**5.1. Random Forest Classifier**

Hypermeter tuning values for Dataset I (Cleveland) && Dataset II (IEEE)

|  |  |  |
| --- | --- | --- |
| Dataset | Classifier | GridsearchCV Hypermeter Tuning Values |
| I | RF | n\_estimators = 500, random\_state = 42, max\_leaf\_nodes = 20, min\_samples\_split = 15 |
| II | RF | ‘criterion’: ‘gini’, ‘max\_depth’: 8, ‘max\_features’: ‘sqrt’, ‘n\_estimators’: 200 |

Thuật toán phân loại RF dự đoán bằng cách trung bình các dự đoán từ các cây quyết định thực của nó. RF là một phương pháp học máy có giám sát dựa trên học tổ hợp. Nó sử dụng bagging để kết hợp nhiều cây quyết định, qua đó cải thiện độ chính xác của dự đoán. Quá trình đào tạo bagging được thực hiện riêng lẻ cho từng cá nhân. Trong quá trình đào tạo, mỗi cây quyết định được đánh giá bằng cách sử dụng các mẫu dữ liệu khác nhau được tạo ngẫu nhiên bằng cách lấy mẫu có hoàn lại từ tập dữ liệu gốc. Khi xây dựng các cây, cũng có một lựa chọn ngẫu nhiên các đặc trưng. Một phiếu bầu đa số được sử dụng để kết hợp các dự đoán của nhiều cây [1].

Thuật toán Random Forest là một phương pháp học máy được sử dụng để giải quyết các bài toán phân loại và hồi quy. Đây là cách Random Forest hoạt động[2][3], được mô tả thông qua các bước:

1. Tạo tập dữ liệu ngẫu nhiên: Random Forest sử dụng phương pháp lấy mẫu bootstrap để tạo ra nhiều tập con ngẫu nhiên từ tập dữ liệu ban đầu. Mỗi tập con bootstrap thường chỉ bao gồm một phần nhỏ của tập dữ liệu gốc, vì vậy sẽ có các điểm dữ liệu không được bao gồm trong mỗi tập con này. Những điểm dữ liệu bị loại bỏ này tạo thành những mẫu "out-of-bag" (OOB), hữu ích trong việc kiểm tra cách cây quyết định được phát triển từ các mẫu trong tập con.
2. Xây dựng cây quyết định: Tại mỗi nút, một tập con ngẫu nhiên các đặc trưng được chọn. Thuật toán thử nghiệm tất cả các ngưỡng có thể để xác định phân chia tốt nhất. Quá trình này được lặp lại cho đến khi đạt được các nút chỉ chứa các trường hợp hoặc các điều kiện dừng đã được định nghĩa trước đó.
3. Tổng hợp kết quả: Khi tất cả các cây quyết định đã được xây dựng, dự đoán được thực hiện bởi từng cây một. Dự đoán cuối cùng được xác định bằng phương pháp bầu chọn đa số giữa tất cả các cây.

Tóm lại, Random Forest là một phương pháp học máy mạnh mẽ, sử dụng lấy mẫu bootstrap để xây dựng nhiều cây quyết định độc lập. Mỗi cây được huấn luyện trên một tập con dữ liệu ngẫu nhiên và một tập con ngẫu nhiên các đặc trưng, và dự đoán cuối cùng được quyết định bằng bầu chọn đa số từ tất cả các cây trong rừng. Đây là cách Random Forest cải thiện độ chính xác và khả năng tổng quát hóa trên các bài toán phân loại và hồi quy.

**5.5. Naive Bayes Classifier**

**5.5.1. Định lý Bayes**

Hypermeter tuning values for Dataset I (Cleveland) && Dataset II (IEEE)

|  |  |  |
| --- | --- | --- |
| Dataset | Classifier | GridsearchCV Hypermeter Tuning Values |
| I | NB | var\_smoothing: 0.1873817422860384 |
| II | NB | ‘var\_smoothing’: 0.1873817422860384 |

Một trong những thuật toán học máy có giám sát (supervised ML) phổ biến nhất cho các bài toán phân loại đa lớp là thuật toán Naive Bayes (NB). Nhiều vấn đề phân loại có thể được giải quyết bằng thuật toán NB, dựa trên định lý Bayes. Khái niệm cơ bản của NB là ước tính xác suất của từng lớp mà chúng ta muốn dự đoán, dựa trên xác suất của từng đặc trưng có mặt trong dữ liệu. Theo phương trình, các mô hình naive giả định rằng các đặc trưng của một mô hình là độc lập với nhau[1].

Trong đó, P(c/x) đại diện cho xác suất hậu nghiệm (posterior probability), là xác suất của một giả thuyết (hoặc lớp) dựa trên dữ liệu quan sát được. Thuật ngữ p(x/c) biểu thị độ khả dĩ (likelihood), tức là xác suất quan sát dữ liệu dựa trên giả thuyết (hoặc lớp). Xác suất tiên nghiệm của lớp được ký hiệu là p(c), thể hiện xác suất quan sát giả thuyết (hoặc lớp) khi không có dữ liệu. Cuối cùng, xác suất tiên nghiệm của dữ liệu, ký hiệu là p(x)biểu thị xác suất quan sát dữ liệu khi không có giả thuyết (hoặc lớp). Thuật toán Naive Bayes (NB) giả định rằng mỗi đặc trưng trong dữ liệu có một điều kiện độc lập về cách xác suất của một kết quả sẽ xảy ra cho mỗi lớp dữ liệu duy nhất trong tập dữ liệu.

**5.5.2. Thuật toán phân loại Naive Bayes**

Phân loại Naive Bayes là một thuật toán phân loại dựa trên tính toán xác suất áp dụng Định lý Bayes. Thuật toán này thuộc nhóm các thuật toán học có giám sát.

A picture containing text

Description automatically generatedLogo, company name

Description automatically generated with medium confidenceMỗi mẫu dữ liệu được biểu diễn bởi X = (x1, x2, ..., xn) với các thuộc tính A1, A2, ..., An và thuộc lớp C1, C2, ..., Cm. Khi có một mẫu X chưa biết, Phân loại Naive Bayes sẽ xác định rằng X thuộc lớp Ci nếu và chỉ nếu:

Diagram

Description automatically generated Theo Định lý Bayes, với giả định rằng P(X) là hằng số đối với tất cả các lớp, chỉ cần tối đa hóa P(X|Ci) x P(Ci). Nếu không biết P(Ci), ta cần giả định rằng P(C1) = P(C2) = ...

= P(Cm) và tối đa hóa P(X|Ci). Nếu biết P(Ci), ta tối đa hóa P(X|Ci) x P(Ci). Tuy nhiên, việc tính toán P(X|Ci) rất phức tạp. Thay vì đó, ta có thể dụng giả định Naïve, cho rằng có sự độc lập giữa các thuộc tính, từ đó có thể tính toán xấp xỉ P(x1|Ci), ..., P(xn|Ci) từ các mẫu huấn luyện:

**6.5. Huấn luyện mô hình và đánh giá**

**RANDOM FOREST**

**Dataset I:**

**A screenshot of a computer

Description automatically generated**

**Dataset II:**

**A screenshot of a computer

Description automatically generated**

Đối với Dataset I, confusion matrix của mô hình cho thấy nó đã dự đoán chính xác 26 trường hợp dương tính (positive cases) và 26 trường hợp âm tính (negative cases). Tuy nhiên, có 9 dự đoán sai, bao gồm 3 false negatives và 6 false positive. Đối với Dataset II, confusion matrix cho thấy mô hình đã dự đoán chính xác 70 trường hợp dương tính và 94 trường hợp âm tính, nhưng cũng mắc 20 dự đoán sai, bao gồm 12 false negatives và 8 false positives.

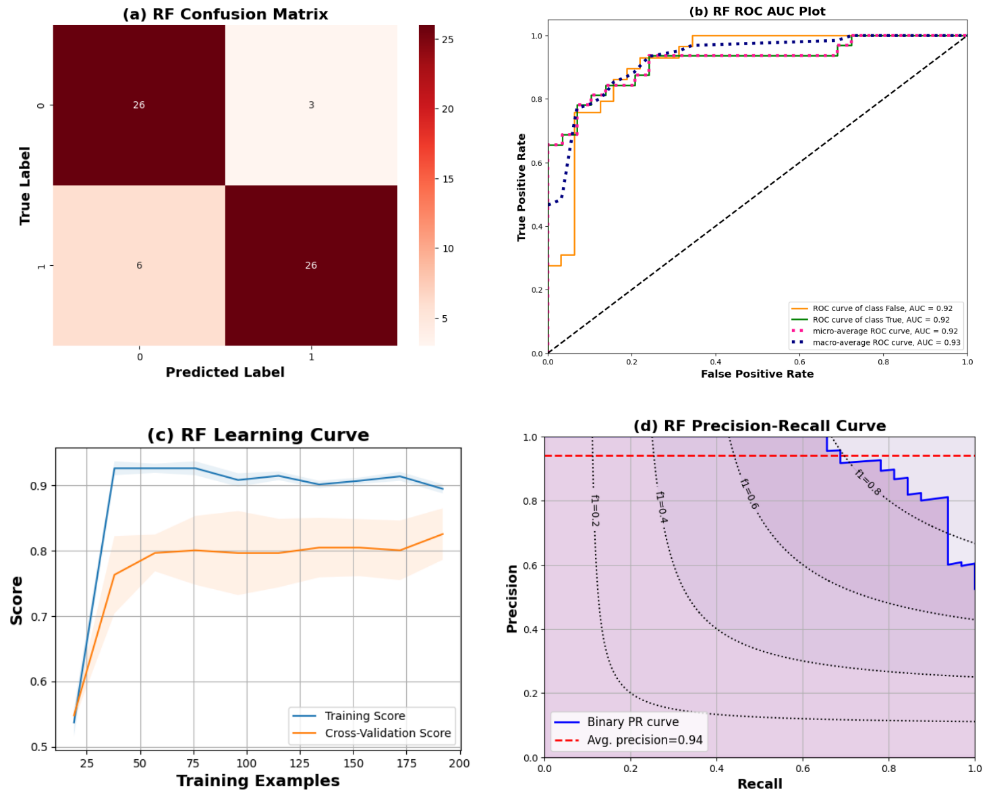
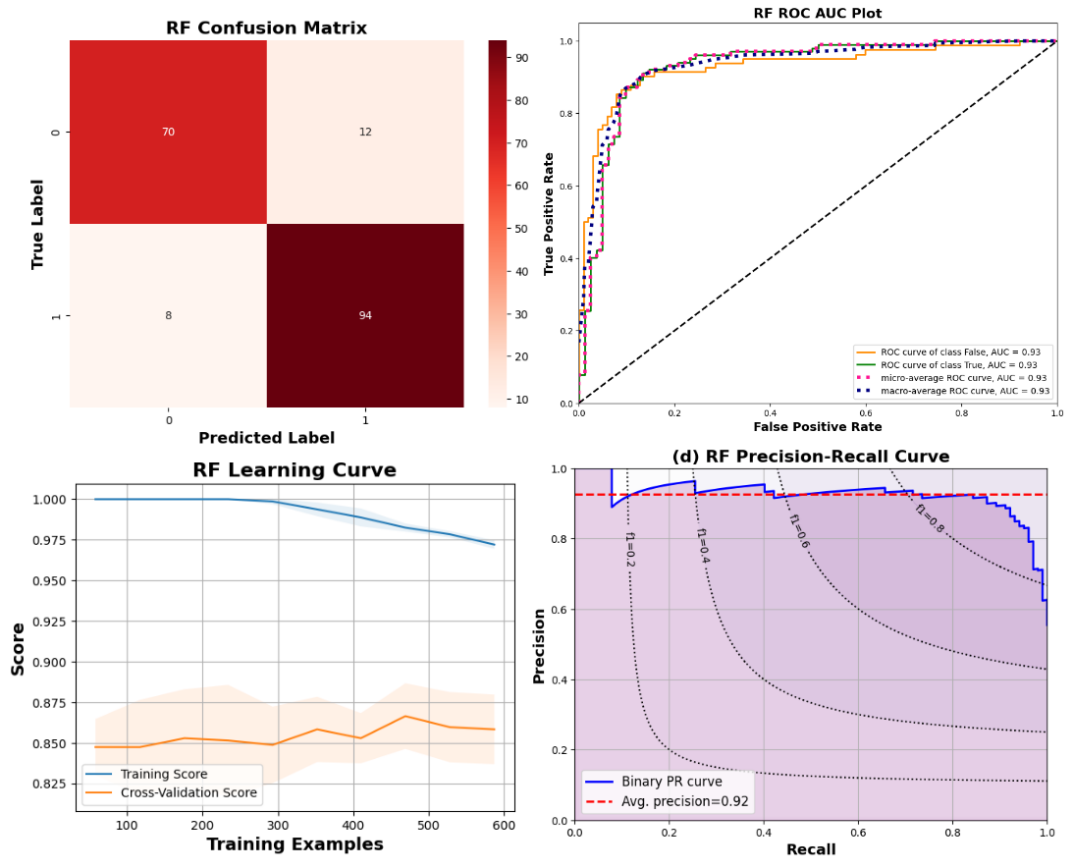
Bảng dưới thể hiện hiệu suất của RF trong việc dự đoán bệnh tim trên hai tập dữ liệu: Dataset I (Cleveland) và Dataset II (IEEE Dataport). Các chỉ số đánh giá mô hình bao gồm precision, recall và F1 score, áp dụng cho cả hai lớp: lớp 0 (không mắc bệnh tim) và lớp 1 (mắc bệnh tim).

Table : Performance measure curve values of RF (Datasets I and II).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model Accuracy | Classes | Dataset I (Cleveland) | | | Dataset II (IEEE Dataport) | | |
| Precision (%) | Recall (%) | F1 Score (%) | Precision (%) | Recall (%) | F1 Score (%) |
|  | 0 | 81 | 90 | 85 | 90 | 85 | 88 |
|  | 1 | 90 | 81 | 85 | 89 | 92 | 90 |
| Accuracy |  |  |  | 85 |  |  | 89 |
| Macro average |  | 85 | 85 | 85 | 89 | 89 | 89 |
| Weighted average |  | 86 | 85 | 85 | 89 | 89 | 89 |

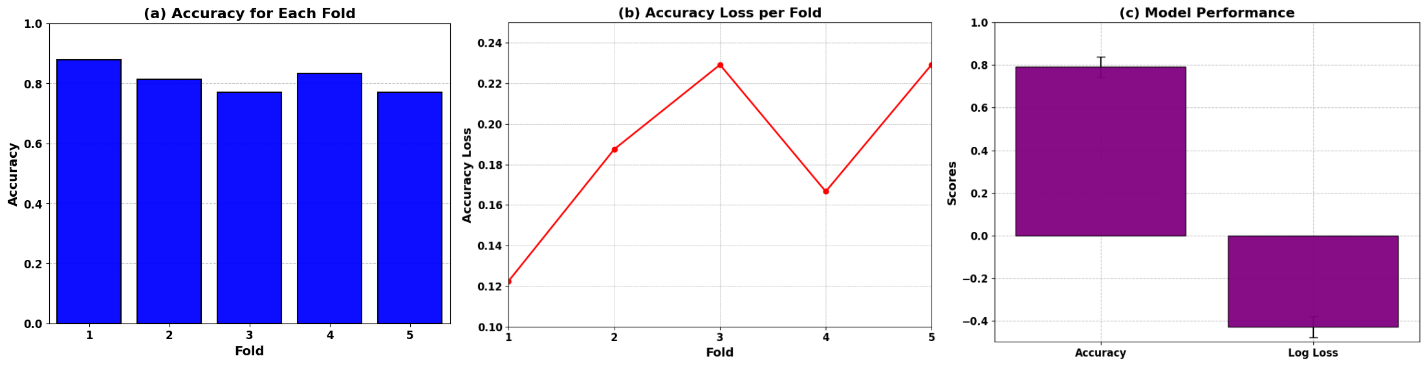
Đối với Dataset I *(Cleveland),* Class 0 có precision là 81%, recall là 90%, F1 score là 85%. Class 1 có precision là 90%, recall là 81%, F1 score là 85%. Độ chính xác tổng thể (accuracy), macro average, và weighted average lần lượt là 85%, 85%, và 85% trên tổng số dataset.

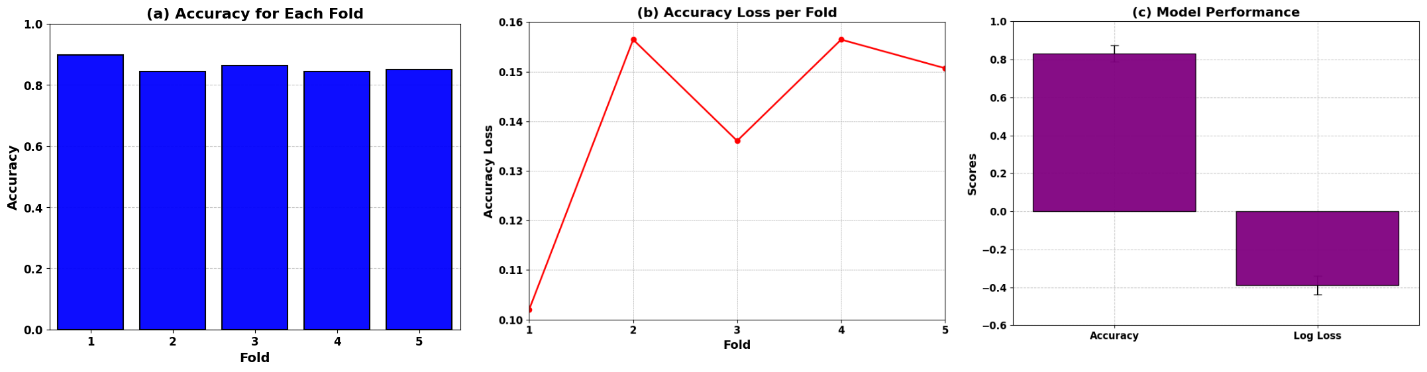
Đối với Dataset II *(IEEE Dataport),* Class 0 có precision là 90%, recall là 85%, F1 score là 88%. Class 1 có precision là 89%, recall là 92%, F1 score là 90%. Độ chính xác tổng thể (accuracy), macro average, và weighted average đều đạt 89% trên toàn bộ dataset.

****

Hình 1 Các đường cong đo lường hiệu suất của RF trên Dataset II.

Hình 2 Các đường cong đo lường hiệu suất của RF trên Dataset I.

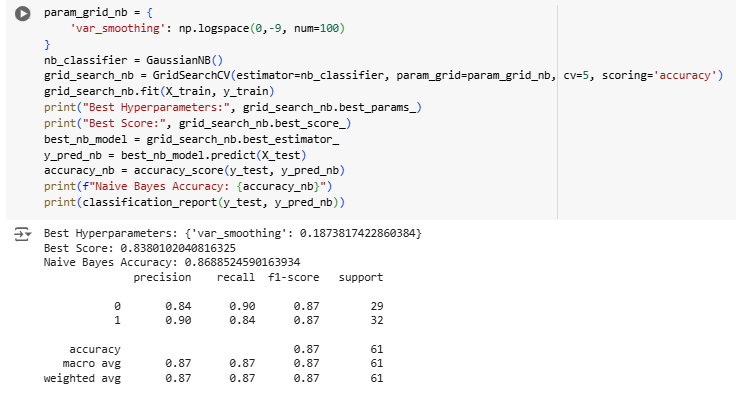
****

****Hình 3 Các biểu đồ độ chính xác 5-fold và giá trị loss của mô hình RF cho Dataset I.

Hình 5 Các biểu đồ độ chính xác 5-fold và giá trị loss của mô hình RF cho Dataset II.

**NAÏVE BAYES**

**Dataset I**

****

**Dataset II**

**A screenshot of a computer program

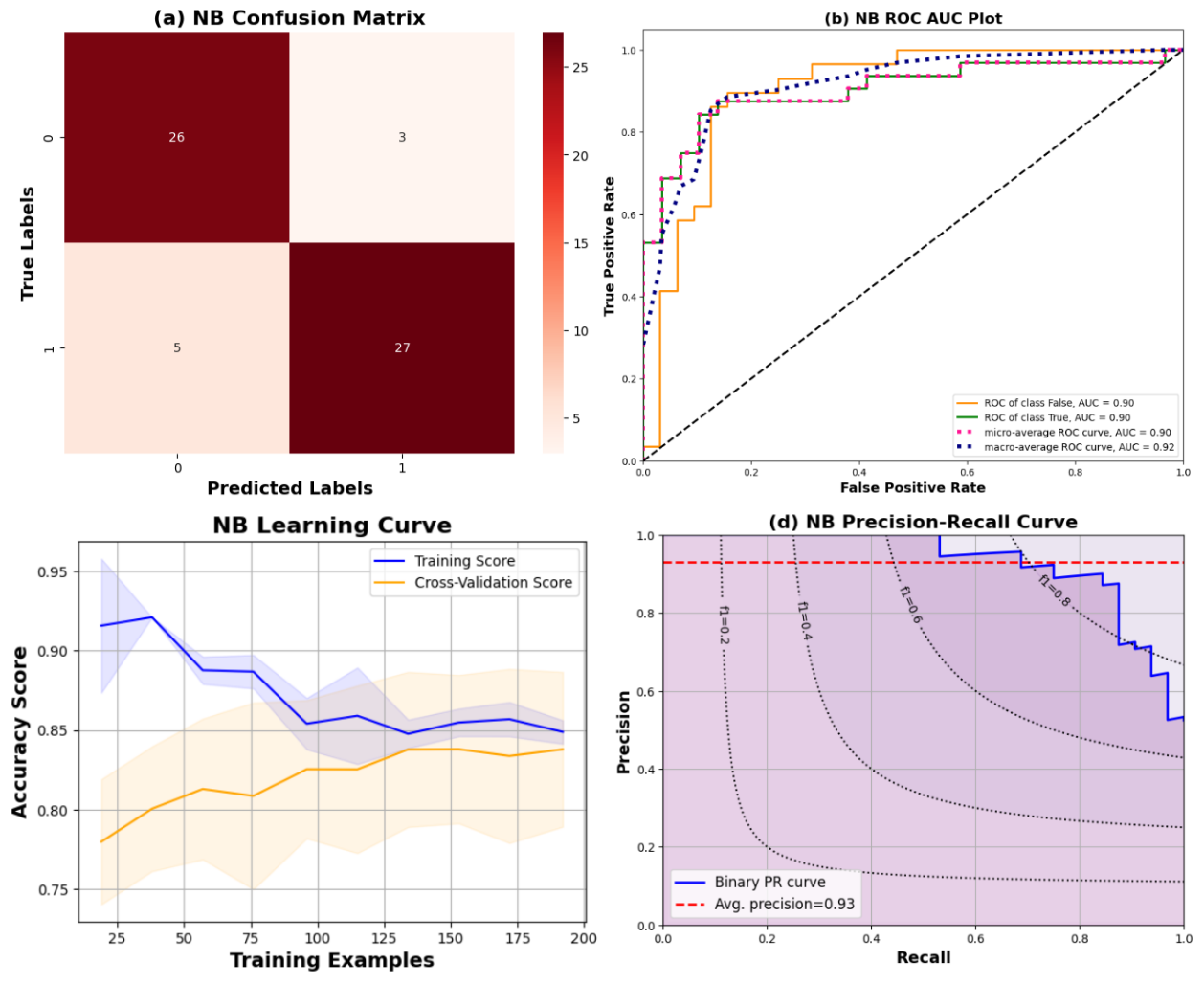
Description automatically generated**

Đối với Dataset I, ma trận nhầm lẫn cho thấy mô hình dự đoán chính xác 26 trường hợp dương tính và 27 trường hợp âm tính, trong khi có 8 dự đoán sai, bao gồm 3 âm tính giả và 5 dương tính giả. Đối với Dataset II, ma trận nhầm lẫn cho thấy mô hình dự đoán thành công 72 trường hợp dương tính và 94 trường hợp âm tính, nhưng cũng có 18 dự đoán sai, gồm 10 âm tính giả và 8 dương tính giả. Bảng trình bày hiệu suất của mô hình NB trong việc dự đoán bệnh tim cho hai tập dữ liệu: Dataset I (Cleveland) và Dataset II (IEEE Dataport). Các chỉ số đánh giá bao gồm độ chính xác (precision), độ nhạy (recall), điểm F1 (F1 score) cho cả hai lớp: 0 (không có bệnh tim) và 1 (có bệnh tim).

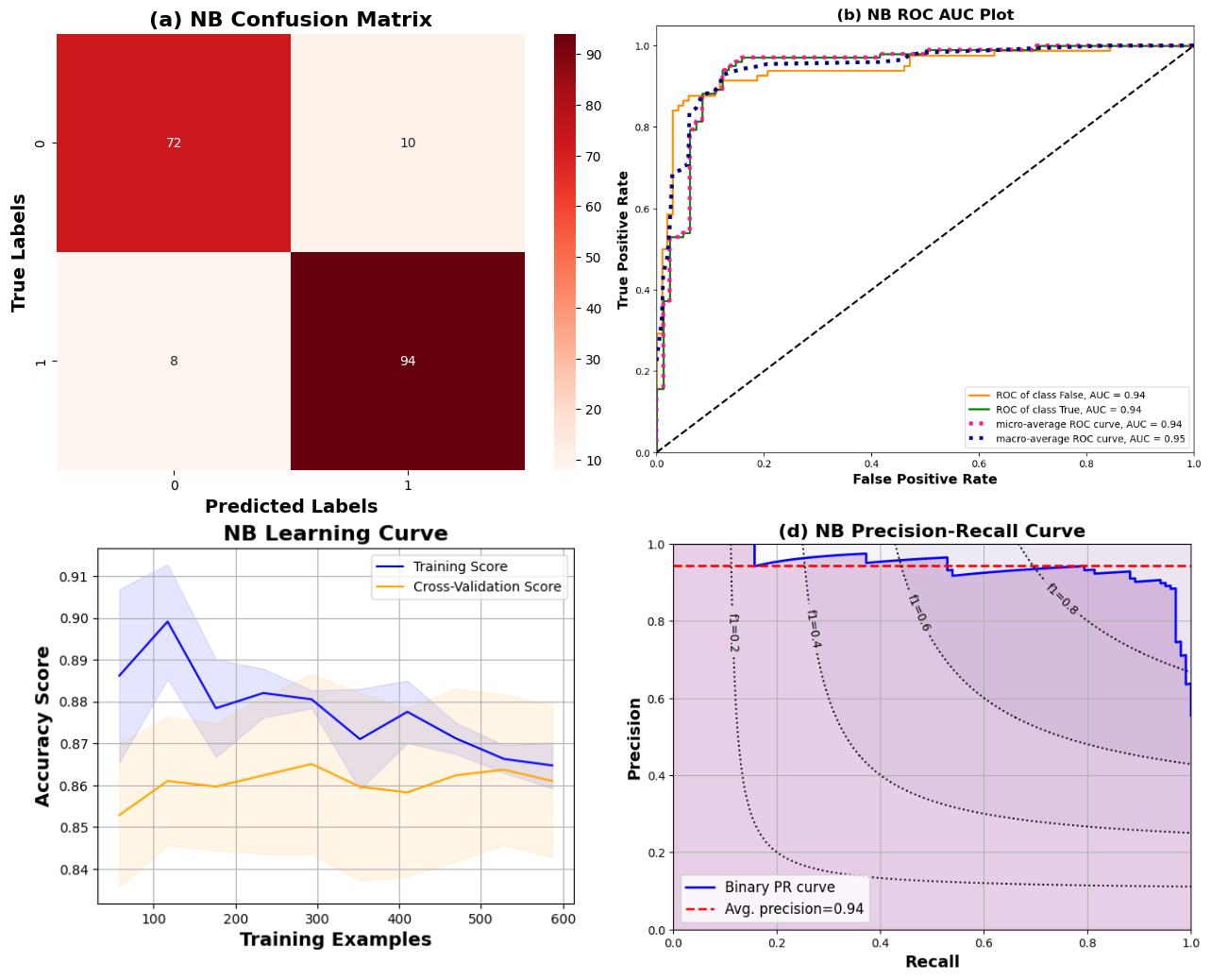
Table : Performance measure curve values of NB (Datasets I and II).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model Accuracy | Classes | Dataset I (Cleveland) | | | Dataset II (IEEE Dataport) | | |
| Precision (%) | Recall (%) | F1 Score (%) | Precision (%) | Recall (%) | F1 Score (%) |
|  | 0 | 84 | 90 | 87 | 90 | 88 | 89 |
|  | 1 | 90 | 84 | 87 | 90 | 92 | 91 |
| Accuracy |  |  |  | 87 |  |  | 90 |
| Macro average |  | 87 | 87 | 87 | 90 | 90 | 90 |
| Weighted average |  | 87 | 87 | 87 | 90 | 90 | 90 |

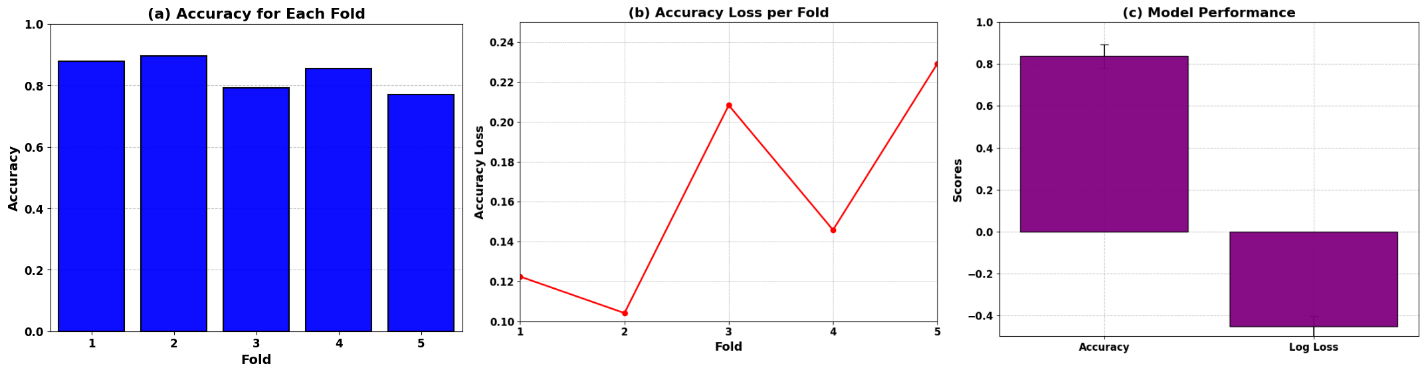
Trong Dataset I, Class 0 có độ chính xác (precision) là 84%, độ nhạy (recall) là 90%, điểm F1 là 87%. Class 1 có độ chính xác (precision) là 90%, độ nhạy (recall) là 84%, điểm F1 là 87%. Tổng số mẫu là 61, độ chính xác tổng thể (overall accuracy), trung bình theo macro (macro average), và trung bình theo trọng số (weighted average) đều là 87%.

Đối với Dataset II, Class 0 có độ chính xác (precision) là 90%, độ nhạy (recall) là 88%, điểm F1 là 89%. Class 1 có độ chính xác (precision) là 90%, độ nhạy (recall) là 92%, điểm F1 là 91%. Với tổng số 184 mẫu, độ chính xác tổng thể (overall accuracy), trung bình theo macro (macro average), và trung bình theo trọng số (weighted average) đều là 90%.

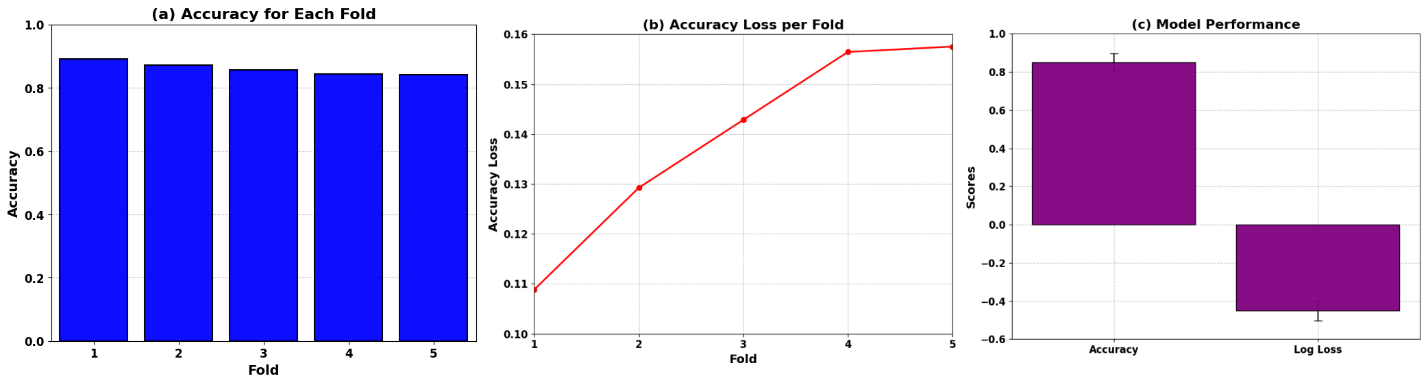
Hình 4 Các đường cong đo lường hiệu suất của NB trên Dataset I.



Hình 5 Các đường cong đo lường hiệu suất của NB trên Dataset II.



Hình 6 Các biểu đồ độ chính xác 5-fold và giá trị loss của mô hình NB cho Dataset I.



Hình 7 Các biểu đồ độ chính xác 5-fold và giá trị loss của mô hình NB cho Dataset II.

[1] [Enhancing Heart Disease Prediction Accuracy through Machine Learning Techniques and Optimization](https://www.mdpi.com/2227-9717/11/4/1210)

[2] [S. J. Rigatti, Random Forest, Journal of Insurance Medicine, 2017, 47(1), 31–39.](https://pubmed.ncbi.nlm.nih.gov/28836909/)

[3] [https://machinelearningcoban.com](https://machinelearningcoban.com/tabml_book/ch_model/random_forest.html)