DỰ ĐOÁN KHẢ NĂNG CHI TRẢ TÍN DỤNG

By
Group 5

Class DAMI330484_21_2_01
Instuctor Quach Dinh Hoang
June 5th, 2022
Faculty of Information Technology
HCMC University of Teachnology and Education

TÓM TẮT

Bài toán phân loại, dự đoán tín dụng của một người luôn là bài toán khó khăn, do tác động bởi nhiều yếu tố bên ngoài (chưa kể sự đúng đắn của dữ liệu mà người vay cung cấp, mất mát dữ liệu, …) và những rủi ro mang lại nếu đưa ra quyết định sai lầm. Việc xây dựng một mô hình tự động đưa ra những quyết định chính xác, rủi ro thấp là cần thiết. Ta sẽ sử dụng một số thuật toán (KNN, Naïve Bayes, Random Forest) và độ đo (F1 score) để xây dựng một mô hình học máy với độ chính xác cao và rủi ro thấp nhất.

1. GIỚI THIỆU

Thị trường tài chính ngày càng phát triển và các ngân hàng đóng một vai trò quan trọng trong các nền kinh tế thị trường. Họ quyết định ai có thể được cấp vốn dựa trên những điều khoản, và điều này ảnh hưởng quyết định có hoặc không cho vay.

Việc cho cấp vốn hoặc cho vay có thể mang lại được nhiều lợi ích cho ngân hàng nếu chọn đúng đối tượng và ngược lại sẽ là rủi ro nếu trao tiền cho một người có vấn đề về khả năng chi trả.

Một bài toán được đặt ra là làm thế nào để có thể quyết định dựa trên những dữ liệu giới hạn mà người mượn cung cấp để ra quyết định tốt nhất.

Ta sẽ lần lượt sử dụng và so sánh các thuật toán K-Nearest Neighbor, Naïve Bayes và Random Forest bằng độ đo F1 score cũng như các phương pháp xử lý mất mát dữ liệu để xác định phương pháp hiệu quả nhất.

2. DỮ LIỆU

Bài toán phân tích sẽ dựa trên tập dữ liệu từ cuộc thi được tổ chức trên diễn đàn kaggle vào năm 2012 www.kaggle.com/competitions/GiveMeSomeCredit

2.1. Mô Tả Tập Dữ Liệu

Thông tin tập dữ liệu bao gồm:

1. Số thứ tự của quan sát

2. Người đó có (hoặc không) khả năng chi trả tín dụng.

3. Tổng số dư thẻ tín dụng và dòng tín dụng cá nhân

4. Tuổi người vay

5. Thu nhập hàng tháng [thuộc tính có mất mát dữ liệu]

6. Thanh toán nợ hàng tháng, tiền cấp dưỡng, chi phí sinh hoạt chia cho tổng thu nhập hàng tháng

7-9. Số lần người vay thanh toán tín dụng trễ hạn:

- *30 – 59 ngày*
- *60 – 89 ngày*
- *90 ngày hoặc hơn*

10. Số khoản vay mở và dòng tín dụng mở

11. Số lượng thế chấp và khoản vay bất động sản

12. Số người phụ thuộc trong gia đình (không bao gồm bản thân) [thuộc tính có mất mát dữ liệu]

Tập dữ liệu đã được chia sẵn thành 2 tập train và test:

- Tập train có 150 000 quan sát
- Tập test có 101 503 quan sát và mất nhãn

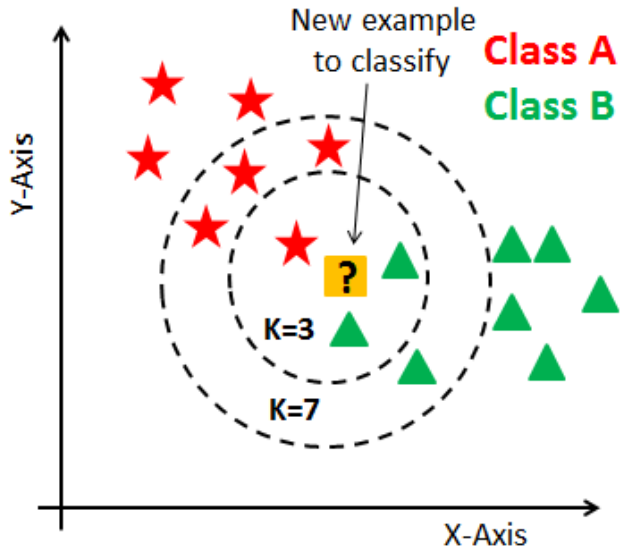
Sau khi xây dựng xong mô hình ta sẽ dự đoán tập test và nộp kết quả lên kaggle để chấm điểm.

2.2. Chuẩn Hóa Dữ Liệu

Ta sẽ chuẩn hóa 2 thuộc tính (thu nhập và tỷ lệ giữa chi trả và thu nhập) có phân bố lệch và giá trị lớn so với các thuộc tính còn lại bằng logarit nepe.

3. PHƯƠNG PHÁP

Thuật toán đầu tiên, K–Nearest Neighbor là một thuật toán đơn giản, chúng cho rằng các dữ liệu cùng 1 lớp sẽ ở gần nhau trong không gian. Chúng sẽ tính toán khoảng cách của K điểm dữ liệu gần nó nhất để phân loại vào lớp nào.



Thuật toán thứ hai, Naïve Bayes dựa trên định lý Bayes. Chúng giả định rằng các biến hoàn độc lập, từ đó tính xác suất có điều kiện để phân loại dữ liệu. Xác suất xảy ra của lớp nào cao hơn, thuật toán sẽ gán lớp đó cho bộ dữ liệu.

$$P(Class \mid X) = \frac{P(X|Class)\cdot P(Class)}{P(X)} \quad [1]$$

2.3. Xử Lý Giá Trị Mất Mát

Trong tập dữ liệu của chúng ta có 2 thuộc tính có sự mất mát dữ liệu (có thể là do người dùng không muốn cung cấp). Ta sẽ xử lý theo 3 hướng:

- Bỏ qua các giá trị bị mất mát
- Thay thế các giá trị mất mát bằng trung vị
- Rời rạc hóa các thuộc tính của tập dữ liệu (cả dữ liệu mất mát và không mất mát).

Cả 3 cách xử lý sẽ được áp dụng cho từng thuật toán để tìm ra phương pháp tốt nhất.

- *Class* là các lớp phân loại của bài toán
- *X* là tập các thuộc tính độc lập với nhau

Vấn đề đặt ra ta là ta sẽ ước lượng $P(Class \mid X)$ thông qua $P(X \mid Class)$. Như giả định ban đầu, thuật toán Naïve Bayes cho rằng các biến thuộc tính độc lập với nhau, do vậy:

$$P(X \mid Class) = \prod_{i=1}^{d} P(Xi \mid Class) \quad [2]$$

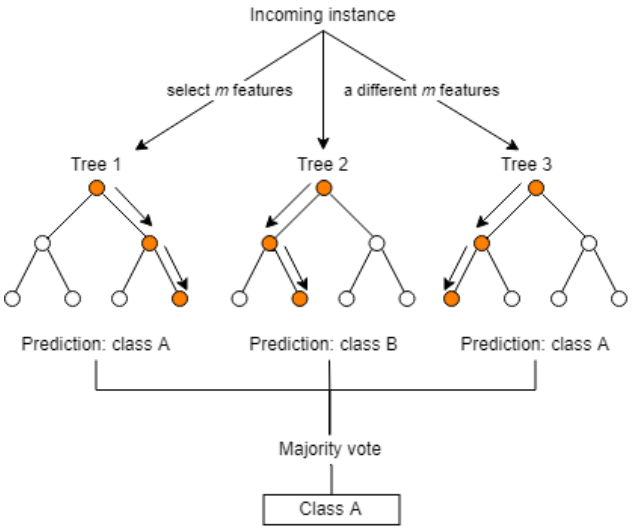
Giờ thì ta có thể ước lượng $P(X \mid Class)$ cho tất cả kết hợp giữa X_i và Class từ tập dữ liệu training. Điểm dữ liệu mới được phân loại vào lớp Y_j nếu [2] đạt cực đại.

Thuật toán thứ ba ta sẽ sử dụng là Random Forest. Thuật toán hoạt động bằng cách xây dựng nhiều cây quyết định, tuy nhiên mỗi cây quyết định sẽ khác nhau do yếu tố ngẫu nhiên:

- Lấy ngẫu nhiên dữ liệu để xây dựng cây quyết định.
- Lấy ngẫu nhiên các thuộc tính để xây dựng cây quyết định.

Sau đó kết quả dự đoán được tổng hợp từ các cây quyết định. Do mỗi cây quyết định trong thuật toán Random Forest không dùng tất cả dữ liệu training, cũng như không dùng tất cả các thuộc tính của dữ liệu để xây dựng cây nên mỗi cây có thể sẽ dự đoán không tốt, khi đó mỗi mô hình cây quyết định không bị overfitting mà có thể bị underfitting, hay nói cách khác là mô hình có high bias.

Tuy nhiên, kết quả cuối cùng của thuật toán Random Forest lại tổng hợp từ nhiều cây quyết định, thế nên thông tin từ các cây sẽ bổ sung thông tin cho nhau, dẫn đến mô hình có low bias và low variance, và mô hình có kết quả dự đoán tốt.



4. THỰC NGHIỆM

Ở phần thực nghiệm, ta sẽ tìm các siêu tham số của từng mô hình bằng cách:

K-Nearest Neighbor: Chỉ có 1 tham số ta cần xác định chọn K bằng cách thử lần lượt: 50, 100, 150 và 200 điểm dữ liệu.

Naïve Bayes: Có 3 siêu tham số, ta cũng sẽ thử lần lượt các siêu tham số:

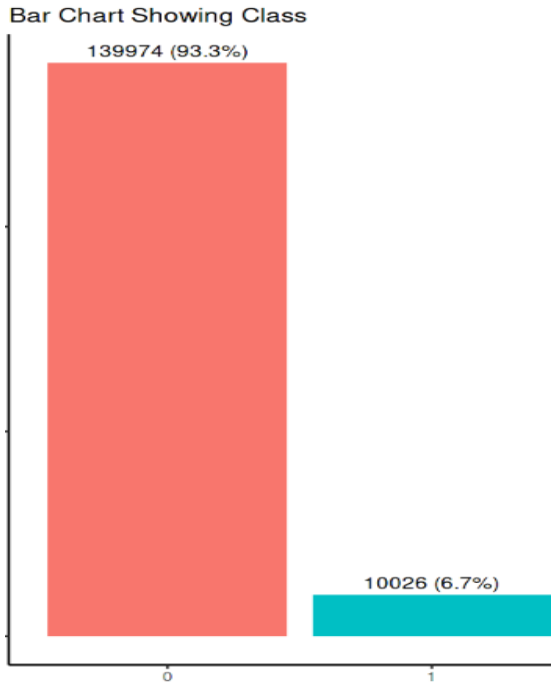
- **Kiểu phân bố** để tính xác suất cho biến liên tục: phân bố kernel và poisson.
- Sử dụng hoặc không **phương pháp làm trơn** (siêu tham số này tránh các xác suất tính được khác 0).
- Tham số **bandwidth – h** cho hàm kernel (KDE)

Random Forest Có 2 siêu tham số ta cần xác định:

- **Số cây quyết định** mà rừng ngẫu nhiên sẽ tạo, thử lân lượt chọn 100, 300, 500 cây (có thể chọn lớn hơn nhưng hiện tại không đủ tài nguyên).
- **Số thuộc tính** để tách và xây dựng cây quyết đi, sử dụng tăng dần từ 4 – 10 thuộc tính để xây dựng.

Việc tìm siêu tham số sẽ tiêu tốn một khoảng thời gian nếu sử dụng toàn bộ toàn bộ dữ liệu. Để nhanh chóng, ta sẽ sử dụng 30% tập huấn luyện ban đầu để huấn luyện và tìm siêu tham số. Do giảm số lượng dữ liệu huấn luyện, nên để kết quả tìm được không phải là ngẫu nhiên ta sẽ sử dụng cross – valiadation với 10-fold.

Xem qua thuộc tính của 2 lớp ta cần phân loại, có thể 2 lớp khá là mất cân bằng.



Trong trường hợp này, không nên sử dụng độ đo accuracy (để tránh bỏ qua có số lượng lớp ít hơn) và độ đo F-measure có vẻ phù hợp cho tình huống này. Cũng cần phải nói thêm:

		Actual	
		No	Yes
Predicted	No	TP	FP
	Yes	FN	TN

Dương tính giả là trường hợp người vay được đoán là sẽ gặp khó khăn tài chính nhưng thật chất là không.

Còn âm tính giả hoàn toàn ngược lại. có thể thấy trường hợp âm tính giả nghiêm trọng hơn rất nhiều. Dương tính giả có làm mất đi khách hàng tiềm năng, còn âm tính giả làm thất thoát tiền của ngân hàng. Dù vậy, cũng cần cân bằng trong việc giảm thiểu số trường hợp của cả hai. Và độ đo F1 sẽ phù hợp với bài toán:

$$F\text{-}measure = \frac{2 \cdot precision \cdot recall}{precision + recall}$$

Trong đó:

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

Độ đo trên sẽ được dùng để so sánh 3 thuật toán: K-Nearest Neighbor, Naïve Bayes, Random Forest.

5. KẾT QUẢ

Sau khi huấn luyện xong mô hình (30% dữ liệu từ tập train) ta sẽ tìm các siêu tham số tốt nhất trước khi dùng chúng huấn luyện cho toàn bộ tập train.

Bảng 1: Kết quả F1 sau khi huấn luyện 30% tập

	REMOVE	MEDIAN	DISCRETE
KNN	0.96 – 0.96	0.96 – 0.96	0.96 – 0.96
Naïve Bayes	0.95 – 0.96	0.96 – 0.97	0.94 – 0.97
Random Forest	0.96 – 0.97	0.96 – 0.97	0.96 – 0.97

Độ chính xác cũng khá cao đối với từng phương pháp tiền xử lý, khoảng dao động cũng không quá nhiều với độ đo F1. Như vậy, ta vẫn chưa có kết luận rõ ràng phương pháp nào tốt hơn nếu chỉ dựa vào F1. Nhưng đối với độ đo AUC thì có sự dao động mạnh. Xem qua bảng sau:

Bảng 2: kết quả AUC sau khi huấn luyện 30% tập dữ liệu training

	REMOVE	MEDIAN	DISCRETE
KNN	0.70 – 0.96	0.68 – 0.95	0.62 – 0.90
Naïve Bayes	0.95 – 0.98	0.95 – 0.98	0.97 – 0.98
Random Forest	0.81 – 0.85	0.77 – 0.82	0.64 – 0.71

**kết quả trên tương ứng với tất cả siêu tham số của mỗi thuật toán và được làm tròn.*

Bằng cách này ta sẽ chọn siêu tham số bằng cách:

1. Nếu độ đo F1 không khác biệt nhiều giữa 2 phương pháp ta sẽ dựa trên độ đo thứ 2 là AUC.

2. Nếu cả 2 độ đo trên vẫn không chênh lệch nhiều ta sẽ chọn mô hình đơn giản hơn hoặc cho khả năng diễn giải tốt hơn.

Từ tiêu chí trên ta sẽ chọn siêu tham như sau:

Bảng 3: Các siêu tham số được lựa chọn

	MEDIAN	DISCRETE

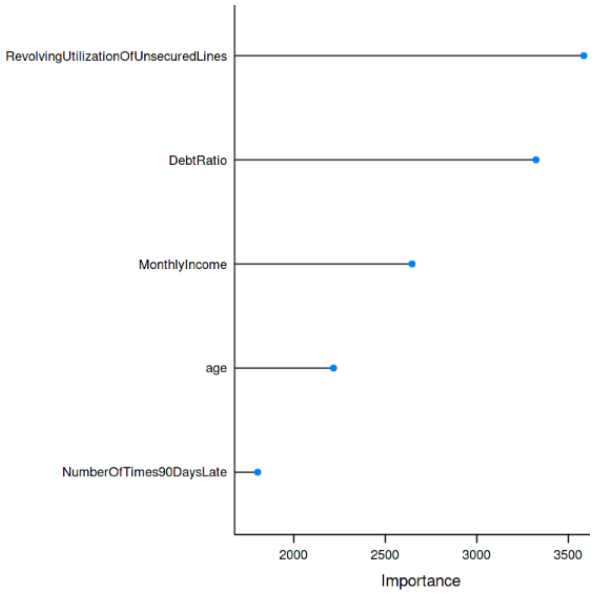
KNN	K = 200	
Naïve Bayes	(True, 0, 0.5)	(True, 0, 3.5)
Random Forest	(500, 4)	(500, 6)

Bằng các siêu tham số ta vừa xác định, ta áp dụng vào mô hình đó để dự doán và nộp kết quả thu được lên kaggle để chấm điểm. Ngoài phương pháp remove phải loại bỏ trước dữ liệu NA, nên không thể nộp lên kaggle (tập Test cũng có NA) để chấm điểm. Các phương pháp còn lại xử lý tương tự đối với tập test. (phương pháp median sẽ sử dụng trung vị của tập train). Và đây là kết quả:

Bảng 4: Điểm Nộp Trên Kaggle (Private score)

	MEDIAN	DISCRETE
KNN	0.81	0.85
Naïve Bayes	0.83	0.82
Random Forest	0.85	0.83

Ta có kết quả của mô hình tốt nhất là KNN và Random Forest. Và đây các đặc trưng quan trọng của mô hình Random Forest:



Lần lượt là *tổng số dư trên thẻ, tỷ lệ chi tiêu, thu nhập* và *tuổi* là các biến quan trọng trong mô hình Random Forest.

6. THẢO LUẬN

Ngoài độ đo F1 score để đánh giá hiệu suất của mô hình, AUC là chỉ số được tính toán dựa trên đường cong ROC (Receiving Operating Curve) nhằm đánh giá khả năng phân loại của mô hình tốt như thế nào. Đường cong ROC tính toán dựa trên 2 chỉ số. Tỉ lệ các trường hợp phân loại đúng:

$$True\ Positive\ Rate = \frac{TP}{Total\ Posive}$$

Và Tỷ lệ dự báo sai các trường hợp thực tế là negative thành thành positive trên tổng số các trường hợp thực tế là negative:

$$False\ Positive\ Rate = \frac{FP}{Total\ Negative}$$

Đây chính là các chỉ số dùng để tính toán hiệu suất phân loại của mô hình. Để hợp chúng lại thành 1 chỉ số duy nhất, ta sử dụng đường cong ROC để hiển thị từng cặp (TPR, FPR) cho các ngưỡng khác nhau với mỗi điểm trên đường cong biểu diễn 1 cặp (TPR, FPR) cho 1 ngưỡng, sau đó tính chỉ số AUC cho đường cong này. Chỉ số AUC chính là con số thể hiện hiệu suất phân loại của mô hình.

Về biến quan trọng, ta có thể dùng các kết quả này để tập trung vào các đặc trưng này trong thực tế để có thể cải thiện quyết định, và giảm đi số đặc trưng không cần thiết để đơn giản mô hình.

7. KẾT LUẬN

Như vậy, nếu F1 là độ đo cho độ chính xác của của mô hình thì AUC cho thấy được sức mạnh phân loại của mô hình.

Tổng thể thuật toán Random Forest và KNN cho kết quả gần như là như nhau và tốt hơn so với các thuật toán còn lại. Nhưng trong thực tế ta sẽ chọn KNN, vì mô hình đơn giản hơn.

Nhưng nếu có đủ tài nguyên cho phép, ta vẫn có thể cải thiện kết quả cho Random Forest bằng cách tăng số lượng cây khởi tạo cho thuật toán. Và đơn giản hóa mô hình bằng cách chỉ sử dụng những biến quan trọng ta tìm được dùng để tách. Điều này có thể làm tăng độ chính xác của thuật toán.

7. PHỤ LỤC

Phụ lục 1: KDE – Kernel Density Estimate

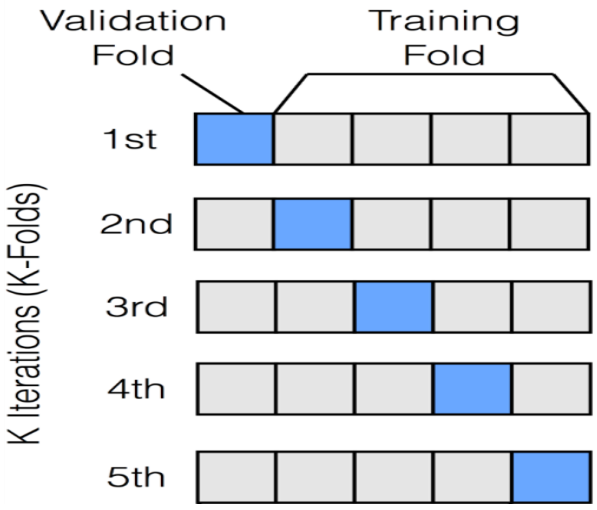
Hàm kernel sẽ giúp xác định hình dạng của đường cong trong khi độ rộng của đường cong được xác định bởi bandwidth – h.

$$\widehat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K\Big(\frac{x - x_i}{h}\Big),$$

Phương pháp KDE sẽ tính tổng của các đường cong chạy dọc theo trục để hình thành nên đường cong mật độ xác suất tổng quát cho dữ liệu.

Phụ lục 2: K – Fold Cross validation

Là một phương pháp đánh giá chéo nhau sau chia thành k phần, để đảm bảo rằng các phần trong tập dữ liệu đều được sử dụng.



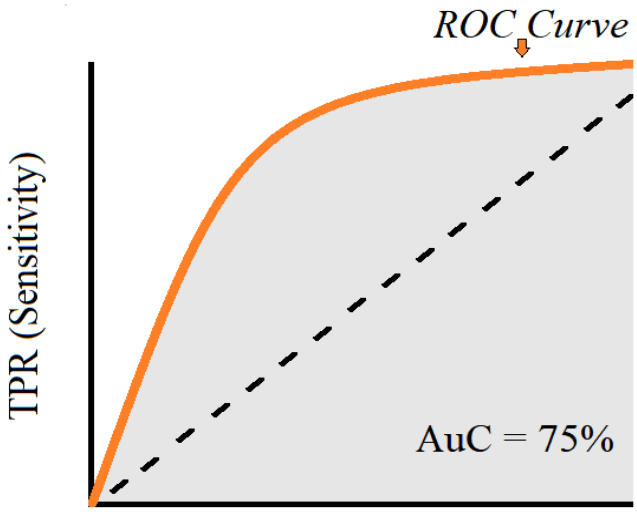
Phụ lục 3: Bảng siêu tham số tốt nhất với độ đo F1

	MEDIAN	DISCRETE
KNN	K = 50	K = 100
Naïve Bayes	(True, 0, 0.5)	(True, 0, 3.5)
Random Forest	(500, 4)	(500, 4)

⇨ Nếu chỉ xét F1 thì siêu tham số sẽ rất khác

Phụ lục 4: Ví dụ ROC Curve, AUC = 75%

Diện tích AUC càng lớn (hay đường ROC càng lồi) chứng tỏ mô hình phân loại khá tốt.



ĐÓNG GÓP

MSSV	STT	TÊN THÀNH VIÊN	ĐÓNG GÓP	TIẾN ĐỘ
19133022	19	Hồng Tiến Hào	- Tiền xử lý dữ liệu - Thuật toán random forest và các vấn đề liên quan. - Tổng hợp và viết báo cáo	100%
18133024	29	Ngô Phi Lít	- Thuật toán KNN và các vấn đề liên quan - Lựa chọn phương pháp đo	100%
19133065	64	Nguyễn Thanh Tùng	- Viết đặc tả dữ liệu - Trực quan hóa - Slide trình bày	100%
17133012	11	Đỗ Lê Tiến Đạt	- Naïve Bayes và các vấn đề liên quan - Các độ đo đánh gia mô hình (F1, AUC)	100%

TÀI LIỆU THAM KHẢO

[1] Quach Dinh Hoang. *Data Mining*, 2022. Slide và video bài giảng

[2] Jiawei Han, Micheline Kamber, Jian Pei. *Data Mining Concepts and Techniques 3rd*, 2011. Nhà xuất bản Morgan Kaufmann.

[3] Mortonkuo. *Top 7 pct Give Me Some Credit Kaggle*, from github.com/mortonkuo/Top-7-pct_Give-Me-Some-Credit_Kaggle

[4] *Give Me Some Credit*, Dataset from www.kaggle.com/competitions/GiveMeSomeCredit

[5] Wikipedia. *Kernel density estimation*, From https://en.wikipedia.org/wiki/Kernel_density_estimation