PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS NÚCLEO DE EDUCAÇÃO A DISTÂNCIA

Pós-graduação Lato Sensu em Inteligência Artificial e Aprendizado de Máquina

Vinicio Silva Lima

PREDIÇÃO DE INSUFICIÊNCIA CARDÍACA

Belo Horizonte Junho de 2022

Vinicio Silva Lima

PREDIÇÃO DE INSUFICIÊNCIA CARDÍACA

Trabalho de Conclusão de Curso apresentado ao Curso de Especialização em Inteligência Artificial e Aprendizado de Máquina, como requisito parcial à obtenção do título de *Especialista*.

Belo Horizonte Junho de 2022

SUMÁRIO

1. Introdução	4
2. Descrição do Problema e da Solução Proposta	5
3. Canvas Analítico	6
4. Coleta de Dados	6
5. Processamento/Tratamento de Dados	8
6. Análise e Exploração dos Dados	12
7. Preparação dos Dados para os Modelos de Aprendizado de Máquina	12
8. Aplicação de Modelos de Aprendizado de Máquina	12
9. Discussão dos Resultados	13
10. Conclusão	13
11. Links	13
12. Referências	13

1. Introdução

O ser humano tem a habilidade de aprender com os problemas e tomar decisões a partir de experiências já vividas. Quando nos referimos a computação, a Inteligência Artificial consegue, por meio de algoritmos complexos de Aprendizado de Máquina, encontrar padrões matemáticos para inferir resultados, de acordo com a necessidade das pessoas. Hoje, com o avanço da Inteligência Artificial, é possível realizar predições se um paciente tem uma pré-disposição a sofrer um ataque cardiovascular baseando-se em dados, tais como: idade, pressão arterial, colesterol, entre outros dados. Apesar disso, por incrível que pareça, ainda sofremos frequentemente com baixas relacionadas a problemas de insuficiência cardíaca, até mesmo em países mais desenvolvidos.

A insuficiência cardíaca aflige ou mata um em cada dois adultos nos Estados Unidos e em outros países desenvolvidos. (Go AS, Mozaffarian D, Roger VL, et al. 2013). Uma das causas dos ataques cardiovasculares é o acúmulo de placas de colesterol nas paredes internas de artérias, conhecido como aterosclerose. Normalmente o desenvolvimento da doença é silencioso e pode desenvolver-se na adolescência ou até mesmo na infância. Quando começa a apresentar sintomas, causa listras esbranquiçadas no revestimento interno das artérias que, com o passar do tempo, transformam-se em bolsas de colesterol que podem inchar-se, dificultando a passagem do fluxo sanguíneo na parede arterial, causando desconforto no peito. Em casos de rompimento das placas de colesterol, pode causar coágulos que, se muito grandes, podem impedir o fluxo sanguíneo e resultar em um ataque cardíaco ou um derrame. (Harvard T.H Chan School of Public Health, 2022).

A prática de atividade física regular reduz o risco de doenças cardíacas, diabetes, acidente vascular cerebral, pressão alta, osteoporose e até mesmo alguns tipos de câncer. (Harvard T.H Chan School of Public Health, 2022). Apesar disso, à medida que vamos envelhecendo, temos uma diminuição de atividade física, nos deixando mais expostos a doenças cardiovasculares (Matthews CE, George SM, Moore SC, et al. 2012).

Com base nisso, entendendo a importância deste tema, este trabalho tem como objetivo aplicar técnicas de Aprendizado de Máquina em uma base de dados de Insuficiência Cardíaca encontrada na plataforma de desafios de Ciência de Dados *Kaggle.* Por meio de modelos de classificação, podemos predizer se os pacientes encontrados no *dataset* possuem insuficiência cardíaca ou não, apresentando todo o

processo de desenvolvimento do estudo como um todo e, desta forma, oferecer métricas de classificação efetivas para avaliar os modelos desenvolvidos.

2. Descrição do Problema e da Solução Proposta

Doenças cardiovasculares levam cerca de 17,9 milhões de vidas a cada ano, representando 31% de todas as mortes em todo mundo. Grande parte desses ataques de insuficiência cardíaca estão relacionados a problemas de acúmulo de placas de colesterol nas artérias. Independentemente do tamanho deste armazenamento de colesterol nas artérias, esses pacientes estão sujeitos a rompimentos inesperados nesses vasos, causando dores no peito e até mesmo derrames.

Não existe uma idade específica para um paciente ter o risco de sofrer de um ataque cardíaco. Infelizmente, muitas dessas vítimas sofrem de hipertensão, diabetes ou até mesmo hiperlipidemia.

Para evitar mais baixas de tantas pessoas, é importante que o diagnóstico de pacientes propensos a sofrerem de ataques cardíacos seja feito com antecedência. Neste requisito, modelos de Aprendizado de Máquina podem apoiar na detecção precoce de possíveis vítimas de doenças cardiovasculares, já que conseguem aprender com os padrões encontrados nas bases de dados e predizer se o paciente corre risco de ter um ataque cardíaco ou não.

Para resolver problemas de classificação, os modelos são comumente de Árvores de Decisão ou de Regressão Logística, por exemplo. Com as predições feitas, os modelos são avaliados através de métricas de classicação (Recall, Precision, F1-Score, etc) e os resultados podem ser compartilhados com uma equipe de médicos, a fim de mostrar que a Inteligência Artificial pode ser uma grande aliada na prevenção de possíveis ataques cardiovasculares.

3. Canvas Analítico

Software Analytics Canvas Project: Predição de Insuficiência Cardíaca

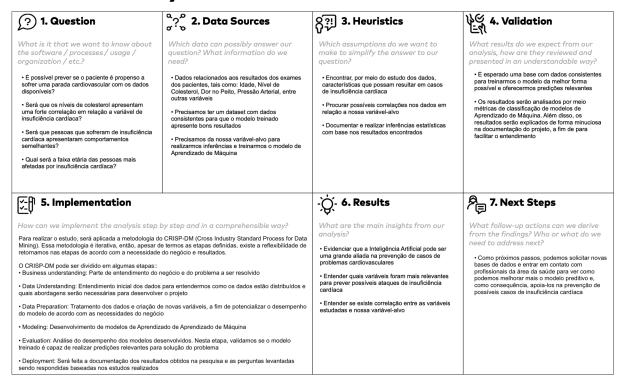


Figura 1: Canvas Análitico

4. Coleta de Dados

A base de dados *Heart Failure Prediction Dataset* foi extraída na plataforma Ka*ggle* por meio do seguinte link:

https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction . A extração foi feita no dia 20/06/2020.

A base é composta por *datasets* de pacientes dos Estados Unidos, Hungria, França e Suíça. Não foi fornecida a data de extração dos dados e a forma de como a base foi obtida.

Nome do dataset: Heart Failure Prediction Dataset

Descrição: *Dataset* com informações de pacientes americanos, suíços, franceses, húngaros com ou sem histórico de insuficiência cardíaca. O objetivo é analisar as variáveis e desenvolver um modelo capaz de realizar predições de possíveis pacientes propensos a sofrer de insuficiência cardiovascular.

Data de publicação: 10/2022

Link: https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction

Nome do Atributo	Descrição	Tipo
Age	Idade do paciente	Inteiro
Sex	Sexo do paciente	String
ChestPainType	Tipo de dor no peito	String
Cholesterol	Colesterol total	Inteiro
ExerciseAngina	Dor no peito proveniente do	String
	exercício	
FastingBS	Glicose no sangue	Inteiro
MaxHR	Máxima de batimento cardíaco	Inteiro
Oldpeak	Depressão ST induzida por	Float
	exercício relativamente	
	sossegado	
RestingECG	Conclusão do	String
	Eletrocardiograma	
ST_Slope	Taxa de frequência cardíaca	String
HeartDisease	Insuficiência Cardíaca	Inteiro
	(variável-alvo)	

5. Processamento/Tratamento de Dados

Para realizar a análise, foi utilizado o *Jupyter Notebook* e as bibliotecas pandas, numpy e category_encoders

Import das bibliotecas

```
import pandas as pd
import numpy as np
from category_encoders.one_hot import OneHotEncoder
executed in 10.4s, finished 00:59:32 2022-06-15
```

Figura 2: import das bibliotecas

Para começar a análise, precisamos entender como os nossos dados estão distribuídos, conforme imagem abaixo:

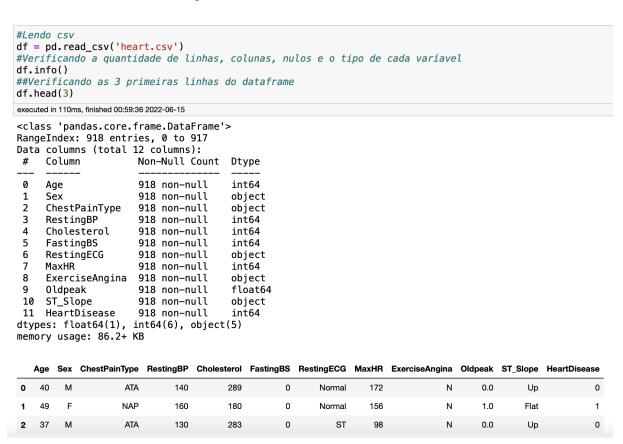


Figura 3: Leitura e entendimento inicial dos dados

Após importarmos o csv, podemos verificar que nossa base não possui variáveis nulas. Como temos atributos do tipo *object*, vamos precisar separar o nosso *dataframe* em duas partes para trabalharmos com as variáveis numéricas separadamente.

Separando as variáveis númericas

```
#Selecionando as variáveis de formato numérico
atributos_numericos = df.select_dtypes(include=['int64', 'float64'])
#Verificando se as variáveis são numéricas
atributos_numericos.info()
executed in 16ms, finished 00:18:26 2022-06-15
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 918 entries, 0 to 917
Data columns (total 7 columns):
     Column
                   Non-Null Count Dtype
0
                                    int64
     Age
                   918 non-null
     RestingBP
                   918 non-null
                                    int64
 1
     Cholesterol
                   918 non-null
                                    int64
     FastingBS
                   918 non-null
                                    int64
     MaxHR
                   918 non-null
                                     int64
     Oldpeak
                   918 non-null
                                     float64
     HeartDisease 918 non-null
                                    int64
dtypes: float64(1), int64(6)
memory usage: 50.3 KB
```

Figura 4: Separação das variáveis numéricas

Para separar as variáveis numéricas foi utilizada a função *select_dtypes* da biblioteca *pandas* e foi feito um filtro para selecionar as que possuem valores inteiros e *float.* Com o resultado atribuído à variável "atributos_numericos", foi utilizada a função *info* para conferir os resultados.

Separando as variáveis em formato string e convertendo-as para formato categórico

```
#Selecionando as variáveis de formato object e transformando em categóricas
atributos_categoricos = df.select_dtypes(include=['object']).astype('category')
#Verificando se as variáveis são categóricas
atributos categoricos.info()
executed in 27ms, finished 00:18:26 2022-06-15
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 918 entries, 0 to 917
Data columns (total 5 columns):
                     Non-Null Count Dtype
    Column
                      918 non-null
     {\tt ChestPainType}
                      918 non-null
                                       category
     RestingECG
                      918 non-null
                                       category
     ExerciseAngina
                      918 non-null
                                       category
     ST_Slope
                      918 non-null
                                       category
dtypes: category(5)
memory usage: 5.3 KB
```

Figura 5: Separação das variáveis categóricas

No caso das variáveis do tipo objeto foi feito uma conversão para formato categórico, pois temos atributos como "Sex" que apresenta resultados provenientes de categorias, como masculino e feminino. Após a atribuição do resultado à variável "atributos_categoricos", realizamos a checagem se o tipo das variáveis foram convertidos para object.



Figura 6: Métricas estatísticas para entendimento de variáveis numéricas

Utilizando nossa variável "atributos_numericos" fizemos a declaração de alguns atributos e adicionamos medidas de dispersão para entendermos melhor os dados que estamos trabalhando. No caso, podemos observar que a variável de *Cholesterol* apresenta um desvio-padrão bem alto e um valor mínimo de 0, o que pode indicar possíveis valores inválidos na base.

Filtrando as colunas numéricas e entendendo um pouco a distribuíção por quartil das variáveis

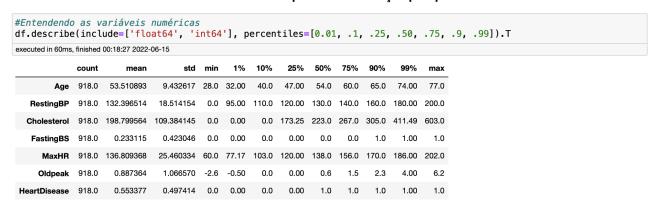


Figura 7: Analisando as variáveis com quebras por quartil

Para nos aprofundarmos um pouco mais nas variáveis, foi aplicada a função describe com quebras por quartis para entendermos melhor como está a dispersão dos atributos. Com base nisso, podemos observar que a variável de *Cholesterol*

provavelmente possui *outliers*, já que o valor máximo encontrado está muito acima da média.

Concatenando os dados categoricos e numéricos



Figura 8: Junção de dataframes

Utilizando as variáveis "atributos_categoricos" e "atributos_numericos" podemos criar um novo *dataframe* concatenando as duas bases. Isso pode ser feito agora sem muitos impeditivos, já que o tipo dos atributos foram tratados anteriormente.

Filtrando colunas categoricas e aplicando o OneHotEncoder para classifica-las numericamente



Figura 9: Aplicando o OneHotEncoder

Com o nosso novo *dataframe*, precisamos separar os valores categóricos novamente e atribuirmos a uma variável temporária chamada "cols_encoding". Feito isso, passamos as variáveis para o *OneHotEnconder* e aplicamos a transformação com o nosso método *fit*, quebrando nossas colunas de acordo com o número de categorias presentes em cada coluna. No caso de "ChestPainType", existiam 4 categorias, então foram geradas quatro colunas distintas para separá-las.

6. Análise e Exploração dos Dados

Nessa etapa você começará a explorar seus dados de uma forma mais analítica, tentando elaborar ideias, levantar hipóteses e começando a identificar padrões em seus dados. Talvez você sinta a necessidade de voltar em passos anteriores, obter mais dados e tratá-los para conseguir responder ao problema proposto. Use e abuse de ferramentas estatísticas consistentes como testes de hipóteses e intervalos de confiança. Plote gráficos que te ajudem a obter insights interessantes: desde os mais simples até gráficos mais sofisticados como boxplots, mapas de calor, etc. Aqui o uso do Python e/ou R e suas poderosas bibliotecas gráficas (Matplotlib, Seaborn, ggPlot2, etc). Apresente trechos de código com as devidas justificativas.

7. Preparação dos Dados para os Modelos de Aprendizado de Máquina

Nesta etapa você deve descrever os tratamentos realizados especificamente para os modelos de Aprendizado de Máquina escolhidos, como por exemplo a criação de atributos, o balanceamento da base de dados (*undersampling* ou *oversampling*), divisão da base em treino, validação e teste, entre outros.

8. Aplicação de Modelos de Aprendizado de Máquina

Nesta seção você deve apresentar os modelos de Aprendizado de Máquina desenvolvidos no trabalho. Mostre partes do código-fonte para ilustrar a implementação de cada modelo, além do pipeline completo do processo. A escolha dos modelos deve ser adequada ao problema proposto. Embora possa ser considerado o uso de ferramentas como Weka, Knime e Orange, por exemplo, encoraja-se a implementação com linguagens como Python ou R. É importante testar mais de um tipo de algoritmo, para que resultados distintos possam ser comparados. Por exemplo, se o trabalho trata de uma classificação, modelos como Árvores de Decisão, Redes Neurais Artificiais e Support Vector Machine poderiam ser utilizados. Além disso, devem ser escolhidas e implementadas as métricas adequadas ao problema proposto, bem como os seus resultados apresentados.

9. Discussão dos Resultados

Nesta seção você deve relatar os resultados alcançados ao final do trabalho. Mostre os resultados das métricas adotadas, seja através de gráficos, tabelas, dentre outros, que permitam a validação do seu trabalho.

10. Conclusão

Nesta seção você deve apresentar um fechamento para o trabalho. É importante apresentar um breve resumo do trabalho, resgatando o problema, como foi tratado e os resultados obtidos, bem como as limitações e perspectivas (trabalhos futuros).

11. Links

Nesta seção você pode disponibilizar *links* para repositórios, como é o caso do GitHub, onde podem ser encontrados o seu projeto, códigos-fonte, vídeos demonstrativos, dentre outros.

12. Referências

Harvard Health Publishing Harvard Medical School. **Heart Health**. 2022. Disponível em: https://www.health.harvard.edu/topics/heart-health Acesso em 02/06/2022

Harvard T.H. Chan School of Public Health. **Heart Disease**. 2022. Disponível em: https://www.hsph.harvard.edu/nutritionsource/disease-prevention/cardiovascular-disease/> Acesso em 02/06/2022

GO AS, Mozaffarian D, Roger VL, et al. **Heart disease and stroke statistics**, 2013. National Library of Medicine. Disponível em:

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5408511/ Acesso em 02/06/2022;

Matthews CE, George SM, Moore SC, et al. 2012. **Amount of time spent in sedentary behaviors and cause-specific mortality in US adults**. National Library of Medicine. Disponível em: https://pubmed.ncbi.nlm.nih.gov/22218159/ Acesso em 02/06/2022