CLASSIFICAÇÃO DE CÂNCER DE PÂNCREAS UTILIZANDO TÉCNICAS DE IMPUTAÇÃO DE DADOS FALTANTES E *UNDERSAMPLING* BASEADO EM CLUSTERIZAÇÃO: uma análise comparativa com diferentes algoritmos de *Machine Learning*

PANCREATIC CANCER CLASSIFICATION USING MISSING DATA IMPUTATION AND CLUSTER-BASED UNDERSAMPLING METHODS: a comparative analysis with multiple Machine Learning algorithms

Wanessa Layssa Batista de Sena
wlbs@a.recife.ifpe.edu.br
Renata Freire de Paiva Neves
renatafreire@recife.ifpe.edu.br

RESUMO

Dados faltantes e desbalanceamento de classes são problemas frequentemente observados em bases de dados associadas a cenários reais, o que inclui a classificação de câncer. Caso estes problemas não sejam endereçados de forma adequada antes da análise, impactos no desempenho de modelos de *Machine Learning* (ML) podem ser observados. Neste artigo, é proposta uma solução combinada a partir da inserção de dados faltantes utilizando a técnica de kNN (*k* vizinhos mais próximos) e *undersampling* baseado em clusterização utilizando *k-means*, com foco na classificação do câncer de pâncreas. Diferentes subconjuntos de dados foram gerados a partir da combinação de diferentes métodos de pré-processamento e o desempenho analisado utilizando um *pipeline* de análise de ML de um estudo prévio. Este pipeline executa dez algoritmos de ML, incluindo *Random Forest*, Máquina de Vetores de Suporte e Redes Neurais Artificiais. Todos os subconjuntos de dados gerados apresentaram um aumento significativo ($p<0,05$ com teste-t de *Student*) no desempenho para a maioria dos algoritmos de ML quando comparados aos resultados obtidos anteriormente quando o *pipeline* foi avaliado pela primeira vez. Os resultados sugerem que kNN e *k-means* são métodos que podem ser utilizados na fase de pré-processamento dos dados para solucionar problemas de dados faltantes e desbalanceamento de classes e melhorar a acurácia da classificação.

Instituto Federal de Pernambuco. Campus Recife. Curso de Tecnologia em Análise e Desenvolvimento de Sistemas. 19 de Junho de 2023.

Palavras-chave: *Machine Learning*; Clusterização *k-means*; kNN; *Undersampling*; Imputação de dados faltantes; Classificação.

ABSTRACT

Missing values and class imbalance are issues frequently found in databases from real-world scenarios, including cancer classification. Impacts on the performance of Machine Learning (ML) models can be observed if these issues are not properly addressed prior to the analysis. In this paper, a combined solution with missing data imputation using kNN (k-nearest neighbors) and cluster-based undersampling using k-means is proposed, focusing on pancreatic cancer classification. Different data subsets were generated by combining different preprocessing methods and the performance was analyzed using a ML analysis pipeline from a previous study. This pipeline implements ten ML classifiers, including Random Forest, Support Vector Machine and Artificial Neural Network. All data subsets presented a significant improvement ($p<0.05$ with Student's T-Test) in the performance of most ML algorithms when compared with the results obtained when the pipeline was first evaluated. Results suggest that kNN and k-means can be used in the data preprocessing phase to overcome missing values and class imbalance issues and improve the classification accuracy.

Keywords: Machine Learning; K-means clustering; kNN; Undersampling; Missing data imputation; Classification.

1 INTRODUÇÃO

O câncer de pâncreas é a sétima causa de morte por câncer em ambos os gêneros, apresentando taxas de incidência e mortalidade semelhantes – 495.773 novos casos e 466.003 mortes registradas em 2020, de acordo com Sung et al. (2021). Projeções indicam que o câncer de pâncreas se tornará a terceira causa de morte por câncer até 2025, ultrapassando o câncer de mama (SUNG et al., 2021). Além das características biológicas inerentes ao câncer de pâncreas, que conferem um comportamento agressivo e um alto potencial metastático a esses tumores, o diagnóstico continua a ser um desafio devido à ausência de métodos sensíveis para a detecção precoce da doença (WINTER et al., 2019).

O uso de Inteligência Artificial (IA) na área da saúde vem crescendo significativamente nos últimos anos, impulsionada principalmente pelo progresso de técnicas analíticas e aumento da disponibilidade de dados associados à assistência médica. Métodos de IA, incluindo *Machine Learning* (ML), podem fornecer informações relevantes a partir de dados de pacientes, ajudando na tomada de melhores decisões clínicas (JIANG et al., 2017).

Entretanto, alguns desafios estão associados ao uso de dados reais de câncer em modelos de ML. Como o número de indivíduos saudáveis é normalmente muito maior do que o número de pacientes com câncer, essa discrepância gera um problema conhecido em ML como desbalanceamento de classe. No desbalanceamento de classe, uma classe é representada por um grande número de

amostras, enquanto a outra classe (normalmente a classe de interesse) é representada por apenas poucas amostras (WEI-CHAO et al., 2017). Essa disparidade entre as duas classes leva a um viés em favor da classe majoritária, causando impactos no desempenho da classificação da classe minoritária (GUZMÁN-PONCE et al., 2020). Além disso, a qualidade dos dados é uma das principais preocupações ao trabalhar com registros de câncer. Como os dados geralmente dependem dos registros médicos dos pacientes, a ocorrência de valores desconhecidos ou ausentes é frequente (YANG et al., 2021). Portanto, superar alguns destes desafios se torna essencial para construir modelos de ML confiáveis que representem a diversidade e a complexidade de dados reais.

O presente trabalho tem como objetivo avaliar diferentes métodos para solucionar os problemas de desbalanceamento de classes e dados faltantes em um conjunto de dados de câncer de pâncreas, e seus efeitos na classificação de diferentes algoritmos de ML. As seções subsequentes deste artigo estão organizado da seguinte forma: seção 2 apresenta uma visão geral dos problemas de desbalanceamento de classes e dados faltantes, além de um resumo de alguns algoritmos de ML; seção 3 apresenta a metodologia de pesquisa; seção 4 descreve os resultados, e seção 5 destaca as conclusões com base nos resultados obtidos.

2 DESENVOLVIMENTO

2.1 O problema de desbalanceamento de classes

O desbalanceamento de classes é um desafio comum a muitas áreas de aplicação reais, incluindo a área de saúde (CHEN; LIU; PENG, 2019; ZHANG; CHEN; ABID, 2019). Um caso prático de desbalanceamento de classes é a classificação de câncer, uma vez que o número de casos negativos (classe majoritária) é muito superior aos casos positivos (classe minoritária). A diferença se torna especialmente evidente ao lidar com tipos de câncer menos frequentes, como é o caso do câncer de pâncreas, com uma taxa de incidência de 2,6% (SUNG et al., 2021). Em ML, o déficit no tamanho da classe minoritária pode limitar o modelo e levar a erros de classificação (VUTTIPITTAYAMONGKOL; ELYAN; PETROVSKI, 2021).

Diferentes estratégias vêm sendo aplicadas para resolver o problema de desbalanceamento de classes. Em especial, abordagens a nível dos dados, que consistem na aplicação de métodos de pré-processamento com o objetivo de reduzir a taxa de desbalanceamento, estão entre as estratégias mais comuns (WEI-CHAO et al., 2017). Este processo pode ser realizado a partir da diminuição do número de instâncias da classe majoritária (*undersampling*) ou o aumento do número de instâncias da classe minoritária (*oversampling*) (GUZMÁN-PONCE et al., 2020). O método de *undersampling* é reportado na literatura como uma melhor alternativa em relação ao *oversampling*, uma vez que o *oversampling* pode aumentar a possibilidade de *overfitting*. Porém, a depender do método utilizado, o *undersampling* também pode ocasionar o *underfitting* (ZHANG; CHEN; ABID, 2019).

Para ultrapassar as limitações do *undersampling*, métodos baseados em clusterização vêm sendo propostos para garantir que dados relevantes não são removidos da classe majoritária (GUZMÁN-PONCE et al., 2020; WEI-CHAO et al., 2017; ZHANG; CHEN; ABID, 2019). Algoritmos de clusterização exploram a

estrutura de distribuição dos dados e definem regras de agrupamento para dados com características similares (AHMED; SERAJ; ISLAM, 2020). Dentre esses algoritmos, o algoritmo de *k-means* é um dos mais utilizados para análise de dados. *K-means* é um método de aprendizagem não-supervisionada que, com base em um dado valor *k*, executa as seguintes etapas: 1) divisão dos dados em *k* clusters; 2) cálculo do centroide para cada *cluster*; e 3) redistribuição dos dados com base no centroide mais próximo (HASSAN et al., 2021). A aplicação do algoritmo *k-means* vem sendo reportada em diferentes contextos (BAI; LIANG; GUO, 2018; QIN et al., 2017). No contexto de *undersampling*, estudos mostram o uso de *k-means* como estratégia única ou combinada com outras técnicas baseadas em clusterização (WEI-CHAO et al., 2017; ZHANG; CHEN; ABID, 2019).

2.2 O problema de dados faltantes

Conjuntos de dados com valores faltantes são comuns e podem causar um impacto significativo na análise de dados. Valores ausentes podem estar presente no conjunto de dados devido a uma série de fatores como, por exemplo, questões não respondidas em um questionário, perda de dados por fatores imprevisíveis, ou altos custos associados à obtenção dos dados (WU et al., 2019). Os problemas causados por dados faltantes durante a análise de ML incluem: aumento no tempo de processamento, complicações durante o processamento e análise dos dados, e dados enviesados (KAISER, 2014). Endereçar esse problema corretamente se torna crucial ao lidar com dados faltantes na classe minoritária, de forma a assegurar que os dados são representativos da diversidade observada no mundo real (CHEN; LIU; PENG, 2019).

Diferentes abordagens para lidar com dados faltantes podem ser aplicadas, incluindo métodos de ML como o kNN. kNN (*k* vizinhos mais próximos) é um método não-paramétrico amplamente utilizado para classificação e regressão. Neste método, *k* vizinhos mais próximos são identificados com base nos dados de treinamento e usados como referência para predição dos dados de teste (PANDEY; JAIN, 2017).

2.3 Algoritmos de ML

Esta subseção traz uma breve descrição dos algoritmos de ML que serão utilizados nas seções subsequentes do presente estudo.

Regressão Logística – Regressão Logística é um modelo estatístico utilizado para problemas de classificação binária. Com base em uma série de entradas, este método utiliza uma função logística (sigmoide) para modelar a saída, retornando valores entre 0 e 1 (SUBASI, 2020).

Árvore de Decisão – Árvore de decisão é um método não-paramétrico de aprendizagem supervisionada aplicado a diferentes áreas, como ML, processamento de imagens, e identificação de padrões (CHARBUTY; ABDULAZEEZ, 2021). Uma árvore de decisão apresenta uma estrutura hierárquica composta de nós e ramos. O processo se inicia em um nó raiz e, em seguida, uma estratégia de dividir-para-conquistar é conduzida para identificar os pontos de divisão ótimos e gerar nós de decisão. O processo é repetido recursivamente até que os nós folhas sejam

gerados, representando todos os possíveis resultados em um determinado conjunto de dados (WEI, 2021).

Random Forest (RF) – RF é um algoritmo de aprendizagem supervisionada amplamente aplicado em problemas de classificação e regressão. De forma similar à árvore de decisão, RF também é um algoritmo baseado em árvore, que recursivamente divide um determinado conjunto de dados em dois grupos até que uma determinada condição de parada seja atingida. Entretanto, ao invés de gerar uma única árvore, o algoritmo calcula a média de previsões com base em várias árvores individuais (SCHONLAU; ZOU, 2020).

Naive Bayes (NB) – NB é um classificador probabilístico baseado no teorema de Bayes. Este método vem sendo aplicado em diversos cenários reais e é amplamente utilizado especialmente por sua simplicidade, acurácia e bom desempenho quando comparado a outros métodos (WICKRAMASINGHE; KALUTARAGE, 2020).

Extreme Gradient Boosting (XGB) – XGB é um algoritmo de ML que usa árvores de decisão *gradient-boosting* para realizar predições. Este modelo foi proposto inicialmente por Chen e Guestrin (2016) e pode ser aplicado tanto em problemas de classificação quanto de regressão. XGB se beneficia de *multithreading* da CPU para computação paralela, o que acelera a sua execução (WEI et al., 2019).

Light Gradient Boosting Machine (LGB) – LBG (também conhecido como LightGBM) é um algoritmo de árvores de decisão *gradient-boosting*, de forma similar ao XGB. LGB é um método com alta precisão e desempenho, aplicado em problemas de *ranking* e classificação (GHORI et al., 2020).

Máquina de Vetores de Suporte (SVM) – SVM é um método popular de aprendizagem supervisionada, utilizado para classificação, regressão e detecção de *outliers*. O principal objetivo do método SVM é encontrar o hiperplano em um espaço n-dimensional que possibilite a segregação de dados em diferentes classes (KURANI et al., 2023).

Redes Neurais Artificiais (ANN) – ANN é um modelo computacional inspirado no cérebro humano. Este método é composto por diversos elementos de processamento, denominados neurônios artificiais (KURANI et al., 2023). Devido à sua capacidade de resolver problemas complexos, ou seja, de grande escala, o uso de ANN vem sendo reportado em diversos estudos relacionados à prática clínica do câncer (DHEEBA; ALBERT SINGH; TAMIL SELVI, 2014; ESTEVA et al., 2017).

3 METODOLOGIA

O objetivo do presente estudo consiste em compreender como os problemas de desbalanceamento de classes e dados faltantes em grandes conjuntos de dados podem ser solucionados, a fim de obter amostras significativas e aumentar a acurácia da classificação. Para realizar esta análise, os métodos estabelecidos por Urbanowicz et al. (2020) foram utilizados, alterando-se apenas a etapa de pré-processamento dos dados. Manter exatamente os mesmos métodos para as outras etapas nos permitiu comparar o desempenho dos modelos de ML em ambos os cenários.

3.1 Conjunto de dados

PLCO *(Prostate, Lung, Colorectal and Ovarian) Cancer Screening Trial* consistiu em um estudo randomizado e controlado, realizado nos Estados Unidos pelo *National Cancer Institute* no período de novembro de 1993 a julho de 2001. Este tinha por objetivo principal determinar a eficácia de procedimentos para a detecção precoce de cânceres de próstata, pulmão, colorretal e ovário, e incluiu cerca de 153 mil homens e mulheres. Apesar de focar nos 4 tipos mencionados anteriormente, o estudo também coletou informações de pacientes com diversos outros tipos de cânceres, incluindo câncer de pâncreas. Questionários foram administrados com o objetivo de coletar dados com base no histórico de saúde, dados demográficos e outros aspectos do estilo de vida dos pacientes (PROROK et al., 2000).

Com base nos dados disponíveis, duas populações foram consideradas no presente estudo: uma população de pessoas que tiveram o diagnóstico de câncer de pâncreas confirmado durante o estudo (n=807) e a segunda de controles saudáveis (n=100.819).

3.2 Seleção de atributos

Tendo como referência o estudo previamente realizado por Urbanowicz et al. (2020), 19 atributos relacionados ao histórico de saúde e hábitos dos pacientes foram selecionados para analisar o conjunto de dados, listados a seguir: **panc_cancer** (caso primário de câncer de pâncreas diagnosticado durante o estudo), **cig_stat** (status atual de consumo de cigarro), **cig_stop** (número de anos desde que o participante parou de fumar), **cig_years** (número de anos de consumo de cigarro), **pack_years** (número total de maços fumados durante o período de consumo de cigarro), **bmi_curr** (valor atual de Índice de Massa Corporal – IMC), **bmi_curc** (valor atual de IMC, de acordo com a categorização da Organização Mundial de Saúde), **diabetes_f** (caso confirmado de diabetes), **panc_fh** (casos confirmados de câncer de pâncreas na família em parentes de primeiro grau), **fh_cancer** (casos confirmados de câncer na família em parentes de primeiro grau), **bmi_20** (IMC aos 20 anos), **bmi_50** (IMC aos 50 anos), **asp** (uso regular de aspirina nos últimos 12 meses), **ibup** (uso regular de ibuprofeno nos últimos 12 meses), **gallblad_f** (já apresentou cálculo biliar ou inflamação na vesícula biliar), **age** (idade ao ingressar no estudo), **race7** (raça/etnia), **marital** (estado civil atual) e **sex** (sexo do paciente).

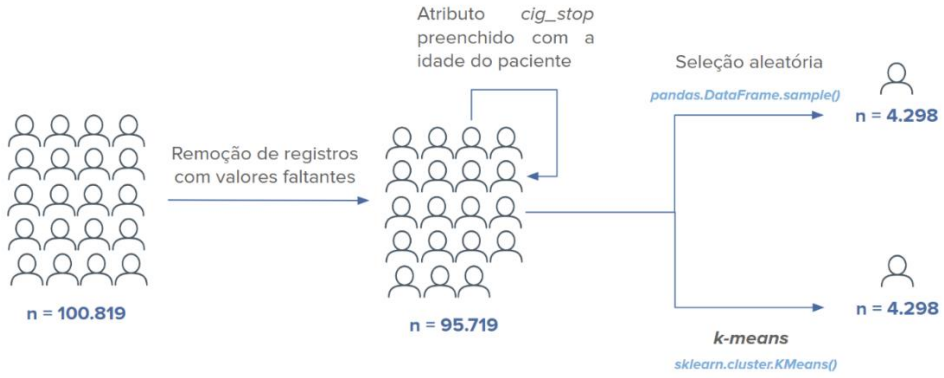
3.3 Pré-processamento dos dados

O algoritmo para pré-processamento dos dados foi implementado utilizando a linguagem de programação *Python*, versão 3.10, a partir de métodos das bibliotecas *NumPy* (versão 1.21.5), *pandas* (versão 1.4.4) e *scikit-learn* (versão 1.0.2).

Devido ao desbalanceamento presente no conjunto de dados PLCO, métodos de *undersampling* foram utilizados para rebalancear a distribuição das classes e igualar o tamanho da população de controles saudáveis com o utilizado por Urbanowicz e colaboradores (2020). A primeira etapa do processo consistiu em remover a maior parte dos registros com valores faltantes, reduzindo a população em 5% (de n=100.819 para n=95.719). As amostras relacionadas a pacientes não-fumantes com o atributo **cig_stop** sem valor associado foram as exceções, uma vez que o

número de anos desde que o paciente parou de fumar não é aplicável neste caso; estas amostras foram mantidas e o atributo preenchido com a idade do paciente. Depois desta etapa, dois métodos foram utilizados para selecionar controles saudáveis: seleção aleatória e clusterização com *k-means*. No presente estudo, diversos *clusters* foram gerados usando *k-means* (*k*=4.298) e a amostra mais próxima do centroide foi selecionada para representar cada grupo. A Figura 1 ilustra as etapas de pré-processamento de dados aplicadas à população de controles saudáveis.

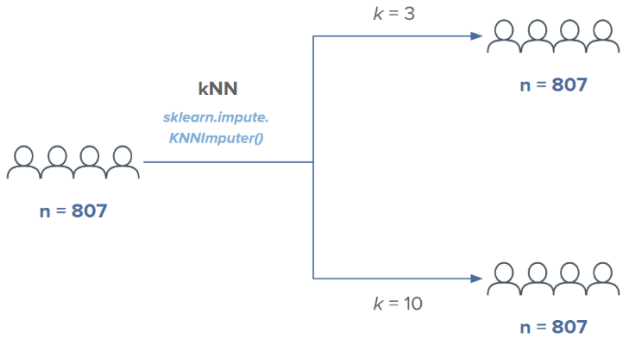
Figura 1 – Estratégia de pré-processamento de dados aplicada à população de pacientes saudáveis, utilizando diferentes métodos de *undersampling*



Fonte: autoria própria

Uma abordagem diferente foi utilizada durante o pré-processamento de dados da população com câncer de pâncreas. Os 807 registros foram mantidos e o algoritmo de kNN foi utilizado para imputação de valores faltantes, a partir da média dos vizinhos mais próximos no subconjunto de treinamento. Diferentes subconjuntos de dados foram gerados alterando-se o número de vizinhos (*k*=3 e *k*=10) utilizado para a imputação de dados, com o objetivo de validar os impactos desta variação na análise final. A estratégia utilizada para o pré-processamento de dados da população de pacientes com câncer de pâncreas está sumarizada na Figura 2.

Figura 2 – Estratégia de pré-processamento de dados aplicada à população de pacientes com câncer de pâncreas, a partir do uso do método de kNN para imputação de dados faltantes



Fonte: autoria própria

Após a combinação das diferentes técnicas descritas acima, 4 subconjuntos de dados foram gerados e usados nas análises subsequentes. Os detalhes são apresentados na Tabela 1.

Tabela 1 – Subconjunto de dados gerados com base em diferentes métodos de pré-processamento

Subconjunto de dados	**População de controles saudáveis**		**População de pacientes com câncer de pâncreas**	
	Número de amostras	**Método de *undersampling***	**Número de amostras**	**Método de imputação de valores faltantes**
S1	4.298	Seleção aleatória	807	kNN, *k*=3
S2	4.298	*k-means*	807	kNN, *k*=3
S3	4.298	Seleção aleatória	807	kNN, *k*=10
S4	4.298	*k-means*	807	kNN, *k*=10

Fonte: autoria própria

3.4 Análise dos dados

Com o intuito de mensurar o impacto das técnicas de pré-processamento descritas na subseção 3.3 no resultado final, as etapas subsequentes foram realizadas utilizando o mesmo *pipeline* de análise de ML de Urbanowicz et al. (2020). Este *pipeline*, que faz uso de diferentes bibliotecas *Python* (*scikit-rebate*, *xgboost* e *lightgbm*, por exemplo) e pode ser executado através de uma aplicação *Jupyter Notebook*, está disponível publicamente em Urbanowicz (2020) e pode ser executado em novos conjuntos de dados a partir de pequenas modificações.

Este *pipeline* é composto de 4 etapas principais:

- Pré-processamento e transformação de atributos;
- Importância e seleção de atributos;
- Modelagem de ML;
- Pós-análise.

Durante a etapa de pré-processamento e transformação de atributos é realizada uma análise exploratória dos dados com o objetivo de entender certas características, incluindo: tipos e correlação de atributos, dimensão dos dados e desbalanceamento de classes. Em seguida, uma limpeza básica dos dados é executada, seguida pela partição de subconjuntos utilizando validação cruzada *K-fold*. Neste procedimento, um conjunto de treinamento é dividido em *K* subconjuntos menores e, enquanto (*K* - 1) subconjuntos são utilizados para construção de um modelo, o subconjunto restante é utilizado para validação. O processo se repete por *K* vezes até que todos os subconjuntos sejam utilizados para validação (JUNG, 2017). No presente estudo foi utilizada validação cruzada *K-fold* com *K*=10.

A próxima fase é a importância e seleção de atributos, que avalia a importância dos atributos antes da execução do algoritmo de ML e remove atributos irrelevantes para cada *k* subconjunto de treinamento, se necessário. Esta última etapa só é importante ao analisar conjuntos de dados com um número grande (>50) de

atributos (URBANOWICZ et al., 2020), portanto não é relevante no presente trabalho.

Uma vez que as etapas anteriores são concluídas, a próxima etapa consiste na modelagem de ML, o ponto central do *pipeline*. Este modelo é composto por 9 algoritmos diferentes: Regressão Logística, Árvore de Decisão, *Random Forest*, *Naive Bayes*, *XGBoost*, *LGBoost*, Máquina de Vetores de Suporte, Redes Neurais Artificiais e *ExSTraCS* (versão 2.0.2.1). *ExSTraCS* é um Sistema Classificador de Aprendizado (LCS), desenvolvido com objetivo de solucionar problemas complexos de classificação e predição a partir da combinação de uma série de heurísticas (URBANOWICZ; MOORE, 2015). O *pipeline* inclui a análise do desempenho do algoritmo *ExSTraCS* antes e após a aplicação de *Quick Rule Filtering* (QRF), um procedimento de compactação (TAN; MOORE; URBANOWICZ, 2013).

Após a execução de cada algoritmo de ML, o desempenho é avaliado a partir do cálculo de métricas que incluem: acurácia balanceada, F1-*Score*, precisão, *recall*, e Característica de Operação do Receptor (ROC) área sob a curva (AUC). A acurácia balanceada corresponde à média entre *recall* e especificidade, calculadas com base no número de verdadeiros positivos (VP), verdadeiros negativos (VN), falsos positivos (FP) e falsos negativos (FN), como mostrado nas Equações 1 e 2, respectivamente (URBANOWICZ; MOORE, 2015). O F1-*Score* é uma medida amplamente utilizada em diversas áreas de ML, sendo aplicada tanto em cenários de classificação binária quanto multiclasses, e representa a média entre precisão, representada na Equação 3, e *recall* (CHICCO; JURMAN, 2020).

$$\textit{recall} = \frac{\sum VP}{\sum VP + \sum FN} \tag{1}$$

$$\text{especificidade} = \frac{\sum VN}{\sum VN + \sum FP} \tag{2}$$

$$\text{precisão} = \frac{\sum VP}{\sum VP + \sum FP} \tag{3}$$

A última etapa do pipeline, pós-análise, sumariza as métricas obtidas para cada algoritmo e gera uma série de arquivos e gráficos que facilitam a análise comparativa do desempenho de todos os algoritmos.

4 RESULTADOS E ANÁLISE

As Tabelas 2 e 3 apresentam as médias de acurácia balanceada e F1-*Score*, respectivamente, para cada subconjunto de dados utilizado no presente estudo, fazendo também a comparação entre as médias obtidas por Urbanowicz et al. (2020). Médias inferiores aos resultados obtidos por Urbanowicz et al. (2020) estão destacadas em vermelho, enquanto médias superiores aos resultados de Urbanowicz et al. (2020) estão destacadas em verde ($p<0,05$ com teste-t de *Student*).

Tabela 2 – Médias de acurácia balanceada (com desvio padrão) utilizando validação cruzada 10-*Fold* para cada subconjunto de dados e algoritmo de ML, em comparação com os resultados obtidos por Urbanowicz et al. (2020)

Algoritmo de ML	**Urbanowicz et al. (2020)**	**S1**	**S2**	**S3**	**S4**
Regressão Logística	0,6795 (0,0328)	0,7762 (0,0191)	0,7352 (0,0276)	0,7712 (0,0263)	0,7359 (0,0259)
Árvore de Decisão	0,6745 (0,0262)	0,7728 (0,0193)	0,7248 (0,0254)	0,7634 (0,0198)	0,7262 (0,0327)
Random Forest	0,6798 (0,03)	0,7806 (0,016)	0,741 (0,0293)	0,7748 (0,0214)	0,7454 (0,0346)
Naive Bayes	0,6053 (0,024)	0,5719 (0,0217)	0,6462 (0,0328)	0,5784 (0,0283)	0,6559 (0,0339)
XGB	0,6854 (0,0276)	0,7838 (0,017)	0,756 (0,0376)	0,7722 (0,0274)	0,7459 (0,0381)
LGB	0,6851 (0,0295)	0,7821 (0,0225)	0,7522 (0,0362)	0,7728 (0,0211)	0,7522 (0,0378)
SVM	0,6761 (0,0218)	0,7758 (0,0202)	0,7364 (0,0266)	0,7737 (0,0221)	0,7364 (0,0421)
ANN	0,5824 (0,0301)	0,7218 (0,0289)	0,7266 (0,0334)	0,7271 (0,0176)	0,7298 (0,0306)
LCS	0,6668 (0,0191)	0,7666 (0,0285)	0,7204 (0,033)	0,7644 (0,0258)	0,7201 (0,0329)
LCS com QRF	0,5579 (0,0164)	0,7067 (0,0326)	0,7126 (0,0327)	0,7139 (0,0301)	0,71 (0,0257)

Fonte: autoria própria

Tabela 3 – Médias de F1-*Score* (com desvio padrão) utilizando validação cruzada 10-*Fold* para cada subconjunto de dados e algoritmo de ML, em comparação com os resultados obtidos por Urbanowicz et al. (2020)

Algoritmo de ML	**Urbanowicz et al. (2020)**	**S1**	**S2**	**S3**	**S4**
Regressão Logística	0,4221 (0,042)	0,523 (0,0233)	0,4894 (0,0348)	0,5239 (0,0336)	0,4958 (0,0286)
Árvore de Decisão	0,4183 (0,0352)	0,4915 (0,0236)	0,5173 (0,0773)	0,4895 (0,0319)	0,5344 (0,0728)
Random Forest	0,4272 (0,04)	0,5131 (0,0206)	0,5558 (0,0402)	0,5158 (0,0344)	0,5687 (0,047)
Naive Bayes	0,3383 (0,0474)	0,2761 (0,039)	0,4241 (0,0655)	0,2874 (0,0518)	0,443 (0,0655)
XGB	0,4317 (0,0363)	0,5054 (0,0184)	0,5555 (0,0477)	0,5039 (0,044)	0,546 (0,0465)
LGB	0,4311 (0,0376)	0,5023 (0,0239)	0,5359 (0,05)	0,5067 (0,0292)	0,5469 (0,0582)
SVM	0,4231 (0,0298)	0,526 (0,0223)	0,5404 (0,0525)	0,5302 (0,0274)	0,5333 (0,0564)
ANN	0,2908 (0,0649)	0,5717 (0,0625)	0,5986 (0,0671)	0,5913 (0,0346)	0,6048 (0,0554)
LCS	0,4215 (0,027)	0,4869 (0,0339)	0,4649 (0,0263)	0,4879 (0,0257)	0,4822 (0,0513)
LCS com QRF	0,2188 (0,0486)	0,5817 (0,065)	0,594 (0,0654)	0,5953 (0,0582)	0,59 (0,0506)

Fonte: autoria própria

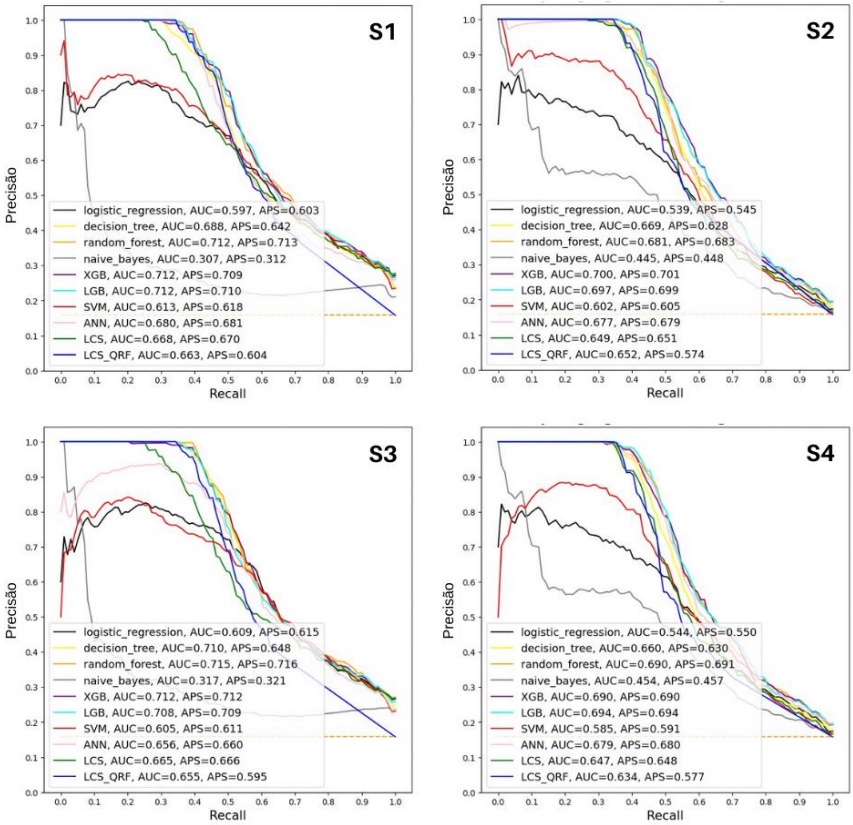
Os resultados apresentados nas Tabelas 2 e 3 mostram que, para todos os subconjuntos de dados, a maioria dos algoritmos de ML teve seu desempenho aumentado significativamente ($p<0,05$ com teste-t de *Student*) quando comparado com as médias obtidas por Urbanowicz e colaboradores (2020). Não foi possível determinar o algoritmo de maior desempenho para cada subconjunto de dados porque estes apresentaram médias similares. Os subconjuntos S2 e S4, gerados utilizando *k-means* como método de *undersampling*, apresentaram desempenhos constantemente positivos para todos os algoritmos de ML quando comparado com Urbanowicz et al. (2020) ($p<0,05$ com teste-t de *Student*). No entanto, o aumento do valor de *k* no método kNN (k=3 para S2 e k=10 para S4) não teve impacto significativo nos resultados.

Estes resultados são consistentes com outros estudos que mostram que métodos de imputação de dados, incluindo kNN, são confiáveis para estimar valores faltantes e podem contribuir para o aumento do desempenho da classificação (KARRAR, 2022; KSIAZEK et al., 2020; WU et al., 2019). Um estudo realizado por Karrar (2022) inseriu, de forma aleatória, valores faltantes em um conjunto de dados para avaliar a precisão da imputação utilizando kNN e obteve uma taxa de acurácia de 89,5% utilizando este método. Wu e colaboradores (2019) compararam diferentes métodos de imputação em um conjunto de dados de câncer de mama e a técnica de kNN apresentou as maiores médias de acurácia de 4 entre os 7 classificadores analisados, em comparação a outras técnicas de imputação. De forma similar, o uso do algoritmo *k-means* para *undersampling*, sozinho ou combinado com outras técnicas baseadas em clusterização, também vem sendo reportado na literatura como uma alternativa para aumentar a diversidade das amostras e reduzir o *underfitting* (PANDEY; JAIN, 2017; TAN; MOORE; URBANOWICZ, 2013).

Em geral, os subconjuntos S1 e S3 também apresentaram melhores resultados quando comparados com os resultados obtidos por Urbanowicz et al. (2020), utilizando seleção aleatória como método de *undersampling*. A exceção foi o algoritmo de *Naive Bayes*, que apresentou desempenho inferior para ambas as métricas quando comparado com Urbanowicz et al. (2020) ($p<0,05$ com teste-t de *Student*). Urbanowicz e colaboradores (2020) resolveram o problema de desbalanceamento de classes no conjunto de dados PLCO a partir da seleção apenas de controles saudáveis com dados de genotipagem disponíveis (n=4.298). Este critério de seleção ocasionou um viés de amostragem (ex. 85% homens, alto número de fumantes), o que pode ter impactado o desempenho dos algoritmos de ML presentes no *pipeline*. Embora a seleção aleatória tenha sido utilizada no presente estudo, não foi observado viés nas populações geradas utilizando este método; além disso, a combinação desta técnica com o kNN para imputação de dados faltantes pode justificar a melhora no desempenho de uma forma geral. O método de seleção aleatória possui um alto grau de incerteza, uma vez que pode gerar boas instâncias e resultar em um modelo com alto desempenho, ou pode potencialmente levar à perda de informações relevantes, impactando o processo de treinamento e o desempenho do modelo (HASANIN et al., 2019).

A Figura 3 traz uma nova perspectiva aos resultados ao trazer uma análise comparativa dos algoritmos de ML para todos os subconjuntos de dados, baseados nas curvas de precisão/*recall*.

Figura 3 – Curvas de precisão/*recall* representando o desempenho dos algoritmos de ML nos subconjuntos de dados S1 a S4, gerados a partir da combinação de diferentes métodos de *undersampling* e imputação de valores. Inclui AUC da precisão/*recall* e *score* de precisão médio (APS)



Fonte: autoria própria

De forma semelhante ao que foi observado com as outras métricas, todos os subconjuntos de dados apresentaram resultados positivos para a maioria dos algoritmos do *pipeline*, com médias similares entres os algoritmos de maior desempenho. A exceção é novamente o algoritmo *Naive Bayes*, que apresentou um desempenho inferior quando comparado com outros algoritmos de ML ($p<0,05$ com teste U de Mann-Whitney). Como o modelo de *Naive Bayes* assume que os atributos utilizados são independentes (WICKRAMASINGHE; KALUTARAGE, 2020), atributos correlacionados utilizados para classificação neste estudo (ex. status atual de consumo de cigarro, número de anos de consumo de cigarro) podem ter causado um impacto negativo no desempenho.

5 CONSIDERAÇÕES FINAIS

No presente estudo foram analisados os efeitos de métodos de imputação de dados faltantes e *undersampling* baseado em clusterização no desempenho de diferentes algoritmos de ML para a classificação do câncer de pâncreas. Os métodos de *k-means* e seleção aleatória foram utilizados para *undersampling* da população de controles saudáveis, enquanto kNN foi utilizado para a imputação de valores faltantes na população de pacientes com câncer de pâncreas. O

desempenho foi analisado utilizando o *pipeline* de ML desenvolvido e utilizado por Urbanowicz et al. (2020). Os resultados apresentados mostram que, para os 4 subconjuntos de dados gerados utilizando diferentes métodos de pré-processamento, houve um aumento significativo no desempenho para a maioria dos algoritmos de ML quando comparado aos resultados obtidos por Urbanowicz et al. (2020). Os resultados obtidos sugerem que os métodos analisados neste estudo podem ser uma alternativa para o aumento da acurácia da classificação.

REFERÊNCIAS

AHMED, M.; SERAJ, R.; ISLAM, S. M. S. The k-means Algorithm: A Comprehensive Survey and Performance Evaluation. **Electronics**, v. 9, n. 8, ago. 2020.

BAI, L.; LIANG, J.; GUO, Y. An Ensemble Clusterer of Multiple Fuzzy k-Means Clusterings to Recognize Arbitrarily Shaped Clusters. **IEEE Transactions on Fuzzy Systems**, v. 26, n. 6, p. 3524-3533, dez. 2018.

CHARBUTY, B.; ABDULAZEEZ, A. Classification Based on Decision Tree Algorithm for Machine Learning. **Journal of Applied Science and Technology Trends**, v. 2, n. 01, p. 20-28, mar. 2021.

CHEN, PH. C.; LIU, Y.; PENG, L. How to develop machine learning models for healthcare. **Nature Materials**, v. 18, p. 410–414, abr. 2019.

CHEN, T.; GUESTRIN, C. XGBoost: A Scalable Tree Boosting System. **Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**, Association for Computing Machinery, p. 785–794, ago. 2016.

CHICCO, D.; JURMAN, G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. **BMC Genomics**, v. 21, n. 6, jan. 2020.

DHEEBA, J.; ALBERT SINGH, N.; TAMIL SELVI, S. Computer-aided detection of breast cancer on mammograms: A swarm intelligence optimized wavelet neural network approach. **Journal of Biomedical Informatics**, v. 49, p. 45–52, jun. 2014.

ESTEVA, A. et al. Dermatologist-level classification of skin cancer with deep neural networks. **Nature**, v. 542, p. 115-118, jan. 2017.

GHORI, K. M. et al. Performance Analysis of Different Types of Machine Learning Classifiers for Non-Technical Loss Detection. **IEEE Access**, v. 8, p. 16033-16048, jan. 2020.

GUZMÁN-PONCE, A. et al. A New Under-Sampling Method to Face Class Overlap and Imbalance. **Applied Sciences**, v. 10, n. 15, p. 5164, jul. 2020.

HASANIN, T. et al. Investigating Random Undersampling and Feature Selection on Bioinformatics Big Data. **2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService)**, p. 346-356, abr. 2019.

HASSAN, N. S. et al. Medical Images Breast Cancer Segmentation Based on K-Means Clustering Algorithm: A Review. **Asian Journal of Research in Computer Science**, v. 9, n. 1, p. 23–38, mai. 2021.

JIANG, F. et al. Artificial intelligence in healthcare: Past, present and future. **Stroke and Vascular Neurology**, v. 2, n. 4, p. 230–243, jun. 2017.

JUNG, Y. Multiple predicting K-fold cross-validation for model selection. **Journal of Nonparametric Statistics**, v. 30, p. 197-215, nov. 2017.

KAISER, J. Dealing with Missing Values in Data. **Journal of Systems Integration**, v. 5, p. 42-51, nov. 2014.

KARRAR, A. E. The Effect of Using Data Pre-Processing by Imputations in Handling Missing Values. **Indonesian Journal of Electrical Engineering and Informatics**, v. 10, n. 2, jun. 2022.

KSIAZEK, W. et al. Development of novel ensemble model using stacking learning and evolutionary computation techniques for automated hepatocellular carcinoma detection. **Biocybernetics and Biomedical Engineering**, v. 40, n. 4, p. 1512-1524, out. 2020.

KURANI, A. et al. A Comprehensive Comparative Study of Artificial Neural Network (ANN) and Support Vector Machines (SVM) on Stock Forecasting. **Annals of Data Science**, v. 10, p. 183–208, fev. 2023.

PANDEY, A.; JAIN, A. Comparative Analysis of KNN Algorithm using Various Normalization Techniques. **International Journal of Computer Network and Information Security**, v. 9, p. 36-42, nov. 2017.

PROROK, P. C. et al. Design of the prostate, lung, colorectal and ovarian (PLCO) cancer screening trial. **Control Clin. Trials**, v. 21, p. 273S-309S, dez. 2000.

QIN, J. et al. Distributed k-Means Algorithm and Fuzzy c-Means Algorithm for Sensor Networks Based on Multiagent Consensus Theory. **IEEE Transactions on Cybernetics**, v. 47, n. 3, p. 772-783, mar. 2017.

SCHONLAU, M.; ZOU, R. Y. The random forest algorithm for statistical learning. **The Stata Journal**, v. 20, n. 1, p. 3-29, mar. 2020.

SUBASI, A. Machine learning techniques. In: ____. **Practical Machine Learning for Data Analysis Using Python**, Academic Press, 2020. cap. 3.

SUNG, H. et al. Global Cancer Statistics 2020: GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries. **CA Cancer J. Clin.**, v. 71, n. 3, p. 209-249, fev. 2021.

TAN, J.; MOORE, J.; URBANOWICZ, R. Rapid Rule Compaction Strategies for Global Knowledge Discovery in a Supervised Learning Classifier System. **ECAL 2013: The Twelfth European Conference on Artificial Life**, p. 110-117, set. 2013.

URBANOWICZ, R. et al. A rigorous machine learning analysis pipeline for biomedical binary classification: application in pancreatic cancer nested case-control studies with implications for bias assessments. **ArXiv**, v. abs/2008.12829v2, set. 2020.

URBANOWICZ, R. ExSTraCS ML Pipeline Binary Notebook, set. 2020. Disponível em: <https://github.com/UrbsLab/ExSTraCS_ML_Pipeline_Binary_Notebook>. Acesso em: 1 out. 2022.

URBANOWICZ, R. J.; MOORE, J. H. ExSTraCS 2.0: Description and Evaluation of a Scalable Learning Classifier System. **Evolutionary Intelligence**, v. 8, n. 2, p. 89-116, set. 2015.

VUTTIPITTAYAMONGKOL, P.; ELYAN, E.; PETROVSKI, A. On the class overlap problem in imbalanced data classification. **Knowledge-Based Systems**, v. 212, jan. 2021.

WEI, L. et al. Gene Expression Value Prediction Based on XGBoost Algorithm. **Frontiers in Genetics**, v. 10, nov. 2019.

WEI, X. A Method of Enterprise Financial Risk Analysis and Early Warning Based on Decision Tree Model. **Security and Communication Networks**, v. 2021, set. 2021.

WEI-CHAO, L. et al. Clustering-based undersampling in class-imbalanced data. **Information Sciences**, v. 409-410, p. 17-26, out. 2017.

WICKRAMASINGHE, I.; KALUTARAGE, H. Naive Bayes: applications, variations and vulnerabilities: a review of literature with code snippets for implementation. **Soft Computing**, v. 25, p. 2277–2293, set. 2020.

WINTER, K. et al. Diagnostic and therapeutic recommendations in pancreatic ductal adenocarcinoma. Recommendations of the Working Group of the Polish Pancreatic Club. **Przeglad gastroenterologiczny**, v. 14, n. 1, p. 1-18, mar. 2019.

WU, X. et al. Imputation techniques on missing values in breast cancer treatment and fertility data. **Health Information Science and Systems**, v. 7, p. 19, out. 2019.

YANG, D. X. et al. Prevalence of Missing Data in the National Cancer Database and Association with Overall Survival. **JAMA Network Open**, v. 4, n. 3, p. e211793, mar. 2021.

ZHANG, J.; CHEN, L.; ABID, F. Prediction of Breast Cancer from Imbalance Respect Using Cluster-Based Undersampling Method. **Journal of Healthcare Engineering**, v. 2019, out. 2019.