SCC5848 - INTRODUÇÃO A CIÊNCIA DE DADOS
Classificação de Doenças Cardíacas

Grupo: 16
Alunos: João Pedro Buzzo Silva - 10425191
Victor Hugo Trigolo Amaral - 12688487

I - Introdução

As doenças cardiovasculares (DCV) são a principal causa de morte no mundo, representando cerca de **32% de todas as mortes globais** em 2019, segundo a Organização Mundial da Saúde (OMS) [2]. Estima-se que mais de **17,9 milhões de pessoas** morram anualmente devido a essas condições, com destaque para infarto do miocárdio, insuficiência cardíaca e outras complicações relacionadas ao coração [2]. Esses números são especialmente alarmantes em países de baixa e média renda, onde mais de **75% das mortes por doenças cardiovasculares** ocorrem [3], refletindo desigualdades no acesso a cuidados preventivos e ao tratamento adequado.

A base de dados *Heart Disease* da UCI Machine Learning Repository [1] foca em fatores que contribuem para doenças cardíacas, incluindo idade, gênero, pressão arterial, colesterol e outros marcadores clínicos. Essa base é amplamente utilizada em estudos de aprendizado de máquina para prever a presença ou ausência de doenças cardíacas em pacientes, contribuindo para avanços em diagnósticos mais precoces e precisos.

No Brasil, as doenças cardiovasculares também são um problema significativo de saúde pública, sendo a principal causa de mortalidade há décadas. Dados indicam que ocorrem cerca de **400 mil mortes** anualmente por DCV no país [3], destacando a importância de esforços preventivos e ferramentas preditivas eficazes. Assim, a análise de dados, aliada a técnicas modernas de ciência de dados e aprendizado de máquina, surge como uma abordagem promissora para apoiar médicos e pacientes no manejo e prevenção de doenças cardíacas.

O dataset coletado era inicialmente dividido em quatro partes, com dados semelhantes coletados de três diferentes países: **Cleveland (EUA)**, **VA Long Beach (EUA)**, **Suíça** e **Hungria**. Nesse sentido, os dados foram coletados e concatenados com o objetivo de obter-se um dataset único.

Visto isso, a exploração dos dados e modelagem foi pautada na comparação de desempenho de classificação dos dois datasets, verificando como a ausência/presença do atributo ‘ca’ influencia nas métricas, e também na comparação entre os diferentes modelos de classificação:

- K-Nearest Neighbors Classifier - KNN
- Decision Tree Classifier - DT
- Support Vector Machine Linear - SVM Linear
- Support Vector Machine Polynomial Kernel (Degree 3) - SVM Poly3

- Multi-Layered Perceptron (32, 16)

Além disso, falaremos mais adiante sobre:

- Alguns artigos com estudos já nessa área, relacionados ao dataset;
- Descrição dos materiais e métodos utilizados na exploração e classificação do conjunto de dados;
- Descrição dos experimentos realizados;
- Análise conclusiva sobre o conjunto de dados, a relevância do atributo **'ca'** e o(s) modelos mais apropriados para esse problema de classificação.

II - Trabalhos Relacionados

Alguns trabalhos de outros autores apresentam resultados interessantes para guiar as decisões tomadas no desenvolvimento deste projeto. Em especial, o conjunto de dados escolhido neste projeto foi utilizado para validar algoritmos de classificação mais complexos e próximos do estado da arte [5]. Além disso, há certo interesse em aferir técnicas mais simples e de mais fácil implementação ao trabalhar com o conjunto de dados *Heart Disease*, a fim de aprendizado e comprovação das técnicas. Portanto, quando comparado com trabalhos recentes, surge interesse em avaliar a qualidade do K-Nearest Neighbors [6] e do Support Vector Machine [7] na classificação do dataset escolhido.

III - Materiais e Métodos

1 - Conjunto de Dados

O conjunto de dados utilizado foi o *Heart Diseases*, fornecido pelo UCI Machine Learning Repository [1]. Ela originalmente conta com 76 atributos de informação sobre pacientes, coletadas de 4 hospitais diferentes, com 920 dados no total. No entanto, como informado no próprio site do UCI, o conjunto de dados foi significativamente reduzido a 14 atributos e é fornecido desta maneira.

O conjunto de dados, no entanto, originalmente não está totalmente padronizado e conta com informações faltantes, sendo representadas por "?", apesar de todas os atributos serem representados numericamente – incluindo as categóricas. A tabela [1] a seguir descreve os atributos.

Tabela [1]: Tipo e descrição dos atributos do conjunto de dados utilizado.

Nome do Atributo	Tipo de Atributo	Descrição
age	Inteiro	Idade do paciente.

sex	Categórico	Sexo do paciente.
cp	Categórico	Tipo de dor.
trestbps	Inteiro	Pressão sanguínea em repouso.
chol	Inteiro	Nível de colesterol sanguíneo.
fbs	Categórico	Indica se o nível de glicemia em jejum é maior do que 120 mg/dL.
restecg	Categórico	Resultado do eletrocardiograma em repouso.
thalach	Inteiro	Máxima frequência cardíaca registrada durante teste de esforço físico.
exang	Categórico	Indica se houve ou não dor induzida pelo esforço físico.
oldpeak	Inteiro	Valor de depressão do segmento ST durante o esforço em relação ao repouso.
slope	Categórico	Inclinação do segmento ST durante o pico do exercício físico.
ca	Inteiro	Número de vasos visualizados por fluoroscopia.
thal	Categórico	Tipo de talassemia.
target	Alvo	Intensidade da doença cardíaca (medido de 0 a 4, sendo 0 a ausência de doença cardíaca).

2 - Exploração e Pré-processamento

Ao avaliar os dados, é possível ver uma predominância de valores faltantes para alguns dos atributos. Em especial, o atributo ‘*ca’* pode indicar a gravidade de uma doença cardíaca e é relativo a um exame que não é mandatório para todo paciente, o que justifica a ausência de dados. Com base nisso, surge o interesse de verificar se a remoção desse atributo prejudica no funcionamento dos modelos de classificação, visto que o conjunto de dados reduz-se a 299 dados, possivelmente perdendo informação. Sendo assim, trabalhar-se-á com dois conjuntos de dados: o primeiro (denominado conjunto 1) contendo todos os atributos, removendo valores nulos que venham a prejudicar os modelos; o segundo (denominado conjunto 2) mantendo todas as linhas, mas removendo o atributo *‘ca’* do conjunto.

As etapas de pré-processamento começaram com a verificação da presença de outliers nos dados dado que estes podem prejudicar o desempenho do modelo. Apesar destes outliers poderem representar a diversidade de organismos e como estes se comportam em cada exame ou medição,também podem indicar falhas de medição ou registro. Em vista da falta de informação a respeito, optou-se por tratar os outliers utilizando cercas baseadas nos quartis de cada atributo.

Também foi feita a transformação de valores nulos no conjunto de dados 2, transformando valores numéricos pela média e valores categóricos pelo mais frequente. Além disso, foi feita a normalização dos dados com base na escala padrão para uma melhor distribuição e visualização de dados.

Além disso, foi feita a remoção de toda linha do conjunto de dados 1 que contivesse algum valor nulo, para uma representação mais adequada de valores reais.

Por fim, foi necessário tratar o atributo alvo, a fim de que ele fosse representado como binário, dado que existem cinco classes no conjunto original. A regra de transformação seguiu como: se a classe original é 0, então o paciente não possui doença cardíaca, sendo inalterado; caso contrário, o paciente possui uma doença cuja gravidade é medida entre 1 e 4, sendo transformada para 1, indicando apenas a presença de uma doença.

3 - Extração de Características

Como previamente citado, o conjunto de dados original continha 76 atributos, que foram reduzidos a 14 antes de serem fornecidos. Portanto, num primeiro momento não há necessidade de remover algum atributo do conjunto.

Contudo, para fins de verificação, fez-se a matriz de correlação dos dados após o pré-processamento, o que indicou que, de fato, em ambos os conjuntos não era necessário extrair características.

4 - Modelos de Classificação

Na seção II, foram abordados dois trabalhos que utilizaram algoritmos simples de classificação [6] [7]. O intuito deste projeto foi comparar estes algoritmos e, para fins de melhor ilustração da comparação, foram adicionados dois outros algoritmos:

- K-Nearest Neighbors [6] comparando 3 vizinhos devido ao pequeno volume de dados;
- Decision Tree com no mínimo 10% de separação de amostras;
- Support Vector Machine linear [7];
- Support Vector Machine polinomial de grau 3;
- Multi-Layered Perceptron (32, 16).

Para treinar os modelos, foi utilizado a técnica de validação cruzada K-Fold com k=10, fazendo o balanceamento do conjunto de treino utilizando o SMOTE e foram extraídas quatro métricas: acurácia, precisão, sensibilidade e f1-score. Para avaliar o desempenho dos classificadores foram analisadas as métricas extraídas, bem como a curva ROC gerada.

5 - Implementação

A implementação utilizou duas bibliotecas:

- Imbalanced Learn:

    - SMOTE.
- Scikit-Learn:
    - KNeighborsClassifier;
    - DecisionTreeClassifier;
    - SVC;
    - MLPClassifier;
    - KFold;
    - roc_curve;
    - auc.

Além disso utilizou-se de bibliotecas para visualização e processamentos dos dados, sendo elas: pandas, numpy, matplotlib.pyplot e seaborn.

IV - Experimentos

1 - Métricas e Desempenhos

Os experimentos realizados foram baseados na execução da classificação binária nos nossos dois conjuntos de dados (dataset sem o atributo ‘ca’, e dataset sem objetos com valores nulos). Os modelos utilizados foram os modelos citados na seção III item 4 e, para cada modelo, foram coletadas as quatro métricas principais, derivadas da matriz de confusão:

- Acurácia
- Sensibilidade (recall)
- Precisão
- F1-score

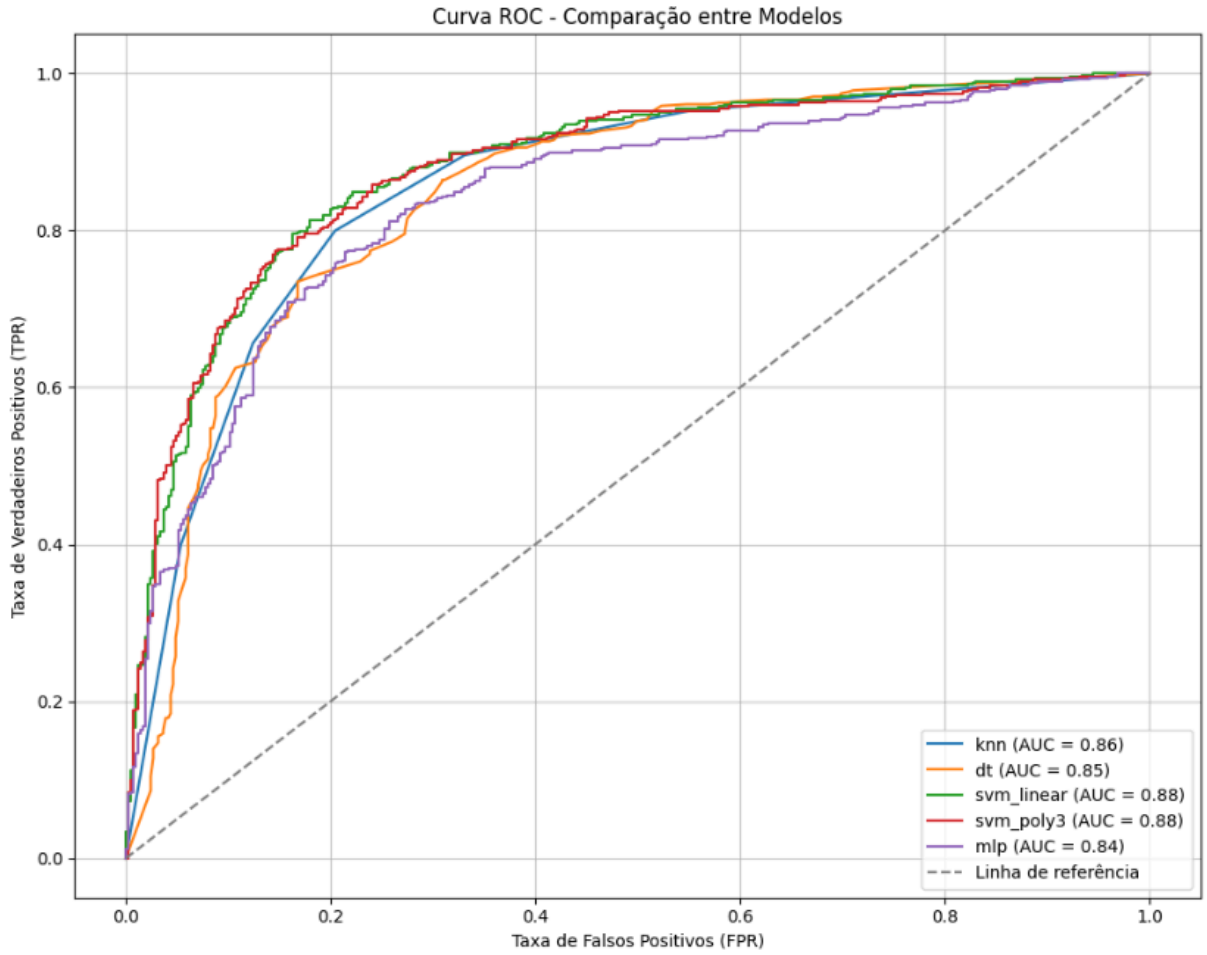
a) Conjunto de dados sem o atributo ‘ca’:

A seguir, temos uma tabela ilustrando os valores médios (em %) de cada métrica para cada modelo com o qual foram realizadas as predições sobre o **conjunto de dados sem o atributo ‘ca’**:

Modelo do Classificador	Acurácia (|%)	Sensibilidade (%)	Precisão (%)	F1-score (%)
K-Nearest Neighbors	79,78 (+- 3,61)	79,66 (+- 8,18)	82,73 (+- 3,91)	80,97 (+- 5,41)
Decision Tree	76,85 (+- 4,18)	77,48 (+- 6,24)	80,41 (+- 7,32)	78,58 (+- 4,51)
SVM linear	81,20 (+- 4,29)	81,30 (+- 8,21)	83,71 (+- 3,52)	82,37 (+- 5,72)

SVM grau 3	80,87 (+- 3,96)	78,92 (+- 8,47)	85,00 (+- 2,82)	81,65 (+- 5,67)
MLP	76,96 (+- 5,62)	77,90 (+- 9,33)	79,37 (+- 6,12)	78,44 (+- 6,99)

Além disso, calculou-se a Curva ROC para os quatro modelos de classificação no **conjunto de dados sem o atributo 'ca'**, que pode ser vista abaixo:



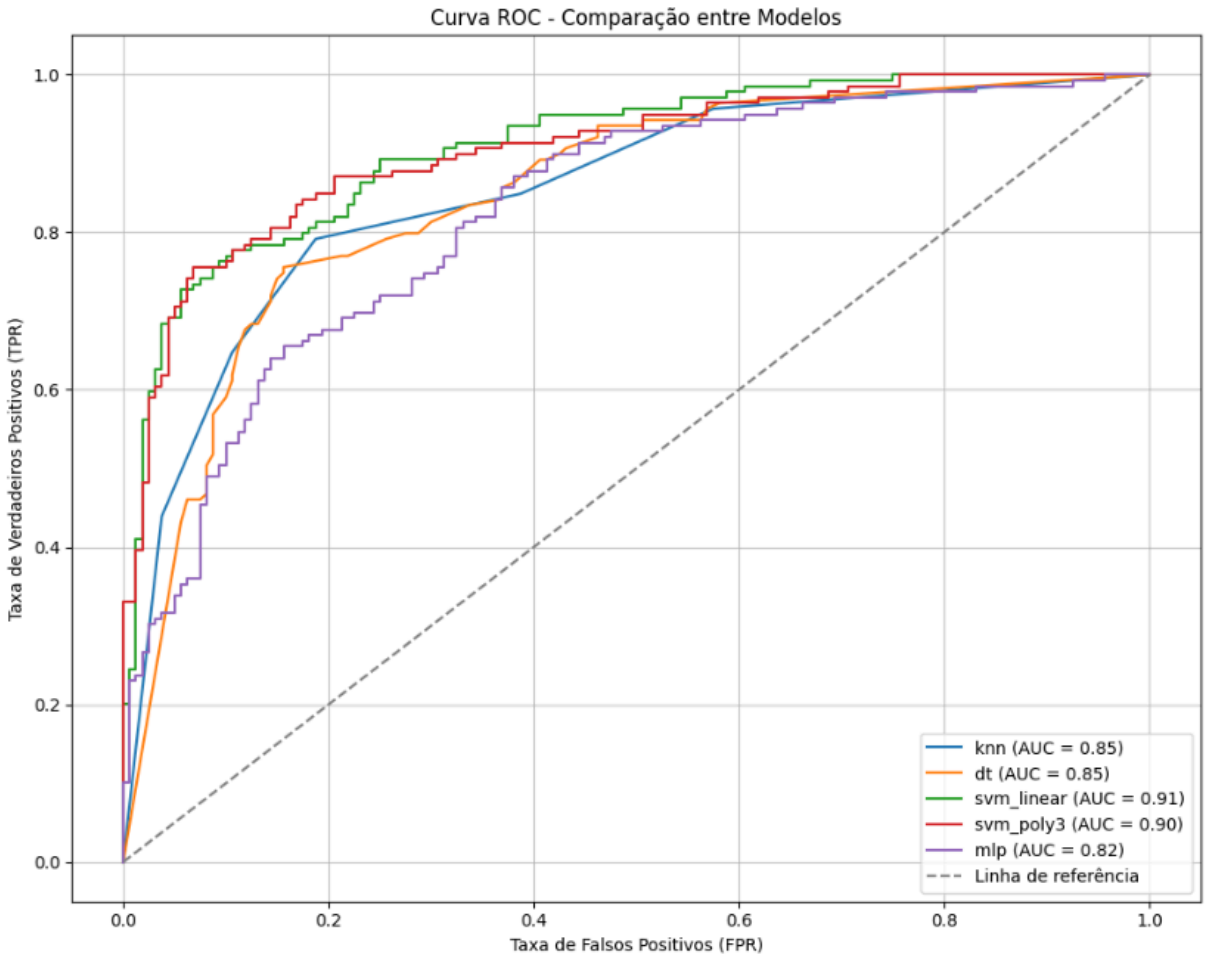
b) Conjunto de dados sem os objetos com valores nulos:

A seguir, temos uma tabela ilustrando os valores médios (em %) de cada métrica para cada modelo com o qual foram realizadas as predições sobre o **conjunto de dados sem os objetos com valores nulos**:

Modelo do Classificador	Acurácia (%)	Sensibilidade (%)	Precisão (%)	F1-score (%)
K-Nearest Neighbors	80,25 (+- 8,66)	79,53 (+- 9,44)	78,35 (+- 10,28)	78,74 (+- 9,25)
Decision Tree	80,28 (+- 5,44)	75,62 (+- 9,48)	80,25 (+- 8,23)	77,44 (+- 7,02)

SVM linear	82,61 (+- 4,89)	79,29 (+- 7,71)	83,26 (+- 7,48)	80,68 (+- 4,51)
SVM grau 3	83,62 (+- 6,21)	76,47 (+- 10,57)	87,09 (+- 8,43)	80,81 (+- 7,47)
MLP	73,60 (+- 8,04)	72,55 (+- 21,02)	74,49 (+- 12,44)	69,55 (+- 15,71)

Além disso, calculou-se a Curva ROC para os quatro modelos de classificação no **conjunto de dados sem os objetos com valores nulos**, que pode ser vista abaixo:



Observação: devido ao não determinismo dos modelos, os valores podem sofrer pequenas variações, em decorrência de uma nova execução.

2 - Resultados

a) Conjunto de dados sem o atributo 'ca':

De acordo com as tabelas, o modelos que apresentaram maiores valores médios de cada métrica foram:

- Acurácia: SVM polinomial de grau 3 (81,83%)
- Sensibilidade: K-Nearest Neighbors (83,63%)
- Precisão: SVM linear (82,39%)

- F1-score: SVM linear (81,45%)

Analisando as curvas ROC dos modelos para esse dataset, pode-se verificar que tanto o SVM linear quanto o SVM polinomial de grau 3 têm valores semelhantes para a área sob a curva (0,88), o que os deixam razoavelmente equilibrados em desempenho de classificação para esse conjunto de dados, sem o atributo ‘ca’.

b) Conjunto de dados sem os objetos com valores nulos:

De acordo com as tabelas, o modelos que apresentaram maiores valores médios de cada métrica foram:

- Acurácia: SVM polinomial de grau 3 (86,25%)
- Sensibilidade: SVM polinomial de grau 3 (91,55%)
- Precisão: SVM linear (82,87%)
- F1-score: SVM polinomial de grau 3 (85,63%)

Analisando as curvas ROC dos modelos para esse dataset, também pode-se verificar que tanto o SVM linear quanto o SVM polinomial de grau 3 têm valores semelhantes para a área sob a curva (0,91), o que os deixam razoavelmente equilibrados em desempenho de classificação para esse conjunto de dados, sem os objetos com valores nulos.

V - Conclusão

Se baseando nos resultados do projeto, pode-se concluir que os modelos SVM linear e SVM polinomial de grau 3 são melhores opções para classificação de diagnósticos de doenças cardíacas. Há de se notar ainda que, uma vez que os dados refletem informações da saúde de um paciente, quanto mais puder-se reduzir a quantidade de falsos negativos, menor a possibilidade de se fornecer um resultado que comprometa o bem-estar do paciente. Sobretudo, é necessário dizer que, apesar do desempenho dos modelos, todo resultado está sujeito a análise de um profissional especializado.

VI - Referências

[1] https://archive.ics.uci.edu/dataset/45/heart+disease

[2] https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)

[3] https://ncdalliance.org/why-ncds/ncds/cardiovascular-diseases

[4]https://www.acritica.com/saude/doencas-cardiovasculares-matam-400-mil-brasileiros-por-ano-1.352544

[5] Dhurandhar, Amit et al. “Leveraging Simple Model Predictions for Enhancing its Performance.” *ArXiv* abs/1905.13565 (2019): n. pag.

[6] Uddin, S., Haque, I., Lu, H. *et al.* Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction. *Sci Rep* 12, 6256 (2022). https://doi.org/10.1038/s41598-022-10358-x

[7] Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., & Lopez, A. (2020). *A comprehensive survey on support vector machine classification: applications, challenges and trends. Neurocomputing.* doi:10.1016/j.neucom.2019.10.118