MINISTÉRIO DA DEFESA

EXÉRCITO BRASILEIRO

DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA

INSTITUTO MILITAR DE ENGENHARIA

(Real Academia de Artilharia, Fortificação e Desenho – 1792)

TRABALHO DE INTELIGÊNCIA ARTIFICIAL

ESTUDO COMPARATIVO DE APLICAÇÃO DE ALGORITMOS CLASSIFICADORES NO CONJUNTO DE DADOS *HEART DISEASE (UCI MACHINE LEARNING REPOSITORY)*

ALUNO:

GABRIEL BOZZA

RIO DE JANEIRO

2019

SUMÁRIO

1.INTRODUÇÃO AO CONJUNTO DE DADOS

Conjunto de dados utilizado: *Heart Disease Data Set* (https://archive.ics.uci.edu/ml/datasets/Heart+Disease)[1][2][3][4]

Número de instâncias: 303

Número de atributos: 75

Data de doação 07/01/1988

Valores ausentes? Sim

Por que escolher este conjunto de dados:

Doença cardíaca descreve uma série de condições que afetam o coração, como doença arterial coronariana, problemas de ritmo cardíaco (arritmias) e defeitos cardíacos congênitos, entre outros.

A doença cardíaca é uma das maiores causas de morbidade e mortalidade entre a população do mundo. A previsão de doença cardiovascular é considerada um dos temas mais importantes no ramo de análise de dados clínicos. A quantidade de dados no setor de saúde é enorme. A mineração de dados transforma a grande coleta de dados brutos de saúde em informações que podem ajudar a tomar decisões.

Isso faz com que a doença cardíaca seja uma grande preocupação a ser tratada. Mas é difícil identificar doenças cardíacas por conta dos vários fatores de risco contribuintes, como diabetes, pressão arterial elevada, colesterol alto, taxa de pulso anormal e muitos outros fatores. Devido a tais restrições, os cientistas se voltaram para abordagens modernas, como mineração de dados e aprendizado de máquina, para prever a doença [5].

Neste trabalho serão aplicadas abordagens de aprendizado de máquina para classificar se uma pessoa está sofrendo de uma doença cardíaca ou não, usando o conjunto de dados: Cleveland Heart Disease, do repositório UCI.

Informações do conjunto de dados:

Este conjunto de dados contém 75 atributos, mas todos os experimentos publicados referem-se ao uso de um subconjunto de 14 deles. Em particular, o banco de dados de Cleveland é o único que tem sido usado por pesquisadores de ML para este *data set* (uma vez que os outros conjuntos de dados disponíveis estão com a maioria dos valores dos atributos ausentes).

O campo *target* (objetivo) refere-se à presença de cardiopatia no paciente. É inteiro valorizado de 0 (sem presença) até 4. Experimentos com o banco de dados de Cleveland concentraram-se em simplesmente tentar distinguir a presença (valores 1, 2, 3 e 4) da ausência (valor 0) de cardiopatia.

Informações sobre os 14 atributos usados:

Nome do atributo	Descrição	Tipo
Idade	Idade em anos do paciente	Inteiro
Sexo	Sexo do paciente: 1 = Homem 0 = Mulher	Categórico (inteiro)
CP	Tipo de dor torácica: 1 = Angina típica 2 = Angina atípica 3 = Dor não anginal 4 = Assintomática	Categórico (inteiro)
trestbps	Pressão sanguínea em repouso (em mm Hg na admissão ao hospital)	Contínuo (float)
Chol	Soro colesterol em mg/dl	Contínuo (float)
FBS	Açúcar no sangue em jejum > 120 mg/dl: 1 = Verdadeiro 0 = Falso	Categórico (inteiro)
restecg	Resultados do eletrocardiograma em repouso: 0 = Normal 1 = Tendo a anomalia da onda de ST-T (inversões da onda de t e/ou elevação ou depressão do ST de > 0, 5 mV) 2 = Mostrando hipertrofia ventricular esquerda provável ou definitiva	Categórico (inteiro)

thalach	Frequência cardíaca máxima alcançada	Inteiro
exang	Angina induzida pelo exercício: 1 = Sim 0 = Não	Categórico (inteiro)
oldpeak	Depressão do ST induzida pelo exercício relativo ao descanso	Contínuo (float)
inclinação	A inclinação do pico do segmento ST do exercício: 1 = Positiva 2 = Plana (inclinação igual a 0) 3 = Negativa	Categórico (inteiro)
CA	Número de vasos principais (0-3) coloridos por flouroscopia	Inteiro
Thal	Tipo de defeito do coração: 3 = Normal 6 = Defeito fixo 7 = Defeito reversível	Categórico (inteiro)
target	Diagnóstico de doença cardíaca (estado angiográfico da doença): 0 = Ausência de cardiopatia (1, 2, 3 ou 4) = Presença de cardiopatia	Categórico (inteiro)

Por que esses parâmetros:

No conjunto de dados real, existiam 75 atributos, mas considera-se apenas os 14 acima porque:

1. **Idade:** A idade é o fator de risco mais importante no desenvolvimento de doenças cardiovasculares.

2. **Sexo:** Os homens estão em maior risco de doenças cardíacas do que as mulheres na pré-menopausa.

3. **Angina (Dor no Peito)**: Angina é a dor no peito ou desconforto causado quando o músculo cardíaco não recebe sangue suficiente rico em oxigênio. Pode parecer uma pressão ou um aperto no peito. O desconforto também

pode ocorrer em seus ombros, braços, pescoço, mandíbula ou costas. A dor da angina pode até parecer indigestão.

4. **Pressão arterial de repouso:** Ao longo do tempo, a pressão arterial elevada pode danificar as artérias que alimentam o coração. Pressão arterial elevada que ocorre com outras condições, tais como obesidade, colesterol alto ou diabetes, aumenta o risco ainda mais.

5. **Soro Colesterol:** Um alto nível de lipoproteína de baixa densidade (LDL) colesterol (o colesterol "ruim") aumenta a chance de que as artérias estejam mais estreitas. Um alto nível de triglicerídeos, um tipo de gordura do sangue relacionada à sua dieta, também aumenta o risco de ataque cardíaco. No entanto, um alto nível de lipoproteína de alta densidade (HDL) colesterol (o "bom" colesterol) reduz o risco de ataque cardíaco.

6. **Açúcar no sangue em jejum**: Não produzir insulina suficiente ou não responder à insulina corretamente faz com que os níveis de açúcar no sangue do corpo subam, aumentando o risco de ataque cardíaco.

7. **ECG de repouso**: Um eletrocardiograma (ECG) registra a atividade elétrica do coração em repouso. Ele fornece informações sobre a frequência e ritmo cardíacos e mostra se há aumento do coração devido à pressão alta (hipertensão) ou evidência de um ataque cardíaco anterior (infarto do miocárdio).

8. **Frequência cardíaca máxima alcançada**: O aumento do risco cardiovascular associado à aceleração da frequência cardíaca é comparável ao aumento do risco observado com pressão arterial elevada. Tem sido demonstrado que um aumento na frequência cardíaca em 10 batimentos por minuto foi associado a um aumento no risco de morte cardíaca em pelo menos 20%, e este aumento no risco é semelhante ao observado com um aumento da pressão arterial sistólica em 10 mm Hg.

9. **Angina induzida por exercício**: A ocorrência de angina após a realização de exercício físico é um fator de risco, pois indica que o coração não está

recebendo sangue rico em oxigênio suficiente durante a realização de exercícios, o que pode indicar a presença de alguma doença cardíaca.

10. **oldpeak e inclinação (Segmento de ST de pico de exercício)**: Em geral, a ocorrência de depressão horizontal ou para baixo do segmento de ST em uma menor carga de trabalho ou frequência cardíaca indica um pior prognóstico e maior probabilidade de doença. Outro achado altamente indicativo de CAD significativo é a ocorrência de elevação do segmento ST > 1 mm (muitas vezes sugerindo a presença de doença cardíaca grave).

11. **CA (Número de vasos principais coloridos por fluoroscopia)**: A fluoroscopia é uma técnica de imagem comumente utilizada na medicina para obter imagens em tempo real em movimento das estruturas internas de um paciente, com a utilização de contrastes e raios-X. Dessa forma, pode-se visualizar alguma região do coração que não está recebendo sangue, o que pode indicar algum problema cardíaco.

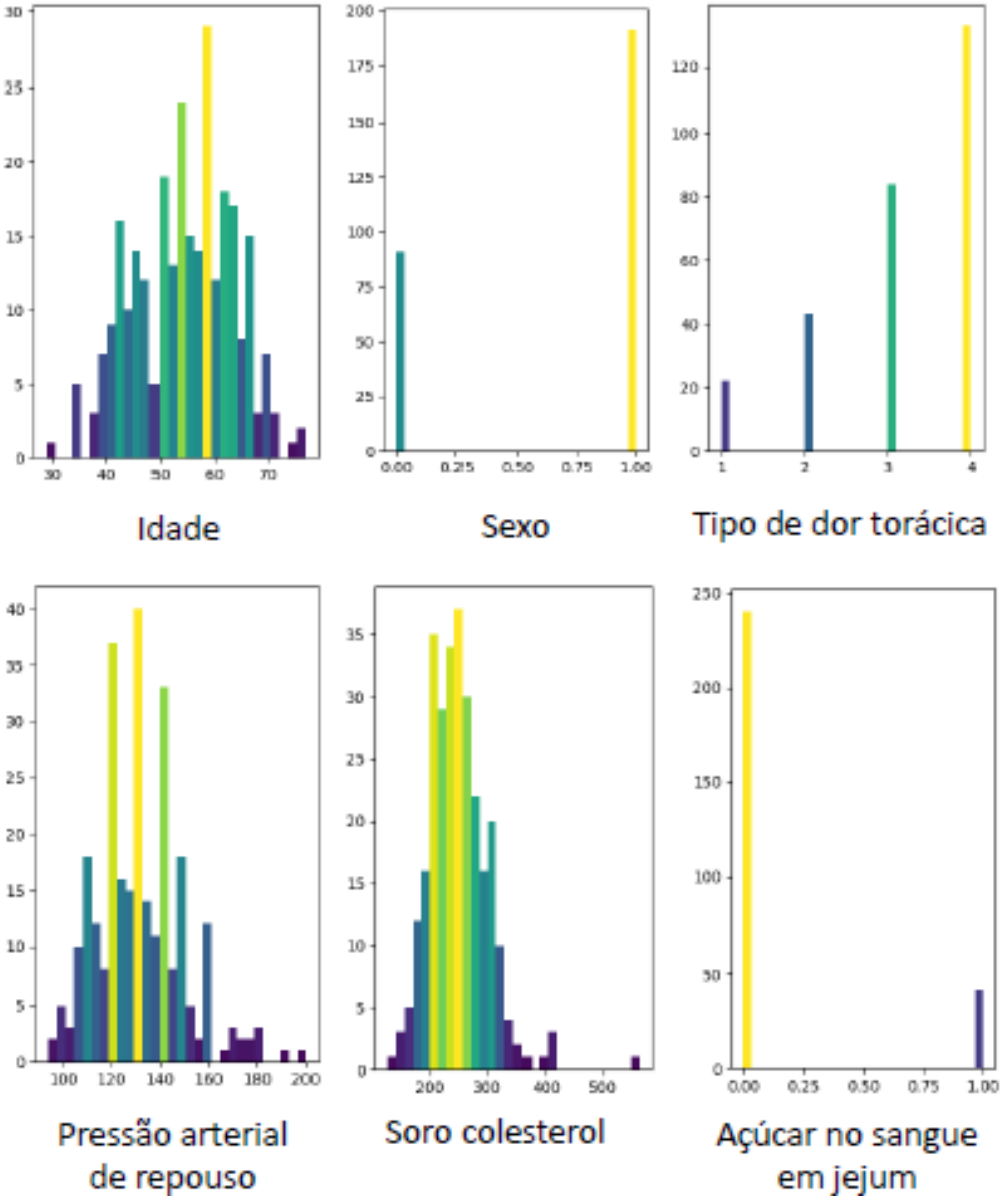
12. **Thal (Teste de Thallium)**: É um teste que mostra como o sangue circula no coração e pode mostrar danos causados ao músculo cardíaco devido a ataques cardíacos anteriores ou doenças cardíacas.

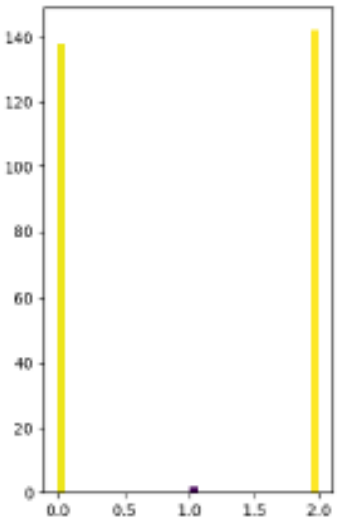
2.VISÃO ESTATÍSTICA

Dentre as 303 instâncias do conjunto de dados, apenas 6 delas tinham valores nulos em alguns de seus atributos.

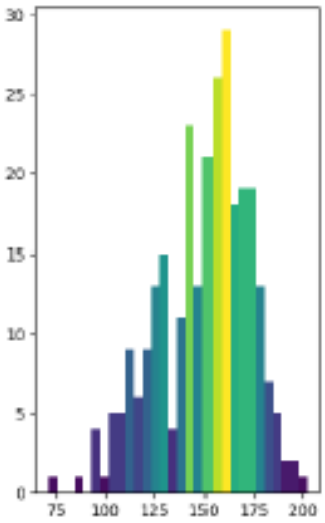
Atributo	Número de dados nulos
CA	4
Thal	2

Abaixo, estão os histogramas que mostram a distribuição geral dos dados por atributo do conjunto de dados analisado:

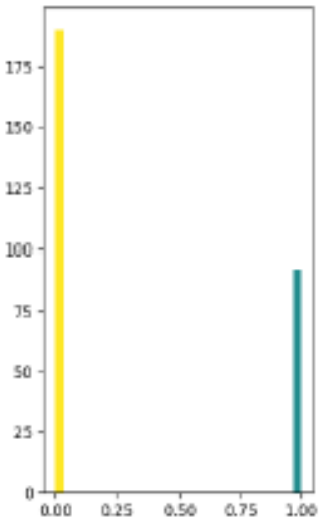




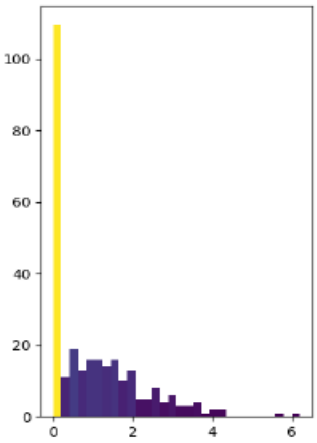
ECG em repouso



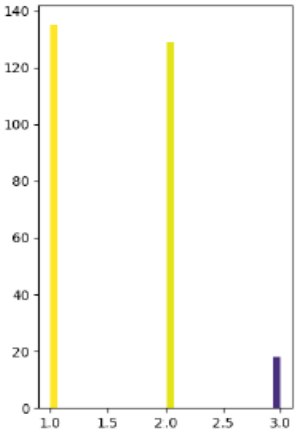
Frequência cardíaca máx.



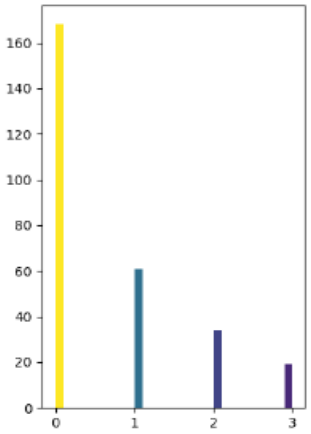
Angina induzida por exercício



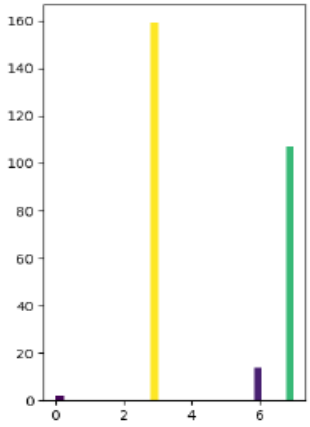
Depressão ST induzida por exercício



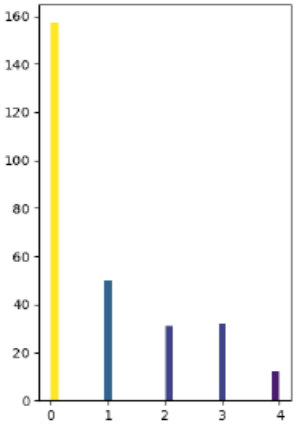
Inclinação ST no pico do exercício



Número de vasos coloridos por fluoroscopia

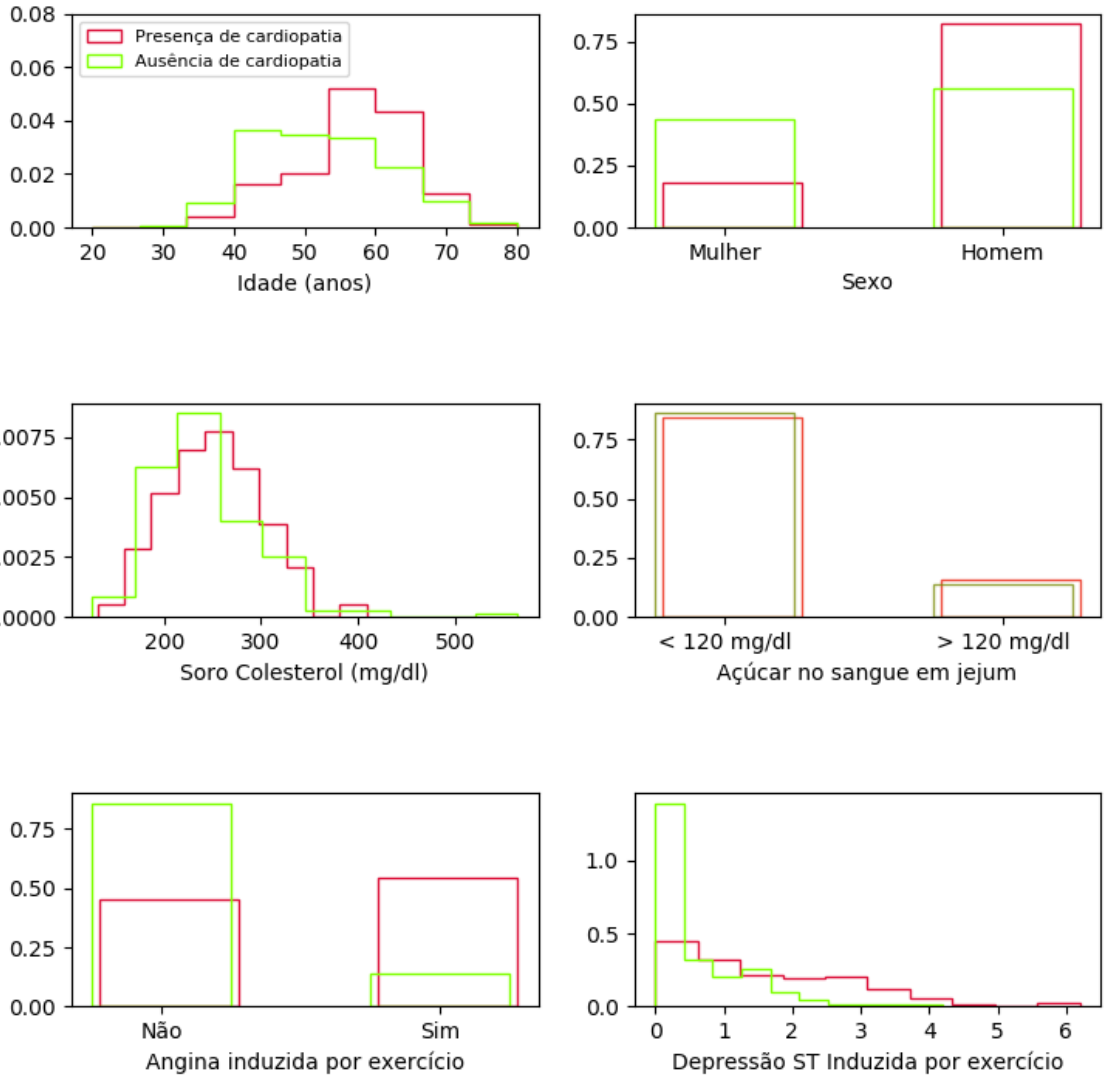


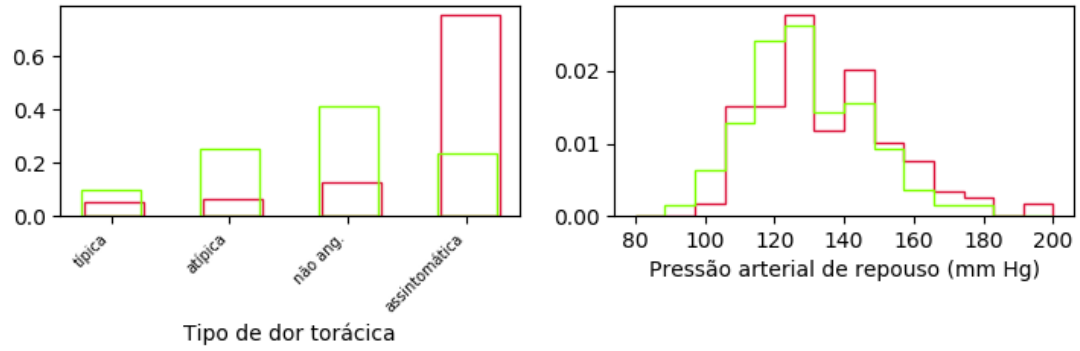
Resultado do teste de Thallium

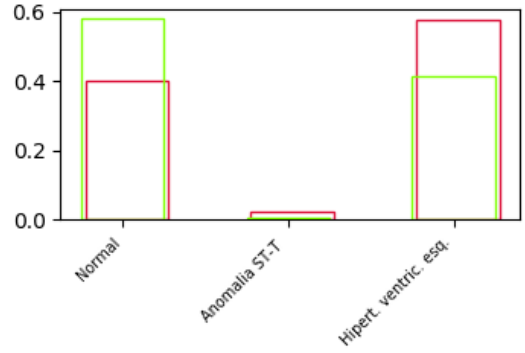


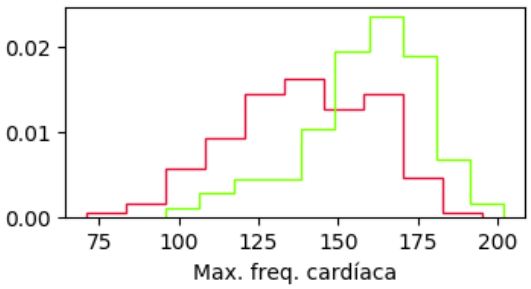
Classificação do risco de ter doença cardíaca

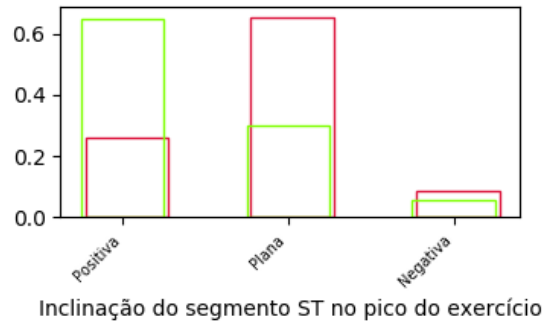
Para visualizar a distribuição dos dados de acordo com a classificação em presença (139 instâncias) ou ausência de cardiopatia (164 instâncias), utilizam-se histogramas dos 13 atributos para as diferentes classificações (0-Ausência de cardiopatia (Verde); 1-Presença de cardiopatia (Vermelho)):

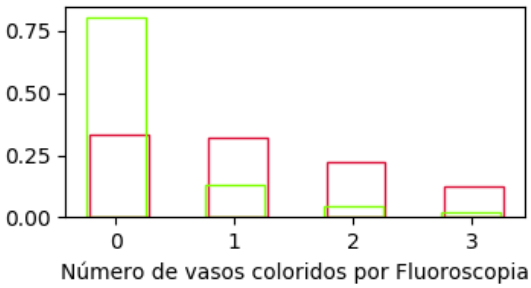


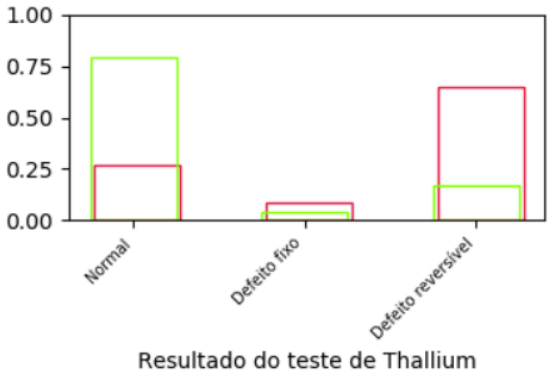












Estes histogramas mostram que as pessoas com doença cardíaca tendem a ser mais velhas e do sexo masculino, têm pressão arterial mais elevada, níveis mais elevados de colesterol e menor frequência cardíaca máxima no teste de Thallium do que as pessoas sem a doença (também, normalmente não apresentam quadro de dores no peito).

Outra maneira de visualizar a distribuição dos dados é utilizando a função describe(), a qual mostra informações estatísticas dos atributos:

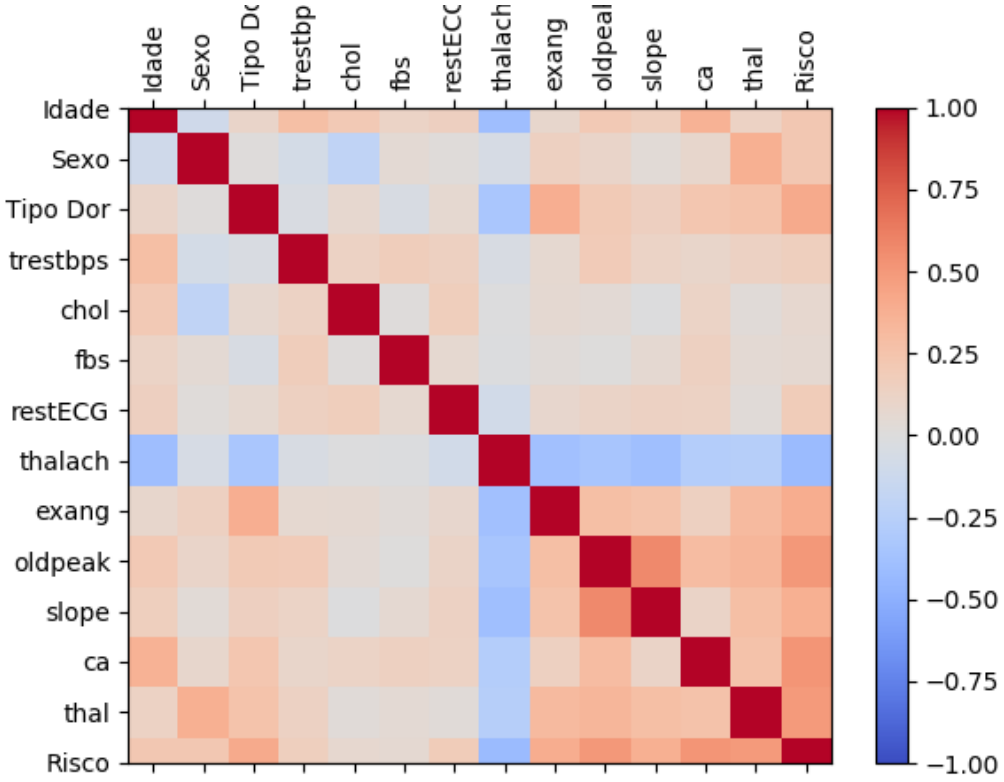
	idade	sexo	cp	trestbps	chol	fbs	restecg
count	303,00	303,00	303,00	303,00	303,00	303,00	303,00
mean	54,44	0,68	3,16	131,69	246,69	0,15	0,99
std	9,04	0,47	0,96	17,60	51,78	0,36	0,99
min	29,00	0,00	1,00	94,00	126,00	0,00	0,00
25%	48,00	0,00	3,00	120,00	211,00	0,00	0,00
50%	56,00	1,00	3,00	130,00	241,00	0,00	1,00
75%	61,00	1,00	4,00	140,00	275,00	0,00	2,00
max	77,00	1,00	4,00	200,00	564,00	1,00	2,00

	thalach	exang	oldpeak	slope	ca	thal	target
count	303,00	303,00	303,00	303,00	299,00	301,00	303,00
mean	149,61	0,33	1,04	1,60	0,67	4,73	0,94
std	22,88	0,47	1,16	0,62	0,94	1,94	1,23
min	71,00	0,00	0,00	1,00	0,00	3,00	0,00
25%	133,50	0,00	0,00	1,00	0,00	3,00	0,00
50%	153,00	0,00	0,80	2,00	0,00	3,00	0,00
75%	166,00	1,00	1,60	2,00	1,00	7,00	2,00
max	202,00	1,00	6,20	3,00	3,00	7,00	4,00

Embora as informações estatísticas da maneira como estão expostas acima possam ser muito úteis, principalmente aliadas ao gráfico do tipo *boxplot*, para o conjunto de dados escolhido a melhor maneira de analisar a distribuição dos dados é através dos histogramas (considerando as categorias e a distribuição geral), uma vez que a maioria dos atributos (9 dos 13 atributos) é do tipo categórico.

Ao analisar o conjunto de dados, é interessante procurar atributos que tenham altas correlações para uma posterior decisão da eliminação de alguns dos atributos utilizados (por serem dados redundantes), dessa forma, tornando o *data set* mais simples para ser processado pelos algoritmos de classificação e entendido por quem utilizará os dados posteriormente.

Para ter uma noção das correlações entre os atributos plotou-se um *heat map* das correlações (matriz de correlação colorida), o qual pode ser visto abaixo:



Pode-se perceber que, no conjunto de dados analisado, nenhum dos pares de atributos não categóricos apresentam correlação positiva alta(>0,5), nem correlação negativa alta(<-0,5), logo não se pode inferir que algum desses atributos seja redundante e necessite ser retirado do conjunto analisado.

3. PRÉ-PROCESSAMENTO

Funções de pré-processamento utilizadas:

Para o tratamento dos dados ausentes, como apenas seis valores estavam faltando no conjunto de dados (4 do atributo ‘ca’ e 2 do atributo ‘thal’) e como o conjunto de dados possui poucas instâncias (303), foi utilizada uma função que substitui os dados ausentes pela média do atributo (função da biblioteca Pandas chamada ‘fillna’ tendo como parâmetro a média do atributo), ao invés de eliminar as instâncias com valores faltando:

```
#df é o nome do datagrama

df['thal'] = df.thal.fillna(df.thal.mean())
df['ca'] = df.ca.fillna(df.ca.mean())
```

Para que todos os atributos tenham o mesmo peso na aplicação dos algoritmos, utiliza-se uma função que normaliza os valores dos atributos (função StandardScaler da biblioteca sklearn):

```
from sklearn.preprocessing import StandardScaler as ss

sc = ss()
std_scale = sc.fit(X_train)
X_train = std_scale.transform(X_train)
X_test = std_scale.transform(X_test)
```

Como todos os atributos do conjunto de dados eram numéricos, até mesmo os categóricos, não foi preciso mapear *strings* em números nem algum outro tipo de tratamento dos dados.

4.PARÂMETROS DOS CLASSIFICADORES

Inicialmente, decidiu-se dividir o conjunto de dados em 2 partes: treino e teste (80:20). O conjunto de treino foi então submetido a uma nova divisão onde um terço dele foi utilizado como conjunto de validação (para estimar as melhores combinações de parâmetros dos classificadores, ou seja, descobrir a combinação dos parâmetros de cada classificador que maximizam a acurácia no conjunto de treino).

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=7)
```

Para encontrar os melhores parâmetros e dividir o conjunto de treino em treino e validação utilizou-se a função do sklearn GridSearchCV() com cross-validation de 3-fold (divide o conjunto em 3 partes iguais e realiza o treino e teste com dois terços e um terço do conjunto, respectivamente, sendo que o processo é repetido até que cada uma das 3 partes originais tenha sido utilizada como teste) para o conjunto de treino (retorna os parâmetros que geraram a maior média de acurácia dentre os 3 *folds*):

```
from sklearn.model_selection import GridSearchCV


tuned_parameters = [{'kernel': ['rbf'], 'gamma': [1e-3, 1e-4], 'C': [1, 10,
100, 1000]}, {'kernel': ['linear'], 'C': [1, 10, 100, 1000]}]
clf = GridSearchCV(SVC(), tuned_parameters,cv=3)
clf.fit(X_train, y_train)
print(clf.cv_results_['params'][clf.best_index_])


tuned_parameters1 =
[{'n_neighbors':[1,2,3,4,5,6,7,8,9,10,15,20,25],'weights':['uniform','distanc
e'],'algorithm':['auto', 'ball_tree', 'kd_tree', 'brute']}]
clf1 = GridSearchCV(KNeighborsClassifier(), tuned_parameters1,cv=3)
clf1.fit(X_train, y_train)
print(clf1. cv_results_['params'][clf1.best_index_])
```

```
tuned_parameters2 = [{'var_smoothing':[1e-3,1e-4,1e-5,1e-6,1e-7,1e-8,1e-9,1e-10]}]
clf2 = GridSearchCV(GaussianNB(), tuned_parameters2,cv=3)
clf2.fit(X_train, y_train)
print(clf2. cv_results_[‘params’][clf2.best_index_])


tuned_parameters3 = [{'criterion':['gini','entropy'],
'splitter':['best','random']}]
clf3 = GridSearchCV(DecisionTreeClassifier(), tuned_parameters3,cv=3)
clf3.fit(X_train, y_train)
print(clf3. cv_results_[‘params’][clf3.best_index_])


tuned_parameters4 = [{'hidden_layer_sizes':[50,100,150,200,250,300],
'activation':['identity', 'logistic', 'tanh', 'relu'], 'solver':['lbfgs', 'sgd', 'adam'],
'alpha':[1e-3,1e-4,1e-5,1e-6]}]
clf4 = GridSearchCV(MLPClassifier(), tuned_parameters4,cv=3)
clf4.fit(X_train, y_train)
print(clf4. cv_results_[‘params’][clf4.best_index_])
```

Parâmetros que maximizam a acurácia

K-NN	{'algorithm': 'auto', 'n_neighbors': 6, 'weights': 'distance'}
Naive-Bayes	{'var_smoothing': 0.001}
DecisionTree	{'criterion': 'entropy', 'splitter': 'best'}
MLP	{'activation': 'identity', 'alpha': 1e-04, 'hidden_layer_sizes': 300, 'solver': 'sgd'}
SVM	{'C': 1000, 'gamma': 0.0001, 'kernel': 'rbf'}

5.DESEMPENHO DOS CLASSIFICADORES

K-NN

Acurácia do conjunto de treino	Acurácia do conjunto de teste
100,00%	80,33%

Matriz de confusão do conjunto de treino – K-NN

Previsão / Realidade	Ausência de doença	Presença de doença
Ausência de doença	129	0
Presença de doença	0	113

Matriz de confusão do conjunto de teste – K-NN

Previsão / Realidade	Ausência de doença	Presença de doença
Ausência de doença	31	8
Presença de doença	4	18

Naive-Bayes

Acurácia do conjunto de treino	Acurácia do conjunto de teste
86,78%	78,69%

Matriz de confusão do conjunto de treino - Naive-Bayes

Previsão / Realidade	Ausência de doença	Presença de doença
Ausência de doença	117	20
Presença de doença	12	93

Matriz de confusão do conjunto de teste - Naive-Bayes

Previsão / Realidade	Ausência de doença	Presença de doença
Ausência de doença	30	8
Presença de doença	5	18

DecisionTree

Acurácia do conjunto de treino	Acurácia do conjunto de teste
100,00%	77,05%

Matriz de confusão do conjunto de treino - DecisionTree

Previsão / Realidade	Ausência de doença	Presença de doença
Ausência de doença	129	0
Presença de doença	0	113

Matriz de confusão do conjunto de teste - DecisionTree

Previsão / Realidade	Ausência de doença	Presença de doença
Ausência de doença	29	8
Presença de doença	6	18

MLP

Acurácia do conjunto de treino	Acurácia do conjunto de teste
86,78%	80,33%

Matriz de confusão do conjunto de treino - MLP

Previsão / Realidade	Ausência de doença	Presença de doença
Ausência de doença	116	19
Presença de doença	13	94

Matriz de confusão do conjunto de teste - MLP

Previsão / Realidade	Ausência de doença	Presença de doença
Ausência de doença	31	8
Presença de doença	4	18

SVM

Acurácia do conjunto de treino	Acurácia do conjunto de teste
85,54%	83,61%

Matriz de confusão do conjunto de treino - SVM

Previsão / Realidade	Ausência de doença	Presença de doença
Ausência de doença	115	21
Presença de doença	14	92

Matriz de confusão do conjunto de teste - SVM

Previsão / Realidade	Ausência de doença	Presença de doença
Ausência de doença	32	7
Presença de doença	3	19

6.ANÁLISE COMPARATIVA DOS DESEMPENHOS

Para analisar os resultados dos classificadores e ranqueá-los de acordo com a utilidade da aplicação deles no conjunto de dados *Heart Disease*, será considerada a taxa de acertos das previsões no conjunto de testes, bem como as taxas de falsos positivos (Pessoas sem doença classificadas como tendo doença cardíaca) e falsos negativos (Pessoas com doença classificadas como não tendo doença cardíaca):

Resultados no conjunto de treino

Classificador	Taxa de acerto (acurácia)	Taxa de falsos positivos	Taxa de falsos negativos	Precisão	Revocação
K-NN	100,00%	0,00%	0,00%	100,00%	100,00%
Naive-Bayes	86,78%	8,26%	4,96%	82,30%	88,57%
DecisionTree	100%	0,00%	0,00%	100,00%	100,00%
MLP	86,78%	7,85%	5,37%	83,19%	87,85%
SVM	85,54%	8,68%	5,78%	81,42%	86,80%

Resultados no conjunto de teste

Classificador	Taxa de acerto (acurácia)	Taxa de falsos positivos	Taxa de falsos negativos	Precisão	Revocação
K-NN	80,33%	13,11%	6,56%	69,23%	81,82%
Naive-Bayes	78,69%	13,11%	8,20%	69,23%	78,26%
DecisionTree	77,05%	13,11%	9,84%	69,23%	75,00%
MLP	80,33%	13,11%	6,56%	69,23%	81,82%
SVM	83,61%	11,47%	4,92%	73,08%	86,36%

Analisando os resultados obtidos pelos classificadores no conjunto de teste, dado que se trata do diagnóstico de uma doença cardíaca, não se pode apenas escolher o melhor classificador baseado na acurácia deste, mas deve-se levar em conta as taxas de falsos positivos e falsos negativos no conjunto de teste, uma vez que a indicação errônea do diagnóstico de uma doença é algo perigoso.

Logo, escolhe-se o classificador que obtém a menor taxa de falsos negativos no conjunto de teste (para desempate utiliza-se a acurácia do conjunto de teste), pois a indicação de que o paciente não tem uma doença cardíaca quando este realmente tem a doença é muito mais grave e perigoso para a saúde deste do que uma indicação de que um paciente saudável pode ter uma doença cardíaca (apenas fará com que este realize outros exames para confirmação ou não da doença).

Classificadores ranqueados

Prioridade de escolha	Classificador	Taxa de acerto (acurácia)	Taxa de falsos positivos	Taxa de falsos negativos	Precisão	Revocação
1	**SVM**	83,61%	11,47%	4,92%	73,08%	86,36%
2	**MLP**	80,33%	13,11%	6,56%	69,23%	81,82%
3	**K-NN**	80,33%	13,11%	6,56%	69,23%	81,82%
4	**Naive-Bayes**	78,69%	13,11%	8,20%	69,23%	78,26%
5	**DecisionTree**	77,05%	13,11%	9,84%	69,23%	75,00%

Como o conjunto de dados possui poucas instâncias, a chance de que os classificadores dessem resultados próximos para o conjunto de teste era mais alta, o que pode ser visto nas tabelas acima (havia apenas 61 instâncias no conjunto de teste).

Assim, para escolher o melhor classificador dentre K-NN e MLP, foi considerado o fato de que, no caso do uso do classificador no mundo real, o número de instâncias disponível deve ser muito maior (ordem de milhares de instâncias), logo a utilização de uma MLP é uma melhor opção em relação ao K-NN, uma vez que redes neurais tendem a ter acurácias muito altas quando são alimentadas com dados suficientes, além disso, pode-se perceber que o K-NN sofreu com o *overfitting*, uma vez que se especializou muito no conjunto de treino e teve um desempenho muito inferior no conjunto de teste.

KNN performa melhor com um número de atributos menor. Quando a quantidade de atributos aumenta, o algoritmo necessita de mais dados para ter uma performance aceitável. Uma grande quantidade de atributos também pode levar ao problema do *overfitting* (como ocorreu com o conjunto de dados escolhido).

Para evitar o *overfitting*, a quantidade de dados deve crescer exponencialmente com o acréscimo do número de dimensões (atributos).

O classificador Bayesiano, como esperado, por supor independência entre todas as variáveis, não teve o melhor desempenho, uma vez que, como pôde ser visto na análise estatística dos dados, alguns dos atributos têm algum grau de correlação (mesmo que não muito significativo→|Corr(X,Y)|<0,5) com os outros.

A máquina de suporte de vetores é uma representação dos dados de treinamento como pontos no espaço separados em categorias por uma lacuna clara que é tão ampla quanto possível. O classificador SVM, por conta de se tratar de uma classificação não linear (a relação entre os 13 atributos é complexa, logo a região de decisão não deve ser linear), teve um bom desempenho se comparado aos outros classificadores (O truque do kernel pode ser usado para adaptar SVMs de modo que se tornem capazes de aprender limites não-lineares da decisão. Um kernel é uma medida da semelhança entre um ponto de dados e todos os outros representados no conjunto de dados.). Também é mais eficiente para a classificação em apenas 2 classes, uma vez que encontrar um hiperplano (ou qualquer outro limiar de decisão) entre apenas 2 regiões é menos complexo do que encontrar vários hiperplanos entre diversas regiões de classes distintas. Além disso, como a região de decisão é relativamente simples, este classificador tem menos problemas com *overfitting*.

O classificador DecisionTree possui regiões de decisão muito complexas, logo tem uma maior probabilidade de *overfitting (early stopping)* e é mais adequado para a classificação em um número maior de classes. Isso pode ser visto nos seus resultados, uma vez que obteve uma acurácia de 100% no conjunto de treino e apenas 77,05% no conjunto de teste (*overfitting* do conjunto de treino). A Árvore de Decisão é simples de entender e visualizar, requer pouca preparação de dados e pode lidar com dados numéricos e categóricos. Porém, pode criar árvores complexas que não generalizam bem, como ocorreu na análise do conjunto escolhido.

As redes neurais são flexíveis e podem ser usadas para problemas de regressão e classificação. Quaisquer dados que possam ser feitos numéricos podem ser usados no modelo, pois a rede neural é um modelo matemático com funções de

aproximação. Funciona dividindo o problema da classificação em uma rede em camadas de elementos mais simples. As redes neurais podem ser treinadas com qualquer número de entradas e camadas, logo era de se esperar que tivesse um bom resultado no conjunto de dados escolhido, como ocorreu. A MLP provavelmente se sairia muito melhor do que os outros classificadores caso o número de instâncias do conjunto de dados fosse grande (milhares de instâncias).

7. SUGESTÕES PARA ANÁLISE POSTERIOR

Para que os classificadores não sejam influenciados pela escolha dos valores mapeados a atributos categóricos, é preciso mapear os valores das classificações dos atributos categóricos a seus respectivos nomes (P.ex o atributo Cp (tipo de dor) deve ter os valores (1,2,3,4) mapeados em (angina_tipica,angina_atípica,não_anginal,assintomático)) e então utilizar a função get_dummies() da biblioteca pandas, a qual cria n colunas(onde n é o número de valores possíveis do atributo categórico) e atribui o valor 1 apenas para a coluna correspondente à classificação e 0 aos restantes, dessa forma, elimina o fato de uma classificação de um atributo categórico ser maior do que outra dividindo esse atributo em n atributos:

Ex:

Instância 1: CP = 4→(angina_tip = 0,angina_atip = 0,não_ang = 0,assint = 1)

Instância 2: CP = 3→(angina_tip = 0,angina_atip = 0,não_ang = 1,assint = 0)

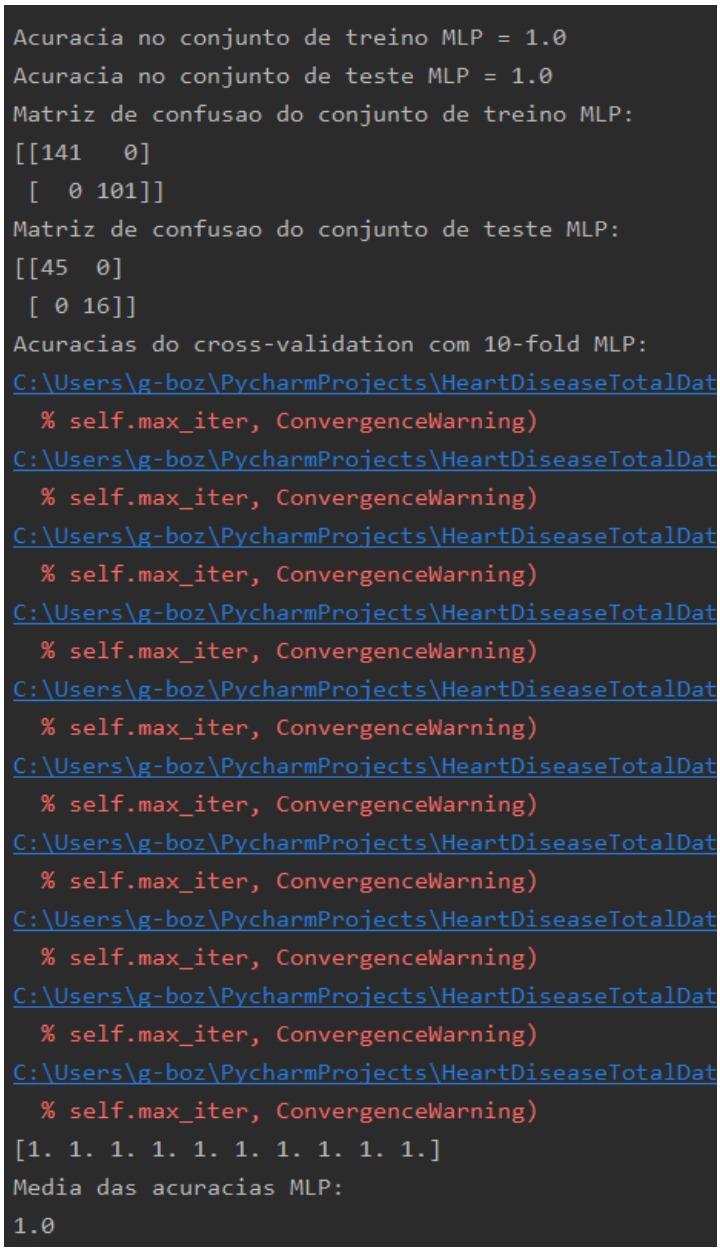
OBS: Eu tentei usar essa função no conjunto de dados, porém, mesmo após realizar o cross-validation as acurácias médias da maioria dos classificadores estavam iguais a 100%, o que eu não consegui explicar, logo supus que estava fazendo algo errado que não sabia detectar (pois não possuo experiência na área) e analisei os dados sem utilizar a função get_dummies()→Os códigos para encontrar os melhores parâmetros para os classificadores e para encontrar os desempenhos dos classificadores utilizando a função get_dummies() também foram enviados (têm o sufixo get_dummies).

```
Acuracia no conjunto de treino SVM = 1.0
Acuracia no conjunto de teste SVM = 1.0
Matriz de confusao do conjunto de treino SVM:
[[141   0]
 [  0 101]]
Matriz de confusao do conjunto de teste SVM:
[[45  0]
 [ 0 16]]
Acuracias do cross-validation com 10-fold SVM:
[1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Media das acuracias SVM:
1.0

Acuracia no conjunto de treino NB = 0.9958677685950413
Acuracia no conjunto de teste NB = 1.0
Matriz de confusao do conjunto de treino NB:
[[141   1]
 [  0 100]]
Matriz de confusao do conjunto de teste NB:
[[45  0]
 [ 0 16]]
Acuracias do cross-validation com 10-fold NB:
[1.         1.         1.         1.         1.         1.
 1.         1.         1.         0.96551724]
Media das acuracias NB:
0.9965517241379309
```

```
Acuracia no conjunto de treino KNN = 1.0
Acuracia no conjunto de teste KNN = 0.8032786885245902
Matriz de confusao do conjunto de treino KNN:
[[141   0]
 [  0 101]]
Matriz de confusao do conjunto de teste KNN:
[[36  3]
 [ 9 13]]
Acuracias do cross-validation com 10-fold KNN:
[0.77419355 0.87096774 0.90322581 0.93548387 0.96774194 0.67741935
 0.8        0.55172414 0.68965517 0.79310345]
Media das acuracias KNN:
0.7963515016685206

Acuracia no conjunto de treino DecisionTree = 1.0
Acuracia no conjunto de teste DecisionTree = 1.0
Matriz de confusao do conjunto de treino DecisionTree:
[[141   0]
 [  0 101]]
Matriz de confusao do conjunto de teste DecisionTree:
[[45  0]
 [ 0 16]]
Acuracias do cross-validation com 10-fold DecisionTree:
[1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Media das acuracias DecisionTree:
1.0
```



```
Acuracia no conjunto de treino MLP = 1.0
Acuracia no conjunto de teste MLP = 1.0
Matriz de confusao do conjunto de treino MLP:
[[141   0]
 [  0 101]]
Matriz de confusao do conjunto de teste MLP:
[[45  0]
 [ 0 16]]
Acuracias do cross-validation com 10-fold MLP:
C:\Users\g-boz\PycharmProjects\HeartDiseaseTotalDat
  % self.max_iter, ConvergenceWarning)
C:\Users\g-boz\PycharmProjects\HeartDiseaseTotalDat
  % self.max_iter, ConvergenceWarning)
C:\Users\g-boz\PycharmProjects\HeartDiseaseTotalDat
  % self.max_iter, ConvergenceWarning)
C:\Users\g-boz\PycharmProjects\HeartDiseaseTotalDat
  % self.max_iter, ConvergenceWarning)
C:\Users\g-boz\PycharmProjects\HeartDiseaseTotalDat
  % self.max_iter, ConvergenceWarning)
C:\Users\g-boz\PycharmProjects\HeartDiseaseTotalDat
  % self.max_iter, ConvergenceWarning)
C:\Users\g-boz\PycharmProjects\HeartDiseaseTotalDat
  % self.max_iter, ConvergenceWarning)
C:\Users\g-boz\PycharmProjects\HeartDiseaseTotalDat
  % self.max_iter, ConvergenceWarning)
C:\Users\g-boz\PycharmProjects\HeartDiseaseTotalDat
  % self.max_iter, ConvergenceWarning)
C:\Users\g-boz\PycharmProjects\HeartDiseaseTotalDat
  % self.max_iter, ConvergenceWarning)
[1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Media das acuracias MLP:
1.0
```

Também seria importante uma análise mais aprofundada dos parâmetros a serem testados nos classificadores, uma vez que, por conta do tempo reduzido, apenas alguns dos parâmetros dos classificadores foram testados (os testes demoram para calcular os melhores parâmetros).

Além disso, uma escolha fina dos atributos a serem utilizados, baseada nas relevâncias e correlações dos atributos faria com que os modelos a serem construídos a partir dos dados fossem mais simples, aumentando a robustez das previsões por não depender de combinações complexas de muitos atributos simultaneamente.

Vale ressaltar que a quantidade de informação necessária para a realização do trabalho era grande e foi difícil extrair qual é a maneira correta (se é que existe uma única maneira) de realizar todos os passos necessários da imensa quantidade de material disponível e das diferentes maneiras de analisar um conjunto de dados. Cada conjunto de dados possui suas especificidades e apenas com experiência na área uma análise mais correta e aprofundada pode ser realizada.

8.REFERÊNCIAS

1.Hungarian Institute of Cardiology. Budapest: Andras Janosi, M.D. 2.University Hospital, Zurich, Switzerland: William Steinbrunn, M.D. 3.University Hospital, Basel, Switzerland: Matthias Pfisterer, M.D. 4.V.A. Medical Center, Long Beach and Cleveland Clinic Foundation:Robert Detrano, M.D., Ph.D

5.Heart Disease Prediction Using Machine learning and Data Mining Technique Jaymin Patel, Prof.TejalUpadhyay, Dr. Samir Patel Department of Computer Science and Engineering, Nirma University, Gujarat, India.

6.https://towardsdatascience.com/introduction-to-data-preprocessing-in-machine-learning-a9fa83a5dc9d-Acessado em 11 de novembro de 2019.

7.https://scikit-learn.org/stable/supervised_learning.html#supervised-learning-Acessado em 14 de novembro de 2019.