Identificação de Doenças Cardiovasculares - Analisando Modelos de Aprendizagem de Máquina

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 – Modelos – Classificação

I. Introdução

As doenças cardiovasculares são um grupo de enfermidades que afetam o coração e os vasos sanguíneos [8]. Elas trazem uma grande preocupação devido à sua alta taxa de mortalidade, é uma das principais causas de morte no mundo. No Brasil, são mais de 1100 mortes por dia, cerca de 1 morte por 90 segundos [4], segundo a Sociedade Brasileira de Cardiologia. Esta estima que aproximadamente, até o final deste ano, 400 mil cidadãos brasileiros morrerão por doenças cardiovasculares.

Os fatores de risco cardiovascular, segundo [2], são idade, sexo e hereditariedade que são imutáveis e alguns deles são relacionados com o estilo de vida levado pela pessoa como obesidade, dislipidemias, diabetes, sedentarismo, tabagismo, hipertensão arterial, estresse e dieta inadequada. Esses fatores frequentemente atuam em conjunto, mesmo que alguns que ocorrem de maneira isolada possam apresentar problemas, o maior risco é quando há vários fatores ao mesmo tempo.

O desenvolvimento e o aprimoramento de modelos de aprendizagem de máquina foi fundamental para poder auxiliar em diagnósticos médicos [11][10]. Para a execução deste, o médico se baseia em uma série de fatores para diagnosticar e classificar um certo problema de saúde. Essa forma de diagnosticar um paciente se assemelha bastante com a maneira que um modelo de aprendizagem aprende e prevê. Hospitais são locais em que são gerados uma ampla quantidade de dados [7] que muitas vezes não são utilizados. Com o avanço da tecnologia, tornou-se mais fácil de coletar dados médicos e organiza-los para alimentar bases de dados e utiliza-las para aprendizagem, ou seja, dando uma utilidade pra essa grande quantidade de informação [9].

Diante do exposto, é visível a seriedade e a frequência com que essas doenças ocorrem, trazendo um alto risco para nossa sociedade [4]. Nosso objetivo nesse artigo é elaborar um método capaz de classificar se um determinado paciente apresenta ou não cardiopatias a partir das suas informações de saúde e estilo de vida por meio de um comparativo entre técnicas de aprendizagem de máquina.

O dataset que será utilizado [5] possui um conjunto de dados de 70000 pessoas, possui uma coluna que indica se certo paciente possui ou não doenças cardiovasculares e dentre seus atributos estão os fatores de risco.

O artigo é organizado da seguinte maneira. Será realizada uma revisão sobre os trabalhos relacionados na Seção 2. Os detalhes sobre a metodologia, modelos e métricas utilizadas estarão compilados na Seção 3. Na Seção 4 iremos prover os experimentos que foram feitos e detalhar como foram executados. E por fim, os resultados serão discutidos.

II. Trabalhos Relacionados

Atualmente existe um grande volume de dados sobre doenças cardiovasculares, as técnicas de aprendizagem de dados podem ser utilizadas em conjunto com esses dados para auxiliar no diagnóstico deste tipo de doença. Apresentamos brevemente três trabalhos que possuem um objetivo similar ao do presente trabalho.

Em [6] é apresentado um estudo comparativo da performance de quatro técnicas para realizar a predição da presença ou ausência de doenças cardiovasculares. Os algoritmos utilizados são; RIPPER, Arvore de Decisão (AD), Perceptron Multicamadas (MLP) e Máquinas de Vetores de Suporte lineares (SVM). Neste estudo foram utilizados as métricas: revocação, taxa de verdadeiros negativos, taxa de falsos positivos, taxa de erro e acurácia. Entre os classificadores o SVM obteve os melhores resultados.

O trabalho de [3] tem como objetivo utilizar um número reduzido de atributos para predizer com precisão a presença de doenças cardiovasculares. Inicialmente eram utilizados treze atributos para realizar a classificação, mas eles conseguiram reduzir este número para onze mantendo valores muito similares de acurácia. O trabalho aplica diferentes algoritmos para a tarefa de classificação e analisa qual dos algoritmos conseguiu

realizar as predições com maior precisão. Eles examinaram os seguintes algoritmos: Naive Bayes(NB), J48 Arvore de Decisão e Bagging. Neste artigo é discutido que na literatura os algoritmos Naive bayes e Bagging tem mostrados bons resultados para a classificação de doenças cardiovasculares. O algoritmo Bagging apresentou o melhor resultado.

[1] apresenta um sistema desenvolvido em Matlab que faz a predição do risco de doenças cardiovasculares. O trabalho faz uso dos maiores fatores de risco associados as doenças cardiovasculares para realizar a predição. A partir dessa premissa eles desenvolveram um modelo híbrido. O modelo é o resultado da combinação de uma rede neural artificial e um algoritmo genético, sendo este último utilizado na inicialização dos pesos para a rede neural. Usando a técnica acima definida o sistema conseguiu atingir 89% de acurácia.

III. Metodologia

Nosso problema resulta numa classificação binária: se uma pessoa tem uma doença cardiovascular (1) ou se uma pessoa não tem essa doença (0). O conjunto de dados, que utilizamos, foi extraído do Kaggle [5]. Esse conjunto foi criado durante exames médicos de 70 000 pessoas, todos os atributos foram coletados durante um único exame. O conjunto contém 34 979 pessoas com doenças cardiovasculares e 35 021 sem.

Os atributos nesse conjunto são: idade, altura, peso, gênero, pressão arterial sistólica, pressão arterial diastólica, colesterol, glucose, fumar, ingestão de álcool e atividade física. É importante notificar que os últimos atributos: fumar, álcool e atividade física, são respostas subjetivas dos pacientes. Além disso, os atributos colesterol e glucose são categorizados em trés niveles relativos: normal, acima do normal, muito acima do normal. Os outros atributos estão em escalas numéricas.

A. Modelos

Separamos o nosso dataset em treino e teste, foi separado 25% dos dados para teste. Utilizaremos modelos de aprendizagem supervisionada, que é quando temos variáveis de entrada (x) e uma variável de saída(y), e então utilizamos o algoritmo para aprender a função de mapeamento da entrada para a saída. Problemas de aprendizagem supervisionada se classificam em problemas de regressão e classificação, este ultimo é o tipo do nosso problema. O código foi feito em Python e pode ser encontrado acessando o seguinte link *https://github.com/nandabezerran/MachineLearningFP*

Os modelos que iremos utilizar serão: Regressão Logística(RL), Análise do Discriminante Gaussiano(AGD), Naive Bayes(NB-G), K-Nearest Neighbors(KNN), que foram implementados ao longo desse trabalho e Árvores de Decisão(AD) e Máquinas de Vetores de Suporte(SVM), que utilizamos as implementações disponibilizadas pela biblioteca scikit-learn. Estes foram selecionados por serem os algoritmos mais comuns para tratar problemas de classificação.

Utilizamos a função grid-search implementada para calcular os hiperparâmetros da Regressão Logística(épocas e o passo de apredizado) e o que está disponível na biblioteca scikit-learn para calcular os hiperparâmetros da Árvore de Decisão. O SVM conseguimos fazer a seleção do parâmetro C e Gama, entretanto não conseguimos rodar para calcular o parâmetro Kernel por causa do tempo que era levado, logo este foi escolhido com base no que é mais comumente utilizado. Para o KNN não foi possível selecionar os hiperparâmetros por grid-search pelo fato do conjunto de dados selecionados conter muitas tuplas. O tempo necessário para rodar seria muito grande, pois os computadores à disposição não possuem um alto poder de processamento. Então utilizamos os valores que são mais frequentemente utilizados por esses algoritmos.

B. Métricas

Foram utilizadas as seguintes métricas para a comparação do desempenho dos algoritmos:

- **Precisão**: pode ser interpretado como: entre os dados classificados como verdadeiros, quantos eram realmente verdadeiros;
- **Revogação**: informa qual porção dos dados verdadeiros foram corretamente classificados;
- ***F1 score***: esta métrica relaciona a precisão e a revocação, combinando-as em num único valor;
- **Acurácia**: ela verifica o quão próximo os valores preditos pelo modelo estão do seu valor real de referência;
- **Curva ROC**: mostra o quão bom o modelo criado pode distinguir entre as classes binárias, podendo analisar também a AUC (Area Under ROC Curve) que resume a curva ROC em um único valor, agregando todos os limiares da ROC.

Mesmo tendo taxas de erros muito similares, classificadores podem cometer erros distintos. Para melhor compreender onde estavam os erros e acertos dos classificadores utilizamos a matriz de confusão, uma vez que está matriz sumariza os acertos e erros de um classificador.

IV. Experimentos

A maioria dos nossos modelos tem hiperparâmetros que precisam ser ajustado. Entretanto, como dito antes nos só podemos executar o grid-search para a Regressão Logística e para a Árvore de Decisão por causa da falta de poder de processamento. Os outros hiperparâmetros foram selecionados de acordo com a nossa experiência e exemplos da disciplina.

Hiperparâmetros:

- **Regressão Logística**: α o Passo de Aprendizagem e λ o Número de Iterações
  - **Valores testados:**
    - α = 0.001, 0.0001, 0.00001
    - λ = 500, 1000, 1250, 1500, 2000
  - **Valores selecionados:** α = 0.001 e λ = 1000
- **SVM - kernel RBF**:
  - **Valores testados:**
    - C = 2^{-1}, 2^{1}, 2^{3}
    - γ = 2^{-12}, 2^{-10}, 2^{-8}, 2^{-6}
  - **Valor selecionado:**
    - C = 2^{1}
    - γ = 2^{-12}
- **KNN**: K o Número de Vizinhos

  - **Valor selecionado:** K = 3
- **Árvores de Decisão**: Critério e Max-Depth
  - **Valores testados:**
    * Critério = Entropy e Gini
    * Max-Depth = 6, 7, 8, 9, 10
  - **Valores selecionados:**
    * Critério = Entropy
    * Max-Depth = 9

V. Resultados

Com os hiperparâmetros selecionados, cada modelo foi aplicado aos dados obtendo os resultados expressos na Tabela 1 abaixo. Nota-se que o modelo que melhor classificou os dados foi a Árvore de Decisão quando se vê que este modelo alcançou 73% em acucária, recall e F1-Score, o que indica melhor performance em geral, pouca ocorrência de falsos negativos e, por ter o F1-Score alto, a relação entre falsos positivos e falsos negativos foi harmônica, o que dá confiabilidade a acurácia.

O modelo de Regressão Logística e a Análise de Discriminante Gaussiano foram os que apresentaram os piores desempenhos com acurácia de 57%. Procurando entender a razão pela qual a regressão logística não alcançou melhores resultados, cogitamos a hipótese de que os dados não eram muito lineares. E, para testar a nossa hipótese aplicamos o Perceptron. Tínhamos conhecimento que, se os dados fossem linearmente separáveis, o Perceptron sempre iria convergir e, com isso em mente, aplicamos a implementação do Perceptron disponibilizada pelo scikit-learn. Pudemos notar que ele também apresentou baixa acurácia, aproximadamente 54% e cometeu erros similares aos da regressão logística, apresentando grande ocorrência de falsos negativos.

Modelo	Acurácia	Precisão	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

Tabela 1: Sumário dos resultados das métricas sobre os algoritmos de aprendizagem para detecção de doenças cardiovasculares

Aplicando a Análise de Discriminante Gaussiano, partimos da hipótese de que as classes respeitam uma distribuição gaussiana ou uma aproximação dela. Porém, os resultados não foram satisfatórios, o que leva a crer que a distribuição das classes não se aproxima muito da fórmula de Gauss. Embora não ocorra muitos casos em que pessoas saudáveis foram classificadas doentes, houve muitos casos em que pessoas doentes foram classificadas saudáveis, o que explica o valor mais baixo do F1-Score.

Modelos	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

Tabela 2: Sumário dos resultados dos modelos para Verdadeiro Positivo(VP), Verdadeiro Negativo(VN), Falso Positivo(FP), Falso Negativo(FN)

O modelo Naive Bayes com probabilidade gaussiana se saiu um pouco melhor obtendo uma acurácia de 66%, com valores semelhantes em Precisão, revocação e F1-Score. O que revela que ele manteve a consistência em suas classificações. Na tabela 2 é possível observar que a quantidade de acertos foi um pouco maior que a quantidade de erros.

O algoritmo de KNN se mostrou um bom modelo para o problema, pois apresentou grande parte de suas classificações corretamente. Acreditamos que sua boa performance tenha sido pelo fato de que esse algoritmo não necessita que os dados sejam lineares. Apesar de ter apresentado bons resultados, acreditamos que ele poderia ter se saído melhor se tivéssemos escolhido um hiperparâmetro por grid-search.

Utilizamos a Máquina de Vetores Suporte fixando o kernel RBF (Radial-Basis Function), pois este é o método mais utilizado, com menos hiperparâmetros a serem ajustados e uma grande capacidade de representação dos dados em altas dimensões. Os melhores hiperparâmetros retornados pelo GridSearch foram um C mediano e um γ bastante pequeno, o que revela que a disposição das classes foi definida por um formato mais simples e com uma tolerância razoável à margem. E é notório sua boa aplicação aos dados pois identificou corretamente muitas pessoas doentes e muitas pessoas saudáveis.

Mas o que se destacou foi a Árvore de Decisão que, por meio de regras lógicas para separação das classes, conseguiu 73% de acurácia. Acreditamos que esse método se ajustou melhor aos dados por ser baseado em regras lógicas, e pela impureza das folhas e o ajustes das divisões dos nós serem feitas por entropia, que é um índice de impureza mais sensível que o índice de Gini pois penaliza mais as impurezas e, por tanto, foi o critério escolhido pelo GridSearch.

VI. Conclusão

Os resultados mostram que o modelo com melhor desempenho foi a Árvore de Decisão tendo a entropia como o índice de impureza, que é um índice mais sensível ao nível de heterogeneidade dos nós. Com isso, conseguimos atingir o nosso objetivo de encontrar um bom método que classifique um paciente em cardiopata ou saudável. Embora tenhamos observado uma dificuldade maior em discernir pessoas doentes pelo número de falsos saudáveis ter se apresentado regularmente maior que os falsos doentes. Ou seja, se uma pessoa foi identificada doente é muito provável que esteja, mas se este for classificado saudável ainda há uma chance de desse resultado estar errado. Por essa razão, futuramente investigaremos novos algoritmos e novas features a serem

considerados para o enriquecimento desse trabalho e obtenção de melhores resultados.

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.