Cardiovascular Diseases (CVDs) prediction with Machine Learning

Hadrien Sevel [315940], Pietro Pecchini [377243], Matthijs Scheerder [377763]
CS-433 Machine Learning - Project 1 - EPFL

***Abstract*—Cardiovascular Diseases (CVDs) have emerged as a leading global cause of mortality in developed countries due to the aging of the overall population, increasingly unhealthy diets and sedentary lives and abuse of harmful substances as alcool and nicotine [1]. By analyzing health and behavioral data collected in the USA by means of telephone surveys and by training a machine learning algorithm on these, it is possible to make early estimation of the risk of Myocardial Infarction or Coronary Heart Disease (MICHD) and therefore proceed with individual prevention path. This article intends to show a procedure to handle the data, choose the right mathematical model and hyperparameters in order to maximize the accuracy of the predictions and therefore being able to identify potentially dangerous situations as early as possible.**

I. Introduction

Heart diseases are a major reason for death in developed countries. This is due to older populations, unhealthy food choices, less active lifestyles, and harmful habits like drinking and smoking. Our goal is to detect these diseases early using data from phone surveys in the USA and machine learning. By doing this, we can help people take steps to prevent these diseases before they become serious.

We have data with 321 different types of information for 328,135 people. Out of them, about 8.8% have had heart diseases. We tested our machine learning model on a part of this data, which has information for 109,379 people. Section II outlines our approach for feature selection, proper training of the logistic regression algorithm with parameter tuning for optimal performance, and improved accuracy using different estimators. The accuracy estimations obtained during the training process are presented and examined in section III, while section IV is dedicated to a comprehensive discussion on the results.

II. Models and Methods

We explain in this section our process to wrangle the data and tune the hyperparameters of our logistic regression model.

A. Data wrangling

Initial exploratory data analysis was conducted to understand the significance and relevance of each feature against the survey used to collect the data [2]. We removed 248 features that were deemed irrelevant (e.g., "Do you live in college housing?") or had more than 50% missing values (e.g., "When was the last time you had your eyes examined by any doctor or eye care provider?" had a response rate of only 0.5%).

For the categorical features, a one-hot encoding approach was employed. By consulting the codebook for each categorical feature, their encodings were determined. For instance, for the feature GENHLTH, "Would you say that in general your health is:", five columns were created corresponding to values 1-5, with binary encodings of 0 and 1. Values labeled as 7, 9, and BLANK (from the codebook) were treated as missing and thus an additional "missing" column was created so that our model can also learn from missing values (since we can not remove them from the test set). An example rule applied in our code for this transformation was:

```
'GENHLTH': {'valid_range': (1, 5),
'invalid_values': {7, 9, np.nan}}
```

As for the numerical features, they were standardized. Taking the feature PHYSHLTH as an example, which asks, "Now thinking about your physical health, which includes physical illness and injury, for how many days during the past 30 days was your physical health not good?": a value of 88 (indicating none) was replaced by 0, and values 77, 99, and BLANK (indicating uncertain or missing values) were replaced by the mean of the feature. Following these transformations, both the training and test set were standardized using the mean and standard deviation of the training set. The rule applied in our code for such transformations was:

```
'PHYSHLTH': {'zero_values': {88},
'invalid_values': {77, 99, np.nan}}
```

Custom rules were thus crafted for each categorical and numerical feature after a thorough study of the codebook to ensure consistent and accurate data processing. Furthermore, individuals with more than 20% of invalid or missing values were removed from the training set.

B. Model training and hyperparameter tuning

Model performance evaluation utilized a 4-fold cross-validation approach with regularized logistic regression (L2 regularization). Various combinations of hyperparameters, namely λ (with 4 different values, 10^{-6}, 10^{-5}, 10^{-4} and 10^{-3}) and γ (with 30 different values, linearly distributed from 0.1 to 3.5), representing the weight of the regularization parameter and the step size for each iteration respectively, were tested. The plots provided showcase the RMSE and F1 scores for different combinations of λ and γ. By comparing these metrics for various hyperparameter combinations, the optimal set of hyperparameters for the model was determined.

III. Results

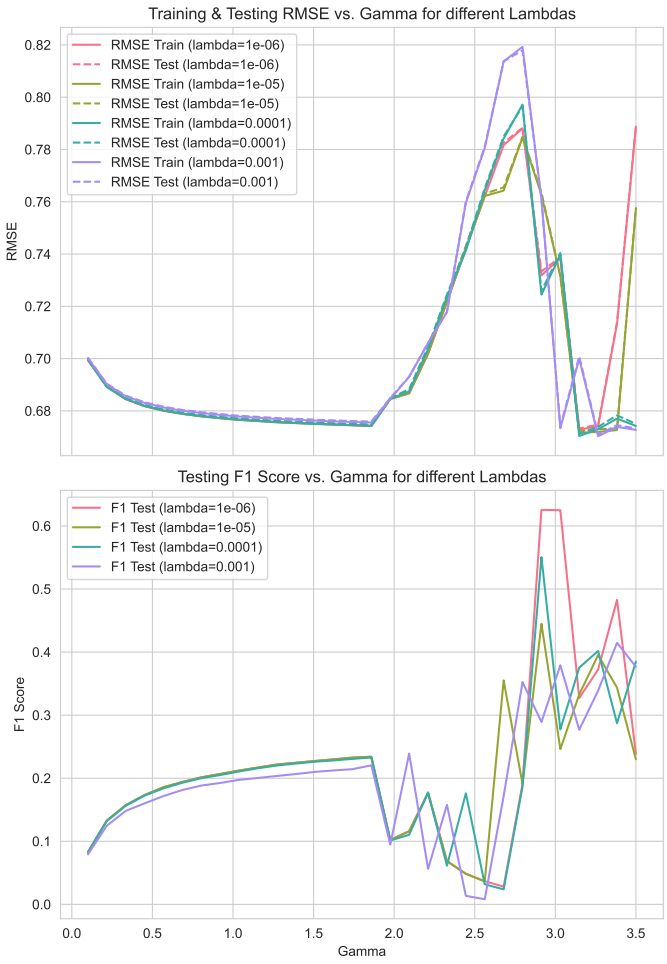

Fig. 1. Training and testing RMSE (top plot) and F1 score (bottom plot) for different (λ, γ) combinations.

Figure 1 shows how RMSE (Root Mean Square Error) for the training and testing sets and F1 score change with different gamma values for various lambda settings. As gamma goes up, both training and testing errors in the RMSE graph tend to go down. But we also need to look at the F1 score, which gauges the equilibrium between precision and recall. Indeed, we want our model to have as few false negatives as possible. In the F1 score graph, there's a clear high point around a gamma of 2.9 for some lambda values. Picking a gamma less than this seemed to make our model fit too closely to our training data. This overfitting is hinted at by having a low error but not a great F1 score, meaning the model was too sure of itself, even when it was wrong. It's important to note that our data is imbalanced – 92% of the people don't have Cardiovascular Diseases (CVDs). So, a model that mostly says "no CVD" can still seem very accurate, but with many false negatives which is not our goal. To better spot actual CVD cases, we adjusted our model: now, scores above 0.40 are seen as having CVD, not just scores above 0.5. By looking at both RMSE and F1 score together, we found that a gamma of 2.9 was the best choice, helping us get a good balance between being right and not missing out on important cases. By then trying our model on the test set, we found that we achieved the best F1 score with lambda equal to 10^{-5}. Additionally, the RMSE blows up for gammas above 3.5, further reinforcing our decision to choose 2.9 as the optimal value.

With these hyperparameters, we obtain a F1 score of 0.752 for the train set and 0.433 for the test set with an accuracy of 0.860.

IV. Discussion

There's a noticeable disparity between the F1 score derived from the training set, which was estimated through cross-validation, and the score achieved on the test set, which was notably lower at 0.433. This difference can be traced back to the discrepancies in how the data was processed for each set. As we looked into in section II-A, certain individuals were excluded from the training set due to a high number of missing responses. Such data cleaning wasn't applied to the test set since we needed to provide predictions for all individuals. This inconsistency in data handling might have introduced biases, negatively affecting the F1 score on the test set. Another cause could be that, despite our efforts to avoid this situation, our model might still be overfitting a bit.

However, it's worth noting that achieving a F1 score of 0.433 alongside an accuracy of 0.860 is a good result, especially given the simplicity of the logistic regression model. These numbers indicate that our model is not only accurate but also succeeds in learning the relationships within the data, producing meaningful results.

Additionally, existing studies on this dataset, as highlighted in [3], indicate that other machine learning approaches, like neural networks, might be more effective for this particular challenge. The current results, although insightful, suggest there's room for exploration with other methodologies to enhance predictive accuracy.

V. Summary

This paper offers readers a detailed look at the research process implemented in order to derive useful information from a dataset and how to use the selected features in order to train a machine learning algorithm and predict outputs. Useful feedback models and methods are largely used, such as cross-validation, and accuracy estimators such as F1 score and RMSE. Particular importance is given to the choice of the optimal parameters: 120 combinations of hyperparameters representing step-size and regulation weights are tested and the results are discussed. The results are relatively satisfying, but in order to obtain greatly better results models based on neural network or other advanced algorithm are needed.

REFERENCES

[1] T. B. Walden R, "Cardiovascular disease," *Herbal Medicine: Biomolecular and Clinical Aspects*, 2011.

[2] "Behavioral risk factor surveillance system," https://www.cdc.gov/brfss/annual_data/annual_2015.html.

[3] A. Teboul, "Building Predictive Models for Heart Disease using the 2015 Behavioral Risk Factor Surveillance… — alexteboul17," https://medium.com/@alexteboul17/building-predictive-models-for-heart-disease-using-the-2015-behavioral-risk-factor-surveillance-b786368021ab, [Accessed 27-10-2023].