Project 1 CS-433 Machine Learning

Adrien Feillard, Lou Fourneaux, Aurélien Laissy
EPFL

***Abstract*—This project's primary objective is to apply basic machine learning concepts to the Behavioral Risk Factor Surveillance System (BRFSS) data set. It involves exploratory data analysis, feature processing, and model analysis to gain insights into the health-related behaviors and conditions of U.S. residents. BRFSS data is a valuable resource for public health research and policy development.**

I. Introduction

The aim of this project is to determine the risk of a person in developing Cardiovascular Diseases (CVD) based on features of their personal lifestyle factors. We will use this data to build a model able to predict as accurately as possible whether given a certain clinical and lifestyle situation a person will develop a coronary heart disease (MICHD) or not. Taking into account that the data set is very imbalanced, our results will be evaluated using f1 score[1].

II. Models and methods

Before advancing to the application of conventional machine learning techniques, a comprehensive data exploration was conducted. This phase facilitated the identification of pertinent features and their inherent characteristics, along with the requisite pre-processing steps for appropriate treatment. Following this initial step, we embarked on evaluating a variety of machine learning models covered in our course, delving into the reasons behind their over/under performance. In the culmination of our project, we implemented cross-validation as a means to gauge our model's effectiveness and its capacity for generalization to new data.

A. Exploratory data analysis

The data set is composed of a training set of 328135 samples (survey respondent) and 320 feature columns (answers to questions). The test set was composed of 109379 samples and 320 features. Each of the samples were assigned the value -1 and 1 for whether they have a MICHD or not. We treated the data taking into account several of their characteristics:

- *Data set balance:* First, it's important to note that there's a significant class imbalance, with only 8.83% of unhealthy persons (represented by 1) in the data.
- *Continuous/categorical features:* some features are categorical (68,125%) and some are continuous (31,175%).They could be processed differently.
- *Missing values:* Some columns have a majority of Nan values, 146 have more than 50% of Nan, and 115 have more than 80% of Nan.
- *Correlated columns:*By visualizing the histogram of the features we noticed that some have similar distributions. In addition, many columns are highly correlated, 70 have more than 50% correlation, and 57 have more than 80% correlation.
- *Exploration of Irrelevant Features:* During our data exploration, we've noticed some features that appear unrelated to our research objectives, particularly in the context of MICHD. This raises the question of their relevance.

B. Feature processing

Pre-processing is necessary in order to get coherent results.

1) *Removing irrelevant features:* We used a qualitative approach for feature selection and removed 65 columns, which constituted about 20.31% of the dataset, due to their poor relevance. For example, one question was about the number of cellphone in a household.
2) *Dealing with unnecessary data columns:* We evaluate the proportion of Nan values within each feature in the training data set. If this proportion exceeds our 80% threshold, we remove the entire feature from both the training and test datasets, as we consider the data within that feature to be too unreliable due to potential noise.
3) *Processing Missing Data in Retained Columns:* The remaining columns were still composed for some of them by NaN values which we decided to replace using the column's median value. This will insure the maintenance of the statistical distribution of the data and avoiding introducing bias into your data set. The median was chosen because of its robustness to outliers.
4) *Dealing with outliers:* We employed the IQR method to eliminate outliers by substituting extreme values with the median. This approach safeguards against the undue influence of extreme values on the final results.
5) *Data standardization:* We make all continuous data scale-invariant by applying the z-score normalization method, ensuring that no single feature dominates the learning process. This step is crucial for optimization in the later stages of our analysis.
6) *Dealing with ill-condition:* We calculated the Pearson correlation coefficients between pairs of columns. Columns with absolute correlation coefficients greater than our threshold of 80% are identified as highly correlated and subsequently removed. This approach helps mitigate multicollinearity issues.
7) *One Hot Encoding:* We initially addressed the presence of categorical data by applying One Hot Encoding.

[1] neptune.ai

However, during the data cleaning process, we consistently encountered a situation where the resulting matrix became singular, meaning that it lacked full rank and couldn't be effectively used for our analysis.

C. Implementations

We implemented the six machine learning methods seen in class. For each of these methods, we implemented a cross-validation and a best parameter selection function in order to tune the hyperparameters and therefore, the weights for each implementation.

- The cross validation prevents over fitting, reduces bias[2] and is necessary for the tuning. The optimal parameters will then, be used to determine the best prediction possible.
- When dealing with the tuning function, we updated the regressions so the gradient descent stops if the loss increases by more than 1 %. This way, the algorithm stops in case of divergence. It also stops the gradient descent if it manages to find a convergence under a threshold. It allows faster computation of the algorithm and avoids non meaningful losses.

Eventually, ridge regression was chosen because of its higher f1-score.

We faced a problem with the least squares tuning process, which resulted in a singular matrix during cross-validation. Despite our efforts to address this by removing low-variance columns and adjusting correlation thresholds, the issue persisted. Thus, we proceeded with least squares without feature expansion.

In the tuning phase, we started from a range of values to optimize: a k-fold of 5, maximum iterations of 1000, lambdas from 1e-4 to 1e-2 and gammas from 0.01 to 0.1. Each of the parameters were optimized with an initial weight initialized using a random matrix and computed without polynomial feature expansion due to memory loss during computation.

Models	γ	λ
Gradient descent	0.015	-
Stochastic gradient descent	0.005	-
Ridge regression	-	0.01
Logistic regression	0.015	-
Regularized logistic regression	0.015	0.0001

TABLE I: Optimal parameters for each regression

D. Classifier

In our binary classification task, where we distinguish between class 1 (unhealthy) and class -1 (healthy), we employ a classifier with a sigmoid function. To enhance the accuracy of positive predictions, the decision threshold was set to 0.55. As opposed to a threshold set to 0.5, 0.55 makes our model more conservative in labeling examples as positive, reducing the number of positive but increasing their precision.

III. Results

We preferred using the F1 score over accuracy for model evaluation because it effectively balances precision and recall. This provides a more robust metric for assessing model performance, particularly on imbalanced datasets. The maximum F1 score we obtained is 37,5% with the ridge regression. The optimal parameters are found by minimizing the loss and not maximizing the f1 score. It explains why some optimal parameters don't yield the optimal F1 score (see fig. 1 where λ= 0.015 is the optimal parameter but λ=0.005 gives the best f1 score).

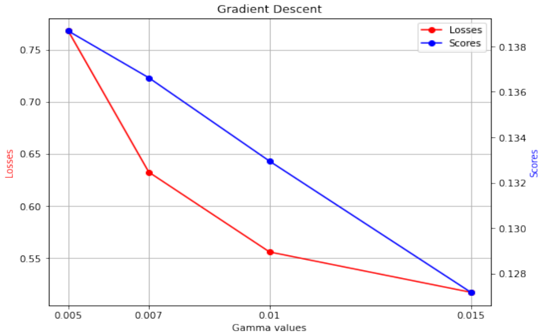

Fig. 1: Gradient descent's loss and scores depending on gamma without polynomial feature expansion

All the f1 scores were calculated with the obtained optimal parameters. We are not over fitting as our model performs similarly for f1 on training (37,5%) and test data (39,1%) on AI crowd.

Models	train F1 score
Gradient descent	0.119
Stochastic gradient descent	0.143
Ridge regression (λ=0.001)	0.375
Logistic regression	0.116
Regularized logistic regression	0.116
Least squares	0.370

TABLE II: f1 score for each regression

IV. Discussion

In our project, data pre-processing emerged as a critical factor influencing our results. Initially, challenges in handling categorical and continuous values differently led to inconsistencies. Moreover, our data cleaning inadvertently retained co-linear columns, although we tried with a wide range of correlation threshold, affecting the importance of key features. A weighted loss function could be implemented to address this. Despite some issues with the least squares method, our overall implementation and parameter tuning functions met our expectations.

V. Summary

In this project, we implemented a Machine Learning pipeline to predict whether some US resident will have MICHD or not. We found that training a ridge regression model with optimized hyperparameters yields the best F1 score 39,1(%) on AI crowd.

[2] CS-433 lecture 4