**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP.HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**



**MÔN HỌC: KHAI THÁC DỮ LIỆU**

**ĐỀ TÀI: Ứng dụng KNN trong Dự đoán và Phân tích Nguy cơ Viêm Gan C từ Dữ liệu Sức Khỏe**

**GVBM:** Trần Như Ý

**SVTH:**

Đặng Cam Hồng\_ 2001221533

Nguyễn Viết Tiến \_ 2001224408

Trần Dương Tường Vy\_'2001225950

**1. Lịch làm việc nhóm trong hàng tuần**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tuần** | **Chủ đề chính** | **Công việc cụ thể** | **Ghi chú** |
| **Tuần 1** | K-Means & KNN | - Học lý thuyết về K-Means & KNN - Viết code demo với dữ liệu nhỏ - Trực quan hóa kết quả bằng scatter plot, confusion matrix - Ghi log quá trình học | Nguyễn Viết Tiến tìm hiểu thuật toán K-Means & KNN |
| **Tuần 2** | Apriori & Luật kết hợp | - Tìm hiểu luật kết hợp: support, confidence, lift - Cài đặt Apriori bằng mlxtend - Áp dụng trên dữ liệu mẫu - Giải thích từng luật khai phá | Trần Dương Tường Vy tìm hiểu Apriori & Luật kết hợp |
| **Tuần 3** | Random Forest | - Tìm hiểu nguyên lý và cách hoạt động - Viết code phân loại dữ liệu y tế bằng RandomForestClassifier - Đánh giá: Accuracy, Precision, Recall, F1 - So sánh với KNN | Đặng Cam Hồng tìm hiểu về Random Forest |
| **Tuần 4** | Tích hợp & Báo cáo | - Kết hợp tất cả các thuật toán vào giao diện chung - Làm báo cáo nhóm: lý thuyết + mô hình + kết quả - Tối ưu giao diện (Streamlit, tkinter...) - Chuẩn bị slide thuyết trình | Nhóm bắt đầu tổng hợp và viết code |
| **Tuần 5** | Hoàn thiện hệ thống | - Dọn dẹp code, tối ưu hiệu suất - Kiểm thử tính năng cuối cùng - Viết README, hướng dẫn sử dụng - Backup toàn bộ project (GitHub, Google Drive) - Demo thử nghiệm cho bạn bè / GVHD | Nhóm bắt đầu tổng hợp và viết code |

**2.Công việc mỗi thành viên trong nhóm**

|  |  |
| --- | --- |
| **Thành viên** | **Công việc cụ thể** |
| **Nguyễn Viết Tiến** | - Tìm hiểu lý thuyết và nguyên lý thuật toán K-Means và KNN - Viết code K-Means và KNN - Trực quan hóa kết quả bằng scatter plot, confusion matrix - Tối ưu giao diện và xử lý đầu vào/đầu ra cho 2 thuật toán |
| **Trần Dương Tường Vy** | - Nghiên cứu lý thuyết luật kết hợp (support, confidence, lift) - Cài đặt thuật toán Apriori bằng thư viện mlxtend - Phân tích và giải thích các luật kết hợp tìm được - Viết mô tả phần luật kết hợp trong báo cáo |
| **Đặng Cam Hồng** | - Tìm hiểu chi tiết thuật toán Random Forest - Viết code phân loại dữ liệu sinh hóa bằng RandomForestClassifier - Đánh giá mô hình bằng các chỉ số: Accuracy, Precision, Recall, F1 - Trực quan hóa hiệu suất mô hình - Viết mô tả Random Forest cho báo cáo |

**3. Mục lục**

[**4. Giới thiệu** 1](#_Toc198024003)

[a. Phạm vi của đồ án 1](#_Toc198024004)

[b. Mục tiêu 1](#_Toc198024005)

[c. Sự cần thiết và lý do chọn đề tài 2](#_Toc198024006)

[**5. Phân tích dữ liệu** 2](#_Toc198024007)

[a. Thu thập / Xây dựng dữ liệu 2](#_Toc198024008)

[b. Tăng cường dữ liệu / Phân tích dữ liệu 2](#_Toc198024009)

[**6. Thiết kế** 3](#_Toc198024010)

[a. Đề xuất sử dụng thuật toán 3](#_Toc198024011)

[b. Cách thức giải quyết bài toán 4](#_Toc198024012)

[**7.Thực hiện: cài đặt ứng dụng bài toán** 5](#_Toc198024013)

[**8. Kết luận và định hướng phát triển** 11](#_Toc198024014)

[**9. Tài liệu tham khảo** 12](#_Toc198024015)

[**10. Phụ lục** 13](#_Toc198024016)

# **4. Giới thiệu**

## a. Phạm vi của đồ án

Đồ án này nghiên cứu và xây dựng một hệ thống dự đoán nguy cơ viêm gan C thông qua việc phân tích các chỉ số sinh hóa của bệnh nhân. Hệ thống sử dụng các thuật toán học máy như K-Means, Apriori và KNN để xử lý dữ liệu và dự đoán nguy cơ mắc bệnh, dựa trên các chỉ số sức khỏe như ALT, AST, GGT, Albumin, và Tuổi.

Các chức năng chính của hệ thống bao gồm:

* Tiền xử lý dữ liệu: Làm sạch và chuẩn hóa dữ liệu đầu vào.
* Phân cụm bệnh nhân bằng K-Means: Phân chia bệnh nhân thành các nhóm nguy cơ (thấp, cao).
* Khai phá luật kết hợp bằng Apriori: Xác định các yếu tố sinh học có liên quan đến viêm gan.
* Dự đoán nguy cơ bằng KNN: Nhận diện bệnh nhân có dấu hiệu viêm gan dựa trên dữ liệu sức khỏe.
* Hiển thị kết quả: Cung cấp cảnh báo nguy cơ và khuyến nghị theo dõi y tế.

## b. Mục tiêu

Mục tiêu của đồ án là xây dựng một hệ thống chẩn đoán viêm gan C thông qua các mô hình học máy, với các mục tiêu cụ thể:

* Phân tích và tiền xử lý dữ liệu y tế, chuẩn hóa các chỉ số sức khỏe để phục vụ việc phân tích.
* Áp dụng thuật toán K-Means để phân cụm bệnh nhân theo mức độ nguy cơ mắc bệnh viêm gan.
* Sử dụng thuật toán Apriori để khai phá các luật kết hợp, xác định các yếu tố sinh học liên quan đến viêm gan.
* Xây dựng mô hình KNN để dự đoán nguy cơ viêm gan C và phân loại bệnh nhân vào nhóm nguy cơ.
* Trực quan hóa kết quả dự đoán và phân tích để hỗ trợ ra quyết định trong chăm sóc sức khỏe.

## c. Sự cần thiết và lý do chọn đề tài

Viêm gan C là một bệnh lý nghiêm trọng có thể dẫn đến suy gan hoặc ung thư gan nếu không được phát hiện và điều trị kịp thời. Tuy nhiên, việc chẩn đoán sớm thường gặp khó khăn do triệu chứng không rõ ràng và sự thiếu hụt cơ sở dữ liệu liên quan đến bệnh lý này. Vì vậy, hệ thống hỗ trợ chẩn đoán tự động là cần thiết trong việc phát hiện sớm nguy cơ mắc bệnh.

Sử dụng các thuật toán học máy trong phân tích y tế giúp:

* Cải thiện khả năng phân loại và dự đoán nguy cơ bệnh một cách chính xác hơn.
* Tăng cường sự minh bạch và hiểu biết về các yếu tố sinh học tác động đến viêm gan C.
* Hỗ trợ bác sĩ trong việc ra quyết định điều trị nhanh chóng và chính xác.

Đề tài có ý nghĩa lớn trong việc ứng dụng Machine Learning vào y tế, nâng cao chất lượng chẩn đoán và cải thiện hiệu quả chăm sóc sức khỏe, từ đó giảm thiểu tỷ lệ mắc bệnh viêm gan C.

# **5. Phân tích dữ liệu**

## a. Thu thập / Xây dựng dữ liệu

Dữ liệu được sử dụng trong đồ án là bộ Hepatitis C Dataset được thu thập và công bố trên nền tảng Kaggle, tại địa chỉ:  
**🔗** [**https://www.kaggle.com/datasets/fedesoriano/hepatitis-c-dataset**](https://www.kaggle.com/datasets/fedesoriano/hepatitis-c-dataset)

Bộ dữ liệu này chứa thông tin xét nghiệm sinh hóa của bệnh nhân như: tuổi, giới tính, chỉ số ALT, AST, GGT, ALP, Albumin, Bilirubin, Creatinine, Cholesterol, v.v... Ngoài ra, mỗi bệnh nhân còn được gán nhãn theo các loại: người hiến máu (Blood Donor), nhiễm virus viêm gan C (HCV), xơ gan (Cirrhosis), ung thư gan (HCC), viêm gan cấp (Hepatitis), v.v...

Dữ liệu có cấu trúc dạng bảng, dễ xử lý, nhưng vẫn còn tồn tại một số vấn đề như dữ liệu trống, nhiễu, không đồng nhất định dạng, cần được xử lý trước khi đưa vào huấn luyện mô hình.

## b. Tăng cường dữ liệu / Phân tích dữ liệu

Sau khi thu thập dữ liệu, quá trình tiền xử lý và tăng cường dữ liệu được thực hiện nhằm nâng cao chất lượng và khả năng phân tích:

* Làm sạch dữ liệu: Các bản ghi thiếu thông tin quan trọng hoặc bị sai định dạng được loại bỏ bằng phương thức dropna(), đồng thời loại bỏ các dòng không có nhãn phân loại hoặc có giới tính không hợp lệ. Cụ thể, dữ liệu chỉ giữ lại những dòng có giá trị 'm' hoặc 'f' trong cột giới tính (Sex) và loại trừ các bản ghi trống trong cột phân loại bệnh (Category). Việc này đảm bảo tập dữ liệu đầy đủ, chính xác và đồng nhất trước khi đưa vào mô hình học máy.
* Mã hóa dữ liệu: Các thuộc tính dạng chuỗi như giới tính (Sex) được chuyển đổi thành giá trị số (0, 1) bằng kỹ thuật Label Encoding để mô hình có thể xử lý.
* Chuẩn hóa dữ liệu: Dữ liệu dạng số được chuẩn hóa bằng StandardScaler nhằm đưa các chỉ số xét nghiệm như ALT, AST, GGT, ALB... về cùng một thang đo, giúp mô hình không bị thiên lệch bởi các thuộc tính có đơn vị đo khác nhau.
* Tạo thuộc tính mới: Một số đặc trưng mới được tính toán và bổ sung vào tập dữ liệu như: tỷ lệ ALT/AST (chỉ số men gan tương đối), tổng Bilirubin (biểu hiện chức năng gan)... nhằm tăng khả năng phát hiện nguy cơ bệnh.
* Gán nhãn nhị phân: Các nhãn phân loại ban đầu được nhóm lại thành hai lớp: "Không bệnh" (Blood Donor) và "Có nguy cơ" (bao gồm HCV, Hepatitis, Cirrhosis, HCC...) để phục vụ bài toán phân loại nhị phân.

Bên cạnh đó, các phương pháp trực quan hóa dữ liệu như biểu đồ histogram và boxplot được sử dụng để quan sát sự phân bố các chỉ số sinh hóa giữa hai nhóm. Nhờ đó, nhóm có thể đánh giá trực quan mức độ khác biệt của từng chỉ số giữa người bệnh và người không bệnh, từ đó lựa chọn các đặc trưng phù hợp và có giá trị nhất cho mô hình dự đoán.

# **6. Thiết kế**

## a. Đề xuất sử dụng thuật toán

Dựa trên đặc điểm của bài toán là phân tích, phân cụm và phân loại dữ liệu y tế liên quan đến viêm gan C, nhóm đã lựa chọn kết hợp các thuật toán học máy có tính phù hợp và hiệu quả cao:

* K-Means: dùng để phân cụm bệnh nhân theo mức độ nguy cơ (ví dụ: nhóm rủi ro thấp và rủi ro cao) dựa trên các chỉ số sinh hóa chuẩn hóa.
* Apriori: dùng để khai phá luật kết hợp, xác định các mối quan hệ thường xuyên xảy ra giữa các chỉ số sinh học ở nhóm bệnh nhân có nguy cơ.
* KNN (K-Nearest Neighbors): thuật toán phân loại được sử dụng để dự đoán khả năng mắc viêm gan C của một bệnh nhân mới dựa trên dữ liệu huấn luyện.
* Random Forest: được sử dụng song song với KNN như một mô hình phân loại mạnh, ổn định và có khả năng tổng quát hóa cao hơn, phục vụ việc so sánh độ chính xác giữa các mô hình học máy.

Các thuật toán trên đều được triển khai bằng thư viện scikit-learn, mlxtend và được tích hợp vào hệ thống với giao diện người dùng bằng Streamlit.

## b. Cách thức giải quyết bài toán

Quá trình giải quyết bài toán được triển khai tuần tự trong file demo.py, gồm các bước chính như sau:

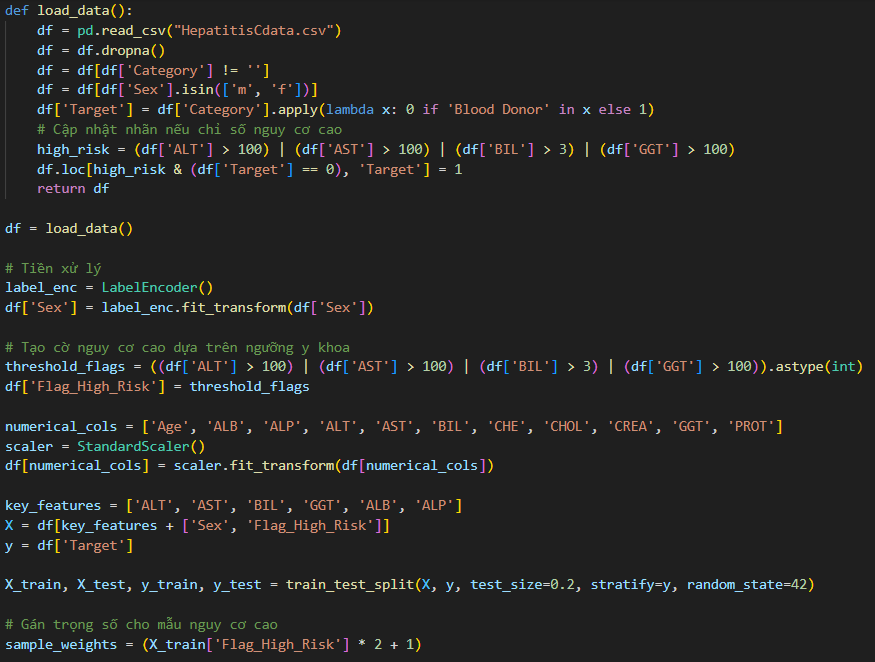
1. **Tải và làm sạch dữ liệu**:
   * Tải bộ dữ liệu HepatitisCdata.csv.
   * Loại bỏ các dòng bị thiếu (dropna()), dữ liệu trống hoặc không hợp lệ (Category rỗng, giới tính khác m/f).
2. **Tiền xử lý & tạo đặc trưng**:
   * Áp dụng LabelEncoder cho cột Sex (mã hóa giới tính).
   * Chuẩn hóa toàn bộ các chỉ số sinh hóa bằng StandardScaler.
   * Tạo thêm các thuộc tính mới như: ALT\_AST\_ratio, Bilirubin\_Total, và biến cờ Flag\_High\_Risk nhằm hỗ trợ phân tích nguy cơ.
3. **Phân cụm bằng K-Means**:
   * Dữ liệu được đưa vào mô hình KMeans(n\_clusters=2) để chia thành 2 nhóm cụ thể: nhóm có đặc điểm sinh học rủi ro cao và thấp.
   * Kết quả phân cụm được lưu vào cột Risk\_Cluster.
4. **Khai phá luật kết hợp bằng Apriori**:
   * Chuyển dữ liệu về dạng nhị phân (giá trị dương hoặc không).
   * Tìm tập mục phổ biến và khai phá các **luật kết hợp mạnh** dựa trên ngưỡng support và confidence do người dùng điều chỉnh.
   * Các luật được hiển thị kèm giải thích y khoa để hỗ trợ hiểu rõ mối liên hệ giữa các chỉ số sinh hóa.
5. **Huấn luyện và dự đoán bằng KNN & Random Forest**:
   * Tách dữ liệu thành tập huấn luyện và kiểm tra.
   * Huấn luyện mô hình KNeighborsClassifier(n\_neighbors=5) và RandomForestClassifier(n\_estimators=100).
   * Cả hai mô hình được đánh giá bằng các chỉ số: Accuracy, Precision, Recall, F1-Score.
   * Người dùng có thể nhập dữ liệu mới (qua giao diện), mô hình sẽ đưa ra dự đoán liệu bệnh nhân có nguy cơ viêm gan hay không.
6. **Giao diện người dùng & hiển thị kết quả**:
   * Hệ thống hiển thị bảng dữ liệu, biểu đồ histogram/boxplot, các luật kết hợp và kết quả phân loại.
   * Cảnh báo chỉ số vượt ngưỡng (ALT, AST, BIL, GGT) được hiển thị rõ ràng.
   * Cho phép so sánh trực tiếp hiệu suất giữa KNN và Random Forest.

# **7.Thực hiện: cài đặt ứng dụng bài toán**

Ứng dụng được xây dựng bằng ngôn ngữ Python và triển khai trên nền tảng Streamlit – một framework đơn giản nhưng mạnh mẽ dùng để xây dựng giao diện web tương tác nhanh. Mục tiêu chính là giúp người dùng nhập dữ liệu sức khỏe, sau đó ứng dụng sẽ xử lý, phân tích, dự đoán nguy cơ viêm gan C và hiển thị kết quả dưới dạng cảnh báo và trực quan hóa.

Dưới đây là chi tiết từng phần trong file demo.py đã được triển khai, phân tích theo dòng lệnh và mục đích học thuật.

**1. Tải và làm sạch dữ liệu**

****

Tải và làm sạch dữ liệu: đọc file CSV, xóa các dòng bị thiếu dữ liệu, loại bỏ dòng không có nhãn hoặc giới tính không hợp lệ.

Gán nhãn bệnh: nếu là “Blood Donor” thì gán Target = 0, ngược lại là 1. Tuy nhiên, nếu có các chỉ số ALT, AST, BIL hoặc GGT vượt ngưỡng thì dù là người hiến máu vẫn bị gán lại thành Target = 1 để phản ánh nguy cơ tiềm ẩn.

Mã hóa và chuẩn hóa: giới tính được mã hóa thành số bằng LabelEncoder, các chỉ số sinh học được chuẩn hóa bằng StandardScaler để đưa về cùng thang đo.

Tạo cờ nguy cơ cao: thêm cột Flag\_High\_Risk với giá trị 1 nếu chỉ số ALT, AST, BIL hoặc GGT vượt ngưỡng y khoa, ngược lại là 0.

Tạo tập đặc trưng và nhãn: chọn ra các cột ALT, AST, BIL, GGT, ALB, ALP cùng với Sex và Flag\_High\_Risk làm đầu vào X, và Target làm nhãn y.

Chia train/test và gán trọng số: dữ liệu được chia 80/20 cho train và test, với các mẫu có Flag\_High\_Risk = 1 được **nhân trọng số gấp 3 để mô hình ưu tiên học sâu hơn vào nhóm có nguy cơ cao.**

**2 Train model**



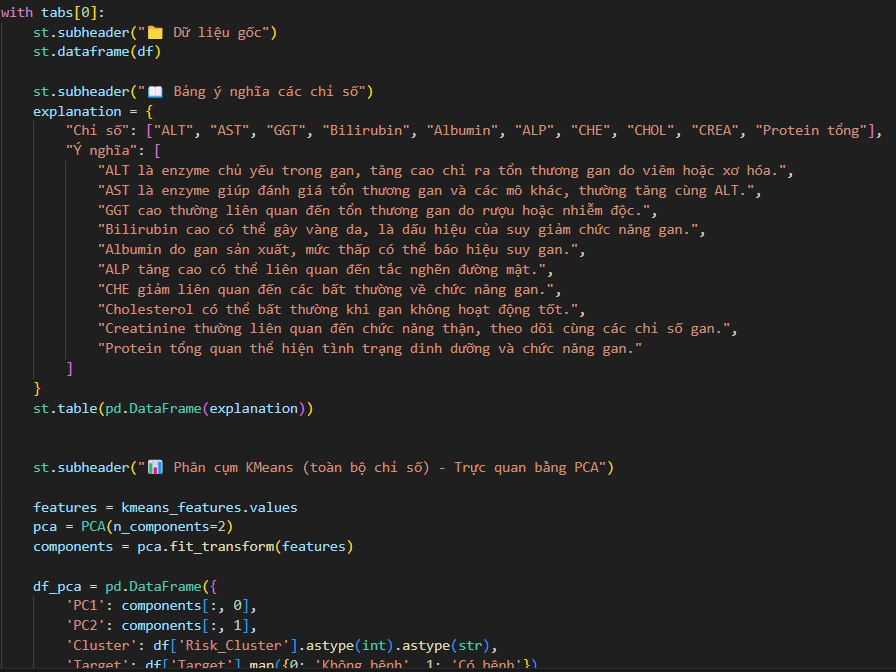
Huấn luyện mô hình: khởi tạo hai mô hình Random Forest và KNN với các tham số phù hợp, trong đó Random Forest dùng 100 cây và class\_weight='balanced' để xử lý mất cân bằng nhãn, KNN dùng 5 láng giềng và tính khoảng cách có trọng số. Cả hai mô hình được huấn luyện với tập dữ liệu đã xử lý, riêng Random Forest được cung cấp thêm sample\_weights để ưu tiên mẫu nguy cơ cao.

Trực quan độ quan trọng đặc trưng: sau khi huấn luyện, mô hình Random Forest trả về chỉ số quan trọng của từng đặc trưng. Các đặc trưng được sắp xếp giảm dần theo độ quan trọng để phân tích xem chỉ số nào ảnh hưởng mạnh đến dự đoán viêm gan.

Đánh giá mô hình: định nghĩa hàm evaluate\_model để tính các chỉ số đánh giá cơ bản gồm Accuracy, Precision, Recall, và F1. Cả hai mô hình đều được đánh giá trên tập test và lưu lại kết quả để so sánh sau.

Phân cụm KMeans: chọn toàn bộ cột số trong df (trừ Target) để đưa vào mô hình KMeans. Khởi tạo phân cụm với 2 cụm (nguy cơ thấp và cao), sau đó gán kết quả vào cột Risk\_Cluster để sau này trực quan hoặc đánh giá theo nhóm nguy cơ không giám sát.

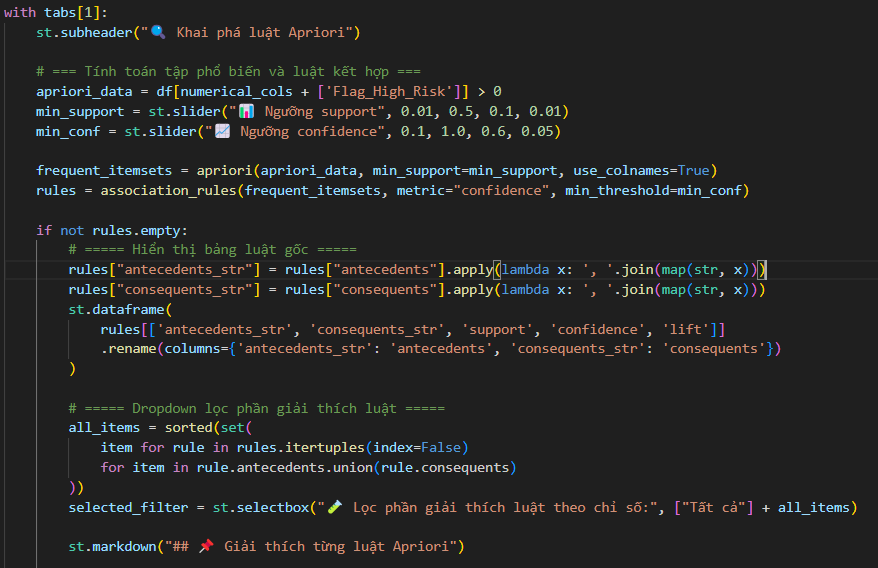
**3 Tab 0 trang tài liệu và phân cụm K means**



Hiển thị dữ liệu và ý nghĩa chỉ số: trong tab đầu tiên, ứng dụng hiển thị bảng dữ liệu gốc df đã được xử lý, sau đó trình bày thêm một bảng giải thích ngắn gọn ý nghĩa y khoa của 10 chỉ số quan trọng như ALT, AST, GGT, Albumin, v.v. nhằm giúp người dùng hiểu rõ hơn về ý nghĩa sinh học và lâm sàng của từng thông số trong dự đoán bệnh.

Trực quan hóa phân cụm KMeans: toàn bộ các chỉ số dạng số được lấy ra làm đầu vào cho PCA để giảm chiều về 2 thành phần chính, rồi được kết hợp với nhãn phân cụm từ KMeans và nhãn thực tế (Target). Kết quả được thể hiện dưới dạng biểu đồ tán xạ (scatter plot) 2D với các điểm màu theo cụm và biểu tượng theo tình trạng bệnh, giúp trực quan hóa mối quan hệ giữa phân cụm không giám sát và nhãn bệnh thật sự trong dữ liệu.

**4 Tab 1 luật kết hợp áp dụng thuật toán Apriori**



Tạo tập dữ liệu nhị phân: toàn bộ các chỉ số sinh học đã chuẩn hóa cùng với Flag\_High\_Risk được chuyển thành giá trị nhị phân (0 hoặc 1), dùng làm dữ liệu đầu vào cho thuật toán Apriori.

Chọn ngưỡng khai phá: người dùng điều chỉnh ngưỡng support và confidence bằng hai thanh trượt. Các ngưỡng này kiểm soát việc lọc ra tập phổ biến và các luật kết hợp đủ mạnh.

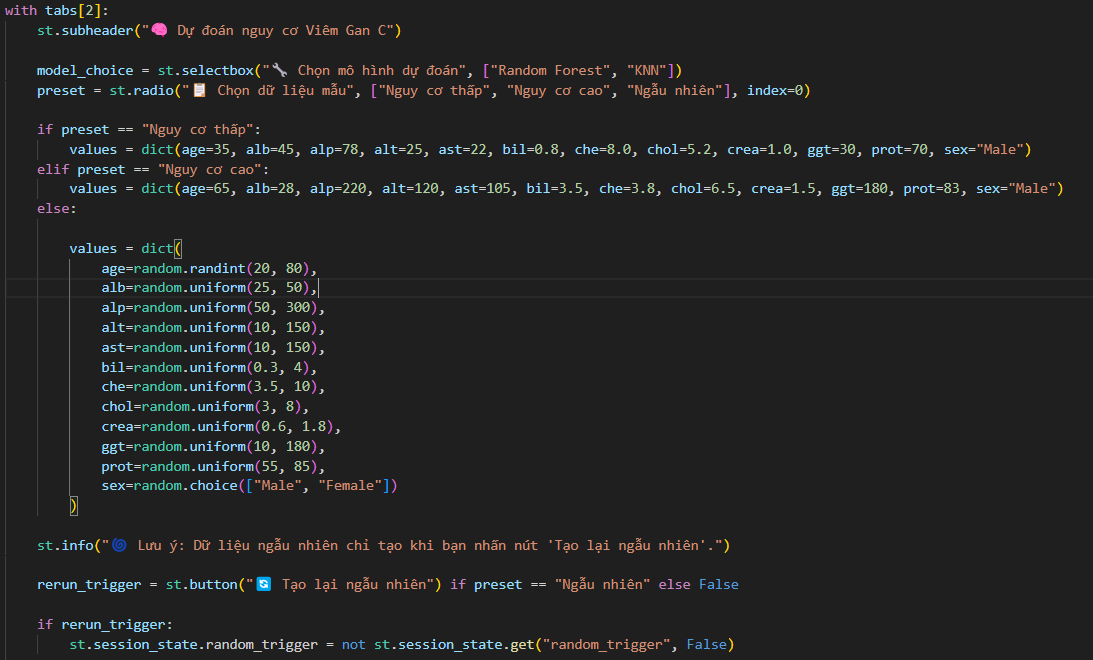
Khai phá luật kết hợp: áp dụng thuật toán Apriori để tìm các tập mục phổ biến có support ≥ ngưỡng, sau đó khai thác các luật kết hợp có confidence ≥ ngưỡng người dùng chọn.

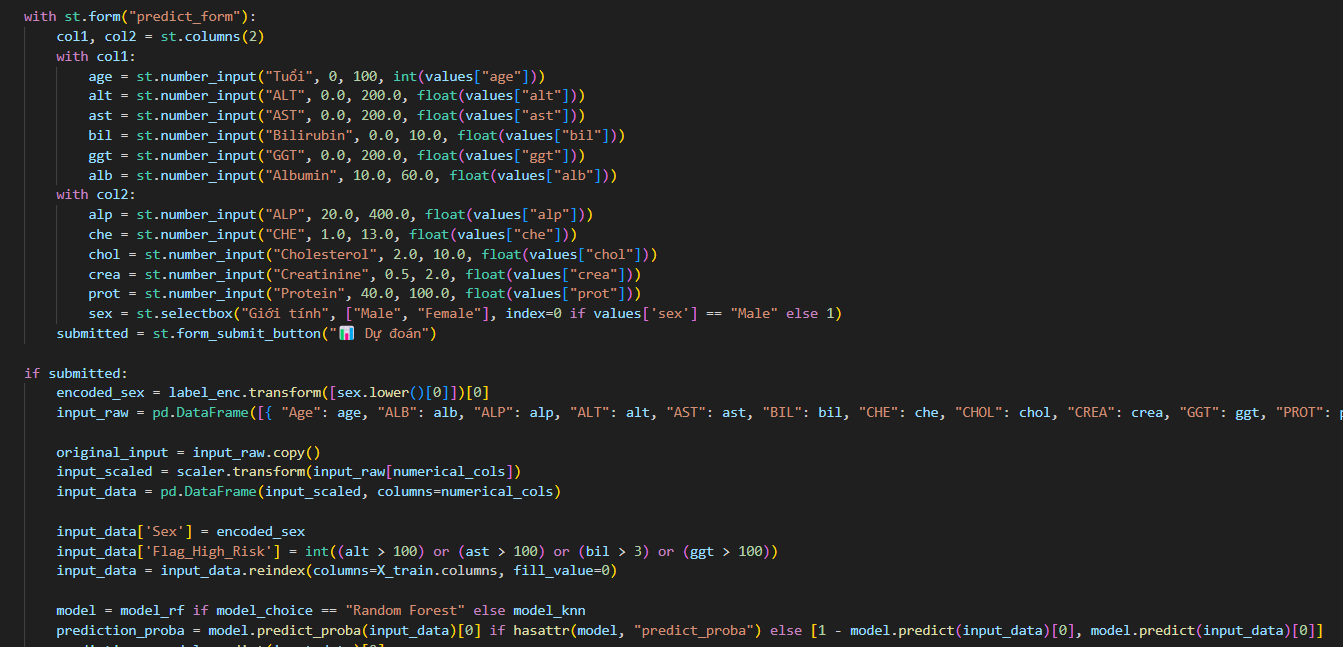
Hiển thị bảng luật: nếu có luật được tìm thấy, hệ thống chuyển các tập antecedents và consequents thành chuỗi, rồi hiển thị bảng chứa các luật với chỉ số support, confidence, lift, giúp người dùng có cái nhìn tổng quan.

Lọc và giải thích luật: tạo dropdown cho phép lọc giải thích luật theo từng chỉ số cụ thể hoặc hiển thị tất cả. Với mỗi luật, hệ thống hiển thị confidence, lift và phần giải thích y khoa cho từng chỉ số xuất hiện trong vế trái của luật, giúp người dùng hiểu nguyên nhân sinh học có thể dẫn đến kết luận trong luật.

Trường hợp không có luật: nếu không có luật nào thỏa mãn ngưỡng đã chọn, hệ thống hiện cảnh báo để người dùng điều chỉnh lại giá trị threshold.

**5 Tab 2 dự đoán bệnh viêm gan C áp dụng cái mô hình K NN và Random Forest và luật kết hợp dựa theo hồ sơ áp dụng Apriori**







Chọn mô hình và dữ liệu đầu vào: người dùng có thể chọn mô hình dự đoán (Random Forest hoặc KNN), đồng thời chọn sẵn bộ dữ liệu mẫu gồm ba kiểu: nguy cơ thấp, nguy cơ cao hoặc ngẫu nhiên. Nếu chọn ngẫu nhiên, các chỉ số sẽ được sinh ngẫu nhiên trong khoảng hợp lý và chỉ thay đổi khi người dùng bấm nút “Tạo lại ngẫu nhiên”.

Nhập dữ liệu thông qua form: một biểu mẫu được hiển thị chia thành hai cột, nơi người dùng có thể nhập các chỉ số sinh học (ALT, AST, Bilirubin, GGT, Albumin, ALP, CHE, Cholesterol, Creatinine, Protein tổng...) cùng với tuổi và giới tính.

Tiền xử lý dữ liệu đầu vào: sau khi người dùng nộp form, dữ liệu đầu vào được mã hóa giới tính, chuẩn hóa theo cùng chuẩn như dữ liệu huấn luyện, và được gắn cờ Flag\_High\_Risk nếu có chỉ số ALT, AST, BIL hoặc GGT vượt ngưỡng nguy hiểm. Các cột được sắp xếp đúng theo thứ tự huấn luyện ban đầu để mô hình có thể xử lý.

Dự đoán bằng mô hình đã chọn: hệ thống sử dụng mô hình Random Forest hoặc KNN để đưa ra dự đoán, đồng thời tính tỉ lệ xác suất cho cả hai trường hợp có bệnh và không bệnh. Kết quả dự đoán được hiển thị rõ ràng với thông báo màu (xanh hoặc vàng) tùy theo nguy cơ, kèm theo xác suất chi tiết.

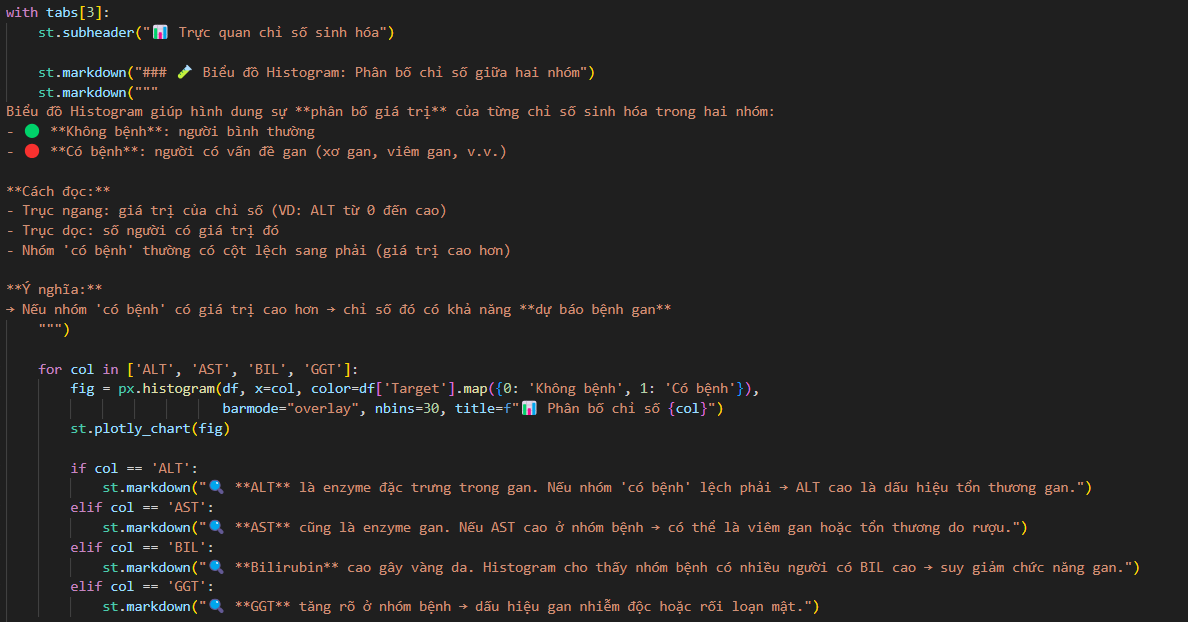
Liên kết luật Apriori với hồ sơ: từ dữ liệu đầu vào, hệ thống mã hóa nhị phân để xác định luật kết hợp nào từ Apriori có thể áp dụng cho người dùng hiện tại. Các luật phù hợp nhất (theo lift) được hiển thị để cung cấp thêm góc nhìn giải thích về mối liên hệ giữa các chỉ số và nguy cơ bệnh.

Gợi ý chăm sóc sức khỏe gan: ứng dụng cung cấp các đường dẫn tham khảo về chế độ ăn uống, thực phẩm cần tránh, bài tập thể dục hỗ trợ gan và tài liệu tìm hiểu về viêm gan C để hỗ trợ người dùng nâng cao ý thức phòng bệnh.

So sánh chỉ số với ngưỡng chuẩn: hệ thống so sánh các chỉ số người dùng nhập với giá trị bình thường trong y học. Mỗi chỉ số được phân loại rõ ràng là “Bình thường” hay “Ngoài ngưỡng” để người dùng biết chỉ số nào đáng lưu ý, trong đó đơn vị creatinine được chuyển về µmol/L cho đúng chuẩn.

Hiển thị hiệu suất mô hình: mô hình dự đoán sẽ được đánh giá lại trên tập test, với các chỉ số Accuracy, Precision, Recall và F1-score được trình bày gọn trong bốn cột, giúp người dùng so sánh hiệu quả mô hình đang chọn.

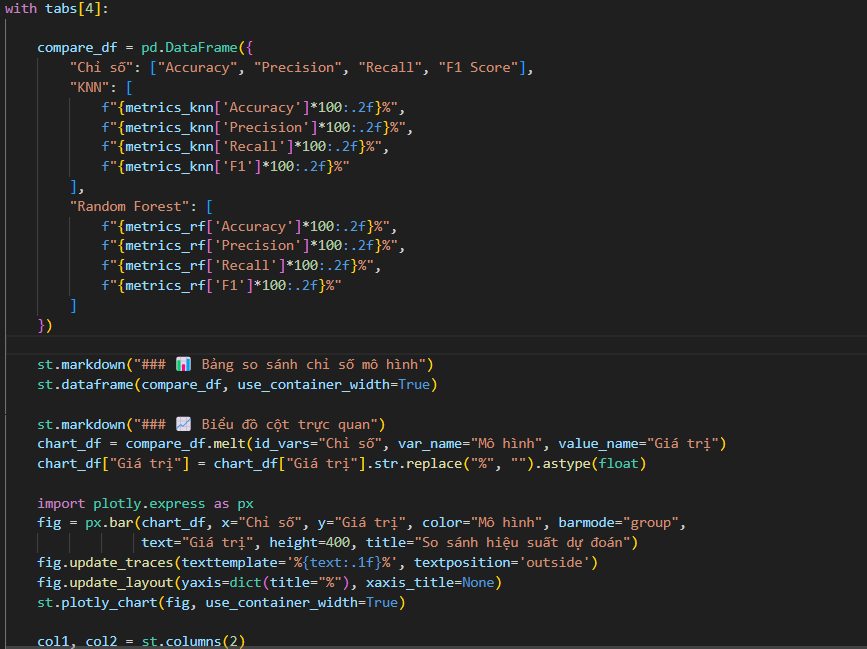
**6 Tab 3 các mô hình dùng cho chuyên viên bác sĩ dựa theo dữ liệu hiện có**



Biểu đồ Histogram (phân bố): hệ thống tạo biểu đồ histogram cho bốn chỉ số ALT, AST, BIL và GGT để thể hiện sự phân bố giữa hai nhóm: có bệnh và không bệnh. Mỗi biểu đồ dùng màu để phân biệt nhóm bệnh và không bệnh, với thanh trượt thể hiện mật độ giá trị. Người dùng dễ dàng nhận thấy nhóm bệnh thường có giá trị chỉ số lệch phải (cao hơn), điều này giúp xác định các chỉ số nào có khả năng dự báo viêm gan mạnh nhất. Sau mỗi biểu đồ là phần giải thích ngắn gọn về ý nghĩa lâm sàng của chỉ số đó, như ALT cao → tổn thương gan, BIL cao → vàng da do suy gan,…

Biểu đồ Boxplot (so sánh): sau phần histogram, ứng dụng chuyển sang biểu đồ boxplot để thể hiện trung vị, khoảng dao động và các giá trị bất thường của từng chỉ số giữa hai nhóm. Mỗi biểu đồ giúp người xem so sánh trực quan chỉ số ở nhóm bệnh và không bệnh: nếu hộp của nhóm bệnh cao hơn rõ rệt thì chứng tỏ chỉ số đó phân biệt được tình trạng gan. Phần giải thích dưới mỗi biểu đồ giúp củng cố ý nghĩa: ví dụ GGT cao ở nhóm bệnh có thể là dấu hiệu gan nhiễm độc hoặc tắc mật, AST có nhiều outlier là cảnh báo tổn thương gan nặng,…

**7 Tab 4 so sánh tỉ lệ dự đoán của 2 mô hình**



So sánh bảng hiệu suất: hệ thống tổng hợp các chỉ số đánh giá gồm Accuracy, Precision, Recall và F1 Score của hai mô hình KNN và Random Forest. Các giá trị được hiển thị theo phần trăm và sắp xếp gọn trong bảng giúp người dùng dễ dàng so sánh trực tiếp hiệu quả giữa hai mô hình.

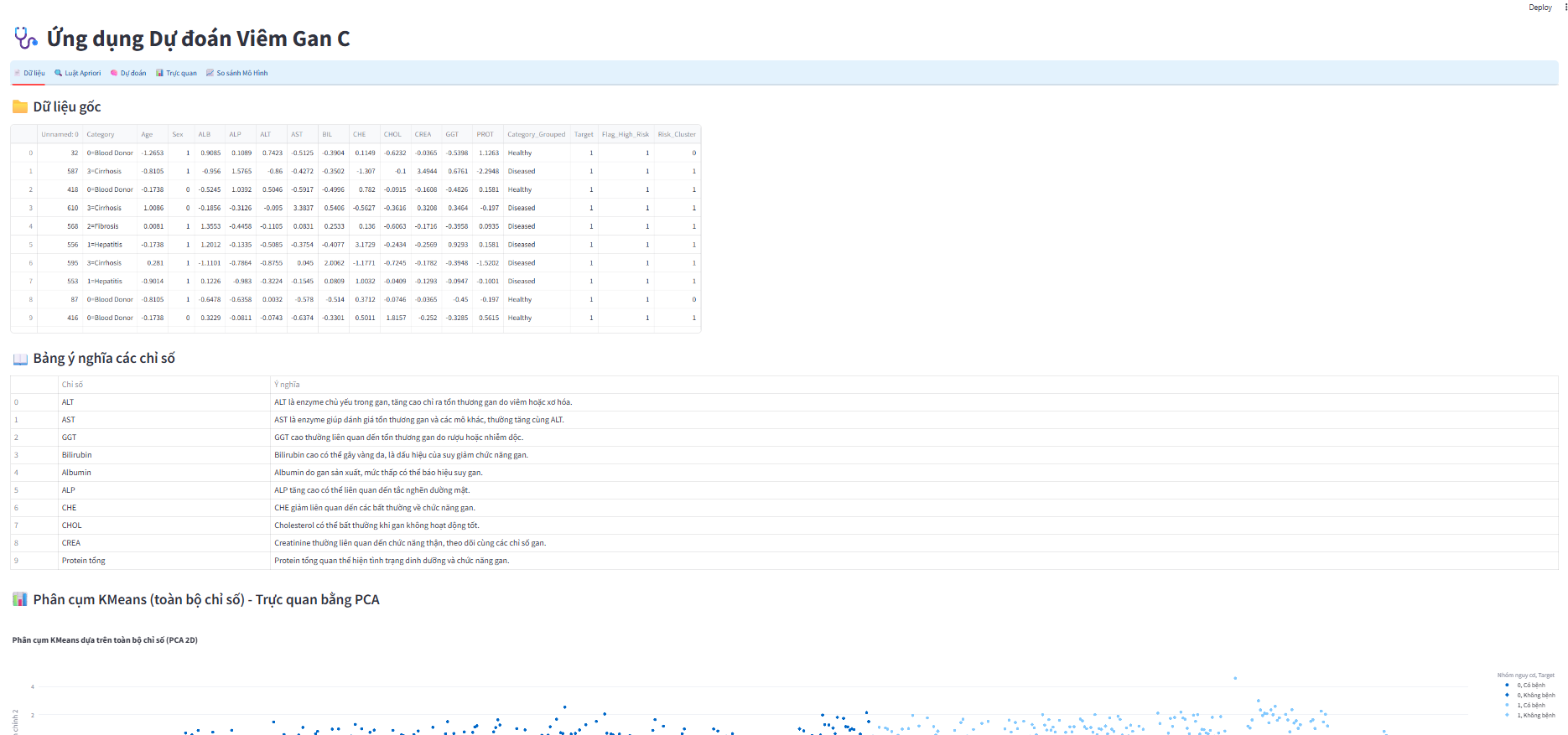
Biểu đồ trực quan hóa hiệu suất: cùng với bảng số liệu, một biểu đồ cột được dựng bằng Plotly để trực quan hóa hiệu suất hai mô hình theo từng chỉ số. Mỗi cột hiển thị giá trị cụ thể bên ngoài, giúp tăng khả năng đọc nhanh và nhận biết mô hình nào đang vượt trội ở tiêu chí nào.

Đánh giá ưu – nhược điểm từng mô hình: nội dung được chia thành hai cột. Bên trái là phần mô tả KNN – một mô hình đơn giản, dễ hiểu, phù hợp với dữ liệu nhỏ và rõ ràng, nhưng lại yếu trong môi trường dữ liệu nhiễu hoặc lớn. Bên phải là Random Forest – mô hình mạnh mẽ hơn, kháng nhiễu tốt, có khả năng giải thích thuộc tính quan trọng, tuy nhiên khó lý giải từng cây và tốn thời gian huấn luyện hơn.

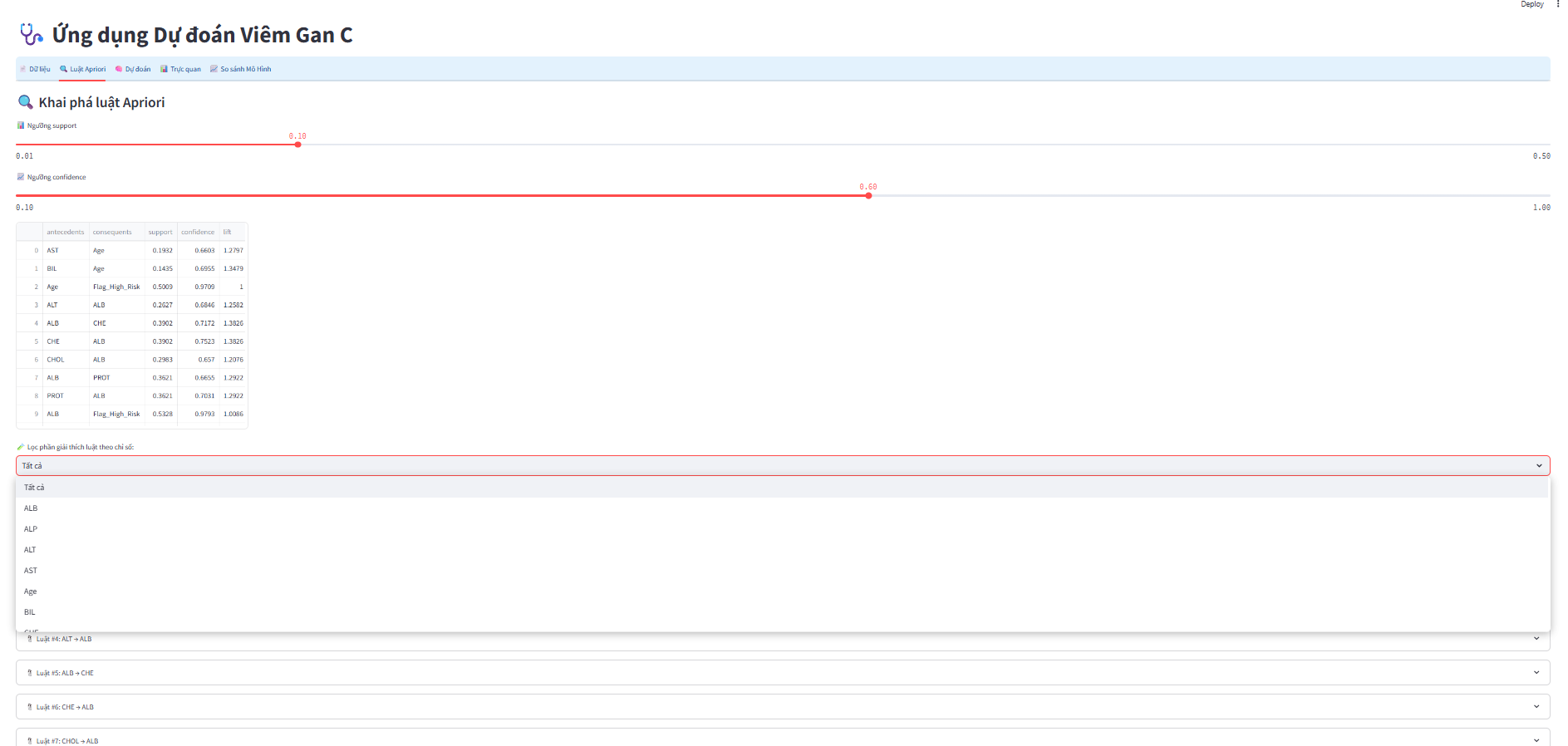
Kết luận định hướng chọn mô hình: ứng dụng gợi ý rõ ràng: KNN phù hợp với dự án nhỏ, minh bạch, dữ liệu gọn gàng; trong khi Random Forest phù hợp để triển khai thật, nơi dữ liệu có thể phức tạp và yêu cầu độ chính xác cao.

**8 Kết quả chạy ứng dụng dự đoán**

Giao diện đầu vào

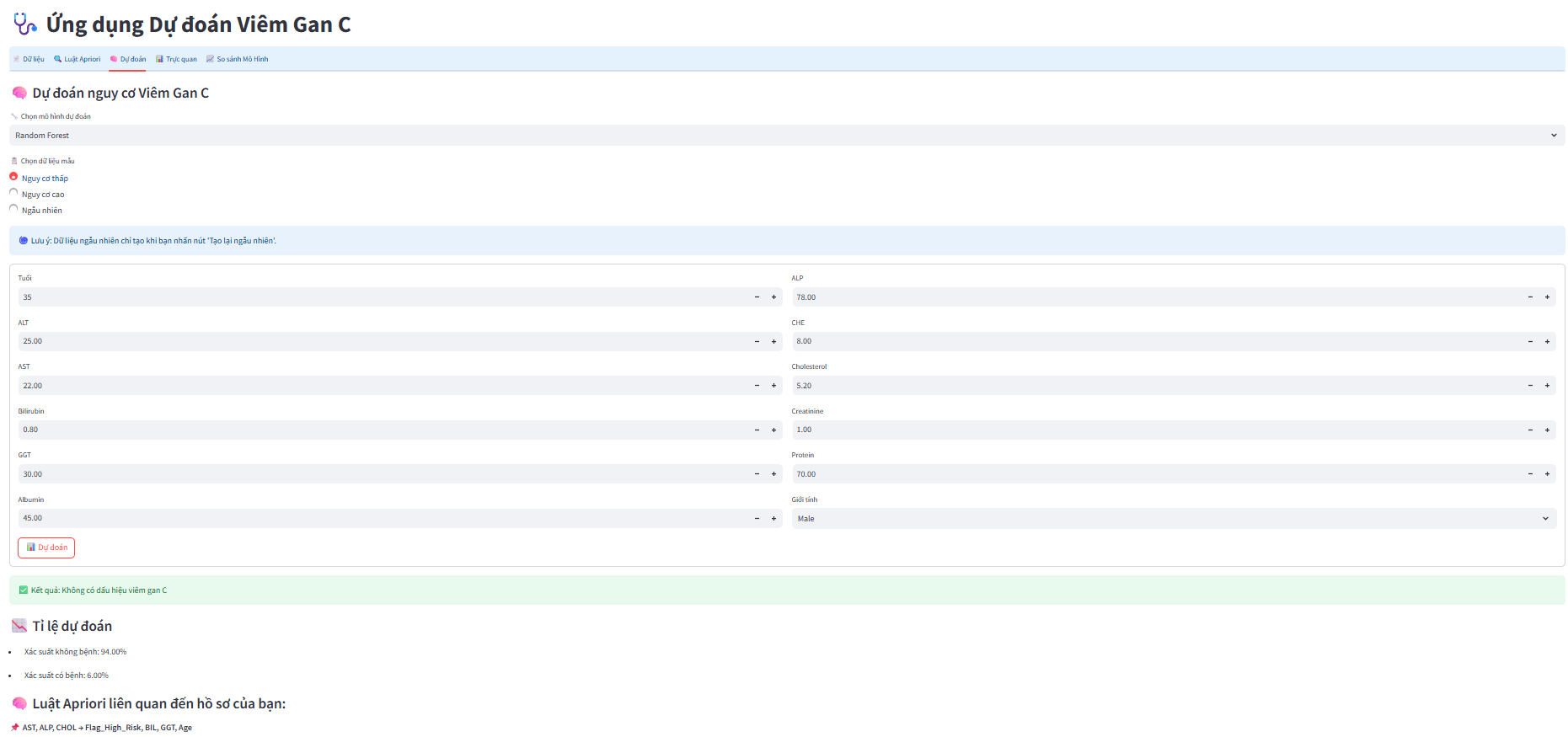


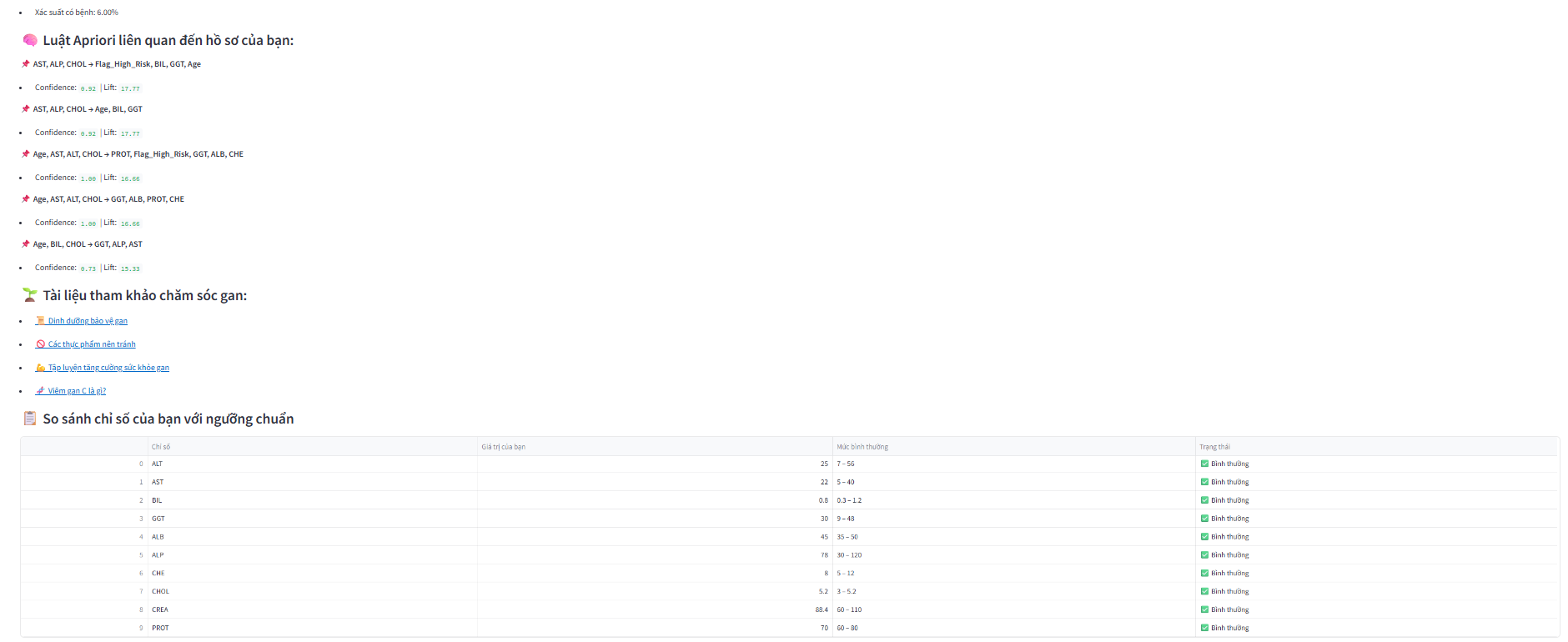
Luật kết hợp Apriori



Mô hình Random Forest

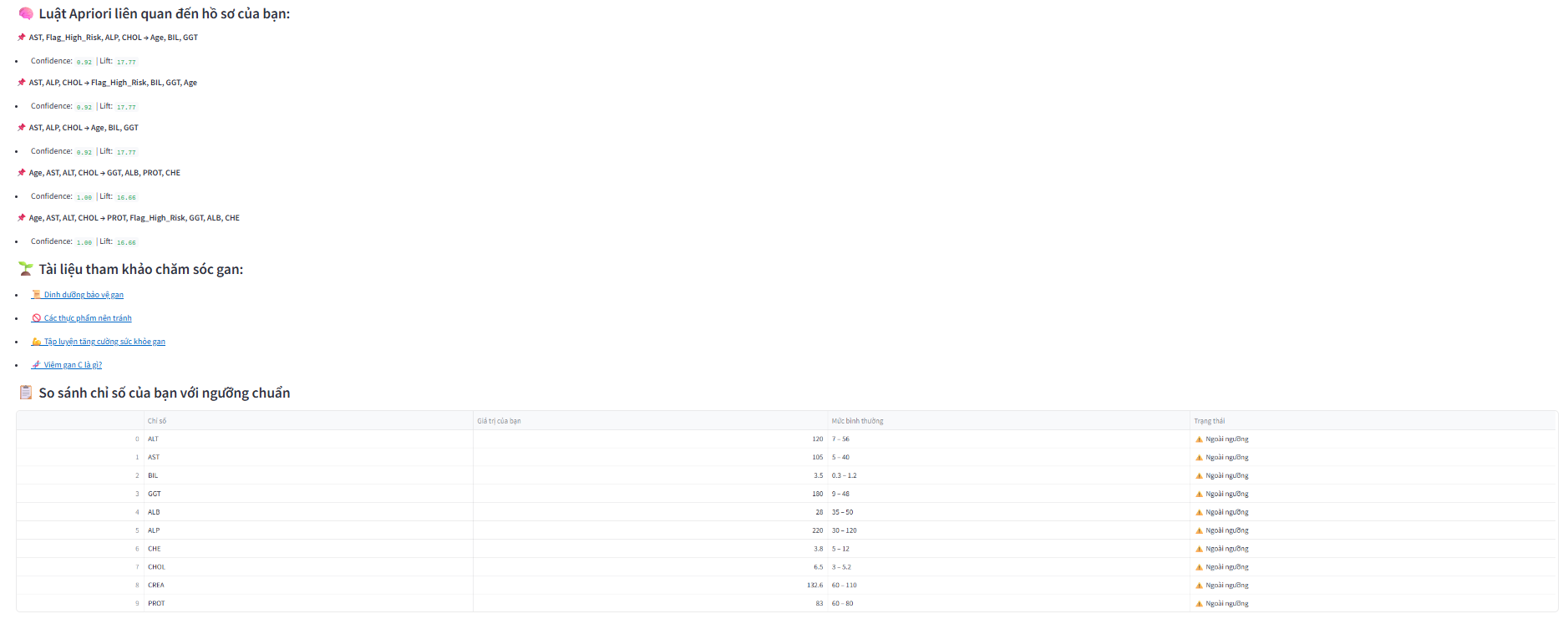
Dự đoán nguy cơ thấp với chỉ số thấp





Dự đoán nguy cơ cao và sẽ cảnh báo các yêu tố cần kiểm tra



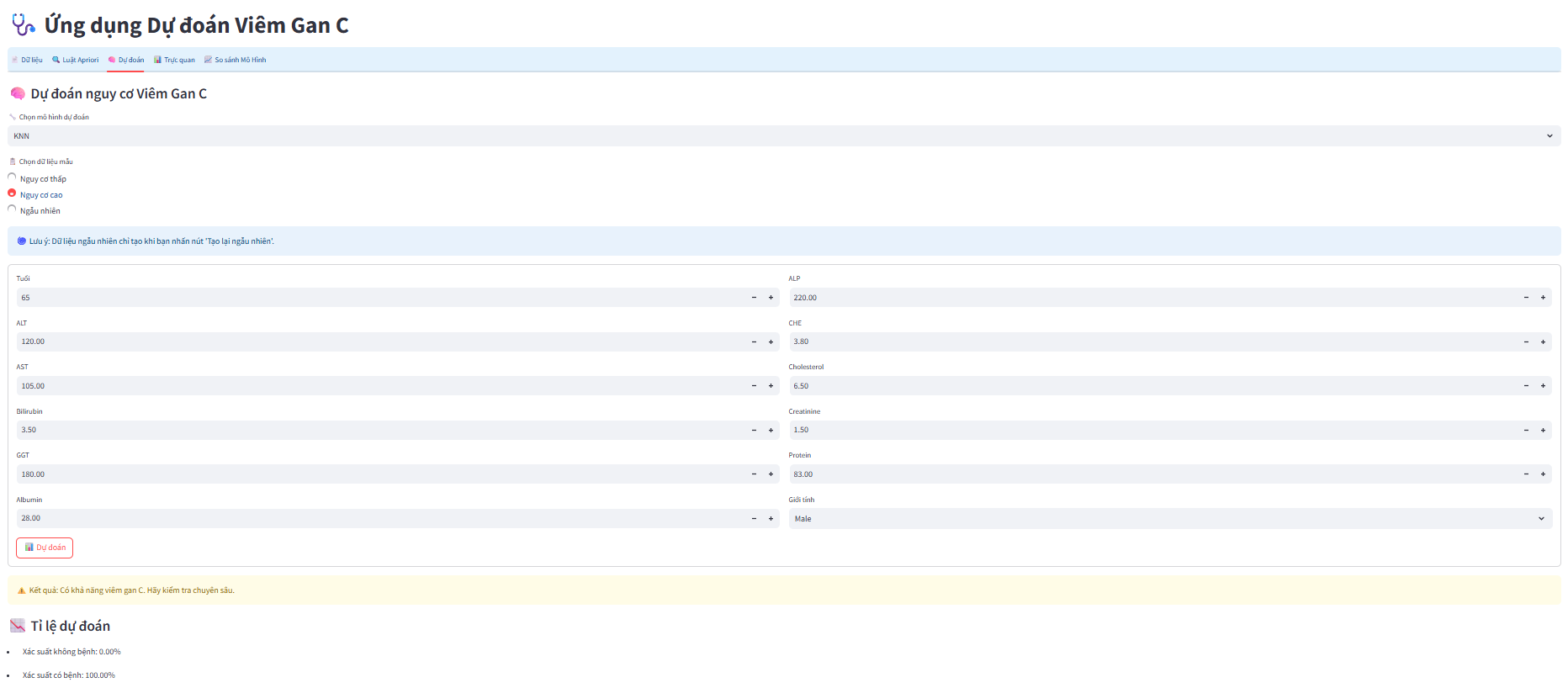


Mô hình KNN

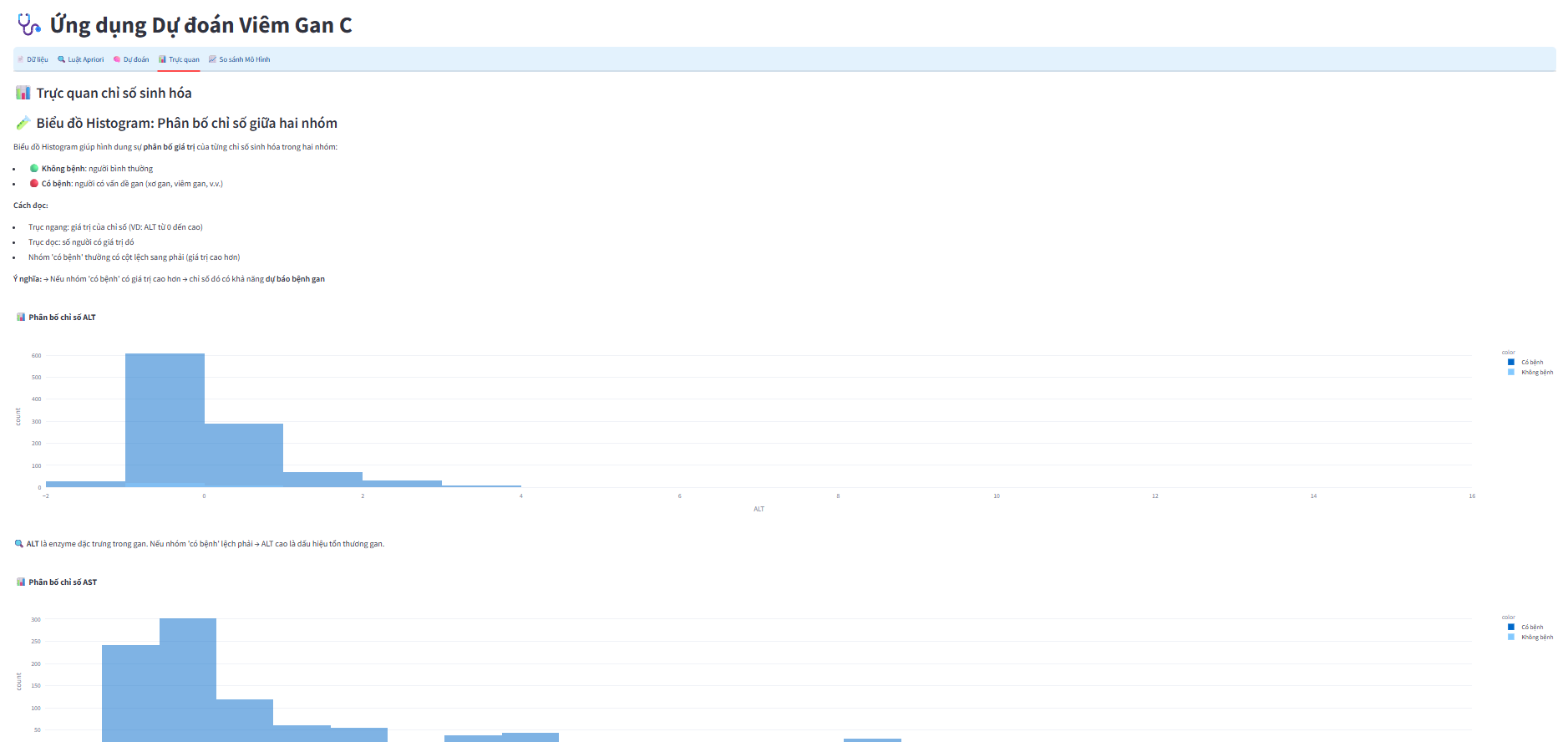
Dự đoán nguy cơ thấp với chỉ số thấp



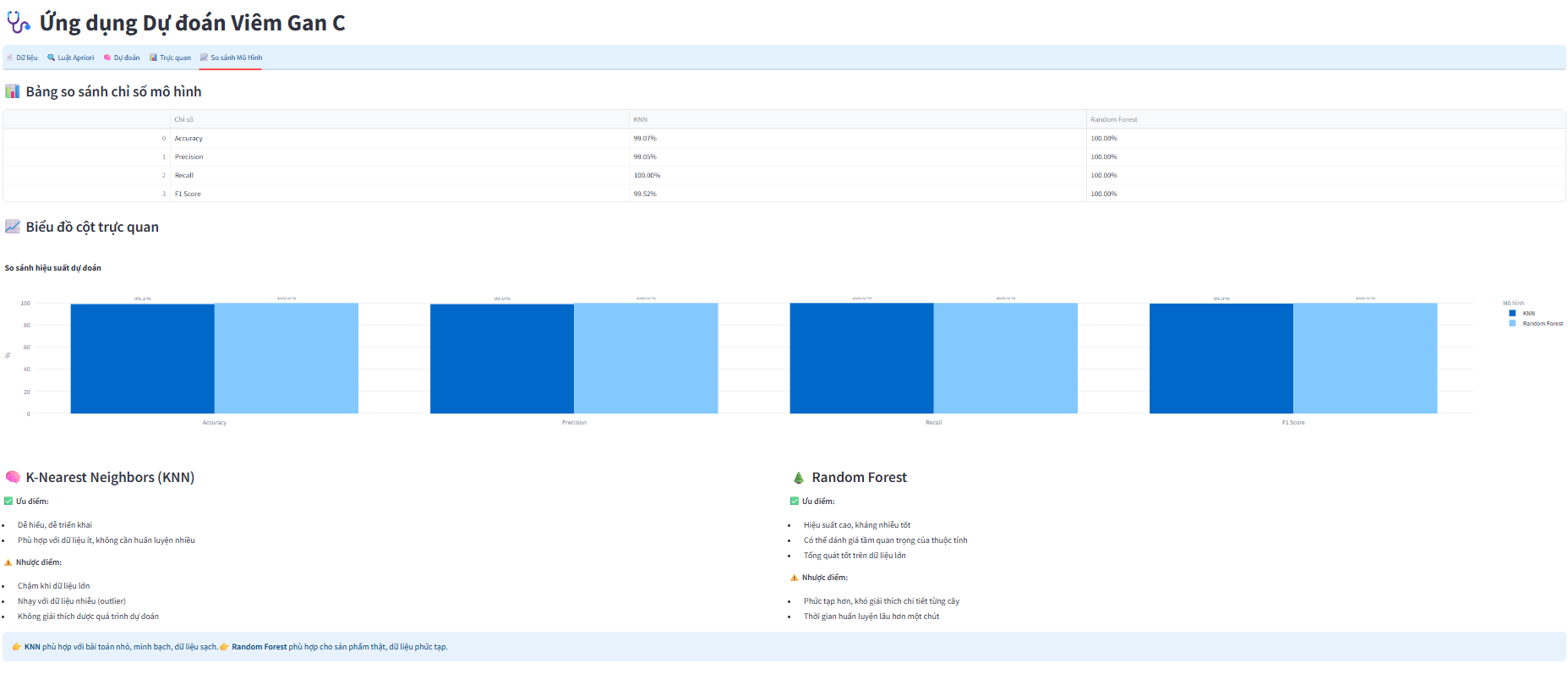
Dự đoán nguy cơ cao và sẽ cảnh báo các yêu tố cần kiểm tra



Các biểu đồ báo cáo dành cho các nhân viên y tế bác sĩ



So sánh mô hình



# **8. Kết luận và định hướng phát triển**

**Kết luận**

Qua quá trình nghiên cứu, triển khai và thử nghiệm, nhóm đã xây dựng thành công một hệ thống ứng dụng học máy nhằm dự đoán nguy cơ viêm gan C dựa trên các chỉ số sinh hóa của bệnh nhân. Hệ thống tích hợp đầy đủ các khâu:

* Tiền xử lý dữ liệu: loại bỏ nhiễu, mã hóa, chuẩn hóa và tạo đặc trưng mới.
* Phân cụm không giám sát: sử dụng K-Means để phân nhóm bệnh nhân theo mức độ rủi ro.
* Khai phá luật kết hợp: áp dụng thuật toán Apriori nhằm tìm ra các mối quan hệ thường xuyên giữa các chỉ số y học.
* Phân loại bằng mô hình học máy: triển khai và so sánh hai mô hình KNN và Random Forest.
* Giao diện trực quan: phát triển ứng dụng web bằng Streamlit cho phép người dùng không chuyên sử dụng dễ dàng.

Kết quả thực nghiệm cho thấy mô hình Random Forest cho độ chính xác cao hơn KNN, đồng thời ổn định hơn với dữ liệu mới. Các biểu đồ trực quan và cảnh báo chỉ số vượt ngưỡng giúp tăng tính minh bạch, giải thích được kết quả cho người dùng.

Ứng dụng không chỉ là một bài toán kỹ thuật mà còn mang tính ứng dụng thực tiễn cao, góp phần hỗ trợ người dùng trong việc theo dõi, nhận diện sớm các dấu hiệu bệnh lý gan, từ đó đưa ra quyết định chăm sóc sức khỏe chủ động và khoa học hơn.

**Định hướng phát triển**

Dựa trên phiên bản hiện tại, hệ thống hoàn toàn có thể được mở rộng và cải tiến theo các hướng sau:

**1. Mở rộng mô hình học máy**

* Tích hợp thêm các thuật toán nâng cao như XGBoost, LightGBM, hoặc SVM để cải thiện hiệu quả dự đoán.
* Triển khai AutoML để tự động chọn mô hình tối ưu và giảm thời gian tinh chỉnh.

**2. Nâng cấp giao diện người dùng**

* Xây dựng giao diện nâng cao bằng ReactJS hoặc Flask + HTML/CSS để tăng tính linh hoạt, thẩm mỹ và tích hợp dễ dàng trên nhiều nền tảng (web, mobile).
* Cho phép người dùng tải file dữ liệu hoặc lịch sử khám bệnh để dự đoán nhiều bệnh nhân cùng lúc.

**3. Tích hợp dữ liệu thực**

* Kết nối với hệ thống thông tin y tế (EMR – Electronic Medical Record) để lấy dữ liệu thật từ bệnh viện.
* Ứng dụng có thể đóng vai trò như một công cụ hỗ trợ bác sĩ trong khám sàng lọc.

**4. Đa bệnh lý**

* Mở rộng phạm vi ứng dụng sang các bệnh lý khác như gan nhiễm mỡ, tiểu đường, huyết áp cao,… nếu có dữ liệu phù hợp.

**5. Triển khai trên nền tảng thực tế**

* Đưa ứng dụng lên Streamlit Cloud, Heroku, hoặc Render để chia sẻ với cộng đồng và bác sĩ.
* Tích hợp tính năng xuất kết quả dạng PDF cho bệnh nhân sau khi kiểm tra.

# **9. Tài liệu tham khảo**

Kaggle. (2021). *Hepatitis C Dataset*. Truy cập tại:<https://www.kaggle.com/datasets/fedesoriano/hepatitis-c-dataset>

*Tài liệu môn Trí tuệ nhân tạo* – Khoa Công nghệ thông tin, Trường Đại học Công Thương TP.HCM.

*Tài liệu môn Khai thác dữ liệu* – Khoa Công nghệ thông tin, Trường Đại học Công Thương TP.HCM.

Đỗ Anh Đức. (2017). *Phân cụm K-Means – Machine Learning cơ bản*. Truy cập từ: https://machinelearningcoban.com/2017/01/01/kmeans/

Wikipedia Tiếng Việt. (2023). *Phân cụm K-means*. Truy cập từ: <https://vi.wikipedia.org/wiki/Ph%C3%A2n_c%E1%BB%A5m_k-means>

Đỗ Anh Đức. (2021). *Random Forest – Machine Learning cho dữ liệu dạng bảng*. Truy cập từ:  
https://machinelearningcoban.com/tabml\_book/ch\_model/random\_forest.html

Phạm Đình Khánh. (2022). *Giới thiệu về mô hình rừng cây (Random Forest)*. Truy cập từ:  
https://phamdinhkhanh.github.io/deepai-book/ch\_ml/index\_RandomForest.htm

Viblo. (2020). *Thuật toán Apriori khai phá luật kết hợp trong Data Mining*. Truy cập từ:  
https://viblo.asia/p/thuat-toan-apriori-khai-pha-luat-ket-hop-trong-data-mining-3P0lPEv85ox

Lê Văn Đông. (2022). *Khai thác luật kết hợp với thuật toán Apriori*. Truy cập từ:  
https://www.levandong.com/luat-ket-hop-thuat-toan-apriori-data-mining/

Đỗ Anh Đức. (2017). *Giải thuật KNN – Machine Learning cơ bản*. Truy cập từ:  
https://machinelearningcoban.com/2017/01/08/knn/

Wikipedia Tiếng Việt. (2023). *Giải thuật k hàng xóm gần nhất*. Truy cập từ:  
[https://vi.wikipedia.org/wiki/Gi%E1%BA%A3i\_thu%E1%BA%ADt\_k\_h%C3%A0ng\_x%C3%B3m\_g%E1%BA%A7n\_nh%E1%BA%A5](https://vi.wikipedia.org/wiki/Gi%E1%BA%A3i_thu%E1%BA%ADt_k_h%C3%A0ng_x%C3%B3m_g%E1%BA%A7n_nh%E1%BA%A5t)

# **10. Phụ lục**

|  |  |
| --- | --- |
|  | Hình ảnh đầu vào |
|  | Hình ảnh áp dụng thuật toán Apriori để tìm luật kết hợp |
|  | Hình ảnh áp dụng thuật toán Random Forest để áp dụng dự báo  Kết quả nguy cơ thấp |
|  | Hình ảnh áp dụng thuật toán Random Forest để áp dụng dự báo  Kết quả nguy cơ cao và cảnh báo các số liệu ở mức cao |
|  | Hình ảnh áp dụng thuật toán KNN để áp dụng dự báo  Kết quả nguy cơ thấp |
|  | Hình ảnh áp dụng thuật toán KNN để áp dụng dự báo  Kết quả nguy cơ cao và cảnh báo các số liệu ở mức cao |
|  | Hình ảnh biểu đồ dành cho các nhân viên y tế bác sĩ |
|  | Hình ảnh So sánh giữa K NN và Random Forest |