Machine Learning - Project 1

Irene Vardabasso - Niccolò Venturini Degli Esposti - Mathilde Chaffard

***Abstract*—This report presents an analyse of a machine learning project aimed at predicting heart attacks. During the project, machine learning methods seen in class were implemented to build a model able to estimate the likelihood of developing Major Ischemic Cardiovascular Heart Disease (MICHD).**

I. Introduction

In this project, our objective was to assess an individual's susceptibility to Cardiovascular Diseases (CVD) by analyzing various lifestyle-related factors. We utilized a feature-rich dataset comprising personal health information and employed binary classification techniques and machine learning algorithms to forecast the likelihood of an individual developing a MICHD. The input data is represented as a feature vector, encapsulating an individual's health-related attributes. Our model's output corresponds to one of two classes: -1 (indicating low risk) or 1 (indicating high risk), facilitating effective risk assessment and early intervention in CVD cases.

II. Models and Methods

A. Data Preprocessing

In the initial phase, we conducted a comprehensive data assessment, recognising that the characteristics of the dataset have a significant impact on the performance and results of predictive models. Therefore, to ensure the accuracy and reliability of the data used to train and test machine learning models, data understanding and cleansing are essential steps. Among our observations, we identified two basic feature types: continuous and categorical. We also recognised that raw data, and consequently derived data, can be contaminated by noise, contain missing values and include features that are inconsistent with the algorithm's objectives. In response to these findings, we carried out a number of data preparation procedures. To deal with missing data, we replaced null values in continuous features with the mean of non-null values for the same feature, while null data in categorical features were replaced with the most common category. We then removed irrelevant columns with high correlation and those with zero standard deviation. Furthermore, given the different scales of different features, we applied data normalisation to ensure equitable feature scaling, thereby reducing the risk of any single feature exerting undue influence. Our experiments also included feature expansion, including the generation of polynomial features of varying degrees, while considering the inclusion of a constant feature. Eventually, we identified a notable class imbalance within the 'y_test' variable, particularly between values of '-1' and '1'. To mitigate the risk of overfitting concerning '-1' values, we introduced an oversampling function that replicates rows, in both 'x_test' and 'y_test', corresponding to '1' values in 'y_test' [1].

B. Model Selection

To find the optimal parameter w^*, we used the logistic regression with the gradient descent. Logistic regression is used for binary classification tasks. It aims to predict whether an observation belongs to class 0 or class 1. To be able to do that, at the very beginning the -1 values are turned into 0 value. In order to find a fix class we used a threshold of 0.5. Then, the classes are mapped back from $\{0, 1\}$ to $\{-1, 1\}$ to comply with the submission format. Furthermore, we decided to add another parameter to our model : l_b, the *loss bias*. The output data contains very few labels 1 (minority class), therefore we re-weight the loss of the label 1 making it bigger. The *loss bias* gives then more importance to minimize the loss of the labels 1. This parameter is commonly used to treat unbalanced dataset as it can influence the model prediction.

C. Parameter Optimization

Optimal values of parameters need to be found to balance the bias and variance in the model. Hyperparameter optimization aims to find hyperparameters that optimize the model performance such as achieving high accuracy and high F1 score. The parameters of our model are the degree (of the polynomials features expansion), γ (the learning rate of the gradient descent) and l_b (the *loss bias*). The algorithmic adjustment of the gradient descent is shown in (1) :

$$w^{t+1} = w^t - \gamma \cdot grad \cdot l_b \tag{1}$$

with $grad$, the gradient of the loss function, defined by :

$$grad = \nabla L(w^t) = \frac{1}{N} \sum_{n=1}^{N} x_n(\sigma(x_n^T w) - y_n) \tag{2}$$

To determine the optimal parameters, we used a k-fold cross-validation approach with k set to 5. Within each of the k iterations, the model is trained on the training set and then evaluated on the test set. The performance evaluation uses the F1-score metric on the test set. These k individual performance scores are then averaged, resulting in a single performance estimate for the model, which in turn guides the determination of the optimal parameter.

D. Performance measurement

To compute the performance of each model, we used the F1-score. This metric is usually used for evaluating performance of classification models and when datasets are imbalanced. F1-score combines two metrics : precision and recall. Precision is the number of true positive predictions divided by the total number of positive predictions (true positives and false positives). And recall is the number of true positive predictions

	Accuracy	F1-score	γ	$degree$	l_b	λ
Experiment 1	0.766	0.360	0.5	2	0.01	0
Experiment 2	0.816	0.381	0.9	2	0.01	0
Experiment 3	0.836	0.422	0.8	2	0.01	0
Experiment 4	0.836	0.419	0.8	2	0.01	0.001
Baseline 0	0.912	0	N/A	N/A	N/A	N/A

TABLE I

TABLE REPORTING ACCURACY AND F1-SCORE CALCULATED BY AICROWD AND THE HYPERPARAMETERS USED FOR THE 6 DIFFERENT EXPERIMENTS, AND THE BASELINE PREDICTION.

divided by the total number of actual positive cases (true positives and false negatives). F1-score is a suitable metric when false negatives and false positives can have significantly consequences. F1-score is defined as the harmonic mean of precision and recall, i.e F1-score is two times the precision multiply by the recall divided by the sum of the precision and the recall. F1-score range from 0 to 1 where a high F1-score indicates a good balance between precision and recall. And a low F1-score indicates a low precision or low recall.

III. RESULTS

Experiment 1 : Logistic regression with 0.75 rate of oversampling. In this experiment, using 91 features, we ran the logistic regression. The rate of oversampling the labels 1 is 0.75, so that the 1s are 0.75% of the total values of -1s. This experiment has a accuracy of 0.766 and a F1-score of 0.360.

Experiment 2 : Logistic regression with 0.6 rate of oversampling. In this experiment, using 91 features, the rate of oversampling the labels 1 is 0.6 (instead of 0.75). This experiment obtained a accuracy of 0.816 and a F1-score of 0.381.

Experiment 3 : Logistic regression with all the columns (all raw data). In this experiment, we ran the logistic regression, starting with the data preprocessing and using for that all the columns of the dataset, i.e all the features. A cross-validation has been done for this experiment. This experiment has a accuracy of 0.836 and a F1-score of 0.422

Experiment 4 : Regularized logistic regression with Lasso. In this experiment we added another hyperparameter λ to our model. The regularization term $\lambda \parallel w \parallel_1$ is added to the calculation of the gradient. We ran the regularized logistic regression, starting with all the columns of the dataset. A cross-validation over the regularization parameter λ has been done. This experiment has a accuracy of 0.836 and a F1-score of 0.419.

IV. DISCUSSION

In our preliminary analysis, we carefully examined the dataset description to inform our feature selection process. We found that the final 91 features, derived from the previous set of 230 features, were of significant value to our model. Given the reasonable number of features, we decided to implement logistic regression as our modelling approach. These selected features were still subjected to the necessary data preprocessing steps. In particular, we undertook a series of experiments to determine whether manual feature selection would provide better predictive results, driven by the supposition that the inclusion of all features could potentially affect the model's ability to learn. Our experiments included various parameter modification, with a particular focus on oversampling rates. Despite our efforts to optimize these parameters, we observed limited improvement. We then revisited the idea of using all 321 features, which required an extensive data preprocessing pipeline, as explained previously. To identify the optimal parameters for this reconfigured model, we performed cross-validation, recognising that the best parameter values may differ from those obtained with the previous model. After running this reconfigured approach, we observed a remarkable increase in the F1-score. This improvement was primarily attributed to the model's intrinsic ability to identify the most pertinent features for the final prediction, outperforming the manual selection process. To deal with high dimensionality and irregular value patterns, we applied Lasso regularisation to our logistic regression model. This technique was intended to counteract overfitting and deal with numerous potentially irrelevant features. However, the results suggest that the lower F1-score achieved with this model configuration may be due to the prior feature selection process, which penalises the action of the Lasso to further refine the feature selection. As a consequence, the model architecture was adapted to logistic regression. The effectiveness of oversampling the '1' values was evaluated empirically through a series of trial and error experiments. Our analysis showed that the optimal oversampling rate for the '1' values was determined to be 0.6 ca, as 58% of the '-1' values. Furthermore, when we cross-validated model with 0.6 rate of oversampling, the loss bias obtained was different from zero. This suggests that our oversampling strategy had not fully addressed the challenges posed by the highly unbalanced dataset. In order to further improve the predictive performance, it is worth considering examining the threshold applied to the sigmoid function. In addition, we suggested implementing a cross-validation procedure to determine the optimal oversampling rate as this could prove in refining our predictive capabilities, even without the use of the loss bias parameter.

V. SUMMARY

To conclude, by learning about our dataset and by testing different kind of hyperparameters and regression algorithms, we improved our machine learning model. We finally chose to use the logistic regression as model and obtained an Accuracy of 0.836 and a F1-score of 0.422.

References

[1] Mohammed, Roweida & Rawashdeh, Jumanah & Abdullah, Malak. (2020). Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results. 243-248. 10.1109/ICICS49469.2020.239556.