác với ví dụ thuộc về lớp c_2

Độ tăng thông tin - Information Gain

- Độ tăng thông tin – IG (Information Gain) của một thuộc tính đối với một tập các ví dụ:
  - Mức độ giảm về Entropy
  - Bởi việc phân chia các ví dụ theo các giá trị của thuộc tính đó
- Information Gain của thuộc tính A đối với tập S

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

Trong đó: $Values(A)$ là tập giá trị có thể của thuộc tính A, và $S_v = \{x | x \in S, x_A = v\}$

- Trong công thức trên, thành phần thứ 2 thể hiện giá trị Entropy sau khi tập S được phân chia bởi các giá trị thuộc tính A

Độ tăng thông tin - Ví dụ

- Hãy tính giá trị độ tăng thông tin của thuộc tính Gió đối với tập dữ liệu S

Ngày	Trời	Nhiệt độ	Độ ẩm	Gió	Chơi tennis
D1	nắng	nóng	cao	yếu	không
D2	nắng	nóng	cao	mạnh	không
D3	u ám	nóng	cao	yếu	có
D4	mưa	trung bình	cao	yếu	có
D5	mưa	lạnh	bình thường	yếu	có
D6	mưa	lạnh	bình thường	mạnh	không
D7	u ám	lạnh	bình thường	mạnh	có
D8	nắng	trung bình	cao	yếu	không
D9	nắng	lạnh	bình thường	yếu	có
D10	mưa	trung bình	bình thường	yếu	có
D11	nắng	trung bình	bình thường	mạnh	có
D12	u ám	trung bình	cao	mạnh	có
D13	u ám	nóng	bình thường	yếu	có
D14	mưa	trung bình	cao	mạnh	không

Độ tăng thông tin - Ví dụ

Hãy tính giá trị độ tăng thông tin của thuộc tính Gió đối với tập dữ liệu S

- Tập dữ liệu S có 2 phân lớp: có $(+)$, không (-)
- Thuộc tính Gió có 2 giá trị: yếu, mạnh
- $S = [9+, 5-]$
- $S_{\text{yếu}} = [6+, 2-]$
- $S_{\text{mạnh}} = [3+, 3-]$

$$IG(S, \text{Gió}) = Entropy(S) - \sum_{v \in \{\text{yếu, mạnh}\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$= Entropy(S) - \frac{8}{14}.Entropy(S_{\text{yếu}}) - \frac{6}{14}.Entropy(S_{\text{mạnh}})$$

$$= 0.048$$

Học cây quyết định - Ví dụ

- Tại nút gốc, thuộc tính nào trong số các thuộc tính {Trời, Nhiệt độ, Độ ẩm, Gió} nên được chọn là thuộc tính kiểm tra?
- Tính:
  - IG (S, Trời) = 0.248
  - IG (S, Nhiệt độ) = 0.029
  - IG (S, Độ ẩm) = 0.151
  - IG (S, Gió) = 0.048
- Thuộc tính Trời có giá trị IG cao nhất. Vì vậy, thuộc tính Trời được chọn làm thuộc tính kiểm tra cho nút gốc!

Học cây quyết định - Ví dụ

- Tại Node1, thuộc tính nào trong số các thuộc tính {Nhiệt độ, Độ ẩm, Gió} nên được chọn làm thuộc tính kiểm tra tiếp theo?
- Lưu ý: Thuộc tính Trời bị loại ra vì nó đã được sử dụng bởi cha của nút Node1 (là nút gốc).
- Tính:
  - IG ($S_{\text{nắng}}$, Nhiệt độ) $= 0.57$
  - IG ($S_{\text{nắng}}$, Độ ẩm) $= 0.97$
  - IG ($S_{\text{nắng}}$, Gió) $= 0.019$
- Vì vậy, thuộc tính Độ ẩm được chọn là thuộc tính kiểm tra cho nút Node1

Bài tập 1

Cho dữ liệu huấn luyện như trong bảng. Màu, Loại, Hãng là các thuộc tính, f là nhãn phân loại.

Màu	Loại	Hãng	f
Trắng	7 chỗ	Toyota	-
Đen	7 chỗ	Honda	+
Trắng	5 chỗ	Honda	-
Đen	5 chỗ	Toyota	+
Đỏ	7 chỗ	Honda	+
Đỏ	5 chỗ	Honda	-
Trắng	5 chỗ	Toyota	+

Hãy xác định nút gốc cho cây quyết định sử dụng thuật toán ID3. Trong trường hợp có nhiều thuộc tính có cùng mức độ ưu tiên thì chọn theo thứ tự từ trái sang phải (Màu, Loại, Hãng).

Các đặc điểm của ID3

- ID3 là thuật toán tìm kiếm cây quyết định phù hợp với dữ liệu huấn luyện
- Tìm kiếm theo kiểu tham lam, bắt đầu từ cây rỗng
- Hàm đánh giá là độ tăng thông tin
- ID3 có khuynh hướng (bias) lựa chọn cây đơn giản
  - Ít nút
  - Các thuộc tính có độ tăng thông tin lớn nằm gần gốc

Training error và Test error (1/2)

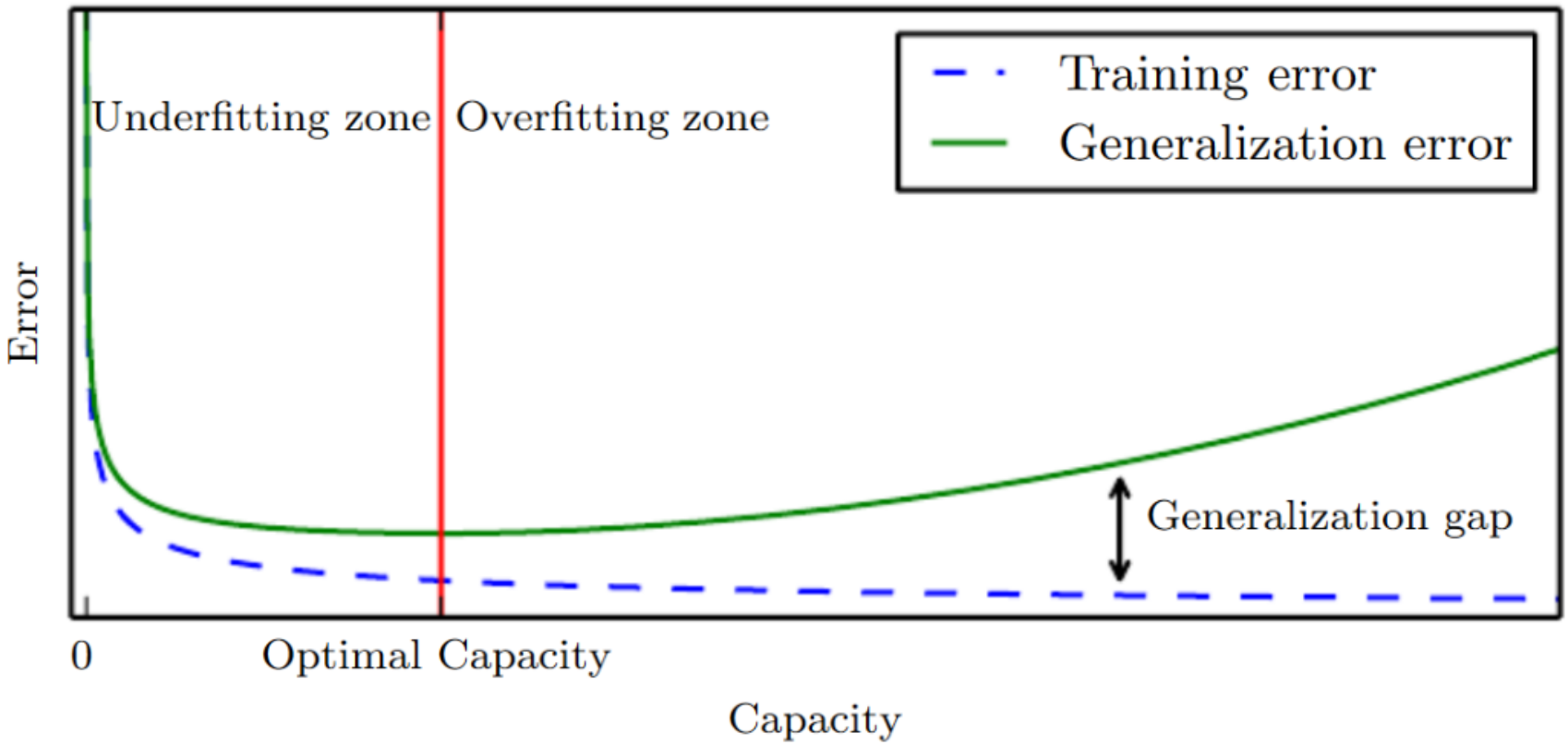
- Training error (lỗi huấn luyện)
  - Là lỗi đo được trên tập dữ liệu huấn luyện
  - Thường đo bằng sự sai khác giữa giá trị tính toán của mô hình và giá trị thực của dữ liệu huấn luyện
  - Trong quá trình học ta cố gắng làm giảm tới mức tối thiểu lỗi huấn luyện
- Test error (lỗi kiểm tra)
  - Là lỗi đo được trên tập dữ liệu kiểm tra
  - Là cái ta thực sự quan tâm

Làm sao ta có thể tác động tới hiệu quả của mô hình trên tập dữ liệu kiểm tra khi ta chỉ quan sát được tập dữ liệu huấn luyện?

Vấn đề quá vừa dữ liệu (overfitting)

- Quá vừa dữ liệu (data overfitting hay overfitting) là vấn đề thường gặp trong học máy và ảnh hưởng nhiều tới độ chính xác của các mô hình.
- Khi xây dựng cây quyết định, thuật toán cố gắng xây dựng cây phù hợp với dữ liệu một cách tối đa.
- Khi cây cho độ chính xác tốt trên dữ liệu huyến luyện nhưng lại cho kết quả không tốt trên dữ liệu kiểm tra => Cây quyết định quá vừa (overfitting) với dữ liệu huấn luyện.

Underfitting và Overfitting



Underfitting: dưới vừa; Overfitting: quá vừa

Generalization error = test error

Capacity: Khả năng của mô hình

Chống quá vừa bằng cách tỉa cây

- Chia dữ liệu thành hai phần
  - Huấn luyện
  - Kiểm tra
- Tạo cây đủ lớn trên dữ liệu huấn luyện
- Tính độ chính xác của cây trên tập kiểm tra
- Loại bỏ cây con sao cho kết qủa trên dữ liệu kiểm tra được cải thiện nhất
- Lặp cho đến khi không còn cải thiện được kết quả nữa

Thuật toán C4.5

Thuật toán C4.5 là cải tiến của thuật toán ID3, thực hiện tỉa các luật như sau:

- Xây dựng cây quyết định cho phép phân loại đúng tối đa dữ liệu huấn luyện
- Biến đổi các cây thành các luật
- Tỉa từng luật bằng cách bỏ bớt các điều kiện thành phần nếu sau khi bỏ độ chính xác tăng lên.
- Sắp xếp các luật sau khi tỉa theo mức độ chính xác trên tập kiểm tra

Sử dụng các thuộc tính có giá trị liên tục

- Tạo ra những thuộc tính rời rạc mới
- Ví dụ, với những thuộc tính liên tục A, tạo ra những thuộc tính Ac như sau:
  - Ac=true nếu $A > c$
  - Ac=false nếu $A <= c$
- Xác định ngưỡng c thế nào?: Thường chọn sao cho Ac đem lại độ tăng thông tin lớn nhất
- Có thể chia thành nhiều khoảng với nhiều ngưỡng

Ví dụ

- Chẳng hạn, nhiệt độ được cho dưới dạng số đo thực như trong ví dụ sau (ở đây nhiệt độ tính bằng độ F):

Nhiệt độ	45	56	60	74	80	90
Chơi tennis	không	không	có	có	có	không

- Xác định những trường hợp hai ví dụ nằm cạnh nhau nhưng có nhãn khác nhau.
- Giá trị trung bình của thuộc tính A của hai thuộc tính như vậy sẽ được sử dụng làm giá trị dự kiến của ngưỡng c

  Ví dụ: (56+60)/2 = 58; (80+90)/2 = 85
- Tính độ tăng thông tin cho từng giá trị dự kiến và chọn c đem lại độ tăng thông tin lớn nhất (Nhiệt độ_{58} và Nhiệt độ_{85})

Các độ đo khác

- Độ đo Information Gain (IG) ưu tiên thuộc tính có nhiều giá trị (thuộc tính có độ tăng thông tin cao nhất với tập dữ liệu)
- Thông tin chia

$$\text{SplitInformation(S,A)} = -\sum_{i=1}^{c} \frac{|S_i|}{|S|} log_2 \frac{|S_i|}{|S|}$$

- Tiêu chuẩn đánh giá thuộc tính

$$\text{GainRatio} = \frac{\text{InformationGain(S, A)}}{\text{SplitInformation(S,A)}}$$

Phân loại Bayes đơn giản

Xác suất hậu nghiệm cực đại (MAP)

- Với một tập các giả thiết (các phân lớp) có thể H, hệ thống học sẽ tìm giả thiết có thể xảy ra nhất (the most probable hypothesis) $h(h \in H)$ đối với các dữ liệu quan sát được D
- Giả thiết h này được gọi là giả thiết có xác suất hậu nghiệm cực đại (Maximum a posteriori – MAP)
- $h_{MAP} = argmax_{h \in H} P(h|D)$
- $h_{MAP} = argmax_{h \in H} \frac{P(D|h).P(h)}{P(D)}$ (Định lý Bayes)
- $h_{MAP} = argmax_{h \in H} P(D|h).P(h)$ ($P(D)$ là như nhau đối với giả thiết h)

MAP - Ví dụ

- Tập H bao gồm hai giả thiết có thể:
  - h_1: Anh ta chơi tennis h_2: Anh ta không chơi tennis
- Tính giá trị của 2 xác suất có điều kiện: $P(h_1|D)$ và $P(h_2|D)$
- Giả thiết có thể nhất $h_{MAP} = h_1$ nếu $P(h_1|D) \geq P(h_2|D)$, ngược lại thì $h_{MAP} = h_2$
- Bởi vì $P(D) = P(D, h_1) + P(D, h_2)$ là như nhau đối với cả hai giả thiết h_1 và h_2, nên ta có thể bỏ qua đại lương $P(D)$
- Vì vậy cần tính hai biểu thức: $P(D|h_1).P(h_1)$ và $P(D|h_2).P(h_2)$, và đưa ra quyết định tương ứng
  - Nếu $P(D|h_2).P(h_1) \geq P(D|h_2).P(h_2)$, thì kết luận là anh ta có đi chơi tennis, ngược lại thì kết luận là anh ta không đi chơi tennis

Phân loại Bayes đơn giản (Naïve Bayes classification) (1/2)

- Biểu diễn bài toán phân loại (classification problem)
  - Một tập học S, trong đó mỗi ví dụ học x được biểu diễn là một vector n chiều: $(x_1, x_2, \ldots x_n)$
  - Một tập xác định các nhãn lớp: $C = \{c_1, c_2, \ldots, c_m\}$
  - Với một ví dụ mới z, cần xác định z được phân vào lớp nào?
- Mục tiêu: Xác định phân lớp có thể phù hợp đối với z:
  - $c_{MAP} = argmax_{c_i \in C} P(c_i|z)$
  - $c_{MAP} = argmax_{c_i \in C} P(c_i|z_1, z_2, \ldots, z_n)$
  - $c_{MAP} = argmax_{c_i \in C} \frac{P(z_1, z_2, \ldots, z_n|c_i).P(c_i)}{P(z_1, z_2, \ldots, z_n)}$

Phân loại Bayes đơn giản (Naïve Bayes classification) (2/2)

- Để tìm được phân lớp có thể có với z
  - $c_{MAP} = argmax_{c_i \in C} P(z_1, z_2, \ldots, z_n | c_i).P(c_i)$
  - $P(z_1, z_2, \ldots, z_n)$ là như nhau với các lớp
- Giả sử trong phương pháp phân loại Naïve Bayes. Các thuộc tính là độc lập có điều kiện với các lớp.

$$P(z_1, z_2, \ldots, z_n | c_i) = \Pi_{j=1}^{n} P(z_j | c_i)$$

- Phân loại Naive Bayes tìm phân lớp có thể nhất đối với z:

$$c_{NB} = argmax_{c_i \in C} P(c_i).\Pi_{j=1}^{n} P(z_j | c_i)$$

Phân loại Bayes đơn giản - Giải thuật

- Giai đoạn học (training phase), sử dụng một tập học đối với mỗi phân lớp có thể (mỗi nhãn lớp) $c_i \in C$
  - Tính giá trị xác suất trước $P(c_i)$
  - Đối với mỗi giá trị thuộc tính z_j, tính giá trị xác suất xảy ra của giá trị thuộc tính đó đối với một phân lớp c_i : $P(z_j|c_i)$
- Giai đoạn phân lớp (classification phase) đối với một ví dụ mới
  - Đối với mỗi phân lớp $c_i \in C$, tính giá trị của biểu thức:

    $P(c_i).\Pi_{j=1}^{n} P(z_j|c_i)$

    Xác định phân lớp của z là lớp có thể nhất c^*

    $c^* = argmax_{c_i \in C} P(c_i).\Pi_{j=1}^{n} P(z_j|c_i)$

Ví dụ

Một sinh viên trẻ với thu nhập trung bình và mức đánh giá tín dụng bình thường sẽ mua một cái máy tính?

Rec. ID	Age	Income	Student	Credit_Rating	Buy_Computer
1	Young	High	No	Fair	No
2	Young	High	No	Excellent	No
3	Medium	High	No	Fair	Yes
4	Old	Medium	No	Fair	Yes
5	Old	Low	Yes	Fair	Yes
6	Old	Low	Yes	Excellent	No
7	Medium	Low	Yes	Excellent	Yes
8	Young	Medium	No	Fair	No
9	Young	Low	Yes	Fair	Yes
10	Old	Medium	Yes	Fair	Yes
11	Young	Medium	Yes	Excellent	Yes
12	Medium	Medium	No	Excellent	Yes
13	Medium	High	Yes	Fair	Yes
14	Old	Medium	No	Excellent	No

Phân loại Bayes đơn giản - Vấn đề

- Nếu không có ví dụ nào gắn với phân lớp c_i có giá trị thuộc tính z_j:
  $=>$ $P(z_j|c_i) = 0$, vì vậy $P(c_i).\Pi_{j=1}^{n} P(z_j|c_i) = 0$
- Giải pháp: Sử dụng phương pháp Bayes để ước lượng giá trị $P(z_j|c_i)$

$$P(z_j|c_i) = \frac{n(c_i, z_j) + mp}{n(c_i) + m}$$

Trong đó:

- $n(c_i)$: số lượng các mẫu gắn với phân lớp c_i
  $n(c_i, z_j)$: số lượng các ví dụ gắn với phân lớp c_i có giá trị thuộc tính z_j
- p: ước lượng đối với giá trị xác suất $P(z_j|c_i)$, $p = 1/k$ với k là giá trị có thể có của thuộc tính z_j
- m: hệ số (trọng số)

Phân loại Bayes đơn giản với dữ liệu liên tục

- Trong trường hợp thuộc tính nhận giá trị liên tục, người ta thường giả sử giá trị đặc trưng liên quan tới mỗi nhãn phân loại tuân theo phân bố Gauss và sử dụng phân bố này để biểu diễn.
- Mô hình này được gọi là Bayes đơn giản Gauss (Gaussian naive Bayes)
- Xác suất thuộc tính x_i nhận giá trị v đối với nhãn phân loại y:

$$P(x_i = v|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp(-\frac{(v-\mu_y)^2}{2\sigma_y^2})$$

- Trong đó: μ_y và σ_y^2 là trung bình và phương sai cho các giá trị của thuộc tính x_i gắn với nhãn phân loại y

Ví dụ

Giả sử cho dữ liệu huấn luyện sau với một thuộc tính liên tục "chiều cao" (cm) và phân loại giới tính có thể nhận giá trị "nam" hoặc "nữ".

Giới tính	nam	nữ	nam	nam	nam	nữ	nữ	nữ	nữ
Chiều cao	175	158	168	171	166	157	161	160	164

Với một người cao 165cm, xác suất người đó là nam hay nữ cao hơn?

Học dựa trên ví dụ

Học dựa trên láng giềng gần nhất

Một số tên gọi khác của phương pháp học dựa trên các láng giềng gần nhất (Nearest neighbors learning)

- Instance-based learning
- Lazy learning
- Memory-based learning

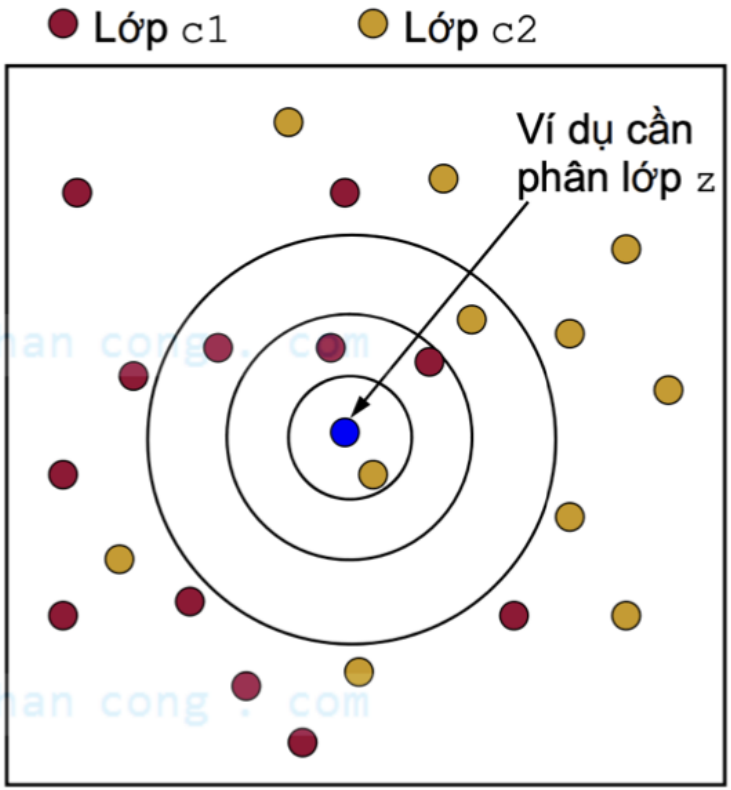
Ý tưởng:

- Với một tập các ví dụ học (huấn luyện)
  - Đơn giản là lưu lại các ví dụ học
  - Chưa xây dựng một mô hình (mô tả) rõ ràng và tổng quát của hàm mục tiêu cần học
- Đối với một ví dụ cần phân loại/dự đoán
  - Xét quan hệ giữa ví dụ đó với các ví dụ học để gán giá trị của hàm mục tiêu (một nhãn lớp, hoặc một giá trị thực)

Học dựa trên láng giềng gần nhất

- Biểu diễn đầu vào của bài toán
  - Mỗi ví dụ x được biểu diễn là một vector n chiều
  - $x = (x_1, x_2, \ldots, x_n)$ trong đó $x_i \in R$ là một số thực
- Có thể áp dụng với các bài toán phân loại hoặc hồi quy

Ví dụ trong bài toán phân lớp



- Xét một láng giềng nhất nhất => Gán z vào lớp c_2
- Xét ba láng giềng gần nhất => Gán z vào lớp c_1
- Xét năm láng giềng gần nhất => Gán z vào lớp c_1

Giải thuật k-NN trong bài toán phân lớp

- Mỗi ví dụ x được biểu diễn bởi hai thành phần:
  - Mô tả của ví dụ: $x = (x_1, x_2, \ldots, x_n)$, trong đó $x_i \in R$
  - Nhãn lớp của ví dụ
- Giai đoạn học (huấn luyện): Lưu lại các ví dụ trong tập dữ liệu huấn luyện
- Giai đoạn phân lớp: Để phân lớp cho ví dụ mới z:
  - Với mỗi ví dụ x thuộc tập dữ liệu huấn luyện, tính khoảng cách giữa x và z
  - Xác định tập láng giềng gần nhất của z => Gồm k ví dụ học trong tập dữ liệu huấn luyện gần nhất với z theo một hàm khoảng cách d
  - Phân z vào lớp chiếm số đông (the majority class) trong số các lớp của các ví dụ học.

Giải thuật k-NN trong bài toán hồi quy

- Mỗi ví dụ x được biểu diễn bởi hai thành phần:
  - Mô tả của ví dụ: $x = (x_1, x_2, \ldots, x_n)$, trong đó $x_i \in R$
  - Giá trị đích mong muốn $y_i \in R$ là một số thực
- Giai đoạn học (huấn luyện): Lưu lại các ví dụ trong tập dữ liệu huấn luyện
- Giai đoạn dự đoán: Để dự đoán giá trị đầu ra cho ví dụ mới z:
  - Với mỗi ví dụ x thuộc tập dữ liệu huấn luyện, tính khoảng cách giữa x và z
  - Xác định tập láng giềng gần nhất của z là $NB(z)$ => Gồm k ví dụ học trong tập dữ liệu huấn luyện gần nhất với z theo một hàm khoảng cách d
  - Dự đoán giá trị đầu ra đối với z:

$$y_z = \frac{1}{k} \sum_{x \in NB(z)} y_x$$

Chọn số láng giềng gần nhất

- Việc phân lớp (hay dự đoán) chỉ dựa trên duy nhất một láng giềng gần nhất thường không chính xác
  - Nếu ví dụ học này là một ví dụ bất thường, không điển hình-rất khác so với ví dụ khác
  - Nếu ví dụ học này có nhãn lớp (giá trị đầu ra) sai-do lỗi trong quá trình thu nhập (xây dựng) tập dữ liệu
- Thường xét $k(> 1)$ các ví dụ học, các láng giềng gần nhất với ví dụ cần phân lớp/dự đoán
- Đối với bài toán phân lớp có 2 lớp, k thường được chọn là một số lẻ, để tránh cân bằng về tỉ lệ ví dụ giữa hai lớp. (Ví dụ: k=3,5,7,...)

Một số hàm tính khoảng cách

- Hàm Eulid

$$d(x,z) = \sqrt{(\sum_{i=1}^{n}(x_i - z_i)^2)}$$

- Hàm Manhattan

$$d(x,z) = \sum_{i=1}^{n}|x_i - z_i|$$

- Một số hàm khoảng cách khác: Hàm Minkowski, Hàm chebyshev, Khoảng cách Hamming, Cosin,...

Bài tập

Cho dữ liệu huấn luyện dưới đây, các dòng A, B, C là thuộc tính, D là nhãn phân loại, E là nhãn dự đoán (hồi quy).

A	2	2	1	1	2	1
B	1	2	1	2	1	1
C	1	2	1	1	2	2
D	+	+	+	+	-	-
E	1.5	1.25	2	2.5	0.5	0.75

Tìm nhãn dự đoán (E)cho mẫu dữ liệu (A=2, B=2, C=1) sử dụng phương pháp k láng giềng gần nhất với k = 3.