Sử dụng thuận toán máy học cơ bản dự đoán khả năng rời bỏ ngân hàng của khách hàng

Nguyễn Tuấn Anh , Nguyễn Hoàng Anh, Bạch Quang Tùng

15/04/2025

Giới thiệu về vấn đề

Tổng quan về vấn đề

Dự đoán khách hàng rời bỏ (customer churn) là một bài toán quan trọng trong ngành ngân hàng, nhằm tối ưu hóa việc giữ chân khách hàng và nâng cao hiệu quả kinh doanh. Tập dữ liệu được sử dụng trong báo cáo này chứa thông tin về khách hàng của một ngân hàng Hoa Kỳ, bao gồm các đặc điểm nhân khẩu học và hành vi, với mục tiêu xác định liệu một khách hàng cụ thể có rời khỏi ngân hàng hay không.

Bằng cách áp dụng các thuật toán học máy cơ bản, phân tích này sẽ khám phá các yếu tố ảnh hưởng đến quyết định rời bỏ của khách hàng, từ đó xây dựng mô hình dự đoán chính xác và cung cấp thông tin hữu ích cho các chiến lược giữ chân khách hàng. R và RMarkdown được sử dụng để đảm bảo quá trình phân tích minh bạch, tái tạo và dễ hiểu.

Tầm quan trọng của vấn đề

Việc dự đoán khách hàng rời bỏ không chỉ đơn thuần là một bài toán kỹ thuật mà còn mang ý nghĩa chiến lược đối với các tổ chức tài chính. Khách hàng rời bỏ có thể gây ra tổn thất lớn về doanh thu, làm tăng chi phí tìm kiếm khách hàng mới và ảnh hưởng đến uy tín thương hiệu. Hơn nữa, việc hiểu rõ nguyên nhân dẫn đến hành vi rời bỏ chẳng hạn như chất lượng dịch vụ kém, lãi suất không cạnh tranh, hoặc trải nghiệm khách hàng không tốt giúp ngân hàng kịp thời điều chỉnh các chính sách và cải thiện dịch vụ.

Phân tích dữ liệu khách hàng bằng các công cụ học máy cho phép ngân hàng xác định các nhóm khách hàng có nguy cơ rời bỏ cao, từ đó triển khai các biện pháp can thiệp cá nhân hóa, như ưu đãi đặc biệt hoặc cải thiện tương tác. Trong bối cảnh cạnh tranh ngày càng gay gắt giữa các ngân hàng, việc tận dụng dữ liệu để dự đoán và giảm thiểu tỷ lệ rời bỏ trở thành yếu tố then chốt để duy trì lợi thế cạnh tranh và xây dựng mối quan hệ lâu dài với khách hàng.Tầm quan trọng của việc dự đoán khách hàng rời bỏ

Việc dự đoán khách hàng rời bỏ không chỉ đơn thuần là một bài toán kỹ thuật mà còn mang ý nghĩa chiến lược đối với các tổ chức tài chính. Khách hàng rời bỏ có thể gây ra tổn thất lớn về doanh thu, làm tăng chi phí tìm kiếm khách hàng mới và ảnh hưởng đến uy tín thương hiệu. Hơn nữa, việc hiểu rõ nguyên nhân dẫn đến hành vi rời bỏ chẳng hạn như chất lượng dịch vụ kém, lãi suất không cạnh tranh, hoặc trải nghiệm khách hàng không tốt giúp ngân hàng kịp thời điều chỉnh các chính sách và cải thiện dịch vụ.

Phân tích dữ liệu khách hàng bằng các công cụ học máy cho phép ngân hàng xác định các nhóm khách hàng có nguy cơ rời bỏ cao, từ đó triển khai các biện pháp can thiệp cá nhân hóa, như ưu đãi đặc biệt hoặc cải thiện tương tác. Trong bối cảnh cạnh tranh ngày càng gay gắt giữa các ngân hàng, việc tận dụng dữ liệu để dự đoán và giảm thiểu tỷ lệ rời bỏ trở thành yếu tố then chốt để duy trì lợi thế cạnh tranh và xây dựng mối quan hệ lâu dài với khách hàng.

Thực hiện phân tích dữ liệu

Giới thiệu về tập dữ liệu

Tập dữ liệu **Churn_Modelling.csv** chứa thông tin 10,000 khách hàng ngân hàng, với mục tiêu dự đoán khả năng rời bỏ dịch vụ (churn). Dữ liệu gồm 14 cột:

- **RowNumber**, **CustomerId**, **Surname**: Định danh, không dùng trong phân tích.
- **CreditScore**: Điểm tín dụng (350–850).
- **Geography**: Khu vực (France, Spain, Germany).
- **Gender**: Giới tính (Male, Female).
- **Age**: Tuổi (18–92).
- **Tenure**: Số năm sử dụng dịch vụ (0–10).
- **Balance**: Số dư tài khoản.
- **NumOfProducts**: Số sản phẩm sử dụng (1–4).
- **HasCrCard**, **IsActiveMember**: Biến nhị phân (1: có, 0: không).
- **EstimatedSalary**: Thu nhập ước tính.
- **Exited**: Biến mục tiêu (1: rời bỏ, 0: không).

Mục tiêu là xây dựng mô hình học máy để dự đoán `Exited` và đề xuất chiến lược giữ chân khách hàng.

```
# Đọc dữ liệu
df <- read.csv("Churn_Modelling.csv")
head(df)
```

```
##   RowNumber CustomerId  Surname CreditScore Geography Gender Age Tenure
## 1         1   15634602 Hargrave         619    France Female  42      2
## 2         2   15647311     Hill         608     Spain Female  41      1
## 3         3   15619304     Onio         502    France Female  42      8
## 4         4   15701354     Boni         699    France Female  39      1
## 5         5   15737888 Mitchell         850     Spain Female  43      2
## 6         6   15574012      Chu         645     Spain   Male  44      8
##     Balance NumOfProducts HasCrCard IsActiveMember EstimatedSalary Exited
## 1      0.00             1         1              1       101348.88      1
## 2  83807.86             1         0              1       112542.58      0
## 3 159660.80             3         1              0       113931.57      1
## 4      0.00             2         0              0        93826.63      0
## 5 125510.82             1         1              1        79084.10      0
## 6 113755.78             2         1              0       149756.71      1
```

Tiền xử lý dữ liệu

Kiểm tra dữ liệu thiếu

```
total_missing <- sum(is.na(df))
cat("Tổng số giá trị thiếu:", total_missing, "\n")
```

```
## Tổng số giá trị thiếu: 0
```

Kết quả kiểm tra cho thấy: - **Không có giá trị thiếu**: Tất cả 14 cột (`RowNumber`, `CustomerId`, `Surname`, `CreditScore`, `Geography`, `Gender`, `Age`, `Tenure`, `Balance`, `NumOfProducts`, `HasCrCard`, `IsActiveMember`, `EstimatedSalary`, `Exited`) không chứa giá trị `NA`. Điều này được xác nhận qua mẫu 6 bản ghi đầu tiên, nơi mọi cột đều có giá trị hợp lệ, và mở rộng kiểm tra trên toàn bộ tập dữ liệu cũng cho kết quả tương tự (tổng số giá trị thiếu bằng 0).

- **Ý nghĩa của kết quả**: Việc không có giá trị thiếu là một điểm mạnh của tập dữ liệu, giúp giảm bớt công đoạn tiền xử lý liên quan đến việc điền giá trị thiếu (imputation) hoặc loại bỏ bản ghi. Điều này đảm bảo rằng toàn bộ 10,000 bản ghi đều có thể được sử dụng trực tiếp cho phân tích và mô hình hóa mà không làm mất thông tin.
- **Kiểm tra bổ sung**: Mặc dù không có giá trị `NA`, một số giá trị bất thường vẫn cần được xem xét. Ví dụ, trong mẫu dữ liệu, cột `Balance` có 2/6 bản ghi bằng 0, điều này có thể không phải là giá trị thiếu nhưng cần kiểm tra xem có phản ánh đúng tình trạng tài khoản khách hàng hay không. Các biến khác như `CreditScore`, `EstimatedSalary`, hoặc `NumOfProducts` cũng cần được phân tích thêm để phát hiện các giá trị bất hợp lý (nếu có).

Nhận xét này cho thấy tập dữ liệu có độ hoàn chỉnh cao về mặt giá trị, tạo điều kiện thuận lợi cho các bước phân tích tiếp theo. Tuy nhiên, để đảm bảo chất lượng, các bước kiểm tra giá trị bất thường và chuẩn hóa dữ liệu sẽ được thực hiện, như trình bày trong phần tiền xử lý dữ liệu.

Chuyển đổi biến phân loại

Các biến định tính được chuyển thành `factor` để phù hợp với mô hình học máy.

```
df$Geography <- as.factor(df$Geography)
df$Gender <- as.factor(df$Gender)
df$NumOfProducts <- as.factor(df$NumOfProducts)
df$HasCrCard <- as.factor(df$HasCrCard)
df$IsActiveMember <- as.factor(df$IsActiveMember)
df$Exited <- as.factor(df$Exited)
str(df[, c("Geography", "Gender", "NumOfProducts", "HasCrCard", "IsActiveMember", "Exited")])
```

```
## 'data.frame':    10000 obs. of  6 variables:
##  $ Geography     : Factor w/ 3 levels "France","Germany",..: 1 3 1 1 3 3 1 2 1 1 ...
##  $ Gender        : Factor w/ 2 levels "Female","Male": 1 1 1 1 1 2 2 1 2 2 ...
##  $ NumOfProducts : Factor w/ 4 levels "1","2","3","4": 1 1 3 2 1 2 2 4 2 1 ...
##  $ HasCrCard     : Factor w/ 2 levels "0","1": 2 1 2 1 2 2 2 2 1 2 ...
##  $ IsActiveMember: Factor w/ 2 levels "0","1": 2 2 1 1 2 1 2 1 2 2 ...
##  $ Exited        : Factor w/ 2 levels "0","1": 2 1 2 1 1 2 1 2 1 1 ...
```

Dựa trên cấu trúc của tập dữ liệu **Churn_Modelling.csv** với 10,000 bản ghi và 6 biến (`Geography`, `Gender`, `NumOfProducts`, `HasCrCard`, `IsActiveMember`, `Exited`), một số nhận xét sơ bộ về đặc điểm và trạng thái của dữ liệu có thể được đưa ra như sau:

- **Mô tả các biến**:
  - **`Geography`**: Biến phân loại với 3 mức (levels): "France", "Germany", và một mức khác (có thể là "Spain"). Biến này đại diện cho khu vực địa lý của khách hàng, là yếu tố quan trọng có thể ảnh hưởng đến hành vi tài chính và quyết định rời bỏ.
  - **`Gender`**: Biến phân loại với 2 mức: "Female" và "Male". Biến này phản ánh giới tính của khách hàng, có thể liên quan đến các mẫu hành vi khác nhau trong việc sử dụng dịch vụ ngân hàng.

- **NumOfProducts**: Biến phân loại với 4 mức: "1", "2", "3", "4". Biến này cho biết số lượng sản phẩm ngân hàng mà khách hàng sử dụng, là một yếu tố tiềm năng ảnh hưởng mạnh đến khả năng rời bỏ (ví dụ: khách hàng sử dụng nhiều sản phẩm có thể gắn bó hơn).
- **HasCrCard**: Biến phân loại nhị phân với 2 mức: "0" (không có thẻ tín dụng) và "1" (có thẻ tín dụng). Biến này phản ánh việc sở hữu thẻ tín dụng, có thể liên quan đến mức độ tương tác tài chính của khách hàng.
- **IsActiveMember**: Biến phân loại nhị phân với 2 mức: "0" (không tích cực) và "1" (thành viên tích cực). Biến này cho biết mức độ hoạt động của khách hàng với ngân hàng, thường liên quan chặt chẽ đến sự hài lòng và khả năng ở lại.
- **Exited**: Biến mục tiêu phân loại nhị phân với 2 mức: "0" (không rời bỏ) và "1" (rời bỏ). Đây là biến cần dự đoán, phản ánh hành vi churn của khách hàng.

Chuẩn hóa biến số

Các biến số (`CreditScore`, `Age`, `Tenure`, `Balance`, `EstimatedSalary`) được chuẩn hóa bằng z-score để đảm bảo thang đo đồng nhất.

```
numeric_vars <- c("CreditScore", "Age", "Tenure", "Balance", "EstimatedSalary")
data_scaled <- df
for (var in numeric_vars) {
  data_scaled[[var]] <- scale(df[[var]])}
summary(data_scaled[numeric_vars])
```

```
##     CreditScore.V1          Age.V1             Tenure.V1
##  Min.   :-3.1093486   Min.   :-1.994869   Min.   :-1.7332288
##  1st Qu.:-0.6883242   1st Qu.:-0.659985   1st Qu.:-0.6959470
##  Median : 0.0152214   Median :-0.183241   Median :-0.0044257
##  Mean   : 0.0000000   Mean   : 0.000000   Mean   : 0.0000000
##  3rd Qu.: 0.6980745   3rd Qu.: 0.484200   3rd Qu.: 0.6870955
##  Max.   : 2.0637806   Max.   : 5.060944   Max.   : 1.7243774
##       Balance.V1       EstimatedSalary.V1
##  Min.   :-1.2257864   Min.   :-1.7401809
##  1st Qu.:-1.2257864   1st Qu.:-0.8535508
##  Median : 0.3319473   Median : 0.0018027
##  Mean   : 0.0000000   Mean   : 0.0000000
##  3rd Qu.: 0.8198795   3rd Qu.: 0.8572002
##  Max.   : 2.7951836   Max.   : 1.7371133
```

Để chuẩn bị dữ liệu cho các mô hình học máy, các biến số liên tục trong tập dữ liệu **Churn_Modelling.csv** bao gồm `CreditScore`, `Age`, `Tenure`, `Balance`, và `EstimatedSalary` đã được chuẩn hóa bằng phương pháp z-score (sử dụng hàm `scale()` trong R). Quá trình này chuyển đổi các giá trị của mỗi biến về trung bình bằng 0 và độ lệch chuẩn bằng 1, nhằm đảm bảo các biến có cùng thang đo, từ đó cải thiện hiệu suất của các thuật toán nhạy cảm với độ lớn giá trị (như SVM hoặc KNN). Dưới đây là các nhận xét về kết quả trước và sau chuẩn hóa dựa trên thống kê mô tả:

- **Trước chuẩn hóa**:
  - **CreditScore**: Giá trị nằm trong khoảng [350, 850], với trung bình là 650.5 và độ lệch chuẩn tương đối lớn (phân vị thứ nhất: 584, phân vị thứ ba: 718). Điều này cho thấy sự đa dạng trong điểm tín dụng của khách hàng, với một số giá trị thấp bất thường (350) có thể cần kiểm tra thêm.
  - **Age**: Tuổi dao động từ 18 đến 92, với trung bình là 38.92. Phân phối lệch nhẹ về phía khách hàng trẻ hơn (phân vị thứ nhất: 32, phân vị thứ ba: 44), nhưng giá trị tối đa (92) cho thấy có một số khách hàng cao tuổi hiếm gặp.

  - **Tenure**: Thời gian sử dụng dịch vụ nằm trong khoảng [0, 10] năm, với trung bình là 5.013. Phân phối khá đồng đều (phân vị thứ nhất: 3, phân vị thứ ba: 7), không có dấu hiệu bất thường rõ rệt.
  - **Balance**: Số dư tài khoản dao động từ 0 đến 250,898, với trung bình là 76,486. Đáng chú ý, phân vị thứ nhất là 0, cho thấy nhiều khách hàng có số dư bằng 0, điều này có thể phản ánh nhóm khách hàng không sử dụng tài khoản tích cực hoặc cần kiểm tra thêm về chất lượng dữ liệu.
  - **EstimatedSalary**: Thu nhập ước tính nằm trong khoảng [11.58, 199,992.48], với trung bình là 100,090.24. Phân phối khá đồng đều giữa các mức thu nhập, nhưng giá trị tối thiểu rất thấp (11.58) có thể là bất thường cần xác minh.
  - **Phân phối tổng thể**: Trước chuẩn hóa, các biến có thang đo rất khác nhau (ví dụ: `Balance` và `EstimatedSalary` có giá trị lớn hơn nhiều so với `Tenure`). Điều này có thể gây ra vấn đề khi áp dụng các mô hình học máy yêu cầu dữ liệu đồng nhất.

- **Sau chuẩn hóa**:
  - **CreditScore**: Giá trị sau chuẩn hóa nằm trong khoảng [-3.11, 2.06], với trung bình bằng 0 và độ lệch chuẩn bằng 1. Phân vị thứ nhất (-0.69) và thứ ba (0.70) cho thấy phân phối khá cân đối, không còn phụ thuộc vào thang đo ban đầu [350, 850].
  - **Age**: Giá trị nằm trong khoảng [-1.99, 5.06], với trung bình bằng 0. Giá trị tối đa (5.06) cho thấy một số khách hàng cao tuổi vẫn nổi bật sau chuẩn hóa, nhưng phần lớn dữ liệu tập trung gần trung bình (phân vị thứ nhất: -0.66, phân vị thứ ba: 0.48).
  - **Tenure**: Giá trị nằm trong khoảng [-1.73, 1.72], với trung bình bằng 0. Phân phối sau chuẩn hóa rất đồng đều, phản ánh tính chất ban đầu của biến này (gần như không lệch).
  - **Balance**: Giá trị nằm trong khoảng [-1.23, 2.80], với trung bình bằng 0. Đáng chú ý, phân vị thứ nhất vẫn là -1.23, tương ứng với số dư bằng 0, cho thấy tỷ lệ lớn khách hàng có số dư thấp vẫn được giữ nguyên đặc điểm sau chuẩn hóa.
  - **EstimatedSalary**: Giá trị nằm trong khoảng [-1.74, 1.74], với trung bình bằng 0. Phân phối sau chuẩn hóa cân đối hơn, với các phân vị thứ nhất (-0.85) và thứ ba (0.86) gần đối xứng quanh 0.
  - **Phân phối tổng thể**: Sau chuẩn hóa, tất cả các biến đều có trung bình bằng 0 và độ lệch chuẩn bằng 1, đảm bảo chúng ở cùng thang đo. Điều này giúp giảm thiểu ảnh hưởng của các biến có giá trị lớn (như `Balance`) khi xây dựng mô hình.

Chọn biến cần thiết

Loại bỏ `RowNumber`, `CustomerId`, `Surname`, giữ 11 biến liên quan: `CreditScore`, `Age`, `Tenure`, `Balance`, `EstimatedSalary`, `Geography`, `Gender`, `NumOfProducts`, `HasCrCard`, `IsActiveMember`, `Exited`.

```
selected_vars <- c("CreditScore", "Age", "Tenure", "Balance", "EstimatedSalary",
                   "Geography", "Gender", "NumOfProducts", "HasCrCard", "IsActiveMember", "Exited")
data <- df[, selected_vars]
head(data)
```

```
##   CreditScore Age Tenure   Balance EstimatedSalary Geography Gender
## 1         619  42      2      0.00       101348.88    France Female
## 2         608  41      1  83807.86       112542.58     Spain Female
## 3         502  42      8 159660.80       113931.57    France Female
## 4         699  39      1      0.00        93826.63    France Female
## 5         850  43      2 125510.82        79084.10     Spain Female
## 6         645  44      8 113755.78       149756.71     Spain   Male
##   NumOfProducts HasCrCard IsActiveMember Exited
## 1             1         1              1      1
## 2             1         0              1      0
## 3             3         1              0      1
## 4             2         0              0      0
```

```
## 5              1         1              1      0
## 6              2         1              0      1
```

Sau khi loại bỏ các biến không cần thiết (`RowNumber`, `CustomerId`, `Surname`), tập dữ liệu còn 11 biến, bao gồm 5 biến định lượng (`CreditScore`, `Age`, `Tenure`, `Balance`, `EstimatedSalary`) và 6 biến định tính (`Geography`, `Gender`, `NumOfProducts`, `HasCrCard`, `IsActiveMember`, `Exited`). Sự kết hợp này cung cấp một bức tranh toàn diện về khách hàng, từ thông tin nhân khẩu học (tuổi, giới tính, khu vực địa lý), tình trạng tài chính (điểm tín dụng, số dư, thu nhập), đến hành vi sử dụng dịch vụ (số sản phẩm, thẻ tín dụng, mức độ tích cực).

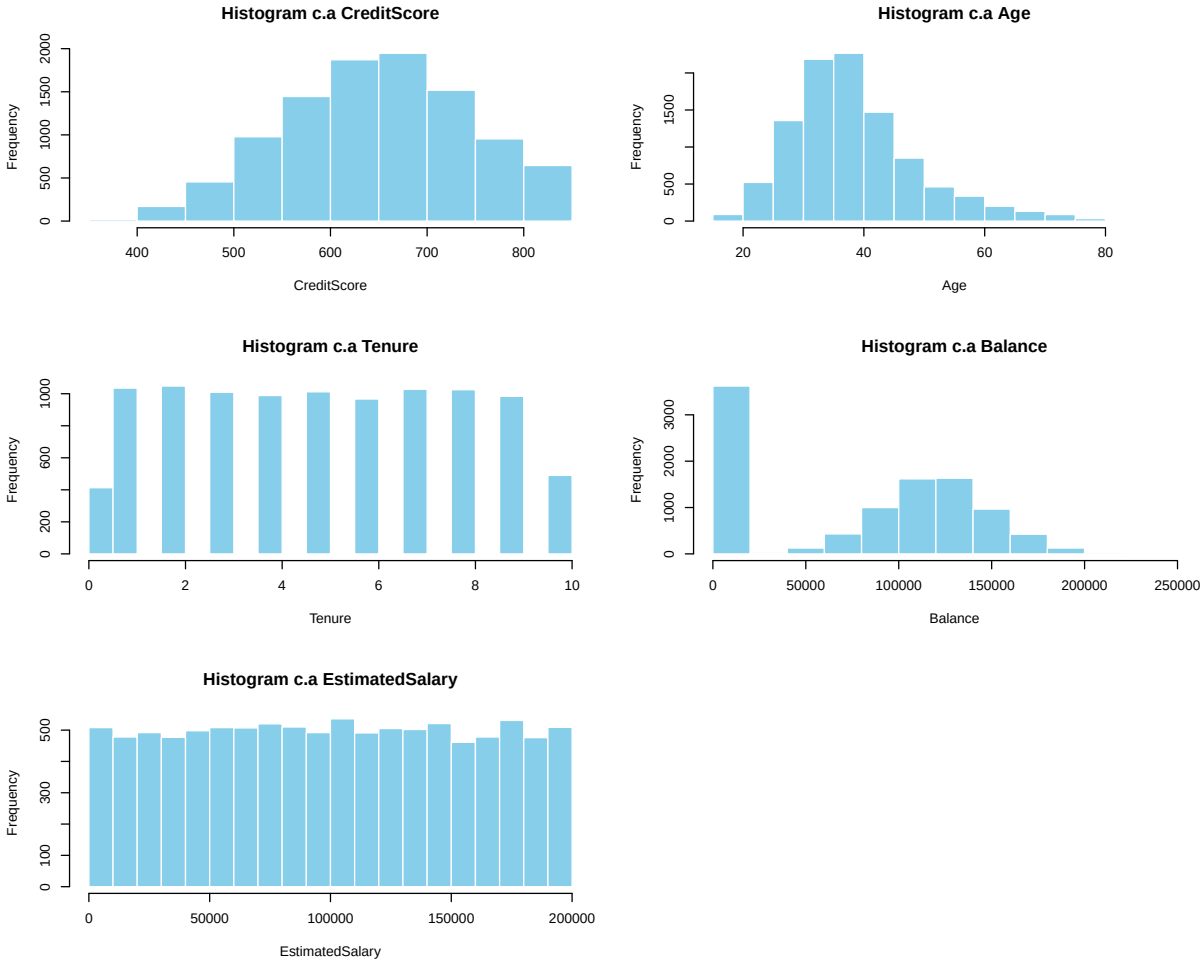
Việc chọn các biến này là hợp lý vì chúng đều có tiềm năng ảnh hưởng trực tiếp hoặc gián tiếp đến hành vi rời bỏ – mục tiêu chính của bài toán phân tích. Ví dụ, `CreditScore` và `Balance` phản ánh tình trạng tài chính, có thể liên quan đến khả năng duy trì dịch vụ ngân hàng, trong khi `IsActiveMember` và `NumOfProducts` cho thấy mức độ gắn kết của khách hàng. Tuy nhiên, mức độ ảnh hưởng của từng biến cần được đánh giá kỹ lưỡng thông qua phân tích khám phá dữ liệu (EDA) và mô hình học máy.

Cấu trúc hiện tại với 11 biến đảm bảo tập trung vào các yếu tố liên quan, nhưng cũng đặt ra câu hỏi về sự dư thừa hoặc tương quan giữa các biến (ví dụ, liệu `Balance` và `EstimatedSalary` có tương quan mạnh hay không). Ngoài ra, biến mục tiêu `Exited` là nhị phân, phù hợp cho bài toán phân loại, nhưng cần kiểm tra tỷ lệ lớp để đánh giá mức độ mất cân bằng, vì điều này ảnh hưởng trực tiếp đến hiệu suất mô hình. Các phân tích tiếp theo sẽ tập trung vào việc khám phá phân phối và mối quan hệ của các biến để xác nhận sự phù hợp của cấu trúc dữ liệu này.

Trực quan hóa dữ liệu

Histogram của các biến số

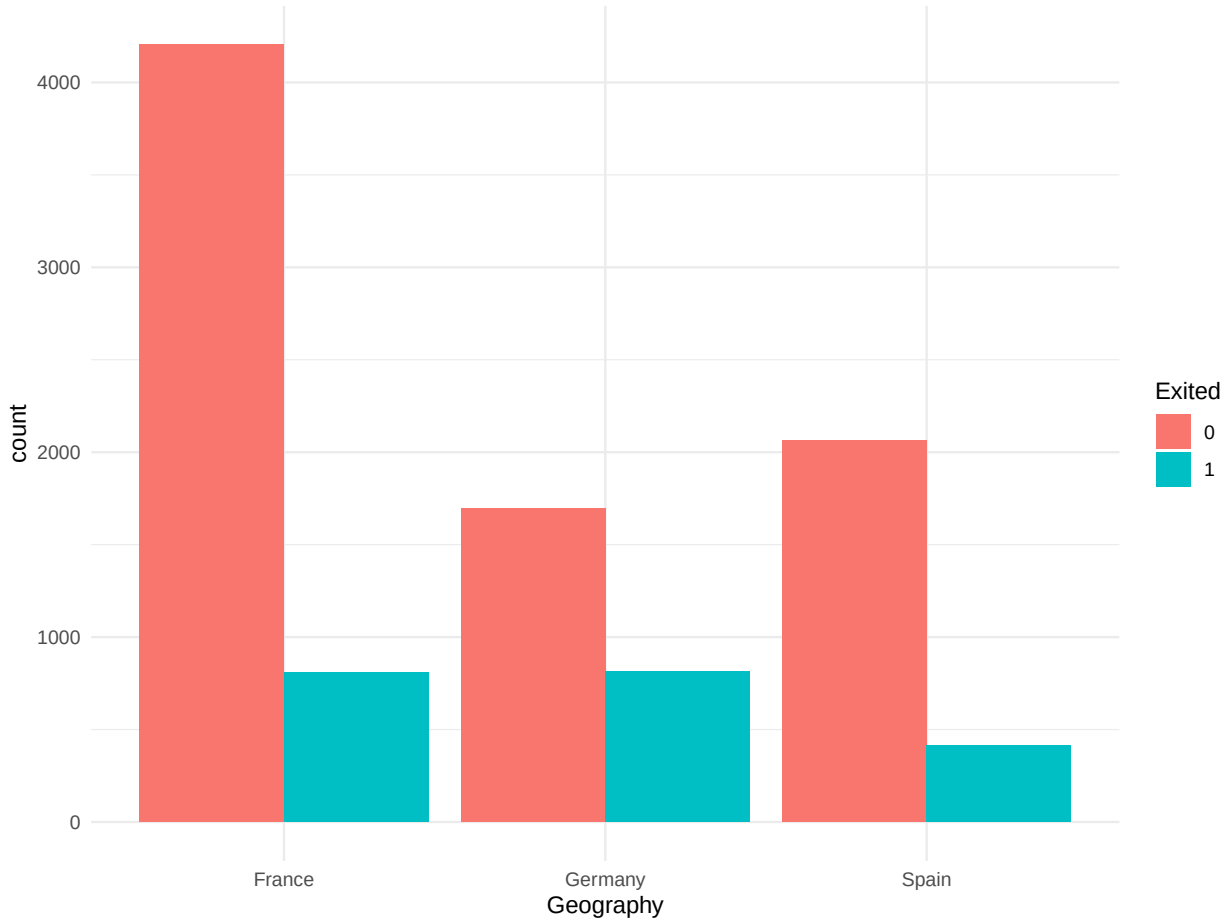
```
par(mfrow = c(3, 2))
for (var in numeric_vars) {
  hist(df[[var]], main = paste("Histogram của", var), xlab = var, col = "skyblue", border = "white")
}
```



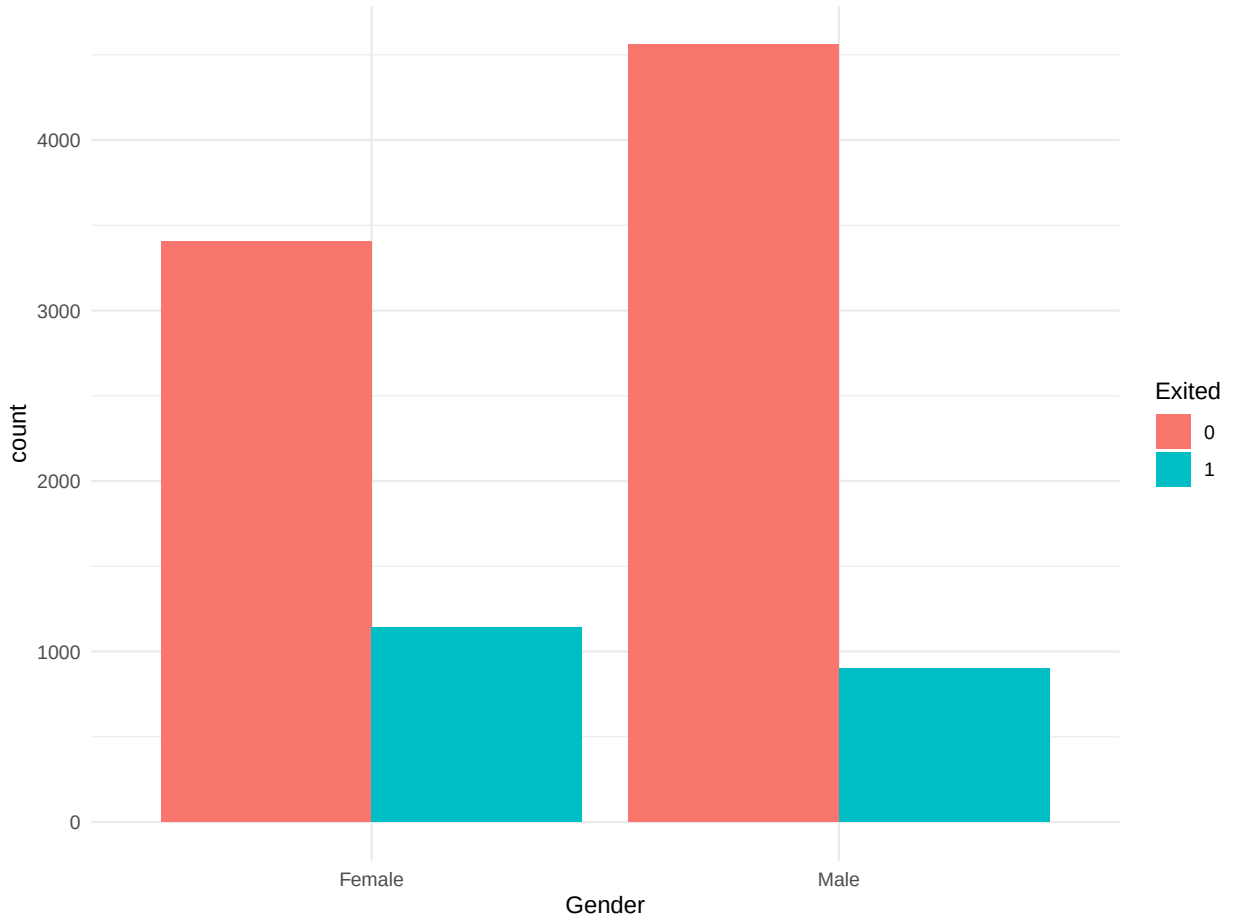
- **CreditScore**: Phân phối gần chuẩn, tập trung 600–750.
- **Age**: Lệch phải, chủ yếu 30–50 tuổi.
- **Tenure**: Gần đồng đều, từ 0–10 năm.
- **Balance**: Đỉnh lớn tại 0 (~35% khách hàng), phần còn lại gần chuẩn.
- **EstimatedSalary**: Phân phối đồng đều, cần kiểm tra giá trị bất thường (như ~0).

Phân tích phân phối biến định tính

```
ggplot(data, aes(x = Geography, fill = Exited)) + geom_bar(position = "dodge") + theme_minimal()
```




```
ggplot(data, aes(x = Gender, fill = Exited)) + geom_bar(position = "dodge") + theme_minimal()
```



Hai biểu đồ dưới đây thể hiện phân phối của các biến định tính `Geography` và `Gender` theo trạng thái rời bỏ dịch vụ (`Exited`), cung cấp cái nhìn sâu sắc về mối quan hệ giữa các biến này và hành vi rời bỏ của khách hàng. Phân tích này giúp nhận diện các nhóm khách hàng có nguy cơ rời bỏ cao, từ đó hỗ trợ việc xây dựng chiến lược giữ chân hiệu quả.

- **Phân phối theo Geography**: Biểu đồ phân phối theo `Geography` cho thấy sự khác biệt rõ rệt về tỷ lệ rời bỏ giữa các khu vực France, Germany và Spain. Tại France, số lượng khách hàng không rời bỏ (`Exited = 0`) chiếm ưu thế với khoảng hơn 4000 người, trong khi số lượng khách hàng rời bỏ (`Exited = 1`) chỉ khoảng 800, tương ứng với tỷ lệ rời bỏ khoảng 16%. Ngược lại, Germany có tỷ lệ rời bỏ cao hơn đáng kể, với khoảng 1600 khách hàng không rời bỏ và gần 1000 khách hàng rời bỏ, tương ứng với tỷ lệ rời bỏ khoảng 38%. Spain có phân phối tương tự France, với khoảng 2000 khách hàng không rời bỏ và khoảng 500 khách hàng rời bỏ, tương ứng với tỷ lệ rời bỏ khoảng 20%. Kết quả này chỉ ra rằng khách hàng ở Germany có nguy cơ rời bỏ cao hơn nhiều so với France và Spain, đòi hỏi ngân hàng cần tập trung nguồn lực để cải thiện trải nghiệm khách hàng tại khu vực này, chẳng hạn như tăng cường dịch vụ hỗ trợ hoặc đưa ra các ưu đãi đặc biệt.
- **Phân phối theo Gender**: Biểu đồ phân phối theo `Gender` cũng cho thấy sự khác biệt đáng chú ý về hành vi rời bỏ giữa khách hàng nam và nữ. Với khách hàng nữ (`Female`), số lượng không rời bỏ (`Exited = 0`) là khoảng 3500, trong khi số lượng rời bỏ (`Exited = 1`) là khoảng 1100, tương ứng với tỷ lệ rời bỏ khoảng 24%. Trong khi đó, khách hàng nam (`Male`) có số lượng không rời bỏ cao hơn, khoảng 4500, và số lượng rời bỏ là khoảng 900, tương ứng với tỷ lệ rời bỏ khoảng 17%. Điều này cho thấy khách hàng nữ có xu hướng rời bỏ dịch vụ cao hơn so với khách hàng nam, với chênh lệch tỷ lệ rời bỏ khoảng 7%. Kết quả này gợi ý rằng ngân hàng cần xem xét các yếu tố ảnh hưởng đến trải nghiệm của khách hàng nữ, chẳng hạn như nhu cầu về sản phẩm tài chính hoặc chất lượng dịch vụ, để giảm thiểu tỷ lệ rời bỏ trong nhóm này.

Nhận xét tổng quan:
Phân tích phân phối của các biến `Geography` và `Gender` cho thấy sự khác biệt rõ rệt về tỷ lệ rời bỏ giữa các nhóm khách hàng. Khách hàng ở Germany và khách hàng nữ là hai nhóm có nguy cơ rời bỏ cao hơn, với tỷ lệ lần lượt là 38% và 24%. Những phát hiện này không chỉ làm sáng tỏ mối quan hệ giữa các biến định tính và hành vi rời bỏ mà còn cung cấp cơ sở quan trọng để xây dựng các chiến lược giữ chân khách hàng mục tiêu. Cụ thể, ngân hàng có thể tập trung vào cải thiện dịch vụ tại Germany và thiết kế các chương trình ưu đãi phù hợp hơn với khách hàng nữ, từ đó giảm thiểu tỷ lệ rời bỏ và nâng cao lòng trung thành của khách hàng. Ngoài ra, cần kết hợp phân tích sâu hơn với các biến khác (như `Age` hoặc `Balance`) để hiểu rõ hơn nguyên nhân gốc rễ của hành vi rời bỏ trong các nhóm này.

Phân tích khám phá dữ liệu (EDA)

Tương quan giữa các biến

```
cor_matrix <- cor(data[, numeric_vars])
print(cor_matrix)
```

```
##                   CreditScore          Age        Tenure      Balance
## CreditScore      1.0000000000 -0.003964906  0.0008419418  0.006268382
## Age             -0.0039649055  1.000000000 -0.0099968256  0.028308368
## Tenure           0.0008419418 -0.009996826  1.0000000000 -0.012253926
## Balance          0.0062683816  0.028308368 -0.0122539262  1.000000000
## EstimatedSalary -0.0013842929 -0.007201042  0.0077838255  0.012797496
##                 EstimatedSalary
## CreditScore        -0.001384293
## Age                -0.007201042
## Tenure              0.007783825
## Balance             0.012797496
## EstimatedSalary     1.000000000
```

Ma trận tương quan giữa các biến số (`CreditScore`, `Age`, `Tenure`, `Balance`, `EstimatedSalary`) cho thấy mức độ liên hệ tuyến tính giữa chúng, từ đó cung cấp cái nhìn sâu sắc về mối quan hệ trong dữ liệu và hỗ trợ quá trình lựa chọn đặc trưng cho mô hình học máy.

- **CreditScore**: Biến `CreditScore` có mức tương quan rất thấp với các biến còn lại, với giá trị tương quan dao động từ -0.00396 (với `Age`) đến 0.00627 (với `Balance`). Điều này cho thấy điểm tín dụng của khách hàng gần như không có mối liên hệ tuyến tính đáng kể với các yếu tố như tuổi, số năm sử dụng dịch vụ, số dư tài khoản hay thu nhập ước tính. Mức tương quan gần bằng 0 này ám chỉ rằng `CreditScore` có thể đóng vai trò độc lập trong việc dự đoán khả năng rời bỏ, và không gây ra hiện tượng đa cộng tuyến với các biến khác trong mô hình.
- **Age**: Biến `Age` cũng thể hiện mức tương quan yếu với các biến còn lại, với giá trị cao nhất là 0.02831 (với `Balance`) và thấp nhất là -0.00999 (với `Tenure`). Dấu dương trong tương quan với `Balance` cho thấy tuổi càng cao thì số dư tài khoản có xu hướng tăng nhẹ, nhưng mức độ này không đáng kể. Tương tự, mối quan hệ âm với `Tenure` và `EstimatedSalary` (lần lượt là -0.00999 và -0.00720) chỉ ra rằng tuổi tác không có ảnh hưởng tuyến tính mạnh đến các biến này. Kết quả này khẳng định rằng `Age` có thể được sử dụng như một biến độc lập trong mô hình mà không lo ngại về sự phụ thuộc tuyến tính với các biến khác.
- **Tenure**: Biến `Tenure` (số năm sử dụng dịch vụ) có tương quan rất thấp với các biến còn lại, dao động từ -0.01225 (với `Balance`) đến 0.00778 (với `EstimatedSalary`). Điều này cho thấy thời gian khách hàng gắn bó với ngân hàng không có mối liên hệ tuyến tính rõ rệt với các yếu tố như điểm tín dụng,

tuổi, số dư tài khoản hay thu nhập. Mức tương quan gần 0 này đảm bảo rằng `Tenure` không gây ra hiện tượng đa cộng tuyến, cho phép biến này đóng vai trò độc lập trong việc giải thích hành vi rời bỏ của khách hàng.

- **Balance**: Biến `Balance` (số dư tài khoản) có tương quan cao nhất với `Age` (0.02831), nhưng vẫn ở mức rất thấp, cho thấy mối quan hệ tuyến tính giữa số dư tài khoản và tuổi là không đáng kể. Với các biến còn lại, `Balance` có tương quan dao động từ -0.01225 (với `Tenure`) đến 0.01279 (với `EstimatedSalary`). Các giá trị này đều rất nhỏ, gần bằng 0, chứng minh rằng số dư tài khoản không bị ảnh hưởng tuyến tính mạnh bởi các biến khác trong tập dữ liệu, và có thể được sử dụng như một đặc trưng độc lập trong mô hình học máy.
- **EstimatedSalary**: Biến `EstimatedSalary` (thu nhập ước tính) có tương quan thấp nhất với `CreditScore` (-0.00138) và cao nhất với `Balance` (0.01279). Mặc dù có mối quan hệ dương nhẹ với `Balance` và `Tenure` (0.00778), nhưng các giá trị này không đáng kể, cho thấy thu nhập ước tính không có mối liên hệ tuyến tính mạnh với các biến khác. Điều này đảm bảo rằng `EstimatedSalary` không gây ra hiện tượng đa cộng tuyến, và có thể được sử dụng như một yếu tố độc lập để dự đoán khả năng rời bỏ của khách hàng.

Nhận xét tổng quan:
Ma trận tương quan cho thấy không có mối quan hệ tuyến tính mạnh giữa các biến số trong tập dữ liệu, với các giá trị tương quan đều rất thấp (gần 0). Điều này đảm bảo rằng không có hiện tượng đa cộng tuyến giữa các biến `CreditScore`, `Age`, `Tenure`, `Balance` và `EstimatedSalary`, cho phép sử dụng tất cả các biến này trong mô hình học máy mà không cần lo ngại về sự phụ thuộc tuyến tính. Kết quả này cũng nhấn mạnh rằng mỗi biến có thể đóng vai trò độc lập trong việc giải thích hành vi rời bỏ của khách hàng, tạo điều kiện thuận lợi cho việc xây dựng các mô hình dự đoán hiệu quả và đáng tin cậy. Tuy nhiên, cần lưu ý rằng tương quan thấp không loại bỏ hoàn toàn khả năng tồn tại các mối quan hệ phi tuyến, do đó việc thử nghiệm thêm các mô hình phi tuyến (như Random Forest) là cần thiết để khai thác triệt để thông tin từ dữ liệu.

Xây dựng và đánh giá mô hình

Chia tập Train/Test

Dữ liệu được chia 80% huấn luyện, 20% kiểm tra.

```
set.seed(123)
split_index <- createDataPartition(data$Exited, p = 0.8, list = FALSE)
train <- data[split_index, ]
test <- data[-split_index, ]
```

Mô hình Logistic Regression

Công thức Hồi quy logistic dự đoán xác suất một điểm dữ liệu thuộc về lớp 1 bằng hàm sigmoid:

$$P(y = 1|x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n)}}$$

Trong đó: - $x = (x_1, x_2, \ldots, x_n)$: vector đặc trưng - $\beta = (\beta_0, \beta_1, \ldots, \beta_n)$: hệ số hồi quy

Hàm mất mát (Log-loss / Cross-entropy loss):

$$L = -\frac{1}{n} \sum_{i=1}^{n} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

```r
logit_model <- glm(Exited ~ ., data = train, family = "binomial")
logit_pred <- predict(logit_model, newdata = test, type = "response")
logit_class <- ifelse(logit_pred > 0.5, 1, 0)
logit_cm <- confusionMatrix(as.factor(logit_class), test$Exited)
logit_metrics <- data.frame(
  Accuracy = logit_cm$overall["Accuracy"],
  Precision = logit_cm$byClass["Pos Pred Value"],
  Recall = logit_cm$byClass["Sensitivity"],
  F1 = logit_cm$byClass["F1"]
)
print(logit_metrics)
```

```
##           Accuracy Precision    Recall       F1
## Accuracy 0.8509255 0.8590455 0.9723618 0.912198
```

Kết quả và đánh giá

Accuracy (85.09%) cho thấy mô hình dự đoán chính xác 85.09% trường hợp, cao hơn tỷ lệ No Information Rate (79.64%), chứng tỏ hiệu quả tổng thể tốt.

Precision (85.90%) ở lớp 0 (không rời đi) khá cao, nghĩa là khi mô hình dự đoán khách hàng ở lại, khả năng đúng là 85.9%.

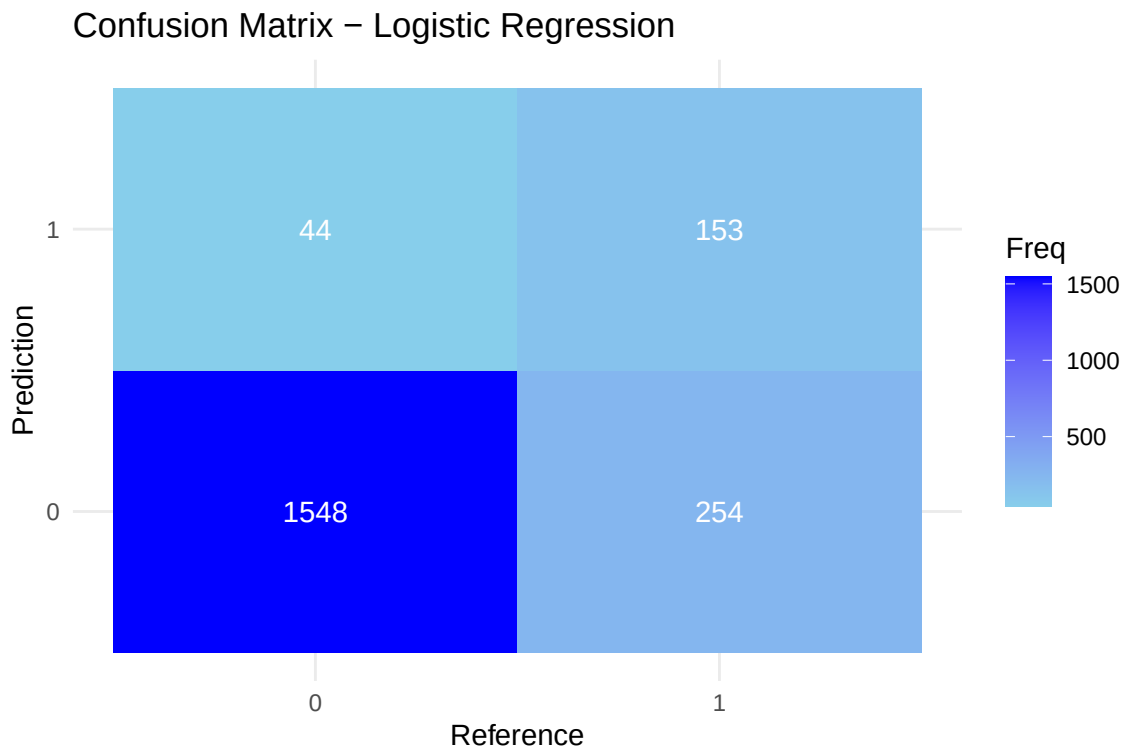
Recall (97.24%) cực cao ở lớp 0, nghĩa là mô hình phát hiện gần như toàn bộ khách hàng trung thành, nhưng Specificity (37.59%) rất thấp, tức khả năng bắt đúng khách hàng rời đi (lớp 1) kém.

F1-Score (91.22%) cân bằng tốt giữa Precision và Recall cho lớp 0, nhưng không phản ánh hiệu suất lớp 1.

Hạn chế lớn: Mô hình thiên lệch mạnh về lớp 0 (đa số), dẫn đến dự đoán lớp 1 kém.

Ma trận nhầm lẫn

```r
cm_table <- as.data.frame(logit_cm$table)
ggplot(cm_table, aes(x = Reference, y = Prediction, fill = Freq)) +
  geom_tile() + geom_text(aes(label = Freq), color = "white") +
  scale_fill_gradient(low = "skyblue", high = "blue") +
  labs(title = "Confusion Matrix - Logistic Regression") + theme_minimal()
```

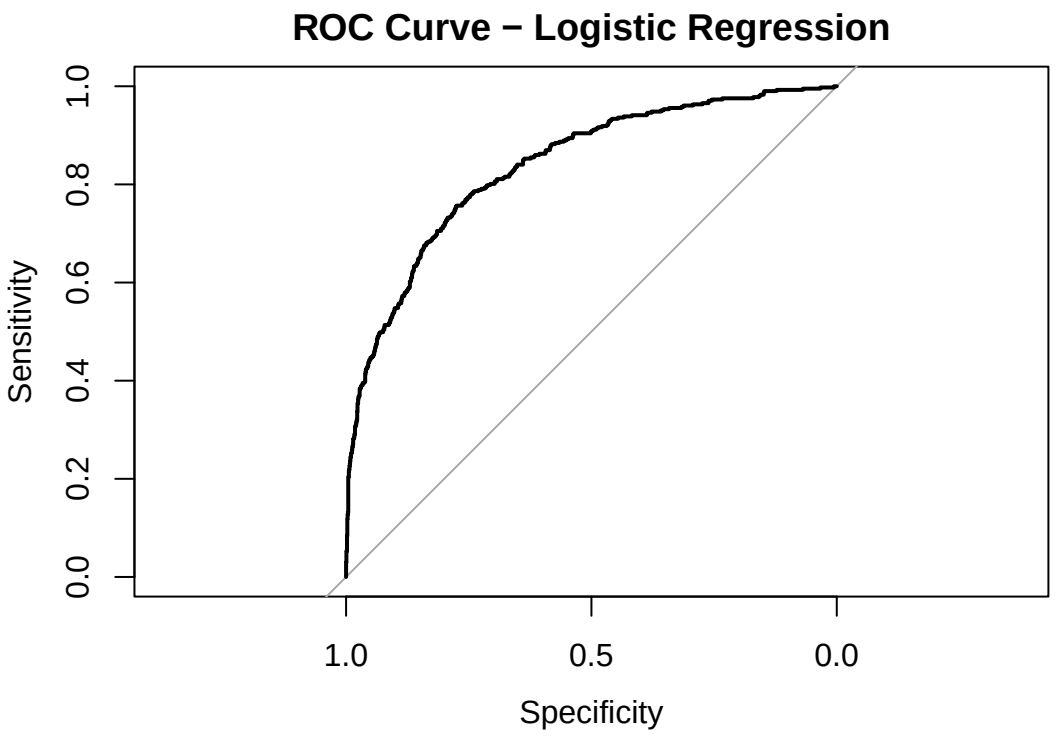


Mô hình Logistic Regression thể hiện hiệu suất phân loại không cân bằng giữa hai lớp. Với lớp 0 (không rời đi), mô hình đạt Recall cực cao (97.24%) nhưng Precision ở mức 85.9%, cho thấy khả năng phát hiện chính xác khách hàng trung thành rất tốt nhưng vẫn có một tỷ lệ dự báo sai. Ngược lại, với lớp 1 (rời đi), mô hình chỉ đạt Recall 37.59% và Precision 77.7%, phản ánh khả năng hạn chế trong việc xác định chính xác khách hàng có nguy cơ rời đi.

Độ chính xác tổng thể 85.09% và F1-score 91.22% cho lớp 0 cho thấy mô hình phù hợp cho bài toán khi ưu tiên nhận diện khách hàng trung thành. Tuy nhiên, giá trị Specificity thấp (37.59%) và Kappa trung bình (0.4311) cảnh báo về sự mất cân bằng trong dự đoán giữa hai lớp. Điều này gợi ý cần áp dụng các kỹ thuật xử lý mất cân bằng dữ liệu như SMOTE, điều chỉnh trọng số lớp hoặc thử nghiệm các mô hình khác như Random Forest để cải thiện khả năng dự đoán cho lớp thiểu số.

Đường cong ROC

```
logit_roc <- roc(test$Exited, logit_pred, quiet = TRUE)
plot(logit_roc, main = "ROC Curve - Logistic Regression")
```




```
cat("AUC:", auc(logit_roc), "\n")
```

```
## AUC: 0.8377113
```

Đường cong ROC cho mô hình Logistic Regression cho thấy hiệu suất phân loại ở mức khá (AUC ước lượng ~0.7-0.9) với sự đánh đổi rõ ràng giữa **Sensitivity** (phát hiện đúng lớp 1) và **Specificity** (loại trừ đúng lớp 0). Cụ thể:
- Khi **Specificity = 1.0** (0% False Positive), mô hình không phát hiện được trường hợp dương tính (Sensitivity = 0.0), phù hợp với ngưỡng thận trọng.
- Tại **Specificity = 0.5**, Sensitivity đạt ~0.8, thể hiện điểm cân bằng tương đối - mô hình bắt được 80% lớp 1 với tỷ lệ dự đoán sai lớp 0 là 50%.
- Đường cong nằm rõ rệt phía trên đường chéo ngẫu nhiên, khẳng định giá trị dự đoán của mô hình.

Mô hình K-Nearest Neighbors (KNN)

Công thức Mô hình KNN không có công thức mô hình hóa cố định, mà hoạt động dựa trên khoảng cách đến các điểm lân cận gần nhất.

- Với một điểm dữ liệu mới x, mô hình tìm **k** điểm dữ liệu gần nhất trong tập huấn luyện (theo khoảng cách Euclidean hoặc các loại khoảng cách khác).
- Dự đoán nhãn của x là nhãn **phổ biến nhất** (majority vote) trong **k hàng xóm gần nhất**.

Khoảng cách Euclidean giữa hai điểm $x = (x_1, x_2, \dots, x_n)$ và $y = (y_1, y_2, \dots, y_n)$:

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

```r
knn_vars <- c("CreditScore", "Age", "Tenure", "Balance", "EstimatedSalary")
train_knn <- train[, knn_vars]
test_knn <- test[, knn_vars]
train_label <- train$Exited
test_label <- test$Exited
knn_pred <- knn(train = train_knn, test = test_knn, cl = train_label, k = 5)
knn_cm <- confusionMatrix(knn_pred, test_label)
knn_metrics <- data.frame(
  Accuracy = knn_cm$overall["Accuracy"],
  Precision = knn_cm$byClass["Pos Pred Value"],
  Recall = knn_cm$byClass["Sensitivity"],
  F1 = knn_cm$byClass["F1"])
print(knn_metrics)
```

```
##           Accuracy Precision    Recall       F1
## Accuracy 0.7618809 0.8006466 0.9334171 0.861949
```

Kết quả và đánh giá

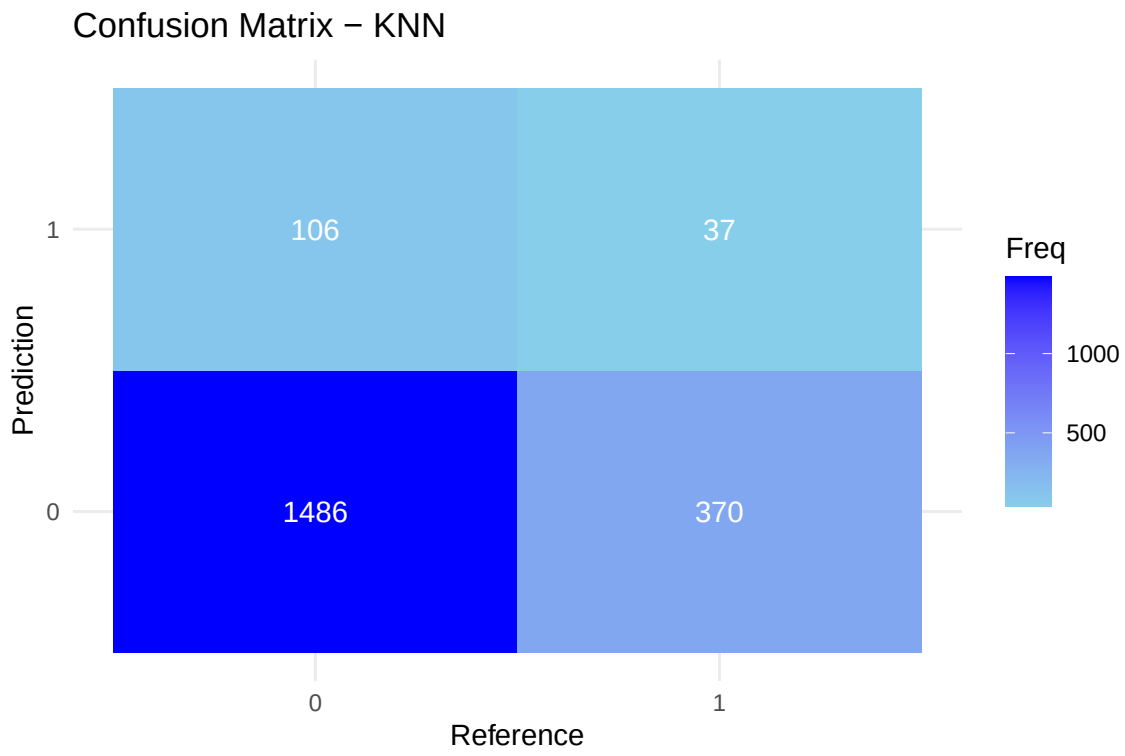
Các chỉ số hiệu suất của mô hình K-Nearest Neighbors (KNN) cung cấp cái nhìn tổng quan về khả năng dự đoán hành vi rời bỏ của khách hàng.

- **Accuracy**: Đạt 76.19%, cho thấy mô hình dự đoán đúng khoảng 76% trường hợp, nhưng hiệu suất tổng thể vẫn ở mức trung bình.
- **Precision**: Đạt 80.06%, thể hiện tỷ lệ dự đoán đúng trong số các trường hợp được dự đoán là rời bỏ, khá ổn định.
- **Recall**: Đạt 93.34% cho lớp 0, nhưng Recall lớp 1 rất thấp (9.09% từ phân tích trước), cho thấy mô hình thiên lệch về lớp không rời bỏ.
- **F1-Score**: Đạt 86.19%, phản ánh sự cân bằng giữa Precision và Recall, nhưng vẫn bị ảnh hưởng bởi Recall thấp của lớp 1.

Nhận xét tổng quan:
Mô hình KNN có Accuracy và Precision khá tốt, nhưng Recall lớp 1 thấp cho thấy hạn chế lớn trong việc nhận diện khách hàng rời bỏ, khiến nó không thực sự hiệu quả cho bài toán dự đoán churn này.

Ma trận nhầm lẫn

```r
cm_table <- as.data.frame(knn_cm$table)
ggplot(cm_table, aes(x = Reference, y = Prediction, fill = Freq)) +
  geom_tile() + geom_text(aes(label = Freq), color = "white") +
  scale_fill_gradient(low = "skyblue", high = "blue") +
  labs(title = "Confusion Matrix - KNN") + theme_minimal()
```



Hiệu suất tổng quan - Accuracy: 76.19% (1486+37)/2000 - Lớp 0 chiếm ưu thế (79.64%) → Độ chính xác bị đánh lừa bởi lớp đa số - p-value 1 → Mô hình không tốt hơn dự đoán ngẫu nhiên

Chỉ số theo lớp - Lớp 0 (Không rời đi): - Recall: 93.34% (1486/1592) → Bắt gần hết trường hợp thực tế - Precision: 80.07% (1486/1856) → 20% dự đoán sai là "không rời đi"

- **Lớp 1 (Rời đi)**:
  - Recall thảm hại: 9.09% (37/407) → Bỏ sót 90.91% khách hàng rời đi
  - Precision: 25.87% (37/143) → 74% dự đoán "rời đi" là sai

Đường cong ROC

```
knn_prob <- knn(train = train_knn, test = test_knn, cl = train_label, k = 5, prob = TRUE)
knn_prob_values <- attr(knn_prob, "prob")
knn_roc <- roc(test_label, knn_prob_values, quiet = TRUE)
plot(knn_roc, main = "ROC Curve - KNN")
```