Machine Learning on Bank Marketing Dataset

1st Đoàn Bảo Long
UIT 21520332
KHDL2021
baolongvncom@gmail.com

2nd Nguyễn Thiện Trí
UIT 21522707
KHDL2021
21522707@gm.uit.edu.vn

3rd Nguyễn Ngọc Lương
UIT 21522311
KHDL2021
21522311@gm.uit.edu.vn

***Tóm tắt nội dung*—Trong lĩnh vực tiếp thị ngân hàng, việc xác định nhóm các đối tượng cần hướng tới là một vấn đề quan trọng. Mục tiêu của chúng em trong bài báo cáo này là áp dụng các phương pháp máy học để xây dựng một mô hình dự đoán khả năng mà một khách hàng đăng kí gửi tiền dài hạn dựa trên bộ Bank Marketing Dataset. Với mô hình này, ta có thể thu gọn được nhóm đối tượng cần hướng đến, từ đó đưa ra phương pháp tiếp thị thích hợp.**

***Index Terms*—preprocessing, machine learning, Data Analysis, Customer Segmentation**

I. GIỚI THIỆU

A. Bộ dữ liệu

Bộ dữ liệu Bank Marketing Dataset được tạo ra bởi Sérgio Moro (ISCTE-IUL), Paulo Cortez (Univ. Minho) và Paulo Rita (ISCTE-IUL) @ 2011 với mục đích nghiên cứu trên lĩnh vực tiếp thị ngân hàng. Bộ dữ liệu được lấy từ trang https://archive.ics.uci.edu/dataset/222/bank+marketing.

Bộ dữ liệu có hơn 450000 mẫu và đã được làm sạch. Với mỗi mẫu trong bộ dữ liệu là thông tin về cuộc gọi tiếp thị và thông tin khách hàng được tiếp thị qua các chiến dịch tiếp thị của một ngân hàng ở Bồ Đào Nha nhằm bán các gói kí gửi dài hạn của ngân hàng (Là việc một khách hàng gửi một số tiền nhất định có kì hạn vào ngân hàng để nhận lãi suất). Nó gồm các thông tin như độ tuổi, giới tính, độ dài cuộc gọi tiếp thị, chiến dịch... và thông tin khách hàng có đăng kí gói kí gửi dài hạn hay không.

Bộ dữ liệu Bank Marketing Dataset được sử dụng rộng rãi trong các nghiên cứu về Machine Learning để xây dựng các mô hình dự đoán và khoa học dữ liệu, đóng góp quan trọng trong việc nghiên cứu và phát triển các chiến lược tiếp thị hiệu quả cho các ngân hàng và các doanh nghiệp khác nhau.

B. Vấn đề nghiên cứu

Vấn đề nghiên cứu là xây dựng một mô hình để dự đoán việc một khách hàng có đăng kí gói kí gửi dài hạn hay không. Việc này có thể tăng hiệu quả cho các chiến dịch tiếp thị bằng cách xác định được những khách hàng có tiềm năng cao sẽ mua gói kí gửi dài hạn, từ đó giúp bên tiếp thị xác định được nhóm đối tượng khách hàng cần hướng đến. Điều này cũng giúp ngân hàng hiểu rõ hơn về khách hàng của mình, từ đó phát triển các chiến lược tiếp thị, các gói kí gửi sau này một cách phù hợp hơn.

C. Mục tiêu của bài báo cáo

Mục tiêu chính của bài báo cáo là phân tích bộ dữ liệu Bank Marketing và báo cáo kết quả của các Mô hình học máy mà nhóm thử nghiệm, đồng thời đánh giá và nhận xét mỗi mô hình.

II. PHÂN TÍCH DỮ LIỆU

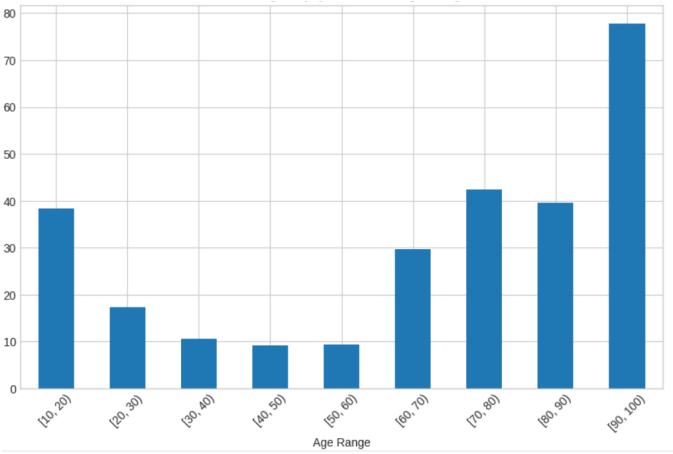
A. Mô tả dữ liệu và các thuộc tính

Bộ dữ liệu chứa 45211 dòng và bao gồm 17 thuộc tính, trong đó có một số thuộc tính quan trọng sau:

- **age**: Độ tuổi của khách hàng (kiểu số).
- **job**: Nghề nghiệp của khách hàng (thuộc tính phân loại, ví dụ: management, technician, ...).
- **marital**: Tình trạng hôn nhân của khách hàng (thuộc tính phân loại, ví dụ: married, single, divorced, ...).
- **education**: Trình độ giáo dục của khách hàng (thuộc tính phân loại, ví dụ: tertiary, secondary, ...).
- **balance**: Số dư trung bình hằng năm trong tài khoản của khách hàng (kiểu số).
- **housing**: Có khoản nợ mua nhà hay không (thuộc tính phân loại, ví dụ: yes, no).
- **contact**: Phương thức liên lạc với khách hàng (thuộc tính phân loại, ví dụ: telephone, cellular, ...).
- **month**: Tháng khi lần liên lạc cuối cùng diễn ra (thuộc tính phân loại, ví dụ: jan, feb, mar, ...).
- **duration**: Thời lượng của cuộc gọi cuối cùng với khách hàng (kiểu số).
- **campaign**: Số lượng cuộc gọi đã thực hiện đối với khách hàng trong chiến dịch tiếp thị này (kiểu số).
- **y**: Kết quả mục tiêu, liệu khách hàng có mở tài khoản gửi tiền gửi hay không (thuộc tính phân loại, ví dụ: yes, no).

B. Khám phá dữ liệu và phân tích đặc trưng

Sau khi vẽ biểu đồ và tiến hành tính toán, quan sát các giá trị như giá trị nhỏ nhất, lớn nhất, trung bình và phân vị cho biến liên tục và số lượng, phân bố, đặc điểm của các nhãn, nhóm rút ra được một số phân tích quan trọng sau về bộ dữ liệu:



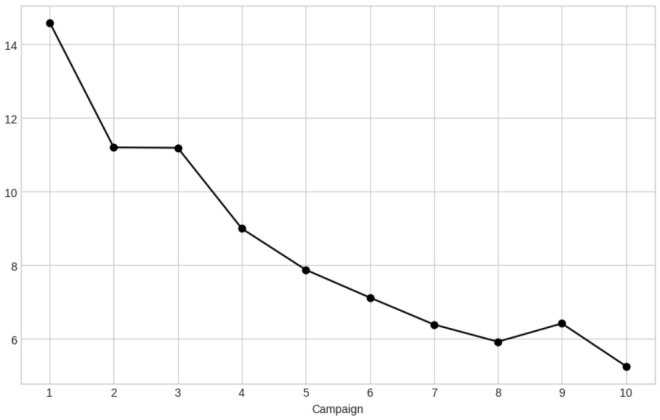
Hình 1. Phần trăm số khách hàng đăng ký theo từng nhóm tuổi

Biểu đồ trên cho thấy tỷ lệ các khách hàng ở nhóm tuổi từ 10 đến 20 và từ 60 đến 100 chiếm đa số trong tập khách hàng đăng ký tiền gửi có kỳ hạn. Vì vậy phía ngân hàng nên cân nhắc tập trung vào 2 nhóm đối tượng này nhiều hơn trong chiến dịch tiếp thị tiếp theo.

```
age          0.025155
balance      0.052838
day         -0.028348
duration     0.394521
campaign    -0.073172
pdays        0.103621
previous     0.093236
y            1.000000
Name: y, dtype: float64
```

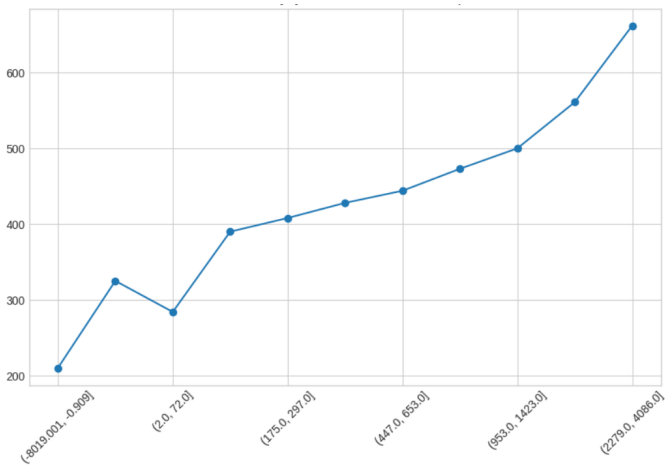
Hình 2. Chỉ số tương quan của các thuộc tính với biến mục tiêu **y**

Từ hình 2 ta nhận thấy rằng thuộc tính duration có tương quan khá cao đối với biến mục tiêu. Giá trị duration được ghi lại sau khi cuộc gọi được thực hiện cho khách hàng. Có thể giải thích lý do tại sao duration có tương quan cao với việc mở một khoản tiền gửi có kỳ hạn là vì khi người tiếp thị bên ngân hàng nói chuyện với khách hàng càng lâu thì khả năng khách hàng mục tiêu mở một khoản tiền gửi có kỳ hạn sẽ càng cao (nghĩa là khách hàng sẽ càng có hứng thú đối với chiến dịch marketing từ ngân hàng).



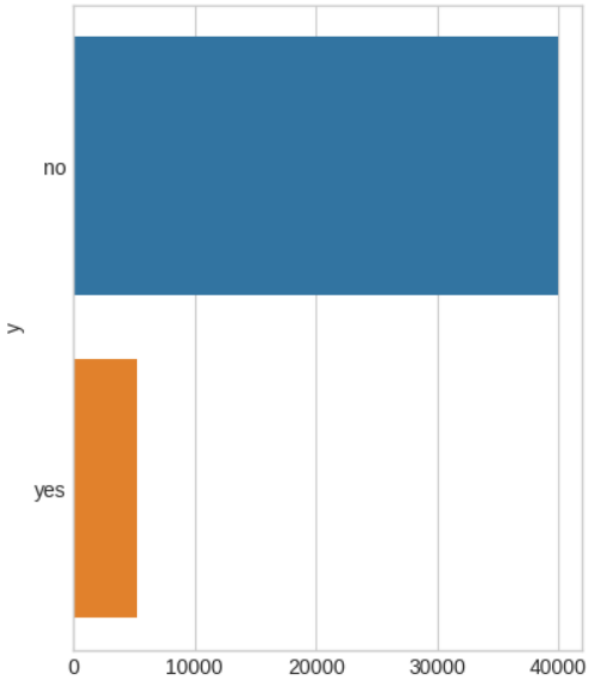
Hình 3. Phần trăm số khách hàng đăng ký theo số lần liên lạc

Ngân hàng cũng nên triển khai một chiến lược mới quy định rằng không nên thực hiện quá 3 cuộc gọi cho cùng một khách hàng để tiết kiệm thời gian và công sức trong việc có được khách hàng đăng ký mới. Nếu người tiếp thị càng gọi nhiều lần cho cùng một khách hàng, thì khả năng họ từ chối đăng ký sẽ càng cao.



Hình 4. Số lượng khách hàng đăng ký dựa trên số dư trong tài khoản

Ta nhận thấy số lượng khách hàng đăng ký tiền gửi tỷ lệ với số dư trung bình có trong tài khoản ngân hàng của họ, đây cũng là một hướng tiếp cận cho ngân hàng khi cần phải chọn ra những khách hàng tiềm năng cho chiến dịch tiếp thị tiếp theo.



Hình 5. Phân bố của 2 nhãn yes và no trong bộ dữ liệu

- Bộ dữ liệu này có sự mất cân bằng cao ở biến mục tiêu, ở đây số lượng mẫu có giá trị biến mục tiêu là 'yes' ít hơn 'no' khi chỉ có khoảng 5000 giá trị 'yes' so với khoảng 40000 giá trị 'no'. Không khó hiểu khi trong thực tế số người đăng ký gửi tiền có kỳ hạn luôn thấp hơn nhiều so với số người không đăng ký. Vì vậy chúng ta sẽ cần phải xử lý việc cân bằng lại bộ dữ liệu trước khi đưa vào các mô hình máy học.

III. Cân bằng dữ liệu

Đây là một bước quan trọng trọng việc xây dựng mô hình dự đoán người sẽ đăng kí gói kí gửi dài hạn của ngân hàng vì tỉ lệ người đăng kí gói kí gửi (nhãn 'yes') sẽ thấp hơn rất nhiều so người không đăng kí (nhãn 'no). Nếu không cân bằng dữ liệu, mô hình sẽ có xu hướng học quá tập trung vào lớp đa số (nhãn 'no'), dẫn đến việc dự đoán nhãn 'yes' gặp khó khắn. Vì trong lĩnh vực tiếp thị ngân hàng, ta cần cố gắng để không dự đoán nhầm những người trên thực tế sẽ đăng kí gói kí gửi. Cho nên chúng ta cần phải cân bằng lại bộ dữ liệu được dùng để huấn luyện. Các phương pháp cân bằng dữ liệu sẽ được dùng bao gồm:

A. Random Undersampling

Random Undersampling là một phương pháp cân bằng dữ liệu trong đó các mẫu từ lớp đa số sẽ được loại bỏ ngẫu nhiên để giảm tỷ lệ mẫu của lớp đa số xuống cùng với lớp thiểu số.

B. SMOTE

SMOTE (Synthetic Minority Over-sampling Technique): Phương pháp này tạo ra các mẫu nhân tạo trong lớp thiểu số bằng cách kết hợp các mẫu gần nhất trong cùng một lớp. Thuật toán cân bằng dữ liệu bằng SMOTE:

1) Xác định lớp thiểu số và lớp đa số trong tập huấn luyện.
2) Chọn một mẫu ngẫu nhiên từ lớp thiểu số.
3) Tìm k-nearest neighbors (k là một tham số đã được định trước) của mẫu đó trong cùng lớp. K-nearest neighbors là các mẫu gần nhất với mẫu đã chọn dựa trên khoảng cách uclidean hoặc khoảng cách Mahalanobis.
4) Chọn một trong các neighbors ngẫu nhiên và tạo ra một mẫu nhân tạo mới. Việc tạo ra mẫu nhân tạo được thực hiện bằng cách lấy một tỷ lệ ngẫu nhiên (thường từ 0 đến 1) và thêm nó vào đại diện của mẫu đã chọn và neighbor đã chọn.
5) Lặp lại các bước 2-4 cho đến khi số lượng mẫu nhân tạo mong muốn đã được tạo ra.

C. ADASYN

ADASYN (Adaptive Synthetic Sampling) cũng là một phương pháp cân bằng dữ liệu giống SMOTE trong đó mẫu từ lớp thiểu số được tạo ra thông qua việc tổng hợp tự động (synthetic samples) dựa trên việc xác định các khu vực hiệu quả để tạo ra mẫu mới.

ADASYN là một biến thể của SMOTE. Khác với SMOTE, ADASYN tạo ra các mẫu nhân tạo không chỉ dựa trên số lượng mẫu gần nhất trong cùng một lớp, mà còn dựa trên độ lệch của mẫu đó. Nghĩa là, ADASYN tăng cường lấy ý tưởng của SMOTE bằng cách tạo ra các mẫu nhân tạo nhiều hơn cho các vùng mẫu thiểu số mà gặp khó khăn hơn trong việc phân loại đúng.

Nhóm sẽ thử nghiệm mỗi phương pháp cân bằng dữ liệu với một số mô hình máy học khác nhau sau đó thống kê kết quả, đánh giá và nhận xét các kết quả thu được.

IV. Các phương pháp máy học được áp dụng

A. Logistic Regression

Mô hình Logistic Regression (hồi quy logistic) là một mô hình thống kê phân loại được sử dụng để dự đoán xác suất của một biến phụ thuộc nhị phân dựa trên các biến độc lập. Nó là một trong những phương pháp phân loại đơn giản và phổ biến nhất trong machine learning và thống kê.

1) Lí do mô hình được chọn: Các điểm mạnh của mô hình Logistic Regression đối với bộ dữ liệu Bank Marketing được phân tích như sau:

- Mô hình Logistic Regression có hiệu suất tốt với các bộ dữ liệu lớn và đa dạng về các thuộc tính đầu vào.
- Mô hình rất thích hợp trong việc xử lí các biến phân loại và biến liên tục.
- Một điểm mạnh lớn của mô hình Logistic Regression là nó cho phép ta giải thích hiểu rõ được tác động của các biến đầu vào đến kết quả dự đoán. Kết quả của mô hình có thể được giải thích bằng cách xem xét hệ số của các biến đầu vào. Điều này giúp ta có thể hiểu rõ hơn về quy luật và mức độ tác động của các yếu tố ảnh hưởng tới đăng kí gói kí gửi dài hạn của khách hàng.

2) Công thức: Logistic Regression mô hình hóa mối quan hệ giữa các đặc trưng đầu vào và xác suất của lớp tích cực bằng hàm logistic (còn được gọi là hàm sigmoid). Hàm logistic được định nghĩa như sau:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

trong đó:

$h_\theta(x)$: xác suất dự đoán của $y = 1$ dựa trên x với tham số θ

θ : vector tham số của mô hình

x : vector đặc trưng đầu vào

3) Hàm mất mát: Để ước lượng các tham số tối ưu θ, chúng ta cần định nghĩa một hàm mất mát để đo lường sự khác biệt giữa xác suất dự đoán và nhãn lớp thực tế. Hàm mất mát cho Logistic Regression còn được gọi là hàm mất mát cross-entropy. Nó được định nghĩa như sau:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log(h_\theta(x^{(i)})) + (1 - y^{(i)})\log(1 - h_\theta(x^{(i)}))\right]$$

trong đó:

$J(\theta)$: hàm mất mát

$y^{(i)}$: nhãn lớp thực tế của mẫu thứ i

$x^{(i)}$: đặc trưng đầu vào của mẫu thứ i

Hàm này đảm bảo rằng giá trị mất mát cao hơn khi mô hình dự đoán sai lớp của mẫu và thấp hơn khi mô hình dự đoán đúng.

4) Tối ưu: Mục tiêu của Logistic Regression là tối thiểu hóa hàm mất mát $J(\theta)$ để tìm ra các giá trị tham số tối ưu θ. Nhóm chọn sử dụng thuật toán tối ưu gradient descent.

Bằng cách lấy đạo hàm của hàm mất mát theo từng tham số θ_j, chúng ta có thể cập nhật các tham số theo từng bước cho đến khi hàm hội tụ. Quy tắc cập nhật cho gradient descent là:

$$\theta_j := \theta_j - \alpha\frac{\partial J(\theta)}{\partial \theta_j}$$

trong đó tham số α xác định tốc độ và độ lớn của việc cập nhật các tham số mô hình trong thuật toán tối ưu hóa.

B. Naive Bayes Classifier

1) Bộ phân lớp Naive Bayes: Bộ phân lớp Naive Bayes là một tập hợp các thuật toán học có giám sát dựa trên việc sử dụng định lý bayes với giả định về sự độc lập mạnh mẽ giữa các đối tượng.

Định lý Bayes phát biểu mối quan hệ sau, cho lớp y và vectơ đặc trưng phụ thuộc từ x_1 đến x_n:

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n \mid y)}{P(x_1, \ldots, x_n)}$$

Giả định ngây thơ: mọi đối tượng đều độc lập với nhau.

$$P(x_i \mid y, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = P(x_i)$$

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)}$$

Bởi vì P(x_1)P(x_2) ... P(x_n) không đổi, ta có thể sử dụng công thức sau:

$$P(y \mid x_1, \ldots, x_n) \propto P(y)\prod_{i=1}^{n} P(x_i \mid y)$$

$$\Downarrow$$

$$\hat{y} = \arg\max_y P(y)\prod_{i=1}^{n} P(x_i \mid y),$$

Ta cần tìm xác suất của bộ đầu vào với tất cả các giá trị của biến y sao cho đầu ra có xác suất tối đa.

Naive Bayes Classifer đã được sử dụng rộng rãi trong các bài toán phân loại dữ liệu, ví dụ như phân loại văn bản, phân loại hình ảnh và các bài toán phân loại khác. Thuật toán này là một trong những thuật toán đơn giản nhưng hiệu quả trong Machine Learning. Có ba loại mô hình Naive Bayes phổ biến trong Machine Learning:

- Multinomial Naive Bayes: Mô hình này được sử dụng chủ yếu cho việc phân loại văn bản, bởi vì nó giả định rằng sự xuất hiện của một từ trong văn bản phụ thuộc vào số lần từ đó xuất hiện trong văn bản. Ví dụ, khi phân loại email là "spam"hay "không phải spam", sử dụng mô hình Multinomial Naive Bayes các thuộc tính sử dụng được là số lần xuất hiện của mỗi từ trong email.
- Gaussian Naive Bayes: Mô hình này được sử dụng chủ yếu trong việc phân loại các tập dữ liệu có giá trị số liên tục. Giả định của mô hình này là giá trị của thuộc tính trong một lớp có phân phối chuẩn. Ví dụ, một bài toán đơn giản có thể là phân loại loại hoa dựa trên các thuộc tính như chiều dài, chiều rộng của cánh hoa.
- Bernoulli Naive Bayes: Mô hình này giống như Multinomial Naive Bayes, nhưng nó được sử dụng cho các tập dữ liệu có giá trị rời rạc và nhị phân. Ví dụ, một bài toán đơn giản có thể là phân loại các email là "spam"hoặc "không phải spam"dựa trên việc sử dụng thuộc tính có mặt hoặc không có mặt của từ khóa trong email.

Trong bài toán này, chúng ta sẽ sử dụng mô hình Gaussian Naive Bayes.

C. RandomForest Classifier

RandomForestClassifier là một mô hình máy học phổ biến được sử dụng cho cả nhiệm vụ phân loại và hồi quy. Nó thuộc họ thuật toán ensemble learning và kết hợp các dự đoán của nhiều cây quyết định để đưa ra các dự đoán chính xác và mạnh mẽ hơn.

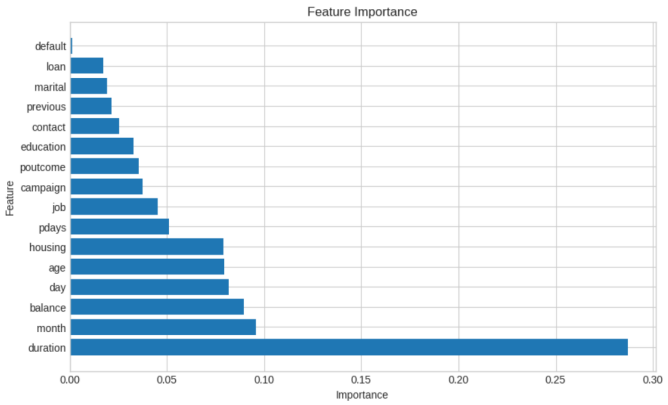
Tổng quan cách hoạt động của bộ phân lớp RandomForestClassifier:

1) Tạo "rừng ngẫu nhiên": RandomForestClassifier tạo ra một tập hợp các cây quyết định. Mỗi cây được xây dựng độc lập bằng cách sử dụng một tập con ngẫu nhiên của dữ liệu huấn luyện. Lấy mẫu ngẫu nhiên được thực hiện cả về điểm dữ liệu và thuộc tính. Quá trình này được gọi là "bagging"và giúp làm tăng tính đa dạng giữa các cây.
2) Xây dựng cây quyết định: Tương tự như cách xây dựng cây quyết định như đã trình bày ở trên.

3) Dự đoán kết hợp: Khi tất cả các cây quyết định được xây dựng, các dự đoán được thực hiện bởi từng cây riêng lẻ. Đối với các nhiệm vụ phân loại, lớp có đa số phiếu bầu trong số các cây sẽ được chọn làm dự đoán cuối cùng.

RandomForestClassifier có một số ưu điểm nổi bật, bao gồm:

1) Có độ chính xác cao nhờ kết hợp được cùng lúc nhiều cây quyết định với nhau, sự giảm phương sai là một trong những lý do chính khiến các RandomForest thường hoạt động tốt hơn các cây quyết định riêng lẻ.
2) Xử lý tốt dữ liệu nhiều chiều.
3) Thuật toán có thể xử lý cả các thuộc tính categorical và numerical mà không yêu cầu feature scaling.
4) Có thể cung cấp các ước tính về tầm quan trọng của các thuộc tính, giúp lựa chọn thuộc tính có tầm ảnh hưởng cao đến biến mục tiêu.



Hình 6. Feature Importance

Kết quả feature importance do bộ phân lớp RandomForest cho thấy thuộc tính **duration** có độ quan trọng cao đến quyết định có đăng ký hay không của một khách hàng, kết quả này khớp với kết quả từ bảng chỉ số tương quan của các thuộc tính đã được trình bày ở mục phân tích dữ liệu.

D. Catboost Classifier

CatBoost là một thuật toán gradient boosting mạnh mẽ, chuyên xử lý các bài toán phân loại trong các tác vụ học máy. Nó là một framework học máy mã nguồn mở do Yandex phát triển và có sẵn để sử dụng trong Python, R và các giao diện command-line.

Bộ phân loại CatBoost được thiết kế đặc biệt cho các vấn đề phân loại và được xây dựng trên nguyên tắc gradient-boosting. Dưới đây là một số ưu điểm chính của bộ phân lớp CatBoost:

- Tự động xử lý các thuộc tính categorical: CatBoost có thể xử lý các thuộc tính categorical một cách hiệu quả mà không cần mã hóa thủ công hoặc mã hóa one-hot.
- Kỹ thuật gradient-boosting: CatBoostClassifier kết hợp nhiều mô hình cây quyết định với nhau để tạo ra một mô hình dự đoán mạnh mẽ. Trong đó thuật toán thực hiện lặp đi lặp lại việc xây dựng các cây quyết định, các cây sau sửa lỗi của các cây trước đó, dần dần cải thiện độ chính xác của mô hình dự đoán một cách tổng thể.
- Tự động xử lý missing values: CatBoostClassifier có thể tự động xử lý các giá trị bị thiếu, bộ phân lớp cho phép chúng ta huấn luyện mô hình một cách trực tiếp trên bộ dữ liệu có missing values.
- Hỗ trợ tăng tốc GPU: CatBoost cung cấp khả năng tăng tốc GPU, cho phép giảm thời gian training và dự đoán nhanh hơn khi sử dụng phần cứng tương thích.

V. Đánh giá kết quả

Nhóm đề xuất sử dụng 4 thuật toán Machine Learning là Logistic Regression, Naive Bayes, Random Forest và CatBoost kết hợp với các phương pháp cân bằng dữ liệu SMOTE, ADASYN, Undersampling. Hai thước đo được sử dụng để đánh giá mô hình là precision và recall được tính theo các công thức (1), (2).

$$Precision = \frac{TruePositive}{FalsePositive + TruePositive} \quad (1)$$

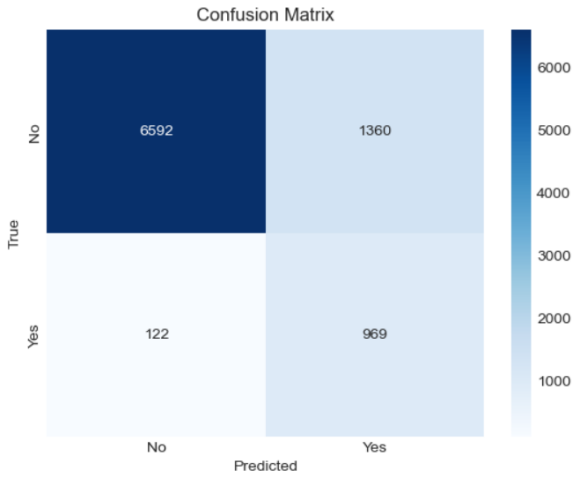
$$Recall = \frac{TruePositive}{FalseNegative + TruePositive} \quad (2)$$

Tiến hành dự đoán trên tập test gồm 9043 mẫu, trong đó có 7952 nhãn 'No' và 1091 nhãn 'Yes'. Kết quả được thu lại vào bảng 1:

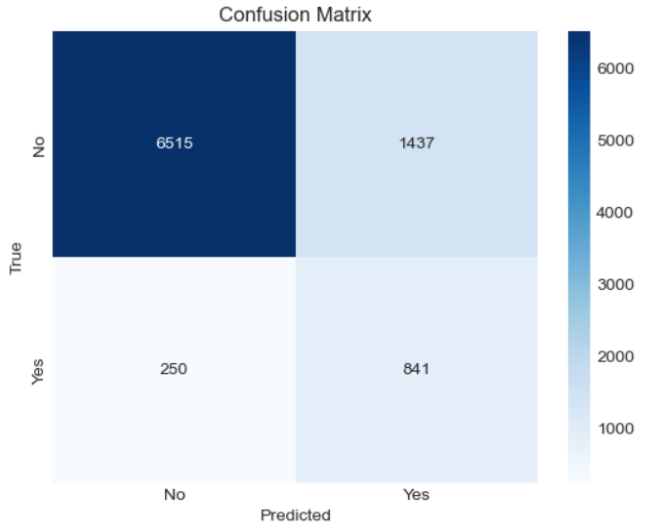
Bảng I
KẾT QUẢ CỦA CÁC MÔ HÌNH

	Logistic Regression		Naive Bayes		Random Forest		CatBoost	
	precision	recall	precision	recall	precision	recall	precision	recall
UnderSampling	0.37	0.77	0.27	0.82	0.42	0.89	0.42	0.88
SMOTE	0.36	0.70	0.22	0.80	0.50	0.67	0.46	0.77
ADASYN	0.34	0.73	0.21	0.81	0.49	0.68	0.44	0.77

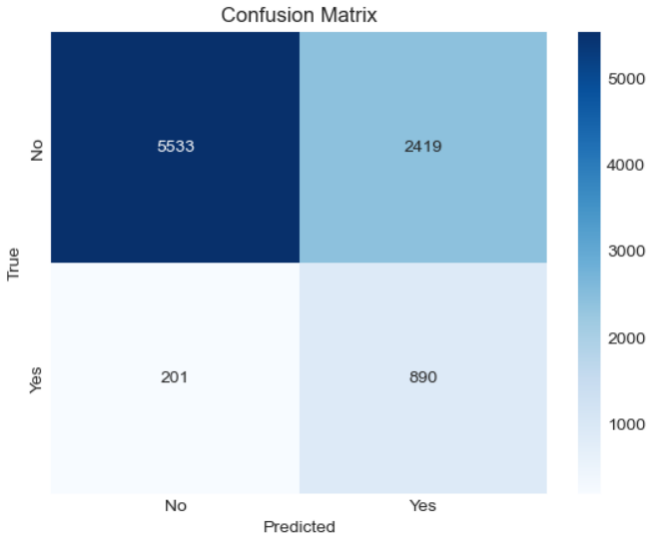
Từ bảng 1 ta có thể thấy phương pháp cân bằng dữ liệu Undersmapling mang lại kết quả tốt hơn so với 2 phương pháp còn lại với độ phủ lên tới gần 90%. Trong đó, mô hình máy học Random Forest đạt kết quả cao nhất với độ phủ 90% và độ chính xác 42%.



Hình 7. Ma trận nhầm lẫn của mô hình Random Forest

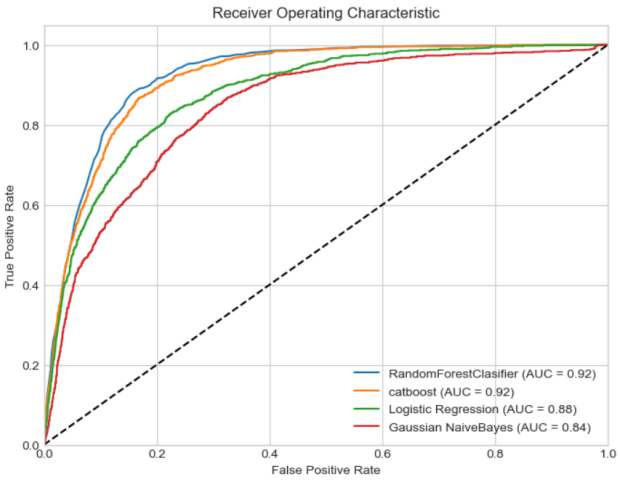


Hình 8. Ma trận nhầm lẫn của mô hình Logistic Regression



Hình 9. Ma trận nhầm lẫn của mô hình Naive Bayes

Qua các hình 7, 8 và 9 ta có thể thấy rằng các mô hình đã phân loại chính xác hầu hết các trường hợp có nhãn 'Yes'. Điều này là rất quan trọng bởi đây chính là tệp khách hàng mà các ngân hàng nhắm đến. Các mô hình có precision rơi vào khoảng 30% - 40%, có nghĩa cứ 3 khách hàng được phân loại 'Yes' thì có 1 khách hàng thực sự đăng kí gói kí gửi dài hạn. Theo đánh giá của nhóm, kết quả phân loại cho tập test trong trường hợp này là một kết quả tốt.



Hình 10. Đường cong ROC

VI. Kết luận

Tổng kết lại, báo cáo đã trình bày những nỗ lực và thành quả trong việc áp dụng các phương pháp học máy để dự đoán khách hàng đăng kí gói kí gửi dài hạn.

Các phương pháp đề xuất hoạt động rất hiệu quả, có độ chính xác cao. Tuy nhiên, việc áp dụng các kết quả này vào trong thực tế cần được thận trọng và cân nhắc.

Trong tương lai, nhóm sẽ tiếp tục nghiên cứu và phát triển các phương pháp học máy mới để nâng cao độ chính xác và độ tin cậy của mô hình. Chúng tôi cũng hy vọng rằng nghiên cứu của nhóm sẽ giúp ích cho những công trình dự đoán khác và tìm kiếm những ứng dụng mới của học máy trong lĩnh vực tài chính-ngân hàng.

Tài liệu

[1] [Moro et al., 2011] S. Moro, R. Laureano and P. Cortez. Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology.

[2] In P. Novais et al. (Eds.), Proceedings of the European Simulation and Modelling Conference - ESM'2011, pp. 117-121, Guimarães, Portugal, October 2011. EUROSIS.