Detecting Cardiovascular Diseases - An Analysis Over Machine Learning Models

Arina de J. A. M. Sanches, Benedikt W. Josef Reppin, Fernanda Bezerra Nascimento, Vitoria Verçosa de Oliveira

***Abstract*—The main focus of this paper is to present and analyze a procedure to train machine learning models focusing on classifying the presence of cardiovascular disease in people. The dataset used, available on Kaggle, presents a group of different people with different kinds of risk factors, which are the major causes of these diseases. It is used 6 traditional machine learning models and they were compared based on the classification of this problem, using the risk factors each patient presents. To best analyze each model capacity were used varied metrics. By the end, the obtained results are examined showing that the Decision Tree is the best performing model for this task.**

***Resumo*—O foco principal deste artigo é apresentar e analisar um procedimento para treinar modelos de aprendizado de máquina com foco na classificação da presença de doenças cardiovasculares em pessoas. O conjunto de dados usado, disponível no Kaggle, apresenta um grupo de pessoas diferentes com diferentes tipos de fatores de risco, que são as principais causas dessas doenças. São utilizados 6 modelos tradicionais de aprendizado de máquina e eles foram comparados com base na classificação desse problema, utilizando os fatores de risco que cada paciente apresenta. Para melhor analisar a capacidade de cada modelo, foram utilizadas métricas variadas. Ao final, os resultados obtidos são examinados, mostrando que a Árvore de Decisão é o modelo com melhor desempenho para esta tarefa.**

Cardiovascular – Models – Classification

I. Introduction

Cardiovascular diseases are a group of diseases that affect the heart and blood vessels [8]. They are of great concern due to their high mortality rate, it is one of the main causes of death in the world. In Brazil, there are more than 1100 deaths per day, about 1 death for 90 [4] seconds, according to the Brazilian Society of Cardiology. This estimates that approximately, until the end of this year, 400 thousand Brazilian citizens will die from cardiovascular diseases.

The cardiovascular risk factors, according to [2], are age, sex, and heredity which are immutable, and some of them are related to the lifestyle led by the person such as obesity, dyslipidemia, diabetes, sedentary lifestyle, smoking, high blood pressure, stress, and inadequate diet. These factors often work together, even though some that occur in isolation may present problems, the greatest risk is when there are several factors at the same time.

The development and improvement of machine learning models were fundamental to be able to assist in medical diagnostics [10][11]. For the execution of this, the doctor relies on a series of factors to diagnose and classify a certain health problem. This way of diagnosing a patient is very similar to the way that a learning model learns and predicts. Hospitals are places where a large amount of data is generated that are[7] often not used. With the advancement of technology, it became easier to collect medical data and organize it to feed databases and use them for learning, that is, giving usefulness to this large amount of information [9].

Given the above, the seriousness and frequency with which these diseases occur are visible, bringing a high risk to our society[4]. Our aim in this article is to develop a method capable of classifying whether a given patient has heart disease or not based on his health and lifestyle information through a comparison between machine learning techniques.

The dataset that will be used [5] has data of 70000 people, has a column that indicates whether or not a certain patient has cardiovascular diseases and among its attributes are the risk factors.

The article is organized as follows. A review will be carried out on the works listed in Section 2. Details on the methodology, models, and metrics used will be compiled in Section 3. In Section 4 we will provide the experiments that were done and detail how they were performed. Finally, the results will be discussed.

II. Related Work

Currently, there is a large volume of data on cardiovascular diseases, data learning techniques can be used in conjunction with this data to assist in the diagnosis of this type of disease. We briefly present three works that have a similar objective to the present work.

In [6] a comparative study of the performance of four techniques to predict the presence or absence of cardiovascular diseases is presented. The algorithms used are; RIPPER, Decision Tree (AD), Multilayer Perceptron (MLP), and Linear Support Vector Machines (SVM). In this study, the following metrics were used: recall, rate of true negatives, rate of false positives, error rate, and accuracy. Among the classifiers, SVM obtained the best results.

The work of [3] aims to use a reduced number of attributes to accurately predict the presence of cardiovascular diseases. Initially, thirteen attributes were used to perform the classification, but they managed to reduce this number to eleven while maintaining very similar values of accuracy. The work applies different algorithms for the classification task and analyzes which of the algorithms managed to make the predictions with greater precision. They examined the following algorithms: Naive Bayes (NB), J48 Decision Tree, and Bagging. In this article, it is discussed that in the literature the Naive Bayes and Bagging algorithms have shown good results for the classification of cardiovascular diseases. The Bagging algorithm showed the best result.

[1] presents a system developed in Matlab that predicts the risk of cardiovascular diseases. The work makes use of the major risk factors associated with cardiovascular diseases to make the prediction. Based on this premise, they developed a hybrid model. The model is the result of the combination of an artificial neural network and a genetic algorithm, the latter being used in the initialization of weights for the neural network. Using the technique defined above, the system was able to achieve 89 % accuracy.

III. Metodology

Our problem results in a binary classification: if a person has cardiovascular disease (1) or if a person does not have that disease (0). The data set, which we used, was extracted from Kaggle [5]. This set was created during medical examinations of 70,000 people, all attributes were collected during a single examination. The set contains 34 979 people with cardiovascular diseases and 35 021 without.

The attributes in this set are age, height, weight, gender, systolic blood pressure, diastolic blood pressure, cholesterol, glucose, smoking, alcohol intake, and physical activity. It is important to notify that the last attributes: smoking, alcohol, and physical activity, are subjective responses of patients. In addition, the cholesterol and glucose attributes are categorized into three relative levels: normal, above normal, much above normal. The other attributes are on numerical scales.

A. Models

We separated our dataset into training and testing, 25 % of the data for testing was separated. We will use supervised learning models, which is when we have input variables (x) and an output variable (y), and then we use the algorithm to learn the input to an output mapping function. Supervised learning problems are classified into regression and classification problems, the latter being the type of our problem. The code was made in Python and can be found by accessing the following link *https://github.com/nandabezerran/MachineLearningFP*

The models that we will use will be: Logistic Regression (RL), Gaussian Discriminant Analysis (AGD), Naive Bayes (NB-G), K-Nearest Neighbors (KNN), which were implemented throughout this work, and Decision Trees (AD) and Support Vector Machines (SVM), which use the implementations provided by the scikit-learn library. These were selected because they are the most common algorithms to deal with classification problems.

We use the implemented grid-search function to calculate the hyperparameters of Logistic Regression (times and learning step) and what is available in the scikit-learn library to calculate the hyperparameters of the Decision Tree. SVM was able to make the selection of the parameter C and Gamma, however, we were unable to run to calculate the Kernel parameter because of the time it took, so it was chosen based on what is most commonly used. For KNN it was not possible to select hyperparameters by grid-search because the selected data set contains many tuples. The time required to run would be very long, as the computers available do not have a high processing power. So we use the values that are most often used by these algorithms.

B. Metrics

The following metrics were used to compare the performance of the algorithms:

- **Precision**: can be interpreted as: among the data classified as true, how much was actually true;
- **Revocation**: informs which portion of the true data has been correctly classified;
- ***F1 score***: this metric relates precision and recall, combining them into a single value;
- **Accuracy**: it checks how close the values predicted by the model are to their real reference value;
- **ROC curve**: shows how good the model created can distinguish between binary classes, being able to also analyze the AUC (Area Under ROC Curve) that summarizes the ROC curve in a single value, aggregating all ROC thresholds.

Even though they have very similar error rates, classifiers can make different errors. In order to better understand where the classifiers' errors and successes were, we used the confusion matrix, since this matrix summarizes the classifier's successes and errors.

IV. Experiments

Most of our models have hyperparameters that need to be adjusted. However, as stated before, we can only perform the grid-search for Logistic Regression and for the Decision Tree because of the lack of processing power. The other hyperparameters were selected according to our experience and examples from the discipline.

Hyperparameters:

- **Logistic Regression**: α the Learning Step and λ the Number of Iterations
  - **Tested Values:**
    - α = 0.001, 0.0001, 0.00001
    - λ = 500, 1000, 1250, 1500, 2000
  - **Selected Values:** α = 0.001 e λ = 1000
- **SVM - RBF kernel**:
  - **Tested Values:**
    - C = 2^{-1}, 2^{1}, 2^{3}
    - γ = 2^{-12}, 2^{-10}, 2^{-8}, 2^{-6}
  - **Selected Values:**
    - C = 2^{1}
    - γ = 2^{-12}
- **KNN**: K the Number of Neighboors
  - **Selected Value:** K = 3
- **Decision Tree**: Criterion e Max-Depth
  - **Tested Values:**
    - Criterion = Entropy e Gini
    - Max-Depth = 6, 7, 8, 9, 10
  - **Selected Values:**
    - Criterion = Entropy
    - Max-Depth = 9

V. Results

With the hyperparameters selected, each model was applied to the data, obtaining the results expressed in Table 1 below. It is noted that the model that best classified the data was the Decision Tree when it is seen that this model reached 73 % accuracy, recall, and F1-Score, which indicates better performance in general, the little occurrence of false negatives, and, due to the high F1-Score, the relationship between false positives and false negatives was harmonious, which gives accuracy and reliability.

The Logistic Regression model and the Gaussian Discriminant Analysis showed the worst performances with an accuracy of 57 %. In an attempt to understand why the logistic regression did not achieve better results, we considered the hypothesis that the data were not very linear. And, to test our hypothesis we applied Perceptron. We were aware that, if the data were linearly separable, Perceptron would always converge, and, with that in mind, we applied the Perceptron implementation provided by scikit-learn. We could notice that it also presented low accuracy, approximately 54 %, and made errors similar to those of logistic regression, with a high occurrence of false negatives.

Model	Accuracy	Precision	Recall	F1-Score
RL	0.57	0.65	0.57	0.51
KNN	0.64	0.64	0.64	0.64
ADG	0.57	0.63	0.58	0.52
NB-G	0.66	0.67	0.66	0.66
AD	0.73	0.74	0.73	0.73
SVM	0.72	0.72	0.72	0.72

Table 1: Summary of the results of the metrics on the learning algorithms for detecting cardiovascular diseases

Applying the Gaussian Discriminant Analysis, we start from the hypothesis that the classes respect a Gaussian distribution or an approximation of it. However, the results were not satisfactory, which leads us to believe that the distribution of the classes is not very close to the Gauss formula. Although there are not many cases in which healthy people were classified as sick, there were many cases in which sick people were classified as healthy, which explains the lower value of the F1-Score.

Models	VP	VN	FP	FN
RL	1904	8181	595	6820
KNN	5455	5815	2929	3301
ADG	2188	7843	843	6626
Naive Bayes - Gaussian	4917	6670	2016	3897
Árvores de Decisão	5942	6903	1865	2790
SVM	5872	6729	2050	2849

Table 2: Summary of model results for True Positive (VP), True Negative (VN), False Positive (FP), False Negative (FN)

The Naive Bayes model with Gaussian probability did a little better, obtaining an accuracy of 66 %, with similar values in Precision, recall, and F1-Score. Which reveals that he maintained consistency in his ratings. Table 2 shows that the number of correct answers was slightly higher than the number of errors.

The KNN algorithm proved to be a good model for the problem, as it presented most of its classifications correctly. We believe that its good performance was due to the fact that this algorithm does not require that the data be linear. Despite having presented good results, we believe that it could have done better if we had chosen a hyperparameter by grid-search.

We used the Support Vector Machine fixing the RBF (Radial-Basis Function) kernel, as this is the most used method, with fewer hyperparameters to be adjusted and a great capacity for representing data in high dimensions. The best hyperparameters returned by GridSearch were an average C and a very small $gamma$, which reveals that the layout of the classes was defined by a simpler format and with reasonable margin tolerance. And its good application to the data is notorious because it correctly identified many sick people and many healthy people.

But what stood out was the Decision Tree which, through logical rules for class separation, achieved 73 % accuracy. We believe that this method was better adjusted to the data because it is based on logical rules, and because the impurity of the leaves and the adjustments of the divisions of the nodes are made by entropy, which is an impurity index more sensitive than the Gini index because it penalizes more the impurities and, therefore, was the criterion chosen by GridSearch.

VI. Conclusion

The results show that the model with the best performance was the Decision Tree with entropy as the impurity index, which is an index more sensitive to the level of heterogeneity of the nodes. With that, we managed to reach our goal of finding a good method that classifies a patient as cardiac or healthy. Although we have observed a greater difficulty in distinguishing sick people because the number of false healthy people has regularly presented itself greater than the false sick. That is, if a person has been identified as sick, it is very likely that they are, but if they are classified as healthy, there is still a chance that this result is wrong.

Referências

[1] S. U. Amin, K. Agarwal, and R. Beg. Genetic neural network based data mining in prediction of heart disease using risk factors. In *2013 IEEE Conference on Information Communication Technologies*, pages 1227–1231, 2013.

[2] Inaê Bispo, Patrícia Santos, Maria Carneiro, Tamiles Santana, Marcos Henrique, Cezar Casotti, Isleide Santos, and José Carneiro. Fatores de risco cardiovascular e características sociodemográficas em idosos cadastrados em uma unidade de saúde da família. *O Mundo da Saúde*, 40:334–342, 09 2016.

[3] Vikas Chaurasia and Saurabh Pal. Data mining approach to detect heart diseases. *International Journal of Advanced Computer Science and Information Technology (IJACSIT)*, 2:56–66, 11 2013.

[4] Sociedade Brasileira de Cardiologia. Cardiômetro - mortes por doenças cardiovasculares no brasil. http://www.cardiometro.com.br. accessed: 10.07.2020.

[5] Kaggle. Cardiovascular disease dataset, 01 2019.

[6] Milan Kumari and Sunila Godara. Comparative study of data mining classification methods in cardiovascular disease prediction. 2011.

[7] Weitong Chen Xuming Han Minghao Yin Lin Yue, Dongyuan Tian. Deep learning for heterogeneous medical data analysis. *World Wide Web*, 03 2020.

[8] Jan A. Olvera Lopez E, Ballard BD. Cardiovascular disease. https://www.ncbi.nlm.nih.gov/books/NBK535419/. accessed: 10.07.2020.

[9] Mohit Sharma Sandeep Kaushik Sabyasachi Dash, Sushil Kumar Shakyawar. Big data in healthcare: management, analysis and future prospects. *Journal of Big Data volume*, 06 2019.

[10] Paul Sajda. Machine learning for detection and diagnosis of disease. *Annual Review of Biomedical Engineering*, 8(1):537–565, 2006. PMID: 16834566.

[11] Swati Shilaskar and Ashok Ghatol. Feature selection for medical diagnosis : Evaluation for cardiovascular diseases. *Expert Systems with Applications*, 40(10):4146 – 4153, 2013.