Machine Learning to predict heart attacks
CS-433 Machine Learning - Project 1

Ahmed Abdelmalek
Mohamed Khaled Miraoui
Néstor Lomba Lomba

Abstract—**We address the task of predicting the risk of developing coronary heart disease using Machine Learning. Our dataset is taken from the Behavioral Risk Factor Surveillance System (BRFSS), which contains health-related information about U.S. residents, including lifestyle factors and clinical data. Our approach start by engineering new features that make sense of the values obtained from the survey, followed by correlation purging and data normalization. Then, we train various algorithms, including those discussed in lectures, while using local validation to avoid overfitting. Using our best model, we are able to achieve an accuracy of 0.885 and a f1-score of 0.447 on AICrowd (Submission #275484).**

I. Introduction

This project explores the application of machine learning techniques to predict the likelihood of coronary heart disease. To achieve meaningful results, we understand and clean the dataset, implement suitable classification models, train these models, and thoroughly validate the results.

This report will detail our **Data process** and outline the **Machine learning methods** utilized. We will then present our **Results** and conclude with a comprehensive **Conclusion**.

II. Data Pre-processing

A. Data expansion

We first start by looking at the documentation of the variables provided in the BRFSS Survey webpage. Going through the codebook pdf, we see that each variable is divided in subcategories depending on the values they have. For example, for the variable DROCDY3_ (Drink-occasions-per-day), the subcategories are: 0 for no drinks, 1-899 for drink-occasions-per-day, and 900 for N/A. This variable is not linear, since 900 does not mean more drink occasions than 3, it just means no answer. We split the features into however many subcategories they have, using the value for range subcategories and binary for singular values. This feature engineering makes us go from 321 to 1868 features.

B. Correlation score

After the data expansion, we now have 1868 features instead of 322. The next step is to filter the useless data features, meaning those who have a **weak correlation with the output**. For example, the smoking rate does influence the risk of heart attack, while the birth date doesn't correlate with coronary heart disease risk at all.

To compute the **correlation**, we use the **Pearson corellation score** : its value comes between -1, and 1. 1 indicates the variable X corellates positively with Y, and -1 indicates that the X negatively corellates with Y. We do this for each feature, and remove features with a low $|r_{min}|$, wich results in less noise and less overfitting.

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

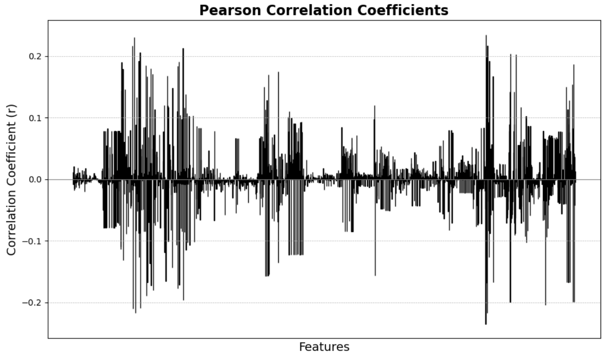

Figure 1. Pearson score for expanded features

C. Sampling and standardizing data

The last step of our data pre-process was to standardize the dataset and creating more balanced training data.

These 2 final steps ensure a better model performance :

- standardizing enables equal contribuition of the features, as well as well as improving computational properties of gradient descent algorythms
- having balanced training data prevents having a model bias toward the majority class, and also ensures better evaluation metrics, so the model performs equally for the 2 classes in the case of a binary classification (or a multimodal as well)

III. Methods

A. Implemented Algorithms

For this project, we implemented several regression techniques to analyze our data. For linear regression methods, we utilized both **gradient descent** and **stochastic gradient descent** algorithms to minimize the mean squared error between the predicted and actual values. We also employed the least squares method using normal equations for an exact solution. To address potential overfitting, we applied ridge regression with an L2 regularization term using normal equations. For classification problems with binary labels, we used logistic regression optimized via gradient descent, and extended it to a regularized version by incorporating an L2 penalty term to further prevent overfitting.

Method	**F1 Score**
Linear regression, gradient descent	0.180
Linear regression, stochastic gradient descent	0.216
Least squares regression, normal equations	0.204
Ridge regression, normal equations	0.206
Logistic regression,gradient descent	0.447
Regularized logistic regression, gradient descent	0.447

Table I
VALIDATION F1-SCORE OF THE IMPLEMENTED ALGORYTHMS

B. Training and Hyperparameter Tuning

Since our dataset for coronary heart disease was unbalanced (most samples did not have coronary heart disease, with significantly fewer positive cases than negative ones, we needed to address this imbalance to improve the model's learning process. To achieve this, we repeated the positive cases in the training data, ensuring that they appeared more frequently and contributed more to the model's training. We controlled this process using a parameter called training positive ratio, which represented the desired ratio of positive cases to the total number of samples in the dataset. We optimized the training positive ratio, testing for different values going from 0.055 to 0.55 and the ratio that gave the best F1 score was 0.22. We can visualise the different f1 scores in the graph below.

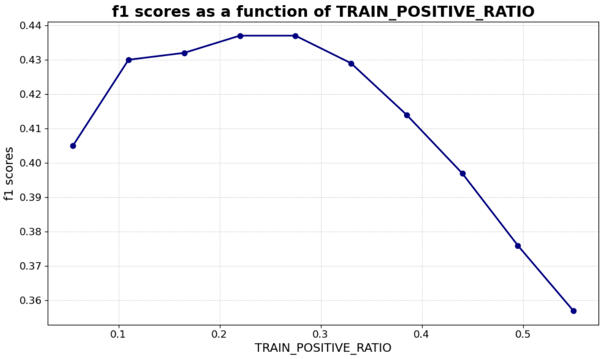

Figure 2. f1 scores for different positive training ratio values

IV. RESULTS

Present key results, including accuracy, F1 scores, and any insights from validation metrics. Use figures as needed.

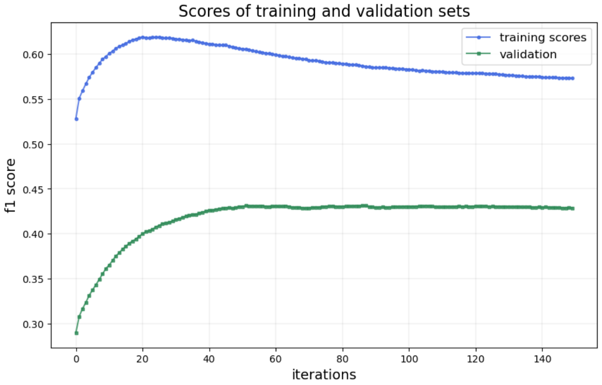

Figure 3. Training and validation f1 scores

The Pearson Corelation score gives **the linear relationship** a variable X and output Y. However, like mentioned in subsection II-A, the data is not encoded into a linear format, thus this score would not be the most appropriate of use. But we can see that for a high value of $|r_{min}|$, we filter more data, wich results in lower f1 scores.

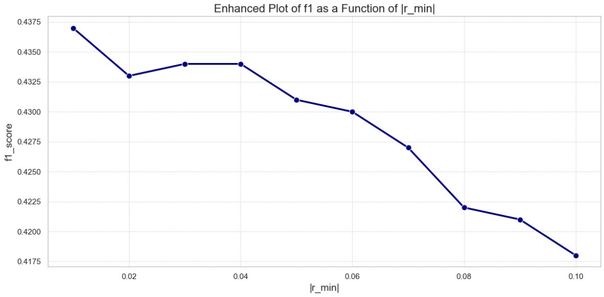

Figure 4. f1 score according to $|r_{min}|$

V. CONCLUSION

Our final model, yielding an F1 score of 0.447 on the AIcrowd leaderboard, demonstrated the significance of balanced data, feature selection, and iterative optimization. This project underscored the impact of methodical data preprocessing and model selection, providing valuable insights into practical machine learning workflows for real-world applications.