CREDIT SCORE CLASSIFICATION

PHÂN LOẠI ĐIỂM TÍN DỤNG

NHÓM 4

PROJECT CONTENT

I Giới thiệu bài toán

II Nghiên cứu liên quan

III Phương pháp

IV Thực nghiệm

V Kết luận

I. Giới thiệu bài toán

1. Điểm tín dụng

Điểm tín dụng (Credit Score) là một con số mô tả khả năng trả được nợ của một người. Điểm tín dụng dựa trên lịch sử tín dụng về các yếu tố như số tài khoản đã mở, tổng mức nợ và lịch sử trả nợ. Người cho vay sử dụng điểm tín dụng để đánh giá xác suất trả nợ đúng hạn của một cá nhân.

2. Mục đích của điểm tín dụng

Điểm tín dụng được tạo ra nhằm mục đích hỗ trợ các tổ chức tài chính trong việc xác định chính sách ngân hàng và chiến lược thương mại của họ. Một mặt, để giảm rủi ro tài chính và mặt khác, để tăng lợi nhuận.

II. NGHIÊN CỨU LIÊN QUAN

II. NGHIÊN CỨU LIÊN QUAN

Nghiên cứu "Dự đoán điểm tín dụng bằng các mô hình sử dụng máy học: Một tập dữ liệu mới"

- Sử dụng bộ dữ liệu bao gồm lịch sử giao dịch thẻ tín dụng và hồ sơ khách hàng
- Nghiên cứu thử nghiệm bằng nhiều thuật toán học máy, bao gồm logistic regression (LR), Decision Tree (DT), random forest (RF), multi layer perceptron (MLP), XGBoost và LightGBM.
- Kết quả thử nghiệm chứng minh rằng MLP vượt trội hơn LR, DT, RF, LightGBM và XGBoost về hiệu suất dự đoán trong tỷ lệ dương tính thực sự, đạt được diện tích ấn tượng dưới đường cong (AUC) là 86,7% và chỉ số accuracy là 91,6%, với recall vượt quá 80%.

II. NGHIÊN CỨU LIÊN QUAN

Nghiên cứu "Phương pháp học máy để dự đoán điểm tín dụng" đề xuất tích hợp các mô hình XGBoost và Support Vector Machine để nâng cao độ chính xác và độ tin cậy của các mô hình chấm điểm tín dụng đánh giá rủi ro. Phương pháp này sử dụng các kỹ thuật học máy để nhận ra các mẫu và xu hướng từ dữ liệu trong quá khứ để dự đoán các sự kiện trong tương lai. Cách tiếp cận được đề xuất được so sánh với các mô hình điểm tín dụng hiện có để xác nhận hiệu quả của nó bằng cách sử dụng năm số liệu đánh giá phổ biến.

III. PHƯƠNG PHÁP

3.1. Các kỹ thuật nền tảng

- Phân tích và xử lý dữ liệu
- Mô hình hóa dữ liệu
- Đánh giá mô hình
- Xử lý mất cân bằng dữ liệu
- Tinh chỉnh mô hình

3.2. Đề xuất phương pháp

Học máy truyền thống

- Cây quyết định (Decision Tree- DT)
- K láng giềng gần nhất (K-Nearest Neighbors - KNN)
- Rừng ngẫu nhiên (Random Forest - RF)
- XGBoost (eXtreme Gradient Boosting - XGB)
- Naive Bayes (NB)

Học sâu

- Mạng nơ-ron tích chập sâu (Convolutional Neural Network - CNN)
- Mạng nơ-ron hồi quy (Recurrent Neural Network - RNN)

Tiến trình
thực hiện
thực nghiệm

Bước 1: Khai thác dữ liệu

Quan sát tổng quan dữ liệu (Hệ tương quan, giá trị thống kê, vấn đề mất cân bằng dữ liệu ở biến mục tiêu).

Bước 2: Chuẩn bị dữ liệu

- Tiến hành tiền xử lý dữ liệu (xóa trùng lặp, kiểm tra dữ liệu trống), biến đổi các trường dữ liệu định danh với phương pháp mã hóa nhãn (LabelEncoder).
- Xóa các đặc trưng không quan trọng
- Sử dụng phương pháp IQR để loại bỏ các giá trị ngoại lệ
- Sử dụng phương pháp Min-Max Scaling để chuẩn hóa

Bước 3: Xây dựng tập dữ liệu huấn luyện và kiểm tra

Bộ dữ liệu được chia thành 2 tập gồm tập huấn luyện 80% và tập dữ liệu kiểm tra 20%.

Bước 4: Xây dựng và huấn luyện các mô hình

Với tập dữ liệu huấn luyện tiến hành xây dựng các mô hình với phương pháp Grid search CV để tìm ra các siêu tham số.

Bước 5: Mô hình thử nghiệm với tập dữ liệu kiểm tra đưa ra đánh giá mô hình.

Bước 6: So sánh các mô hình dựa trên hiệu suất các chỉ số và đưa ra mô hình tốt nhất

IV. THỰC NGHIỆM

4.1. Miêu tả dữ liệu

Tập dữ liệu bao gồm 28 đặc trưng với 100,000 quan sát

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	ID	Customer_ID	Month	Name	Age	SSN	Occupation	Annual_Income	Monthly_Inhand_	Num_Bank_	Num_Credit_	Interest_Ra	Num_of_Lc	Type_of_Lc	Delay_from	Num_of_De
2	0x1602	CUS_0xd40	January	Aaron M	23	821-00	Scientist	19114.12	1824.843333	3	4	3	4	Auto Loan,	3	7
3	0x1603	CUS_0xd40	February	Aaron M	23	821-00	Scientist	19114.12		3	4	3	4	Auto Loan,	-1	
4	0x1604	CUS_0xd40	March	Aaron M	-500	821-00	Scientist	19114.12		3	4	3	4	Auto Loan,	3	7

Q	R	S	T	U	V	W	X	Y	Z	AA	AB
Changed_C	Num_Credi	Credit_Mix	Outstandin	Credit_Utili	Credit_Hist	Payment_o	Total_EMI_	Amount_in	Payment_B	Monthly_Ba	Credit_Score
11.27	4	_	809.98	26.82262	22 Years an	No	49.574949	80.415295	High_spent	312.49409	Good
11.27	4	Good	809.98	31.94496	NA	No	49.574949	118.28022	Low_spent_	284.62916	Good
_	4	Good	809.98	28.609352	22 Years an	No	49.574949	81.699521	Low_spent_	331.20986	Good

Trường dữ liệu “Credit Score”: được sử dụng để đánh giá khả năng thanh toán của khách hàng và phản ánh lịch sử tín dụng của họ, có ba nhóm giá trị: “Good”(tốt), “Standard” (trung bình), và “Poor” (kém).

Thông qua việc phân loại khách hàng thành các nhóm dựa trên biến "Credit_Score", các tổ chức tài chính có thể chấm điểm tín dụng khách hàng dựa trên lịch sử tín dụng và các yếu tố liên quan.

4.2. Tiền xử lý dữ liệu

01 Loại bỏ dữ liệu không cần thiết

02 Xử lý các dữ kiệu NaN và dữ liệu chứa giá trị không chính xác

03 Sử dụng phương pháp IQR để loại bỏ các giá trị ngoại lệ

04 Mã hóa dữ liệu

05 Xử lý mất cân bằng dữ liệu

06 Chuẩn hóa dữ liệu

07 Sử dụng phương pháp One - Hot Encoding để biến đổi các giá trị

08 Xác định dữ liệu tập huấn và kiểm tra

01 Loại bỏ trường dữ liệu không cần thiết

. Tiền xử lý dữ liệu

```
#loại bỏ các cột không cần thiết
d_col = ['ID','Customer_ID','Month','Name','SSN','Monthly_Inhand_Salary','Num_Bank_Accounts','Num_Credit_Card',
         'Interest_Rate','Num_of_Loan','Type_of_Loan','Changed_Credit_Limit','Num_Credit_Inquiries','Credit_Mix',
         'Credit_Utilization_Ratio','Amount_invested_monthly']
drop_train = train.drop(d_col , axis=1).copy()
drop_train
```

	Age	Occupation	Annual_Income	Delay_from_due_date	Num_of_Delayed_Payment	Outstanding_Debt	Credit_History_Age
0	23	Scientist	19114.12	3	7	809.98	22 Years and 1 Months
1	23	Scientist	19114.12	-1	NaN	809.98	NaN
2	-500	Scientist	19114.12	3	7	809.98	22 Years and 3 Months
3	23	Scientist	19114.12	5	4	809.98	22 Years and 4 Months

Payment_of_Min_Amount	Total_EMI_per_month	Payment_Behaviour	Monthly_Balance	Credit_Score
No	49.574949	High_spent_Small_value_payments	312.49408867943663	Good
No	49.574949	Low_spent_Large_value_payments	284.62916249607184	Good
No	49.574949	Low_spent_Medium_value_payments	331.2098828537912	Good
No	49.574949	Low_spent_Small_value_payments	223.45130972736786	Good

02 Xử lý các dữ liệu NaN và dữ liệu chứa giá trị không chính xác

03 Sử dụng phương pháp IQR để loại bỏ các giá trị ngoại lệ.

IQR là sự khác biệt giữa tứ phân vị thứ nhất Q1 và tứ phân vị thứ ba Q3:

$$IQR=Q3-Q1$$

Age	0
Occupation	0
Annual_Income	0
Delay_from_due_date	0
Num_of_Delayed_Payment	7002
Outstanding_Debt	0
Credit_History_Age	9030
Payment_of_Min_Amount	0
Total_EMI_per_month	0
Payment_Behaviour	0
Monthly_Balance	1200
Credit_Score	0

Hình 1. Số lượng giá trị NaN trong các trường dữ liệu

04 Mã hóa dữ liệu

Chuyển đổi dữ liệu từ chuỗi thành số và thay đổi kiểu dữ liệu dữ liệu để dễ dàng phân tích hơn, tăng cường khả năng xử lý

Age	object
Occupation	object
Annual_Income	object
Delay_from_due_date	object
Num_of_Delayed_Payment	object
Outstanding_Debt	object
Credit_History_Age	object
Payment_of_Min_Amount	object
Total_EMI_per_month	object
Payment_Behaviour	object
Monthly_Balance	object
Credit_Score	object

Dữ liệu trước xử lý

Age	float64
Occupation	object
Annual_Income	float64
Delay_from_due_date	float64
Num_of_Delayed_Payment	float64
Outstanding_Debt	float64
Credit_History_Age	float64
Payment_of_Min_Amount	object
Total_EMI_per_month	float64
Payment_Behaviour	float64
Monthly_Balance	float64
Credit_Score	object

Dữ liệu sau xử lý

Hình 2. Các trường dữ liệu trước và sau khi thay đổi kiểu dữ liệu

05 Xử lý mất cân bằng dữ liệu

Sử dụng module Random Over Sampler xử lý sự mất cân bằng dữ liệu trong tập huấn luyện

Dữ liệu trước xử lý

Dữ liệu sau xử lý

Hình 3. Tỷ lệ các quan sát trong trường dữ liệu “Credit_Score” trước và sau khi cân bằng

06 Chuẩn hóa dữ liệu

Sử dụng phương pháp Min-Max Scaling để chuẩn hóa các biến số, chuẩn hóa được xác định theo công thức:

$$x_scaled = \frac{x - x_min}{x_max - x_min}$$

trong đó:

- x là giá trị ban đầu
- x_scaled là giá trị sau khi áp dụng Min-Max Scaling (nằm trong khoảng 0 đến 1)
- x_min, x_max lần lượt là giá trị nhỏ nhất, giá trị lớn nhất của trường dữ liệu trong tập dữ liệu.

Age	Annual Income	Delay from due date	Num of Delayed Payment	Outstanding Debt	Credit History Age	Total EMI per month	Monthly Balance
0.07031	0.083668	0.111111	0.002500	0.162020	0.67062	0.000602	1.0
0.10937	0.192389	0.166667	0.000909	0.121012	0.79228	0.000229	1.0
0.10937	0.192389	0.111111	0.000455	0.121012	0.79525	0.000229	1.0

Hình 4. Dữ liệu sau chuẩn hóa

07 Sử dụng phương pháp One-Hot Encoding để biến đổi các giá trị của cột nghề nghiệp “Occupation” thành các biến giả định dạng số.

Occupation_Mechanic	Occupation_Media_Manager	Occupation_Musician	Occupation_Scientist	Occupation_Teacher	Occupation_Writer
0	0	0	1	0	0
0	0	0	1	0	0
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	1	0

Hình 5. Dữ liệu sau khi sử dụng phương pháp One - Hot Encoding

08 Xác định dữ liệu tập huấn và kiểm tra

Bộ dữ liệu được chia theo tỷ lệ 80:20 với tập huấn luyện (80%) và tập kiểm tra (20%)

4.3. Các độ đo đánh giá hiệu năng

Phương pháp Accuracy (độ chính xác)

Cách đánh giá này đơn giản tính tỉ lệ giữa số mẫu dự đoán đúng và tổng số mẫu trong tập dữ liệu. Công thức:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Giả sử độ accuracy = 90% có nghĩa là trong số 100 mẫu thì có 90 mẫu được phân loại chính xác.

Phương pháp Precision

Precision được định nghĩa là tỷ lệ giữa số lượng mẫu được dự đoán khớp mẫu trên tổng dự đoán khớp mẫu và dự đoán đúng nhưng sai

$$\text{Precision} = \frac{TP}{TP + FP}$$

Phương pháp Recall

Recall được định nghĩa là tỷ lệ giữa số lượng mẫu được dự đoán khớp mẫu với tổng dự đoán khớp mẫu và dự đoán sai nhưng đúng

$$\text{Recall} = \frac{TP}{TP + FN}$$

Precision cao đồng nghĩa độ chính xác của các mẫu đúng là cao. Recall cao đồng nghĩa với việc bỏ sót các mẫu thực sự positive là thấp. Một mô hình phân lớp tốt là mô hình có cả Precision và Recall đều cao, tức càng gần một càng tốt.

Phương pháp F1 - Score

F-1 score là trung bình điều hòa (harmonic mean) của precision và recall, có tầm quan trọng tương tự như với FNs và FPs. Cụ thể:

$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

F-1 score càng cao tương ứng precision và recall càng cao, mô hình phân loại càng tốt.

4.4. Các tham số và môi trường cài đặt

Môi trường Phần mềm

Ngôn Ngữ Lập Trình: Sử dụng Python trong môi trường Jupyter Notebook, một công cụ phổ biến trong khoa học dữ liệu cho phép thực hiện mã, ghi chú và hiển thị kết quả một cách tương tác.

Thư Viện Sử Dụng

- pandas và numpy cho việc xử lý và phân tích dữ liệu.
- matplotlib và seaborn cho việc trực quan hóa dữ liệu.
- sklearn cho các mô hình học máy như Decision Tree, KNN, Random Forest, SVM và Adaboost, cũng như các công cụ tiền xử lý dữ liệu như MinMaxScaler.
- tensorflow.keras được sử dụng cho việc xây dựng và huấn luyện mô hình mạng nơ-ron tích chập (CNN) và mạng nơ-ron hồi quy (RNN).

4.5. Các phương pháp cơ sở

Thực hiện so sánh 7 mô hình dưới cùng một điều kiện thí nghiệm để đảm bảo công bằng, tính hiệu quả và chính xác trong so sánh:

Phương pháp phân lớp: Decision Tree, K-Nearest Neighbor, Random Forest, XGBoost, Naive Bayes

Mô hình học sâu: Convolutional Neural Network, Recurrent Neural Network

Các thí nghiệm được thực hiện trên cùng một tập dữ liệu, sử dụng cùng một quy trình tiền xử lý và chia dữ liệu để đảm bảo tính khách quan. Mỗi phương pháp được đánh giá dựa trên các chỉ số như accuracy, precision, recall, và F1-score để có cái nhìn toàn diện về hiệu suất của chúng.

4.6. Kết quả và phân tích

- Các kỹ thuật học máy truyền thống
- Các kỹ thuật học sâu
- So sánh các mô hình

Các kỹ thuật học máy truyền thống

Thử nghiệm các mô hình học máy truyền thống: Decision Tree (Cây quyết định), K-Nearest Neighbor (K- láng giềng gần nhất), Random Forest (Rừng ngẫu nhiên), XGBoost, Naive Bayes và có kết quả được trình bày trong bảng dưới đây:

	Model	Accuracy	Precision	Recall	F1-score
1	**Decision Tree**	0.855382	0.854924	0.855382	0.851688
2	**K-Nearest Neighbor**	0.712090	0.707868	0.712090	0.702121
3	**Random Forest**	**0.885779**	**0.887773**	**0.885779**	**0.882678**
4	**XGBoost**	0.835985	0.837708	0.835985	0.830886
5	**Naive Bayes**	0.617953	0.611433	0.617953	0.585627

Kết quả đánh giá các mô hình học máy truyền thống

Qua bảng phân tích của các mô hình học máy truyền thống khi sử dụng lần lượt các chỉ số đánh giá hiệu suất Accuracy, Precision, Recall, F1-score và kết quả cho thấy mô hình Random Forest (Rừng ngẫu nhiên) có hiệu suất đạt tốt nhất khi các chỉ số đều đạt trên 88%, cao hơn các mô hình còn lại khoảng từ 0-30% tùy vào từng phương pháp so sánh. Bên cạnh đó, Naive Bayes và K-Nearest Neighbor hai mô hình có hiệu suất được nhìn nhận là thấp nhất trong bảng phân tích với kết quả lần lượt là hơn 60% và hơn 70% ở cả bốn chỉ số đánh giá.

	Model	Accuracy	Precision	Recall	F1-score
0	Decision Tree	0.855382	0.854924	0.855382	0.851688
1	K-Nearest Neighbor	0.712090	0.707868	0.712090	0.702121
2	Random Forest	0.885779	0.887773	0.885779	0.882678
3	XGBoost	0.835985	0.837708	0.835985	0.830886
4	Naive Bayes	0.617953	0.611433	0.617953	0.585627

Các kỹ thuật học sâu

Xây dựng mô hình Convolutional Neural Network (Mạng neural tích chập sâu) và Recurrent Neural Network (Mạng nơ ron hồi quy), thu được kết quả sau:

	Model	Accuracy	Precision	Recall	F1-score	Thời gian
1	**CNN**	0.686358	0.679902	0.686358	0.677860	**1 phút 56s**
2	**RNN**	0.675702	0.671962	0.675702	0.663117	**10 phút 43s**

Kết quả đánh giá các mô hình học sâu

Hai mô hình học sâu đưa ra kết quả là hiệu suất ở cả 2 phương pháp đánh giá xấp xỉ nhau. Tuy nhiên, mô hình Convolutional Neural Network đã có mức % nhỉnh hơn một chút so với mô hình còn lại ví dụ như Accuracy của CNN chỉ cao hơn 1% so với RNN. Quan sát kết quả thu được có thể thấy RNN đưa ra kết quả không cao và thời gian thực nghiệm dài hơn CNN.

So sánh các mô hình

	Model	Accuracy	Precision	Recall	F1-score
1	**DT**	0.855382	0.854924	0.855382	0.851688
2	**KNN**	0.712090	0.707868	0.712090	0.702121
3	**RF**	**0.885779**	**0.887773**	**0.885779**	**0.882678**
4	**XGB**	0.835985	0.837708	0.835985	0.830886
5	**NB**	0.617953	0.611433	0.617953	0.585627
7	**CNN**	0.686358	0.679902	0.686358	0.677860
8	**RNN**	0.675702	0.671962	0.675702	0.663117

Dựa vào bảng tổng hợp các chỉ số đánh giá hiệu suất của các mô hình trên, ta nhận thấy rằng mô hình Random Forest có tỷ lệ hiệu suất cao nhất trong 7 mô hình còn lại lần lượt với tất cả các chỉ số đánh giá đều trên 88%.

Thông qua biểu đồ trên ta nhận thấy rằng các phương pháp đánh giá trên mô hình RF có hiệu suất cao nhất (khoảng 89%), tiếp đó là mô hình DT với hiệu suất chỉ kém 2-3%. Ngoài ra, mô hình XGB có hiệu suất cao thứ ba (khoảng 84%). Với phương pháp đánh giá hiệu suất bằng 4 chỉ số accuracy, precision, recall, f1-score thì mô hình RNN có kết quả hiệu suất là thấp nhất (trên 60%).

V - Kết luận

Bài báo cáo thực nghiệm so sánh nhiều cách tiếp cận khác nhau của mô hình chấm điểm tín dụng. Việc đánh giá trên nhiều mô hình lựa chọn ra mô hình có khả năng chấm điểm tín dụng chính xác nhất. Các thực nghiệm đã chỉ ra rằng mô hình Random Forest được huấn luyện vượt trội hơn các mô hình khác với các chỉ số đánh giá hiệu năng accury, precision, recall, f1-score đều trên 88%. Đối với phương pháp xây dựng tính năng, mỗi phương pháp thể hiện thế mạnh của mình khi được đào tạo bởi các mô hình khác nhau.

THANK YOU

for listening