MAIS 202
Team #3
Anastasiia Nemyrovska, Shirley Ding, Michelle Sayeh, Lucie Shi

MAIS 202 - PROJECT DELIVERABLE 3

1. Final Training Results

a. Change of Dataset

For our machine learning model, we initially selected a dataset with a variety of detailed medical metrics, such as “EKG results” and “Slope of ST”. We realize that many of these features would not be familiar or accessible to a typical user without prior extensive medical testing. Given our goal to create a user-friendly web application that allows anyone to assess their heart disease risk, we decided to change our dataset to one featuring more common health and lifestyle metrics. This makes our application accessible to a wider audience.

Our new dataset Heart Disease Prediction includes factors that users are more likely to know or estimate about themselves, such as blood pressure, cholesterol, heart rate, smoking habits, alcohol intake, exercise hours, family history of heart disease, and diabetes status.

b. Results

The final and preliminary results after the change of dataset compare as follows:

Confusion Matrix

Final: [[113, 9], [18, 104]]

Preliminary: [[28, 3], [3, 26]]

The new model deals with a significantly larger dataset, as evident from the confusion matrix values.

Precision

Final: 0.92

Preliminary: 0.89

The new model shows improved precision, indicating better positive prediction accuracy.

Recall

Final: 0.85

Preliminary: 0.89

The recall is slightly lower in the new model, meaning it is identifying slightly fewer true positives.

Specificity

Final: 0.93

Preliminary: 0.90

The new model has a slightly better specificity, meaning it is better at identifying true negatives.

F1 Score

Final: 0.89

Preliminary: 0.89

The F1 score is roughly equivalent, suggesting the balance between precision and recall remains consistent.

Conclusion

The new model demonstrates a slight improvement in precision and specificity, which is beneficial for reducing false positives and negatives. However, there's a minor drop in recall, which might affect the detection of all true positive cases. With consistent F1 scores, the model shows robustness and balanced performance. Overall, the new model seems to have enhanced precision and specificity while maintaining a balance, even with a larger dataset.

c. Metrics

The logistic regression model demonstrates strong overall performance, as indicated by a confusion matrix with high true positives and true negatives (see Figure 1), resulting in a precision of 0.92, recall of 0.85, and specificity of 0.93. The precision-recall curve (see Figure 2) shows consistently high precision across recall values, with only minor trade-offs at higher recall levels, while the ROC curve (see Figure 3) highlights excellent discriminatory power, maintaining a low false positive rate.

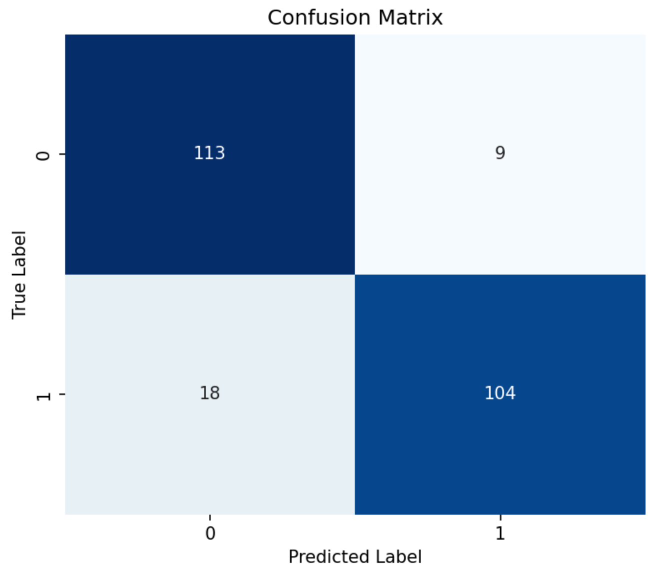

Figure 1. Confusion matrix indicates a strong performance with a high number of correct predictions (both positive and negative). The number of false negatives (18) is slightly higher compared to false positives (9).

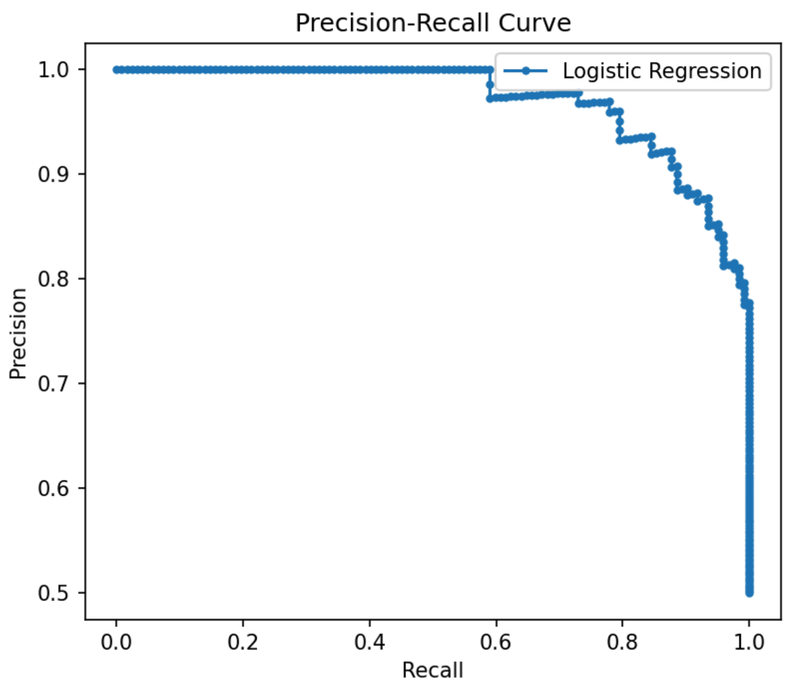

Figure 2. Precision-Recall Curve that remains high across the range of recall values reflects a consistently high balance between precision and recall for the model.

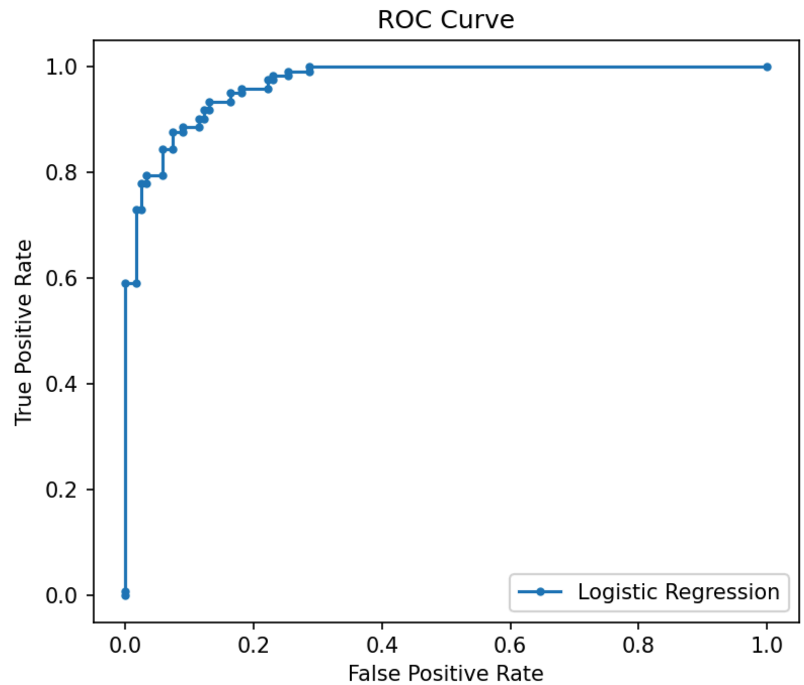

Figure 3. The ROC Curve shows an excellent performance, with the model achieving high true positive rates (TPR) while maintaining low false positive rates (FPR).

2. Final Demonstration Proposal

a. Final product

Our final product is a Heart Health Predictor, a web application designed to assess the likelihood of heart disease based on user-provided inputs. The application includes three key components:

- Landing Page: An introductory page explaining the purpose and utility of the tool.
- Assessment Page: A form that collects user inputs such as age, gender, cholesterol level, blood pressure, heart rate, and lifestyle factors (e.g., smoking, exercise, family history, etc.).
- Results Page: Displays the predicted likelihood of heart disease as a percentage, based on the provided inputs.

The integration approach involves capturing user inputs from the assessment page, feeding them into a machine learning model (built using scikit-learn) hosted on a Flask backend, and presenting the prediction on the results page.

b. Choice of Stacks and Technologies

The following technologies were chosen for their functionality, accessibility, and our familiarity:

- Python: For developing and training the machine learning model using scikit-learn.
- Flask: To build the backend that processes user input, runs the prediction model, and serves results.
- HTML & CSS: To design a user-friendly front end with intuitive forms and visually appealing pages.
- scikit-learn: For training and deploying the logistic regression model used for predictions.

This stack is lightweight, efficient, and well-suited for the scale and scope of this project, allowing seamless integration between the frontend, backend, and machine learning components.

c. Experience with Technologies

We are already familiar with Python, HTML, and CSS, having used them in prior coursework and projects. Although we have limited direct experience with Flask, we are leveraging the workshop recording provided in the course materials, along with online resources and documentation, to learn and implement Flask effectively.