TRƯỜNG ĐẠI HỌC VINH

ĐÁNH GIÁ
MÔ HÌNH PHÂN LỚP DỮ LIỆU

Phan Anh Phong, PhD.
Vinh University

1

Nội dung

- Đặt vấn đề
- Đánh giá mô hình phân lớp
- Cách phân chia dữ liệu
- Một số chỉ số (độ đo) đánh giá
- Bài tập
- Thảo luận

2

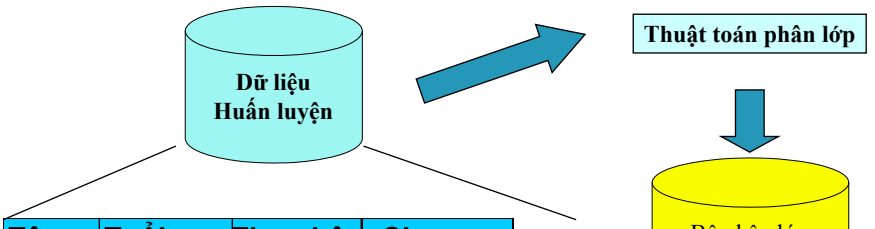
Phân lớp dữ liệu

- Cho 1 tập các đối tượng, mỗi đối tượng được xác định bởi tập THUỘC TÍNH (ĐẶC TRƯNG) và 1 THUỘC TÍNH PHÂN LỚP đối tượng
- Tìm 1 MÔ HÌNH (model) để phân loại các đối tượng dựa vào các đặc trưng đó

- Mục tiêu: Đánh giá hiệu năng mô hình phân lớp ?

3

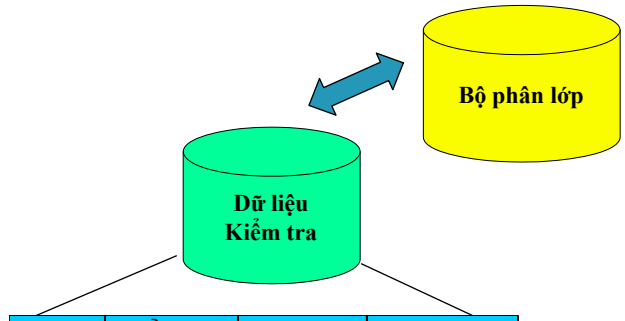
Xây dựng mô hình phân lớp



Tên	Tuổi	Thu nhập	Cho vay
Xuân	30	5M	no
Hạ	40	10M	yes
Thu	35	20M	yes
Đông	25	7M	yes
Tây	45	10M	no
Bắc	50	15M	no

4

Đánh giá mô hình phân lớp

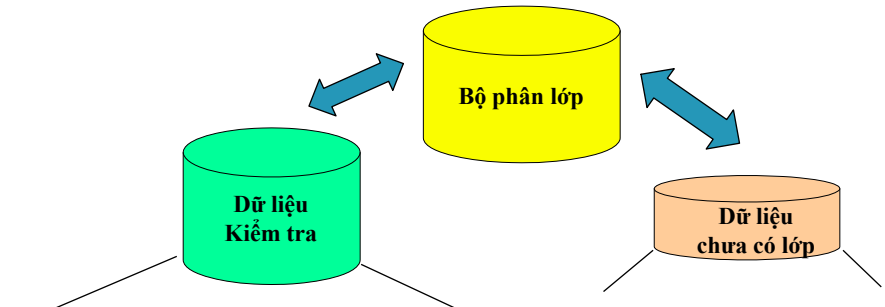


Tên	Tuổi	Thu nhập	Cho vay
Tùng	25	10M	no
Trúc	35	6M	yes
Cúc	40	15M	yes
Mai	28	9M	yes

Hiệu năng
như thế nào

5

Sử dụng mô hình phân lớp



Tuổi	Thu nhập	Cho vay
25	10M	
35	6M	
40	15M	
28	9M	

Có cho vay?
YES/NO?

6

Hiệu năng mô hình phân lớp

- Tiêu chí đánh giá
- Hiệu năng của mô hình phân lớp
  - Dựa vào dữ liệu kiểm tra
  - Các độ đo/chỉ số đánh giá
  - Cách tiếp cận phân chia dữ liệu
- Độ đo đánh giá hiệu năng
  - Về độ chính xác chung (Accuracy)
  - Đánh giá theo Precision và Recall
  - Đánh giá độ chính xác theo ma trận nhầm lẫn (Confusion matrix)
  - Các độ đo khác

7

Tiêu chí đánh giá

- Tính chính xác (Accuracy)
- Tính hiệu quả (Efficiency): Chi phí về thời gian và tài nguyên (bộ nhớ) cần thiết cho việc huấn luyện và kiểm thử hệ thống
- Khả năng xử lý nhiễu (Robustness): Khả năng xử lý (chịu được) của hệ thống đối với các ví dụ nhiễu (lỗi) hoặc thiếu giá trị
- Khả năng mở rộng (Scalability): Hiệu năng của hệ thống (vd: tốc độ học/phân lớp) thay đổi như thế nào đối với kích thước của tập dữ liệu tăng
- Khả năng diễn giải (Interpretability): Mức độ dễ hiểu (đối với người sử dụng) của các kết quả và hoạt động của hệ thống
- Mức độ phức tạp (Complexity): Mức độ phức tạp của mô hình hệ thống (hàm mục tiêu) học được

8

Đánh giá mô hình phân lớp

- Trả lời những câu hỏi :
  - Mô hình đã được huấn luyện thành công hay chưa?
  - Mức độ thành công của mô hình tốt đến đâu?
  - Cần tối ưu như thế nào? Kỹ thuật? Tham số?
  - Khi nào nên dừng quá trình huấn luyện?
  - Khi nào nên cập nhật mô hình?

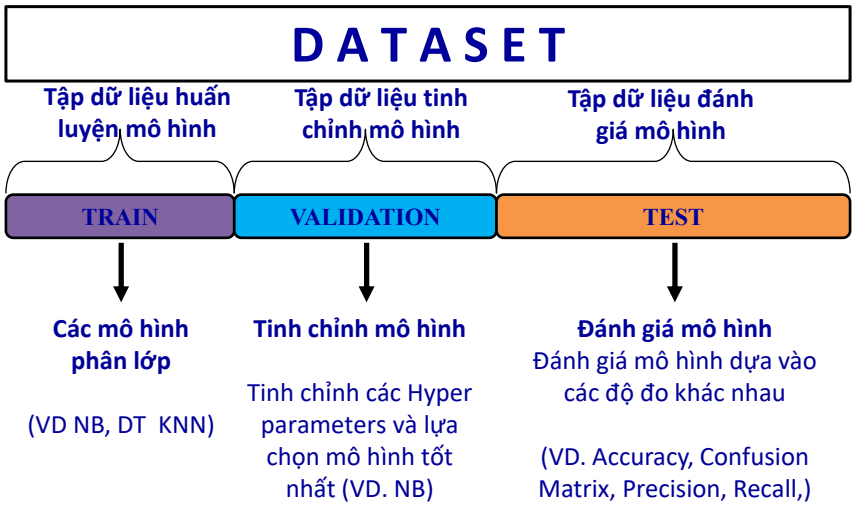
9

Cách phân chia dữ liệu đánh giá

- Bộ dữ liệu cỡ lớn
  - Chia 3 phần bằng nhau, 2 dùng huấn luyện, 1 dùng để kiểm tra đánh giá độ chính xác
  - Hoặc chia 3 ngẫu nhiên k lần, mỗi lần thực hiện như trên, sau đó tính trung bình cộng độ chính xác
- Bộ dữ liệu cỡ vừa
  - Cross-validation (k-fold, ở đây k = 10 thường được dung, cũng có thể chọn k = 5
- Bộ dữ liệu cỡ nhỏ
  - Leave – one - out

10

Cách chia dữ liệu khác



11

Cách chia dữ liệu khác

- Một bộ dữ liệu có thể chia 2 hoặc 3 : huấn luyện mô hình, tinh chỉnh mô hình (fine-tuning) và đánh giá mô hình.
- Tùy thuộc tính chất của bộ dữ liệu (số mẫu dữ liệu lớn hay nhỏ, có cân bằng hay không…) => có nhiều cách để phân chia khác
  - Hold-out/Repeated Hold-out (thường được sử dụng khi dataset lớn)
  - K-Fold
  - Leave-one-out
  - Statified sampling (dùng cho imbalanced dataset)
  - Bootstrap sampling
  - …

12

Các độ đo hiệu năng

- Accracy (độ chính xác): là tỉ lệ giữa số mẫu dữ liệu được dự đoán đúng và tổng số mẫu được kiểm tra.
  - Nhược điểm của Accuracy là chỉ cho ta biết độ chính xác khi dự báo của mô hình, nhưng không thể hiện mô hình đang dự đoán sai như thế nào
  - Accuracy lộ rõ hạn chế khi được sử dụng trên bộ dữ liệu không cân bằng (imbalanced dataset)
  - Ví dụ: Cho mô hình phân lớp A, và 10 bản ghi kiểm tra. Mô hình A phân lớp đúng 7 bản ghi thì độ chính xác của bộ phân lớp này là 70%
  - Cách này mang tính chung chung mà không chú tâm vào từng lớp

13

Hiệu năng mô hình phân lớp

- Ví dụ

Lớp thực tế\Lớp dự đoán	Mua_laptop = yes	Mua_laptop = no	Tổng
Mua_laptop = yes	**15**	**5**	**20**
Mua_laptop = no	**3**	**7**	**10**
Tổng	**18**	**12**	**30**

Độ chính xác là :
Accuracy = (15+7)/30 = 22/30 ≈ 73,33%

14

Các độ đo hiệu năng

- Ma trận nhầm lẫn (Confusion Matrix)

		Predicted Class: Positive	Predicted Class: Negative	
Actual Class	Positive	True Positive (TP)	False Negative (FN) **Type II Error**	**Sensitivity** $\frac{TP}{(TP+FN)}$
	Negative	False Positive (FP) **Type I Error**	True Negative (TN)	**Specificity** $\frac{TN}{(TN+FP)}$
		Precision $\frac{TP}{(TP+FP)}$	**Negative Predictive Value** $\frac{TN}{(TN+FN)}$	**Accuracy** $\frac{TP+TN}{(TP+TN+FP+FN)}$

15

Các độ đo hiệu năng

- Precision cho biết trong số các mẫu dữ liệu được mô hình phân lớp vào lớp Positive, có bao nhiêu mẫu thực sự thuộc lớp Positive.
- Recall cho biết được có bao nhiêu mẫu dữ liệu thực sự ở lớp Positive được mô hình phân lớp đúng trong mọi mẫu dữ liệu thực sự ở lớp Positive.
- Precision và Recall có giá trị trong [0,1], hai giá trị này càng gần với 1 thì mô hình càng chính xác. Precision càng cao đồng nghĩa với các điểm được phân loại càng chính xác. Recall càng cao cho thể hiện cho việc ít bỏ sót các điểm dữ liệu đúng.

16

Độ đo đánh giá mô hình

		Nhãn lớp thực tế	
		Positive	Negative
Phân lớp dự đoán	Positive	20	70
	Negative	80	930

- Accuracy = (20+930)/(20+80+70+930) = 86.36%
- TP_{rate} = Recall = 20/(20+80) = 20%
- Accuracy = (10+930)/(10+90+70+930) = 85.45%
- TP_{rate} = Recall = 10/(10+80) = 10%
- Accuracy = (00+930)/(00+100+70+930) = 84.54%
- TP_{rate} = 00%

17

Các chỉ số theo ma trận nhầm lẫn

		Nhãn lớp thực tế	
		Positive	Negative
Phân lớp dự đoán	Positive	TP (True positie)	FP (False Possitive)
	Negative	FN (False Negative)	TN (True Negative)

$accuracy = \frac{TP + TN}{TP + FN + FP + TN}$	
$True\ Positive\ Rate = TP_{rate} = Sensitivity = Recall = \frac{TP}{TP + FN}$	$Positive\ Predictive\ Value = PP_{value} = Precision = \frac{TP}{TP + FP}$
$True\ Negative\ Rate = TN_{rate} = Specificity = \frac{TN}{TN + FP}$	$Negative\ Predictive\ Value = NP_{value} = \frac{TN}{TN + FN}$
$False\ Positive\ Rate = FP_{rate} = \frac{FP}{TN + FP}$	$F\text{-}measure = (1 + \beta^2) \cdot \frac{Precision \cdot Recall}{(\beta^2 \cdot Precision) + Recall}$
$False\ Negative\ Rate = FN_{rate} = \frac{FN}{TP + FN}$	$G\text{-}mean = Geometric\ mean = \sqrt{Sensitivity \cdot Specificity}$

18

Cách chia dữ liệu

- ***Hold-out/Repeated Hold-out*** **(thường được sử dụng khi dataset lớn)**
- Tập dữ liệu (*data*) sẽ được chia thành 2 tập con *datatrain* và *datatest* không giao nhau (|*datatrain*| >>|*datatest*|).

- Tập huấn luyện *datatrain*: để huấn luyện hệ thống
- Tập kiểm thử *datatest*: để đánh giá hiệu năng của hệ thống sau khi đã được huấn luyện
- Yêu cầu:
  - Dữ liệu thuộc tập kiểm thử *datatest* không được sử dụng trong quá trình huấn luyện hệ thống.
  - Dữ liệu thuộc tập huấn luyện *datatrain* không được sử dụng trong quá trình đánh giá hệ thống sau khi huấn luyện.

19

Cách chia dữ liệu

- ***K-Fold***
- Tập dữ liệu (*data*) được chia thành k tập con không giao nhau (gọi là "fold") có kích thước xấp xỉ nhau.
- Mỗi lần lặp, một tập con trong *k* tập sẽ được dùng để làm tập kiểm thử, *(k-1)* tập còn lại sẽ được sử dụng làm tập huấn luyện.
- *k* giá trị lỗi (mỗi giá trị tương ứng với mỗi "fold") sẽ được tính trung bình cộng để thu được giá trị lỗi tổng thể.
  Ví dụ: ta có thể chia data thành 10 hoặc 5 folds *(k = 10 hoặc k = 5)*

- Thông thường mỗi tập con (fold) được lấy mẫu phân tầng (xấp xỉ phân bố lớp) trước khi áp dụng quá trình đánh giá Cross validation
  => Phù hợp khi ta có tập dữ liệu *data* vừa và nhỏ.

20

Cách chia dữ liệu

- **leave-one-out**
  - Số lượng các nhóm folds bằng kích thước của tập dữ liệu *(k = |data|)*
  - Mỗi nhóm fold chỉ bao gồm 1 ví dụ
  - Khai thác tối đa tập dữ liệu ban đầu
  - Không có bước lấy mẫu ngẫu nhiên
  - Chi phí tính toán cao => Phù hợp khi ta có tập dữ liệu data (rất) nhỏ.

21

Cách chia dữ liệu

- **Statified sampling (dùng cho imbalanced dataset)**
- Được sử dụng khi các tập ví dụ có kích thước nhỏ hoặc không cân xứng (unbalanced datasets).
- Ví dụ: có ít hoặc không có các ví dụ với một số lớp
- Mục tiêu: Phân bố lớp (Class distribution) trong tập huấn luyện và tập kiểm thử phải xấp xỉ như trong tập toàn bộ các ví dụ (*data*)
- Stratified sampling là một phương pháp để cân xứng về phân bố lớp
- Đảm bảo tỉ lệ phân bố lớp trong tập huấn luyện và tập kiểm thử sẽ là xấp xỉ nhau
- => Phương pháp này không áp dụng được cho bài toán học máy hồi quy (vì giá trị đầu ra của hệ thống là một giá trị số, không phải là một nhãn lớp)

22

Cách chia dữ liệu

- **Bootstrap sampling**
- Phương pháp này sử dụng việc lấy mẫu lặp lại để tạo nên tập huấn luyện.
- Giả sử toàn bộ tập *data* bao gồm *n* ví dụ
- Lấy mẫu có lặp lại *n* lần đối với tập *data* để tạo nên tập huấn luyện *datatrain* gồm *n* ví dụ:
  - Từ tập *data*, lấy ngẫu nhiên một ví dụ *x* (nhưng không loại bỏ *x* khỏi *data*)
  - Đưa dữ liệu *x* vào trong tập huấn luyện
  - Lặp lại các bước trên *n* lần, ta có *n* dữ liệu trong tập *datatrain*
    => Sử dụng dữ liệu tập *datatrain* để huấn luyện hệ thống.
    => Sử dụng tất cả các dữ liệu thuộc *data* nhưng không thuộc tập huấn luyện (*datatrain*) để tạo nên tập test.
- Xác suất để 1 ví dụ không được chọn vào tập huấn luyện là (1-1/n).
- Xác suất để một ví dụ (sau khi lấy mẫu lặp lại – bootstrap sampling) được đưa vào tập kiểm thử là: (1-1/n)^n => Phù hợp với tập dữ liệu có kích thước (rất) nhỏ

23

Bài tập -

- **Tính Accuracy cho thuật toán KNN với Leave-One-Out (LOO)**
- Cho một tập dữ liệu gồm 10 mẫu hoa tulip, mỗi mẫu được mô tả bởi ba đặc trưng: chiều cao, đường kính hoa và màu sắc. Trong đó màu sắc là thuộc tính phân lớp

ID	Chiều cao (cm)	Đường kính hoa (cm)	Màu sắc
1	20	3	Đỏ
2	30	4	Vàng
3	25	2	Vàng
4	22	3	Đỏ
5	32	5	Vàng
6	28	2	Vàng
7	24	3	Đỏ
8	35	5	Vàng
9	27	2	Vàng
10	23	3	Đỏ

Sử dụng thuật toán KNN với k = 3 và phương pháp Leave-One-Out (LOO) để tính toán accuracy

24

Bài tập (con.)

Sử dụng thuật toán KNN với k = 3 và phương pháp Leave-One-Out (LOO) để tính toán accuracy

1. **LOO:** Lặp lại 10 lần, mỗi lần loại bỏ một mẫu hoa làm mẫu test và sử dụng 9 mẫu còn lại để huấn luyện mô hình.
2. **Dự đoán:** Sử dụng mô hình KNN được huấn luyện để dự đoán loại hoa cho mẫu test đã được loại bỏ.
3. **So sánh:** So sánh dự đoán của mô hình với loại hoa thực tế của mẫu test.
4. **Tính toán accuracy:** Tính tỷ lệ giữa số dự đoán chính xác và tổng số mẫu test (10 mẫu).

- **Chẳng hạn:**
- Lần 1: Loại bỏ mẫu 1, huấn luyện mô hình với 9 mẫu còn lại, dự đoán loại hoa cho mẫu 1.
- Lần 2: Loại bỏ mẫu 2, huấn luyện mô hình với 9 mẫu còn lại, dự đoán loại hoa cho mẫu 2.

...

- Lần 10: Loại bỏ mẫu 10, huấn luyện mô hình với 9 mẫu còn lại, dự đoán loại hoa cho mẫu 10.
- **Sau 10 lần lặp, ta có thể tính toán accuracy trung bình của mô hình KNN với k = 3 và LOO.**