UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

GABRIEL MOREIRA BERETTA

Comparação de estratégias computacionais para integração de dados ômicos na classificação de subtipos de tumores

Monografia apresentada como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação

Orientador: Profª. Drª. Mariana Recamonde-Mendoza

Porto Alegre
2023

RESUMO

Atualmente, o uso de Aprendizado de Máquina (AM) para auxílio ao diagnóstico de doenças tem sido recorrente devido aos seus excelentes resultados. Essa técnica tem se tornado fundamental para se trabalhar com dados ômicos, de alta dimensionalidade e complexidade, na elaboração de modelos clínicos preditivos. Na busca por modelos de alta sensibilidade e precisão, múltiplos dados ômicos são utilizados em conjunto, havendo uma crescente necessidade em se explorar análises computacionais integrativas. Dessa forma, muitos métodos de integração para dados ômicos têm surgido e avaliá-los tem se transformado em uma tarefa cada vez mais difícil. Neste trabalho, avaliamos as três principais estratégias de integração de dados ômicos: estratégia de estágio inicial, intermediário e final; bem como alguns algoritmos para integração de dados para estratégia de estágio intermediários. Para tal, utilizamos três algoritmos de AM para avaliar os resultados de cada uma das estratégias de integração de dados ômicos: (i) Árvores de Decisão; (ii) Florestas Aleatórias; (iii) Máquinas de Vetores de Suporte (SVM). Além disso, selecionamos dois algoritmos de integração de dados ômicos, o *Neighborhood based Multi-Omics Clustering* (NEMO) e o *Cancer Integration via Multikernel Learning* (CIMLR), para executar a integração dos dados na estratégia de estágios intermediários. Os resultados obtidos neste trabalho apontaram que a estratégia de estágio inicial, apesar de muito criticada, tem certa vantagem sobre as demais para os modelos testados. Entretanto, os resultados obtidos com as outras estratégias não são ruins, indicando que, devido aos testes realizados com modelos específicos, não é possível chegar a uma conclusão definitiva sobre qual estratégia é a melhor. Na estratégia de estágio intermediário, também reforçamos a ideia de que alguns modelos específicos de AM desempenham melhor que outros, pois possuem certa dificuldade em utilizar suas transformações. Finalmente, apesar das diversas críticas existentes sobre a estratégia de estágio inicial (*e.g.,* inconsistências nos dados, alta dimensionalidade), seus resultados foram os melhores dentre todas, reforçando a ideia de que ela não pode ser descartada como uma possibilidade. Além disso, reforça também que quando o objetivo é identificar padrões ou relacionamentos que abrangem várias ômicas, a estratégia de estágio inicial é uma das mais úteis. Contudo, não podemos ignorar os benefícios das demais estratégias.

Palavras-chave: Aprendizado de máquina. dados multi-ômicos. estratégias de integração de dados. bioinformática.

Comparison of computational strategies for integrating omic data in the classification of tumor subtypes

ABSTRACT

Currently, the use of Machine Learning (ML) to aid in the diagnosis of diseases has been recurrent due to its excellent results. This technique has become fundamental for working with omic data, of high dimensionality and complexity, in the elaboration of predictive clinical models. In the search for models with high sensitivity and precision, multiple omic data are used together, with a growing need to explore integrative computational analysis. Thus, many integration methods for omic data have emerged and evaluating them has become an increasingly difficult task. In this work, we evaluate the three main omics data integration strategies: early integration, intermediate integration, and late integration strategies, as well as some data integration algorithms for intermediate integration strategies. To do so, we use three ML algorithms to evaluate the results of each of the omics data integration strategies: (i) Decision Trees; (ii) Random Forests; (iii) Support Vector Machines (SVM). In addition, we select two omics data integration algorithms, the Neighborhood-based Multi-Omics Clustering (NEMO) and the Cancer Integration via Multikernel Learning (CIMLR), to perform data integration in the intermediate integration strategy. The results obtained in this work showed that the early integration strategy, despite being heavily criticized, has some advantage over the others for the tested models. However, the results obtained with the other strategies are not bad, indicating that, due to the tests carried out with specific models, it is not possible to reach a definitive conclusion about which strategy is the best. In the intermediate integration strategy, we also reinforce the idea that some specific ML models perform better than others, as they have some difficulty in using their transformations. Finally, despite the various criticisms about the early integration strategy (*e.g.,* inconsistencies in data, high dimensionality), its results were the best among all, reinforcing the idea that it cannot be discarded as a possibility. Furthermore, it also reinforces that when the goal is to identify patterns or relationships that span multiple omics, the early integration strategy is one of the most useful. However, we cannot ignore the benefits of the other strategies.

Keywords: machine learning, multi-omics data, data integration strategies, bioinformatics.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AM	Aprendizado de Máquina
BRCA	Breast invasive carcinoma
COAD	Colon Adenocarcinoma
mRNA	Messenger RNA
miRNA	MicroRNA
CNV	Copy number variation
SVM	Support vector machine
RF	Random Forest
DT	Decision Tree
MCC	Matthews correlation coefficient
SIMLR	Single-cell Interpretation via Multi-kernel Learning
CIMLR	Cancer Integration via Multi-kernel Learning
SNF	Similarity Network Fusion
NEMO	Neighborhood based Multi-Omics clustering
CCA	Canonical Correlation Analysis
PCA	Principal Component Analysis
KNN	K-Nearest Neighbors
TCGA	The Cancer Genome Atlas
A1BG	Alpha-1-Beta-Glycoprotein
A2M	Alpha-2-Macroglobulin
AACS	Acetoacetyl-Coenzyme A Synthetase
VP	Verdadeiro positivo
FP	Falso positivo
VN	Verdadeiro negativo
FN	Falso negativo

SUMÁRIO

1 INTRODUÇÃO

A caracterização de componentes em sistemas celulares é fundamental para entender melhor os processos fisiológicos e o desenvolvimento de doenças. A fim de fomentar descobertas a respeito das moléculas biológicas e de como as mesmas estão contribuindo para a função e dinâmica de um organismo, diversas tecnologias foram desenvolvidas para uma avaliação abrangente ou global de um conjunto de moléculas, ou alterações moleculares — como transcritos, proteínas, e mecanismos de metilação de DNA[1]. Tecnologias de sequenciamento e análise molecular de alto rendimento (*High-throughput technologies*) têm se estabelecido como ferramentas de grande potencial para viabilizar uma medicina mais precisa (HASIN; SELDIN; LUSIS, 2017). Estas tecnologias geram os chamados dados ômicos — como genômica, transcriptômica, proteômica e epigenômica — que ajudam a compreender os mecanismos moleculares do estabelecimento ou progressão de doenças, identificar biomarcadores, ou prever desfechos clínicos, como sobrevida ou eficácia de terapias (KARCZEWSKI; SNYDER, 2018; OLIVIER et al., 2019).

Conforme será apresentado em mais detalhes no Capítulo 2, cada tecnologia ômica é desenvolvida para explorar um nível molecular de forma extraordinariamente detalhada. Por exemplo, enquanto a genômica é capaz de analisar mutações no DNA e variações no número de cópias (CNVs, de *copy number variation*), a transcriptômica mensura os níveis de moléculas de RNA em escala global para avaliar quais transcritos estão presentes e o quão expressos eles estão em determinada amostra ou condição (KARCZEWSKI; SNYDER, 2018). Embora individualmente cada tecnologia ômica (e os respectivos dados ômicos produzidos) tenha contribuído para avanços na medicina e na prática clínica, sabe-se que cada tecnologia individualmente não pode capturar toda a complexidade biológica da maioria das doenças humanas (HASIN; SELDIN; LUSIS, 2017; KARCZEWSKI; SNYDER, 2018).

Muitas das doenças mais prevalentes, como diabetes, câncer e doença de Alzheimer, são resultados de uma combinação de múltiplos fatores genéticos em combinação com estilo de vida e fatores ambientais — sendo chamadas, portanto, de doenças multifatoriais ou doenças complexas. Nestes casos, a combinação de dados ômicos faz-se necessária, pois a maior complexidade dos processos moleculares subjacentes a estas doenças requerem mais de um tipo de análise ômica para produzir uma visão mais precisa

[1] A metilação do DNA é uma modificação química da molécula do DNA, envolvendo a adição de um grupo metil, e está associada a mecanismos de silenciamento genético que impedem a expressão de determinado(s) gene(s).

da biologia e da doença.

A integração de múltiplos dados ômicos é considerada uma estratégia promissora para ajudar no diagnóstico de doenças e na compreensão de fatores como biomarcadores associados ao desenvolvimento ou progressão de doenças (HASIN; SELDIN; LUSIS, 2017; KARCZEWSKI; SNYDER, 2018; CAI et al., 2022). Trabalhos anteriores demonstram que a acurácia da classificação de estágio tumoral é maior ao se integrar diferentes dados ômicos (MA et al., 2020), e que esta abordagem denominada multi-ômicas também é efetiva para a identificação de genes biomarcadores de prognóstico em pacientes com câncer de pulmão (ASADA et al., 2020).

Do ponto de vista metodológico, a integração de dados ômicos impõe diversos desafios para a concepção e avaliação de estratégias, especialmente devido à alta dimensionalidade e variabilidade desses tipos de dados. Nos últimos anos, muitos métodos e estratégias para a integração de dados ômicos foram propostos. As principais estratégias utilizadas atualmente para integração dos dados ômicos são: (i) a integração de estágio inicial (*Early Integration*), que consiste na simples concatenação dos dados ômicos; (ii) a integração de estágio intermediário (*Middle Integration*), baseada em métodos de transformação dos dados e em representações intermediárias, como grafos; e a (iii) integração de estágio final (*Late Integration*), baseada na agregação de saídas de modelos preditivos. Cada uma dessas estratégias apresenta benefícios diferentes. Por exemplo, a integração de estágio inicial possui a vantagem da simplicidade de emprego de método de aprendizado de máquina com os dados gerados (REEL et al., 2020). Já a integração de estágio intermediário pode ser utilizada para combinar uma ampla gama de ômicas sem acarretar um grande aumento da dimensionalidade dos dados, mas introduz uma complexidade no processo devido à dificuldade na manipulação dos dados transformados. Por fim, a integração de estágio final tem a vantagem de ser facilmente aplicada para combinar modelos baseados em diferentes tipos de ômicas, onde cada modelo é desenvolvido a partir de um grupo de pacientes com as mesmas informações sobre as doenças (HE et al., 2016).

No entanto, devido à ampla variedade de métodos propostos e a uma falta de critérios performáticos e bons padrões de referência, a tarefa de avaliar e comparar estes métodos de integração de dados ômicos não é trivial. Alguns métodos foram desenvolvidos para aplicações em problemas específicos, outros possuem o uso restrito para alguns tipos de dados ômicos ou uma determinada quantidade de conjuntos de dados ômicos, e nem todos disponibilizam uma implementação funcional ou documentação completa que possa subsidiar seu emprego em uma avaliação experimental. Assim, esclarecer as vanta-

gens e desvantagens de métodos e estratégias de integração de dados ômicos e comparar experimentalmente seu desempenho para tarefas de interesse segue sendo um desafio.

Neste trabalho, nosso objetivo é avaliar e comparar as estratégias de integração de dados ômicos considerando os três estágios mencionados: (i) inicial, (ii) intermediário e (iii) final. A escolha da análise experimental ao nível de estratégia, e não de método, visa proporcionar maior flexibilidade na escolha de métodos que possam atender plenamente as necessidades em termos de tipo de tarefa e natureza e diversidade dos dados ômicos usados, bem como prover um direcionamento inicial para classes de métodos de integração que possam ser mais promissoras. Como estudo de caso, o presente trabalho aborda o problema de diagnosticar subtipos de tumor em câncer de mama e de cólon, usando dados disponibilizados publicamente pelo consórcio internacional The Cancer Genome Atlas (TCGA) (TOMCZAK; CZERWIŃSKA; WIZNEROWICZ, 2015) para treinamento de modelos preditivos com aprendizado de máquina (AM).

Comparamos os resultados obtidos a partir das três estratégias de integração de dados nos dois tipos de câncer analisados, utilizando três algoritmos de classificação de aprendizado de máquina: (i) florestas aleatórias (*Random Forest*, RF); (ii) árvores de decisão (*Decision Tree*, DT); e (iii) máquinas de vetores de suporte (*Support Vector Machine*, SVM). Através dos experimentos realizados, buscamos quantificar e avaliar quais das estratégias de integração de dados ômicos trazem os melhores resultados e qual a influência dos modelos de aprendizado de máquina nesse processo.

O trabalho está organizado como segue: O Capítulo 2 apresenta todo o embasamento teórico e conhecimento necessário para compreender o presente estudo; o Capítulo 3 sumariza a literatura relacionada à pesquisa desenvolvida; o Capítulo 4 apresenta a metodologia do trabalho; o Capítulo 5 apresenta os experimentos e resultados; e o Capítulo 6 apresenta as conclusões e propostas para trabalhos futuros.

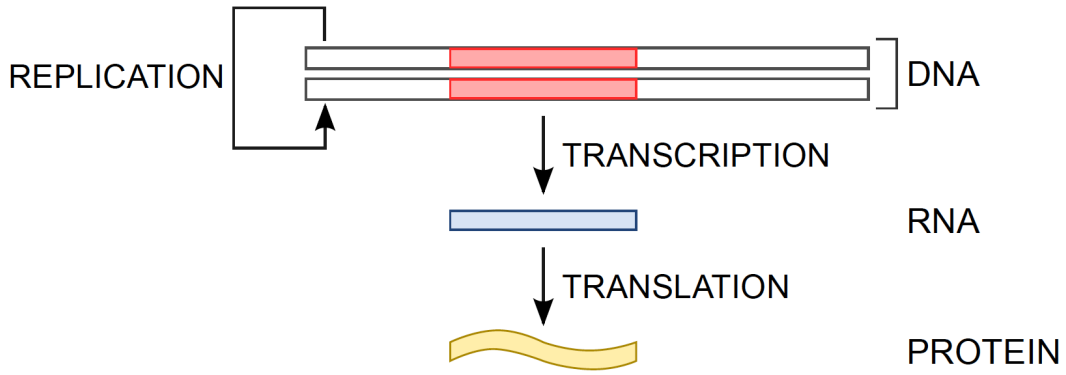
2 REFERENCIAL TEÓRICO

Neste capítulo são abordados os principais conceitos revisados a partir de estudos validados que dão embasamento teórico para o presente trabalho. O capítulo inicia revisando conceitos biológicos fundamentais para compreensão do trabalho, e na sequência apresenta os conceitos computacionais envolvidos com o tema da pesquisa.

2.1 Referencial Biológico

A genômica é uma área de pesquisa relativamente recente, que visa estudar todos os genes de um indivíduo (conhecido como genoma), incluindo as interações destes genes entre si e a influência exercida pelo ambiente no qual o indivíduo vive[1]. Um gene é definido como uma sequência de DNA que carrega as instruções necessárias para direcionar as atividades das células e as funções do corpo. Ao passar pelo processo de expressão gênica, conforme definido pelo Dogma Central da Biologia Molecular (Figura 2.1), um gene produz o seu produto final, o qual pode ser tanto uma molécula de RNA mensageiro (mRNA, de *messenger RNA*) que será posteriormente traduzida em uma proteína funcional (conhecido como gene codificante), ou uma molécula de RNA funcional (também conhecida como não codificante) (SALZBERG, 2018). Embora não exista um consenso universal sobre o número de genes codificantes de proteína no genoma humano, estima-se que os seres humanos possuam em torno de 20 mil genes (SALZBERG, 2018).

Figura 2.1 – Dogma central da biologia molecular



Fonte: Mendoza (2014)

Dessa forma, a análise do processo de expressão gênica é de suma importância

[1] <https://www.genome.gov/about-genomics/fact-sheets/Genetics-vs-Genomics>

para compreender o estado funcional do organismo e investigar as causas e consequências de doenças. Atualmente, existem fenômenos, moléculas e modificações químicas da estrutura do DNA envolvidos com a expressão gênica que são muito utilizados para o estudo de doenças complexas, por terem um papel já reconhecido no desenvolvimento ou progressão das mesmas. A análise e monitoramento destes fatores é usualmente realizada através da mensuração dos seus produtos funcionais, buscando a concentração e intensidade com que um produto gênico está sendo produzido por meio de tecnologias de alto rendimento.

A possibilidade de avaliar padrões globais de expressão gênica, interrogando conjuntos inteiros de RNAs codificantes ou não-codificantes, proteínas, e do próprio genoma, foi introduzida pelo advento das tecnologias ômicas (KARCZEWSKI; SNYDER, 2018). Enquanto a genômica foi o primeiro campo das ômicas a surgir e representa, atualmente, o mais maduro deles em termos de tecnologias para mensuração e protocolos de análises de dados, outros campos foram se desenvolvendo e focando na análise de diferentes moléculas ou fenômenos, conforme revisado por Hasin, Seldin and Lusis (2017). A integração de diferentes tipos de dados ômicos no estudo de doenças — uma abordagem conhecida como multi-ômicas — é frequentemente usada para elucidar as alterações genéticas e as vias e processos biológicos diferem entre os grupos de doença e controle, determinando padrões moleculares associados a doenças.

Nas seções a seguir, serão detalhados a abordagem multi-ômica e os tipos de dados ômicos explorados no presente trabalho.

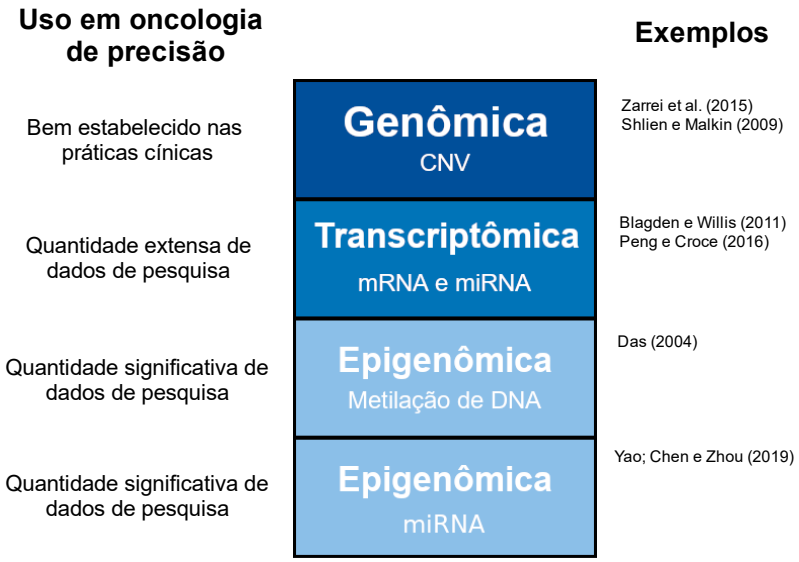
2.1.1 Dados ômicos

Os chamados dados ômicos, são dados de difícil análise que se originam de pesquisas relacionadas ao genoma, transcriptoma, proteoma e diversos outros tipos de moléculas que constituem as células de um organismo. O termo "ômico" é utilizado para se referir às tecnologias usadas para estudar em larga escala (isto é, o conjunto completo no organismo) a atividade e as relações entre as moléculas, sendo elas genes, proteínas, RNAs codificantes (mRNA) e RNAs não codificantes (*e.g.,* microRNAs), entre outras.

Na Figura 2.2, ilustramos alguns dos principais campos das ciências ômicas, focando nos tipos de dados ômicos que serão explorados no presente trabalho. Adicionalmente, a Figura ressalta o nível de aplicação em pesquisas relacionadas à oncologia de precisão, conforme revisado por Olivier et al. (2019). Atualmente, devido à dificuldade de

analisar os dados ômicos, existe uma forte tendência de utilizar algoritmos de aprendizado de máquina, bem como modelos matemáticos e outras formas de inteligência artificial na busca por padrões nesses tipos dados. Contudo, buscando resultados mais confiáveis, costuma-se juntar diversos tipos de dados ômicos diferentes para serem utilizados em análises de padrões, esses dados ômicos unidos também são comumente chamados de dados multi-ômicos.

Figura 2.2 – Resumo de pesquisas em tecnologias ômicas sobre câncer e distúrbios humanos.



Fonte: Adaptado de Olivier et al. (2019)

2.1.2 Transcriptômica

Transcriptômica é uma área de estudo cientifico responsável por analisar e avaliar o conjunto de elemento transcritos, ou seja, RNAs mensageiros, RNAs ribossômicos, RNAs transportadores, bem como RNAs não-codificantes (por exemplo, microRNAs).

O RNA mensageiro (mRNA) é um ácido ribonucleico, que tem como objetivo principal a transferência de informações do DNA para geração de um produto funcional em forma de proteína. Assim, participando diretamente no processo de tradução da sequência de nucleotídeos como um transportador de informações. Além disso, o mRNA é uma molécula muito importante em pesquisas e tratamentos de câncer (BLAGDEN; WILLIS, 2011), ela leva a informação do gene e serve como molde para a produção de

proteína. Sendo assim, se existe algum erro nesse processo, proteínas importantes não são sintetizadas ou são produzidas em excesso, podendo levar ao desenvolvimento de tumores.

Outra molécula importante é a de microRNA (miRNA), que são pequenos RNAs não codificantes que funcionam como silenciadores da expressão gênica. O silenciamento da expressão ocorre mediante uma ligação física entre o miRNA e o mRNA, causando inibição da tradução ou degradação do transcrito de mRNA. Importante salientar que, quando genes são silenciados por miRNAs, a expressão de outros genes e proteínas podem aumentar ou diminuir em decorrência disto, realçando o quão minuciosas são as ações de um miRNA. Dessa forma, um simples distúrbio causado por alguma doença pode desregular a expressão dos miRNAs, que acaba sendo muitas vezes facilmente notada, isso acontece muito em pacientes com tumores, facilitando a distinção entre células cancerígenas (PENG; CROCE, 2016).

2.1.3 Epigenômica

A epigenômica é o estudo de um conjunto de elementos, processos e mecanismos, que causam alterações no fenótipo de organismos sem alterar a sequência de DNA. Um mecanismo epigenético é responsável por gerar mudanças na leitura de genes, pois fazem alterações químicas em sequências de DNA ou proteínas.

Um dos principais mecanismos epigenéticos é a metilação de DNA que dificulta o acesso a posições de fita de DNA onde o grupo metil é aplicado. Essa modificação causa efeitos diferentes dependendo do organismo e da posição onde o grupo metil é colocado, além de poder ser herdada ou adquirida.

A modificação química da metilação de DNA é um dos grandes aliados nas pesquisas sobre câncer, pois atua indiretamente na expressão dos genes (DAS, 2004), facilitando a identificação de desequilíbrios no padrão de metilação, assim associados ao desenvolvimento de tumores.

A expressão do miRNA possui um papel importante na epigenética também, pois, como dito antes, ela modula genes se encaixando com mRNAs sem alterar as sequências de DNA. Além disso, mecanismos epigenéticos, como a metilação de DNA e modificação de RNA, também podem modificar as moléculas de miRNA (YAO; CHEN; ZHOU, 2019).

2.1.4 Genômica

Na genômica, o foco está no sequenciamento e análise do genoma dos organismos. Neste tipo de ômica, um dos principais fenômenos utilizados para avaliação do comportamento de doenças é a variação do número de cópias (CNV, de *Copy number variation*). A variação do número de cópias, consiste em um tipo de varição estrutural, onde regiões do genoma são repetidas e, por ser uma variação estrutural, esse fenômeno acaba afetando de forma considerável os pares de bases dos cromossomos de um organismo. De acordo com estudos, aproximadamente dois terços do genoma humano inteiro é composto de repetições (ZARREI et al., 2015). Além disso, o CNV desempenha um papel importante na geração de variação populacional, assim como em fenótipos de doenças em nós humanos. O CNV vária para cada indivíduo e afeta uma fração maior do genoma do que um único nucleotídeo, por isso o CNV vem sendo estudado cada vez mais para análise de doenças como câncer e Alzheimer (SHLIEN; MALKIN, 2009).

2.2 Referencial Computacional

Nos últimos anos, estudos sobre a aplicabilidade de aprendizado de máquina (AM) para auxiliar no diagnóstico de doenças complexas a partir de dados ômicos têm ganhado destaque devido aos seus bons resultados na área da saúde. Diversos métodos e algoritmos são utilizados, principalmente, para classificar e prever informações como estágio da doença, gravidade do quadro, tipo de doença e, em alguns casos, até o tempo de vida das pessoas (conhecido como análise de sobrevida). Muitos desses estudos utilizam métodos de aprendizado supervisionado e, no mínimo, um dos três estágios de integração de dados ômicos: o estágio inicial, o estágio intermediário, o estágio final ou uma combinação entre eles (KARCZEWSKI; SNYDER, 2018).

Existem três categorias principais de algoritmos de aprendizado de máquina: aprendizado supervisionado, aprendizado não-supervisionado e aprendizado por reforço. Cada uma dessas categorias de algoritmos tem um foco diferente. No caso dos algoritmos de aprendizado supervisionado, suas tarefas mais comuns são predição e classificação de dados. O usuário fornece pares de entrada e saída conhecidos, geralmente na forma de vetores, de modo que cada saída receba um rótulo, que pode ser um valor numérico (para regressão) ou uma classe (para classificação). Os demais tipos de aprendizado lidam com tarefas que não possuem nenhum tipo de supervisão em relação a rótulos espera-

dos (aprendizado não-supervisionado) ou nas quais a supervisão é dada em termos de um feedback avaliando o quão boa foi uma ação para um determinado estado e uma tarefa alvo (aprendizado por reforço). Neste trabalho, focaremos em algoritmos de aprendizado supervisionado para tarefas de classificação. A combinação de formas de integrar dados ômicos, juntamente com algoritmos de classificação de aprendizado de máquina, tem trazido resultados incríveis para identificação de características do câncer (RAJAMOHANA et al., 2020), bem como para a predição e avaliação do Alzheimer (KAVITHA et al., 2022).

As seções a seguir revisam os aspectos teóricos dos algoritmos de aprendizado supervisionado e das estratégias para integração de dados ômicos empregadas neste trabalho.

2.2.1 Algoritmos de aprendizado supervisionado

O aprendizado supervisionado é uma técnica fundamental de AM que visa ensinar um modelo a reconhecer padrões em dados e fazer previsões precisas a partir de novos dados. Esse tipo de aprendizado envolve a utilização de dados rotulados, ou seja, dados que já possuem uma classificação ou rótulo conhecido, e a partir desses dados, o modelo é treinado para prever a classificação de novos dados. Uma das principais tarefas de aprendizado supervisionado é a classificação, onde o modelo é treinado para categorizar instâncias em classes predefinidas com base em atributos (ou características) dos dados. Quando existem três ou mais classes possíveis, o problema é denominado de classificação multiclasse. Existem muitos algoritmos de aprendizado que desenvolvem modelos para tratar o problema de classificação multiclasse, mas vamos nos ater apenas aos algoritmos utilizados nesse trabalho: árvores de decisão, máquinas de vetores de suporte, e florestas aleatórias.

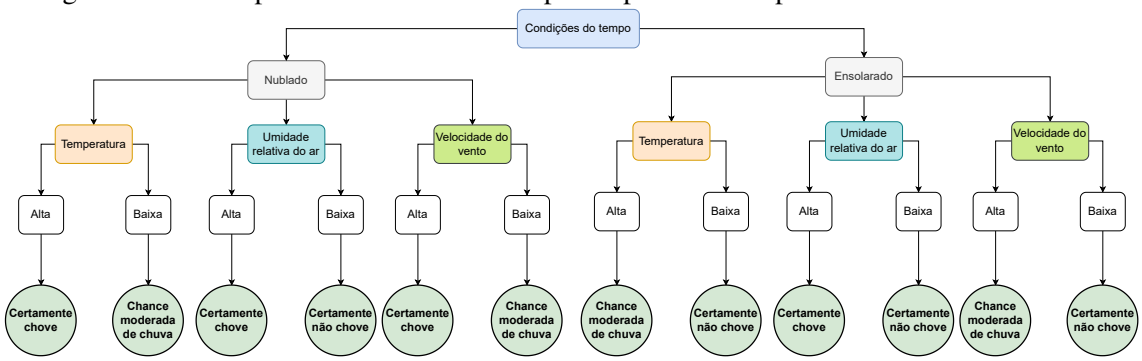
2.2.1.1 Árvores de decisão

O algoritmo de árvores de decisão é um algoritmo que consiste em inferir uma divisão dos dados de treinamento com base nos valores dos atributos disponíveis. O conhecimento é representado por meio de uma árvore, uma estrutura de dados capaz de representar uma sequência de decisões organizadas de forma hierárquica, a partir de um teste inicial denotado pelo nó raiz da árvore. Cada decisão leva a um novo teste ou a uma

classificação final para a instância. As folhas da árvore representam o resultado da classificação, e cada ramo é um teste realizada pelo algoritmo sobre um determinado atributo, visando classificar a instância. Na Figura 2.3, apresenta-se um exemplo de árvore de decisão, onde a porcentagem de umidade relativa do ar e a característica de nuvens claras ou escuras são utilizadas para identificar diferentes classes para as chances de chuva.

Algoritmos de árvores de decisão são fáceis de entender e interpretar, possuem uma complexidade logarítmica e conseguem lidar tanto com dados categóricos como numéricos (SANI; LEI; NEAGU, 2018). Entretanto, esses algoritmos podem ser instáveis, uma vez que pequenas variações nos dados podem resultar em uma árvore completamente diferente. Além disso, os algoritmos de árvores de decisão são baseados em algoritmos heurísticos, ou seja, as decisões tomadas não podem garantir que o resultado seja o globalmente ótimo. A regra de divisão das árvores de decisão, como citado anteriormente, segue uma regra heurística de olhar sempre para o próximo passo a ser dado. Assim, para cada decisão feita, o sistema escolhe o teste que maximiza ou minimiza a função heurística, classificando os algoritmos de árvores de decisão como gulosos.

Figura 2.3 – Exemplo de árvore de decisão para o problema de prever as chances de chover.



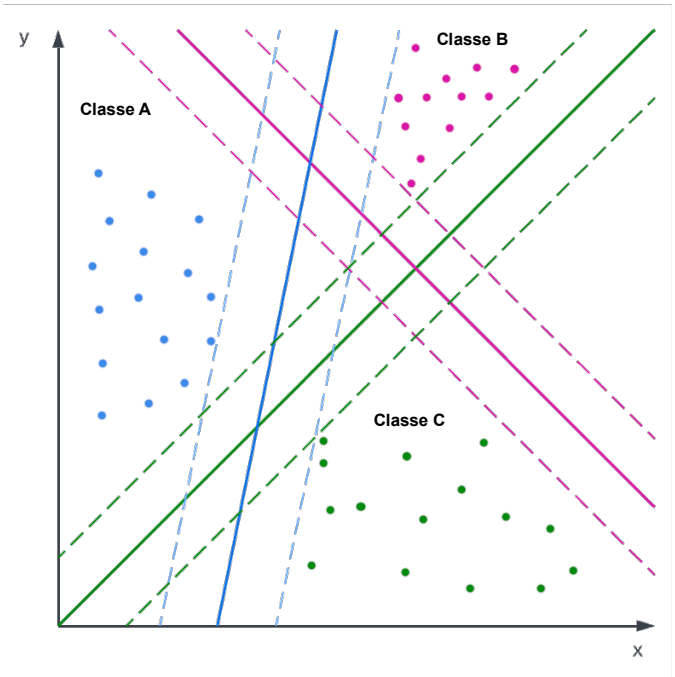
Fonte: O Autor

2.2.1.2 Máquinas de vetores de suporte

Máquinas de vetores de suporte (SVMs) são algoritmos de aprendizado supervisionado desenvolvidos originalmente para classificação binária, que visam encontrar o melhor hiperplano que divide duas classes. Este hiperplano é encontrado a partir dos vetores de suporte, os quais denotam os pontos de classes distintas que são mais próximos entre si no espaço de entrada. Além disso, um passo importante do algoritmo é maximizar a distância entre o hiperplano e os vetores de suporte –distância comumente chamada de "margem". Ainda que a formulação clássica do algoritmo seja de um classificador linear

binário, utilizando abordagens heurísticas como as *One-vs-one* ou *One-vs-rest*, é possível executar a classificação multiclasse utilizando SVMs (MAYORAZ; ALPAYDIN, 2006). O método heurístico utilizado nesse trabalho para executar a classificação multiclasse no SVM foi o *One-vs-one*, que consiste em quebrar o problema de classificação multiclasse em vários subproblemas de classificação binária. Um resultado deste processo pode ser visualizado na Figura 2.4.

Figura 2.4 – Exemplo de SVM com hiperplanos e suas respectivas margens para um problema de classificação multiclasse usando a abordagem *one-vs-one*.



Fonte: O Autor

Outra característica relevante de SVMs, que visa aumentar seu poder de modelagem de problema, é a utilização de funções de kernel para transformação o de dados. Ou seja, SVMs são modelos que se utilizam destas funções para projetar os dados para diferentes espaços de dimensão, usualmente visando aumentar o número de dimensões de modo a tornar problemas complexos mais fáceis de solucionar. Para exemplificar, basicamente essa função de kernel transforma problemas não lineares em problemas lineares, entretanto com um número maior de espaços de dimensão, assim facilitando o problema. Desta forma, a aplicação do algoritmo envolve a seleção da função de kernel usada para a transformação dos dados, sendo funções bem utilizadas as lineares, polinomiais e radiais. Cada kernel possui uma vantagem de uso distinta, auxiliando muito quando existem dados muito complexos. Entretanto, escolher um bom kernel para uma SVM é uma tarefa difícil, sendo um importante hiperparâmetro do algoritmo. Em alguns casos, especialmente bases muito grandes de dados, a aplicação de alguns tipos de kernel pode fazer com que

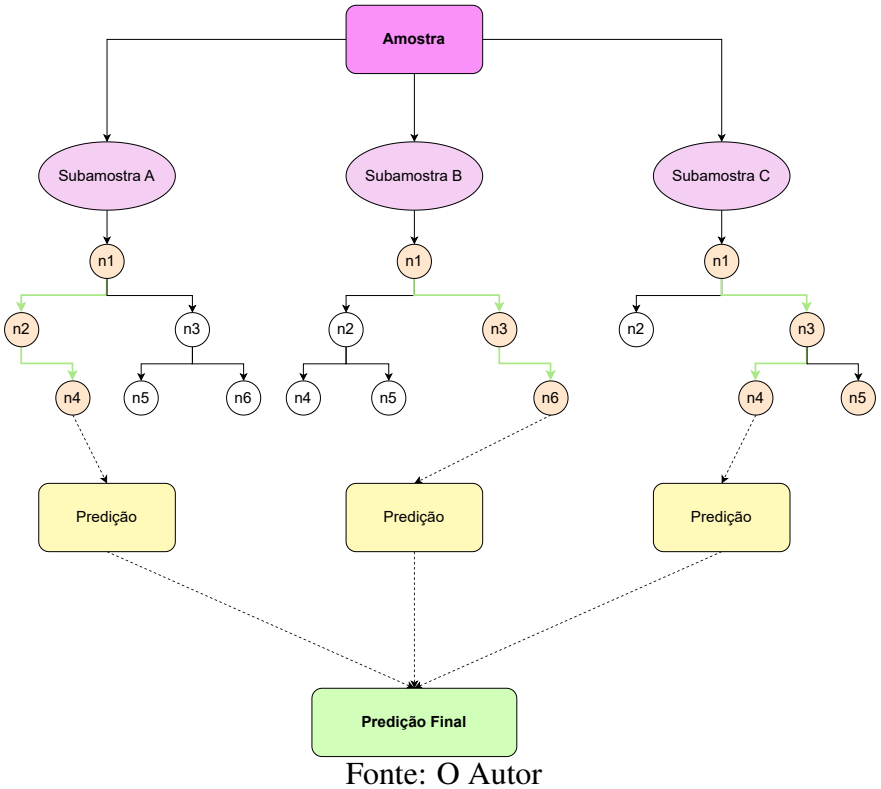
a complexidade e o tempo de execução do algoritmo cresça rapidamente.

2.2.1.3 Florestas aleatórias

O algoritmo de Florestas aleatórias utiliza-se do conceito de árvores de decisão, treinando um conjunto dessas árvores em tempo de treinamento e introduzindo comportamentos aleatórios neste processo a fim de gerar uma diversidade entre as árvores. A aleatoriedade do algoritmo está relacionada ao processo de procura pelo melhor atributo para fazer a partição dos nós nas árvores, sendo utilizado apenas um subconjunto aleatório dos atributos disponíveis. Além disso, o algoritmo usa a abordagem de *bootstrap*, para gerar amostras com reposição. Cada amostra aleatória será usada como base de treinamento para uma respectiva árvore.

Florestas aleatórias, quando usadas para classificação, têm como saída a classe selecionada pela maioria das árvores de decisão – isto é, realiza uma agregação por votação majoritária. Sendo assim, o algoritmo de florestas aleatórias é um exemplo de algoritmo de aprendizado *ensemble*, que ao gerar e combinar diferentes árvores de decisão, visa obter predições com maior acurácia e estabilidade que o algoritmo de árvores de decisão.

Figura 2.5 – Exemplo do algoritmo de florestas aleatórias



Fonte: O Autor

O algoritmo de florestas aleatórias, além de ser simples e acessível, possui uma certa resistência natural ao *overfitting*, pois se existe uma quantidade significativa de árvo-

res na floresta o classificador dificilmente causa *overfitting* no modelo (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). O algoritmo possui uma alta precisão, sendo geralmente bem mais eficiente que o algoritmo de árvores de decisão.

2.2.1.4 Hiperparâmetros de algoritmos de aprendizado

Hiperparâmetros utilizados para controlar como os modelos de AM aprendem a partir dos dados, ajustando o funcionamento dos algoritmos a partir do ajuste do valor destes hiperparâmetros. Os hiperparâmetros dos modelos mudam conforme os algoritmos utilizados, e são tão importantes para o desenvolvimento de modelos preditivos, que uma etapa comum de todo processo de aprendizagem é a otimização desses hiperparâmetros. Uma das abordagens mais simples é denominada busca em grade (*Grid search*), que determina diferentes combinações de todos os hiperparâmetros, e treina e avalia o desempenho do algoritmo com cada uma destas combinações, retornando a combinação com o melhor desempenho entre todas elas.

Considerando os algoritmos revisados nas seções anteriores, listamos abaixo os principais hiperparâmetros envolvidos, com uma breve descrição de cada:

- **Árvores de decisão e Florestas Aleatórias**: dois importantes hiperparâmetros são o *max_depth*, que determina a profundidade máxima da árvore, e *min_samples_split*, que determina o número mínimo de amostras para considerar realizar a divisão das mesmas através da inclusão de um nó de este. Ambos hiperparâmetros permitem aplicar estratégias de pré-poda a fim de controlar a complexidade da árvore de decisão gerada.
- **Máquinas de vetores de suporte**: importantes hiperparâmetros do SVM são o tipo de Kernel aplicado, o parâmetro de regulação C relacionado à penalidade do termo de erro, e também o fator gamma, que determina a extensão da influência de um único exemplo de treinamento no ajuste da fronteira de decisão do modelo.

2.2.2 Avaliação de modelos preditivos

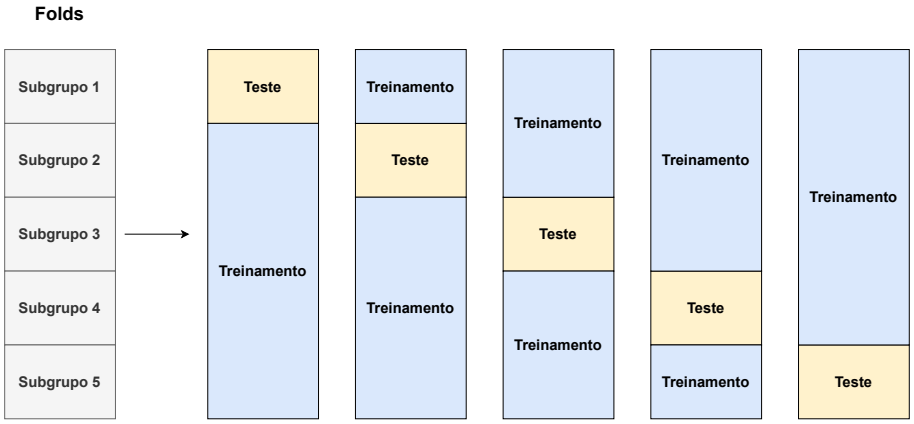
A avaliação do poder preditivo de modelos de classificação requer determinar uma forma de dividir os dados originais, a fim de permitir o treinamento e teste do modelo, bem como um conjunto de métricas de desempenho a serem empregadas para analisar a qualidade das predições realizadas. As seções a seguir revisarão os conceitos de vali-

dação cruzada *k*-fold aninhada e de diversas métricas de desempenho para classificação utilizadas neste trabalho.

2.2.2.1 Validação cruzada k-fold aninhada

A validação cruzada *k-fold* consiste em dividir os dados em *k* subconjuntos disjuntos, de mesmo tamanho, para depois escolher um deles para teste, enquanto os outros *k-1* restantes servirão para treinamento do modelo, como mostrado na Figura 2.6. O processo repete-se *k* vezes, alternando entre os subconjuntos de teste utilizados a cada iteração. No fim do processo, calcula-se uma estimativa de desempenho a partir da média e desvio padrão ao longo das *k* iterações

Figura 2.6 – Exemplo de validação cruzada *k-fold*



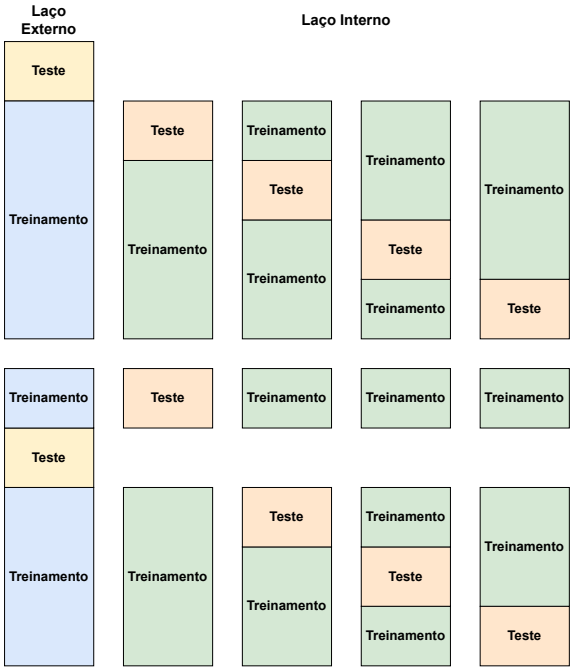
Fonte: O Autor

A estratégia de divisão de dados denominada validação cruzada *k-fold* aninhada nada mais é do que o procedimento tradicional de validação cruzada *k-fold*, onde se aninham o procedimento de otimização de hiperparâmetros com o procedimento de avaliação e seleção de modelo. Essa estratégia evita o viés otimista do desempenho do modelo e o *overfitting* dos dados de treino. No contexto deste trabalho, utilizamos uma validação cruzada aninhada estratificada, ou seja, os subconjuntos criados no processo de divisão de dados respeitam a distribuição original de classes.

Portanto, a estratégia completa da validação cruzada aninhada estratificada funciona por meio de dois laços, um interno e outro externo. No laço externo, dados entram em uma validação cruzada estratificada separando os dados em *k* subconjuntos. No laço interno, os *k-1* subconjuntos (os subconjuntos de treino do laço externo) entram em um procedimento de otimização de hiperparâmetros, que usa outra validação cruzada estratificada que separa esses dados de treino do laço externo em *k-1* subconjuntos novamente.

Por fim, para cada iteração do laço externo, o algoritmo selecionará o melhor modelo que vier do laço interno, e esse modelo será avaliado no subconjunto de teste do laço externo, assim como exemplificado na Figura 2.7. De acordo com trabalhos anteriores, a validação cruzada *k-fold* aninhada pode reduzir o viés de avaliação da validação cruzada *k-fold* quando a mesma é utilizada para otimização de hiperparâmetros e avaliação de modelos (VARMA; SIMON, 2006).

Figura 2.7 – Exemplo de execução de validação cruzada aninhada



Fonte: O Autor

2.2.2.2 Métricas de desempenho para classificação

A matriz de confusão é uma matriz que exibe a distribuição das instâncias conforme as classes reais e as classes previstas pelos modelos. A matriz de confusão mais simples possui dimensionalidade dois por dois, que compara os valores reais e preditos como na Figura 2.8. As principais métricas que uma matriz de confusão apresenta diretamente são as métricas de verdadeiros positivos, falsos positivos, verdadeiros negativos e falsos negativos, definidas a seguir:

- Verdadeiro positivo (**VP**): previsão positiva correta.
- Falso positivo (**FP**): Exemplo da classe negativa com previsão positiva incorreta.
- Verdadeiro negativo (**VN**): previsão negativa correta.

- Falso negativo (**FN**): Exemplo da classe positiva com previsão negativa incorreta.

Estas métricas podem, por sua vez, ser usadas para o cálculo de outros indicadores mais complexos de qualidade de predições de modelos. Assim, através da matriz de confusão podem ser tiradas diversas métricas de desempenho distintas. Entretanto, um dos seus principais objetivos da matriz de confusão é apresentar uma indicação de qualidade dos modelos de forma mais visual.

Figura 2.8 – Exemplo de matriz de confusão binária

		Classe verdadeira	
		Classe A	Classe B
Classe predita	Classe A	VP	FP
	Classe B	FN	VN

Fonte: O Autor

Para o caso de problemas com múltiplas classes, a matriz de confusão pode ser generalizada para uma dimensão *N* por *N,* onde *N* é o número de classes do problema.

Figura 2.9 – Exemplo de matriz de confusão para classificação multiclasse

		Classe verdadeira			
		Classe A	Classe B	Classe C	Classe D
Classe predita	Classe A	VP	E_{BA}	E_{CA}	E_{DA}
	Classe B	E_{AB}	VP	E_{CB}	E_{DB}
	Classe C	E_{AC}	E_{BC}	VP	E_{DC}
	Classe D	E_{AD}	E_{BD}	E_{CD}	VP

Fonte: O Autor

Exemplificando, na Figura 2.9, o E_{AB} são amostras da *Classe A* que foram incorretamente classificadas como da *Classe B*. Sendo assim, os falsos negativos da *Classe A* são determinados pela soma dos E_{AB}, E_{AC} e E_{AD}, que representam todas amostras da *Classe A* que foram incorretamente classificadas como B, C ou D.

$$FN_{\mathrm{A}} = E_{\mathrm{AB}} + E_{\mathrm{AC}} + E_{\mathrm{AD}} \tag{2.1}$$

Já o caso dos falsos positivos, pode ser definido a partir da soma dos elementos de uma linha. Usando a *Classe A* como exemplo novamente, temos como resultado a seguinte equação para determinação dos falsos negativos:

$$FP_{\mathrm{A}} = E_{\mathrm{BA}} + E_{\mathrm{CA}} + E_{\mathrm{DA}} \tag{2.2}$$

Com as informações contidas na matriz de confusão, podemos calcular todas as métricas de avaliação utilizadas nesse trabalho, as quais serão definidas a seguir.

- **Acurácia (ACC)**: Divisão entre amostras corretamente classificadas e todas as amostras classificadas. Ele fornece a acurácia geral do modelo, ou seja, a fração do total de amostras classificadas corretamente pelo modelo.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.3}$$

- **Sensibilidade**: Calculado a partir da razão entre o número de amostras positivas corretamente classificadas e o número total de amostras positivas. Seu principal objetivo é servir como uma taxa de verdadeiros positivos, mede a completude do modelo.

$$Sensibilidade = \frac{TP}{TP + FN} \tag{2.4}$$

- **Especificidade**: Razão entre o número de amostras negativas corretamente classificadas e o número total de amostras negativas. A especificidade mede a proporção de verdadeiros negativos que são identificados corretamente pelo modelo.

$$Especificidade = \frac{TN}{TN + FP} \tag{2.5}$$

- **Acurácia balanceada (ACC balanceada)**: Calculada como a soma da sensibilidade e especificidade dividida por dois. Tem um objetivo muito similar que a acurácia normal, entretanto é uma métrica mais adequada para dados com classes desbalanceadas.

$$ACC\ balanceada = \frac{Sensibilidade + Especificidade}{2} \tag{2.6}$$

- **Precisão**: Calculado como a divisão entre os verdadeiros positivos sobre a soma dos verdadeiros positivos e falsos positivos. Métrica utilizada para informar quantas das

instâncias preditas são de fato da classe positiva, mede a exatidão do modelo.

$$Precisão = \frac{TP}{TP+FP} \tag{2.7}$$

- **F1-Score**: Média harmônica ponderada entre precisão e sensibilidade, constrói uma relação de *tradeoff* entre a precisão e a sensibilidade do modelo.

$$F1\text{-}Score = 2 \times (\frac{Precisão \times Sensibilidade}{Precisão + Sensibilidade}) \tag{2.8}$$

- **Coeficiente de correlação de Matthews (MCC, de *Matthews correlation coefficient*)**: O MCC é usado como uma medida de qualidade de uma classificação. Geralmente considerado uma medida equilibrada, podendo ser usada mesmo em classes de tamanhos diferentes. Esse coeficiente é uma razão que fica entre 0 e 1, sendo calculado da seguinte maneira:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP) \times (TP+FN) \times (TN+FP) \times (TN+FN)}} \tag{2.9}$$

Ressaltamos que exceto pelas métricas de acurácia e acurácia balanceada, que são calculadas a partir do desempenho global do modelo, isto é, considerando todas as classes, as demais métricas são calculadas por classe em problemas multiclasse. Assim, realiza-se o cálculo de uma média entre os valores para a N classes a fim de obter um valor geral como desempenho do modelo. Dentre as diferentes abordagens possíveis para obter a média, destacamos a macro-média, calculada como a média aritmética entre a métrica extraída para as N classes possíveis.

Por fim, é importante salientar que quando existem altos valores de sensibilidade e baixos de precisão, os modelos têm a tendência de produzir falsos positivos. Por outro lado, quando existem altos valores de precisão e baixos valores de sensibilidade, a tendência é que o modelo produza falsos negativos (FAWCETT, 2004).

2.2.3 Estratégias de integração de dados ômicos

Grande parte das pesquisas sobre dados ômicos aplicados ao estudo de doenças humanas utilizam uma das três estratégias comuns para integração desses tipos de dados no desenvolvimento de modelos preditivos ou descritivos: (i) integração de estágio inicial, (ii) integração de estágio intermediário ou (iii) integração de estágio final. A moti-

vação para integrar dados ômicos, conforme mencionado no Capítulo 1, deve-se ao fato de que nenhuma tecnologia ômica individualmente consegue capturar toda a complexidade biológica da maioria das doenças humanas, por serem doenças multifatoriais (HASIN; SELDIN; LUSIS, 2017; KARCZEWSKI; SNYDER, 2018).

Algumas estratégias integram os dados todos em um único conjunto grande, logo após o pré-processamento, outras preferem utilizar-se de algoritmos para transformar os dados em outros tipos de representações, buscando facilitar a aprendizagem dos modelos. Entretanto, existem casos que existe preferencia em rodar os dados separadamente nos modelos e agregar o resultado deles com algum tipo de função, como médias, modas ou até mesmo por soma de porcentagens. Em alguns casos, vemos também combinações dessas estratégias, como agregar os dados depois do pré-processamento e executar algoritmos para transformação desses conjunto de dados em outras estruturas. A escolha dessas estratégias trazem ganhos importantes para a aprendizagem dos modelos, pois podem melhorar desempenho, velocidade, precisão, acurácia e diversas outras métricas.

2.2.3.1 Integração de estágio inicial

A mais simples das estratégias de integração, a integração de estágio inicial, consiste na concatenação de todos os conjuntos de dados ômicos em um único conjunto para ser aplicado no modelo de aprendizado de máquina. Nessa estratégia de integração de dados ômicos existe o aumento do número de variáveis (*i.e.,* atributos), consequentemente um conjunto de dados mais complexo é construído, o que acaba tornando o conjunto de dados de entrada mais ruidoso, aumentando a dimensionalidade do problema, deixando os dados sucessíveis a *"maldição da dimensionalidade"* [2]. Entretanto, suas principais vantagens são a fácil aplicabilidade, simplicidade de uso, e o fato ao utilizar a combinação de variáveis de ômicas, os modelos de aprendizado de máquina podem descobrir diferentes relações entre os dados, o que em alguns casos podem trazer muitos benefícios ao aprendizado (SPICKER et al., 2008).

2.2.3.2 Integração de estágio intermediário

O objetivo da estratégia de integração de estágio intermediário é utilizar-se de outras estruturas de dados para representação dos dados ômicos, focando em estruturas que tragam algum tipo de facilidade para o processo de aprendizagem de modelos. Geral-

[2] A maldição da dimensionalidade implica que para um dado tamanho de amostras, existe um número máximo de características a partir do qual o desempenho do classificador irá degradar, ao invés de melhorar.

mente são utilizados algoritmos para gerar essas outras formas de representação. Neste trabalho, alguns algoritmos de integração intermediária de dados ômicos foram utilizados, são eles:

- **NEMO**: *Neighborhood based Multi-Omics clustering* é um algoritmo de *clustering* de dados multi-ômicos cujo objetivo é ser simples, porém preciso (RAPPOPORT; SHAMIR, 2019). Sua principal inspiração vem de um algoritmo desenvolvido em 2014 chamado de *Similarity Network Fusion* (SNF) (WANG et al., 2014). O NEMO funciona em três passos simples:
  - **Passo um**: O algoritmo constrói para cada ômica uma matriz de similaridade entre pacientes ou amostras.
  - **Passo dois**: As matrizes dessas diferentes ômicas são integradas em uma matriz única.
  - **Passo três**: É feita a tarefa de agrupamento dessas redes construídas (*network clustering*).

  A entrada para o NEMO é um conjunto de matrizes de dados ômicos com mesmo número de amostras ou pacientes. A sua medida de similaridade, que compõe a principal atividade do passo um, é baseada na função de base radial (BUHMANN, 2003). Após a construção da matriz de similaridade (S_l) de cada uma das ômicas, começa a construção das matrizes de similaridades relativas (RS_l) para cada ômica, que se utilizam da matriz S_l para sua composição. A matriz RS_l mede as similaridades de cada amostra em relação a seus vizinhos mais próximos como uma probabilidade de transição entre amostras, ou seja, quanto maior a probabilidade de se mover entre amostras, maior sua similaridade. Por fim, calcula-se a matriz média de similaridade relativa (ARS_l), que nada mais é que uma mistura das distribuições da matriz RS_l. Com a matriz ARS_l, os grupos são calculados utilizando um algoritmo de *Spectral clustering* e o número de cada grupo é calculado como na equação 2.10, onde a são os autovalores da matriz ARS_l.

$$\arg\max_i (a_{i+1} - a_i) \times i \tag{2.10}$$

- **CIMLR**: *Cancer Integration via Multi-kernel Learning* é um algoritmo voltado para identificação de subtipos de câncer, o qual aprende a medir a similaridade entre cada par de amostras em um conjunto de dados multi-ômicos combinando kernels gaussianos por tipo de dados. Dessa forma, são construídas diversas repre-

sentações complementares dos dados de entrada. Inspirado no algoritmo *Single-cell Interpretation via Multi-kernel Learning* (SIMLR), o CIMLR não assume igual importância para cada tipo de dado e consegue incorporar completamente genomas e diversas dimensões para muitos tipos de dados. Algoritmos de aprendizado por Múltiplos kernels são um conjunto de métodos que aprendem por meio de uma combinação linear ou não linear de kernels. As principais características desses tipos de algoritmos são a capacidade de reduzir viés por meio da seleção de kernels e hiperparâmetros de um conjunto maior de kernels. Esses algoritmos são bons para problemas que possuem diferentes fontes de dados como sons, imagens ou vídeos, pois para cada tipo de dado, pode-se construir kernels diferentes que podem ser combinados posteriormente.

Entretanto, os principais problemas dessa estratégia estão na dificuldade de aplicá-la, visto que sua complexidade pode ser tão grande quanto se queira, pois tudo depende dos algoritmos escolhidos para fazer a transformação dos dados. Outro grande problema é a possível perda de algumas informações devido a generalizações ou reinterpretações feitas nos dados durante as transformações, que podem dificultar no treinamento dos modelos.

2.2.3.3 Integração de estágio final

A estratégia de integração de estágio final consiste em aplicar um modelo de aprendizado de máquina para cada conjunto de dados ômicos separadamente, e posteriormente utilizar uma função que combine as predições realizadas pelos diferentes modelos. Por exemplo, tendo como base os dados ômicos utilizados no presente trabalho (CNV, mRNA, miRNA e metilação de DNA), esta estratégia usaria os dados de CNV como entrada do Modelo 1, os dados de mRNA como entrada do Modelo 2, os dados de miRNA como entrada do Modelo 3 e os dados de metilação de DNA como entrada Modelo 4. Por fim, as predições feitas por cada um dos modelos seriam unidas com uma função pré-definida e retornadas como resultado principal. O problema com essa estratégia está na falha em capturar relações que possam existir entre ômicas, uma vez que o aprendizado é executado separadamente para cada tipo de dado. Ou seja, em nenhum momento os modelos dividem conhecimento e assim não conseguem usufruir de possíveis complementariedades entre os dados.

3 TRABALHOS RELACIONADOS

Estratégias de integração de dados ômicos são o ponto central de discussão de muitos artigos atuais focados no estudo de doenças complexas. Esta abordagem possui alta popularidade principalmente pela importância da estratégia de integração de dados nos resultados da execução de qualquer modelo de aprendizado de máquina. Nas literaturas atuais, podemos ver um esforço muito grande na tentativa de explicar benefícios e malefícios de cada tipo de estratégia de integração, bem como para classificar essas estratégias.

Um bom exemplo é o trabalho de Picard et al. (2021), que revisa diversas pesquisas relacionadas ao tema com intuito de explicar e classificar, de forma geral, cada uma das estratégias. Outros trabalhos focam mais em um tipo de estratégia, fazendo testes e tentando explicar minúcias relacionadas aos ganhos e perdas de utilizar aquela estratégia. Como exemplo, citamos o trabalho de Duan et al. (2021), o qual foca exclusivamente na estratégia de estágio intermediário, mais especificamente, comparando experimentalmente algoritmos para integração de dados ômicos ao nível intermediário. Adicionalmente, ressaltamos também o trabalho Reel et al. (2021), um importante estudo das principais formas de aplicação de aprendizado de máquina para análise de dados ômicos, com explicações bem detalhadas e ilustrativas sobre a importância do AM na medicina.

No trabalho de Duan et al. (2021), citado anteriormente, o objetivo dos autores é construir uma análise comparativa minuciosa de diversos métodos de integração de dados ômicos para estratégia de estágio intermediário na tarefa de atribuir os subtipos de cânceres. Os autores testaram dez métodos de integração de dados ômicos diferentes, entre eles os algoritmos NEMO e CIMLR usados no presente trabalho. Para cada método, três métricas foram avaliadas, a acurácia, robustez e eficiência computacional, e os algoritmos que obtiveram melhores resultados foram NEMO e SNF, ambos algoritmos baseados em transformação usando matrizes de similaridade.

Outro trabalho relacionado a integração de estágio intermediário é o de Cai et al. (2022), que apontou ótimos resultados em algoritmos que usam análise de correlação canônica (CCA, de *Canonical Correlation Analysis*) e análise de componentes principais (PCA, de *Principal Component Analysis*), onde o primeiro deles apresentou acurácia de 80%. No trabalho de Rappoport and Shamir (2019), que apresentou o algoritmo NEMO, foi constatado que além do algoritmo ser mais simples, ele executa em muitos casos na

metade do tempo, tendo resultados muito parecidos com os demais que faziam parte do grupo de comparação. Outro importante trabalho é o de Wang et al. (2014), que propôs o algoritmo de *Similarity Network Fusion* (SNF) avaliado em diversos outros estudos, além de ter sido usado como inspiração principal para o desenvolvimento do NEMO e vários outros métodos.

No trabalho de Rappoport and Shamir (2018) sobre algoritmos de *Clustering*, os autores mencionam problemas devido a dimensionalidade dos dados relacionados à estratégia de integração de estágio inicial. Além disso, em Picard et al. (2021), também são apontados os mesmos problemas de dimensionalidade, apesar de apresentar também vantagens claras como a fácil aplicabilidade e fácil criação de relação das ômicas pelos modelos treinados com esta abordagem. Nesse trabalho, os autores também apontaram algumas tendências a melhores resultados na estratégia de estágio intermediário, assim como ressaltaram benefícios e malefícios da estratégia de estágio final. Entretanto, no estudo de Reel et al. (2021), são apresentadas também desvantagens da estratégia de estágio intermediário, principalmente em algoritmos que utilizam Kernels devido sua demanda computacional. No trabalho de Zitnik et al. (2019) são feitas também análises sobre de que forma os dados são interpretados em cada uma das estratégias, bem como desafios encontrados. Além disso, os autores reforçam a ideia de que cada estratégia deve ser aplicada conforme o contexto e o problema para se obter melhores resultados.

Como mencionado anteriormente, muitos trabalhos explicam de forma geral cada uma das estratégias de integração de dados ômicos, trazendo aspectos teóricos que possam representar vantagens ou desvantagens de cada estratégia. Outros trabalhos focam em realizar uma análise experimental centrada em diferentes métodos para uma mesma estratégia de integração. Entretanto, observamos a falta de estudos que realizam a proposta desse trabalho, de realizar uma avaliação experimental geral, uma avaliação geral de cada uma das estratégias comparadas entre si, verificando ganhos e perdas e validando o impacto da escolha da estratégia para o resultado dos modelos. Na Tabela 3.1 temos uma comparação entre os trabalhos relacionados, se eles abordam ou não alguma estratégia e

Tabela 3.1 – Tabela de comparação entre os trabalhos

	estratégias			
Trabalho	**Estágio Inicial**	**Estágio Intermediário**	**Estágio final**	**Comparação experimental entre estratégias**
Picard et al. (2021)	Sim	Sim	Sim	Não
Duan et al. (2021)	Não	Sim	Não	Não
Rappoport and Shamir (2018)	Sim	Não	Sim	Não
Reel et al. (2021)	Não	Sim	Não	Não
Zitnik et al. (2019)	Sim	Sim	Sim	Não
Cai et al. (2022)	Não	Sim	Não	Não
Wang et al. (2014)	Não	Sim	Não	Não

também se eles comparam as estratégias de forma direta. Assim, notamos que nenhum dos trabalhos mencionados realiza uma comparação experimental entre as três estratégias de integração de dados mencionadas.

4 METODOLOGIA

Baseando-se nos trabalhos relacionados, foi desenvolvida uma metodologia englobando três pipelines de execução, um para cada tipo de estratégia de integração de dados ômicos. Os dados utilizados na execução dos pipelines foram coletados do trabalho Duan et al. (2021), focado na comparação de estratégias de integração intermediária em tarefa de agrupamento para análise de subtipos de tumor. No trabalho original, os dados passaram por um pré-processamento bem rigoroso seguindo protocolos padrões da Bioinformática para preparação de dados ômicos. Estas etapas de pré-processamento podem ser consultadas em detalhes na referência do trabalho original (DUAN et al., 2021). Portanto, no escopo deste trabalho, o pré-processamento dos dados foi simplificado.

Para o desenvolvimento dos pipelines foram utilizados a linguagem Python com bibliotecas de aprendizado de máquina do Scikit-learn. Os algoritmos de integração de dados ômicos que fazem parte da estratégia de estágio intermediário foram executados separadamente, pois a maioria deles era desenvolvido na linguagem R. As seções a seguir descreverão todos os detalhes metodológicos do trabalho, importantes para avaliação dos experimentos e resultados.

4.1 Coleta e pré-processamento de dados

Como dito anteriormente, os dados usados nesse trabalho fazem parte de um fração dos dados gerados e utilizados no artigo de Duan et al. (2021). Dentre os nove tipos de câncer com dados pré-processados disponibilizados pelos autores, foram selecionados apenas dois tipos: os dados de câncer de mama (BRCA) e de câncer de cólon (COAD). O conjunto de dados tem natureza multi-ômica e contém quatro tipos de atributos ômicos para o mesmo conjunto de amostras (isto é, instâncias): CNV, Metilação de DNA, expressão de mRNA e expressão de miRNA. No trabalho original, os dados passaram por um minucioso tratamento para pré-processamento dos mesmos. Em suma, os dados selecionados passaram, inicialmente, por um processo de filtragem, retirando todos os atributos e amostras que possuíam mais de 20% dos valores ausentes. Posteriormente, foram selecionadas amostras do mesmo grupo de pacientes e para imputação dos dados faltantes, foi utilizado o algoritmo de *K-nearest neighbor* (KNN). Por fim, foram removidos os ruí-

dos causados pela variação nos grupos de amostras, os chamados *Batch Effects* [1]. Este processo é descrito em mais detalhes no artigo original (DUAN et al., 2021).

Sobre os atributos ômicos utilizados no trabalho, cada um deles é fruto de diversas análises, técnicas e métodos para suas medições, como exemplo temos:

- **Dados de CNV**: Os dados de CNV utilizados no trabalho são frutos de diversos processos e técnicas utilizando o sequenciamento de DNA e microarranjos para identificar regiões de genes que são repetidas e inferir o número de cópias dessas repetições. Como exemplo, temos o gene humano A1BG (de *Alpha-1-Beta-Glycoprotein*), que pode apresentar alterações no número de cópias em diferentes indivíduos. A análise de CNV em A1BG pode servir como base para o estudoda relação entre essa variação genética e doenças como câncer.
- **Dados de Metilação de DNA**: Os dados de Metilação de DNA utilizados no trabalho passam por diversos passos para sua medição. Um exemplo desses passos são as técnicas de microarranjos usados para medir os níveis de metilação do DNA em locais específicos do genoma, onde a saída do fluxo desse processo são os valores beta, que representam o nível de metilação em cada região genômica. Exemplo dessa medição de metilação de DNA é o gene humano AACS (de *acetoacetyl-coenzyme A synthetase*), que ao passar pelo processo de metilação altera seus valores de expressão.
- **Dados de mRNA**: Os dados de mRNA passam por um grande processo de análise quantitativa e qualitativa que geram diversos atributos. Dentre essas medições está o nível de expressão dos mRNAs. Por exemplo, temos mRNAs que codificam proteínas como A2M (de *Alpha-2-Macroglobulin*).
- **Dados de miRNA**: Os dados de miRNA usados no trabalho também passam por muitas análises quantitativas e qualitativas que dão origem aos valores dos seus atributos. Dentre essas análises está o sequenciamento de RNA, a partir do qual é possível obter valores da expressão de vários miRNAs distintos como o *hsa-let-7a-1* [2], um miRNA que se expressa em uma variedade de tecidos humanos.

No desenvolvimento do nosso trabalho, foi adicionada uma etapa extra de pré-processamento, referente à normalização dos dados com o método de *z-scores* para elimi-

[1] *Batch effects* ocorrem quando fatores não biológicos em um experimento causam mudanças nos dados produzidos pelo experimento.

[2] O nome *hsa-let-7a-1* é derivado de uma combinação de vários elementos. "hsa"significa Homo sapiens; "let-7"se refere a uma família de miRNAs; "a"refere-se a subfamília; "1"indica que é o primeiro membro identificado da subfamília.

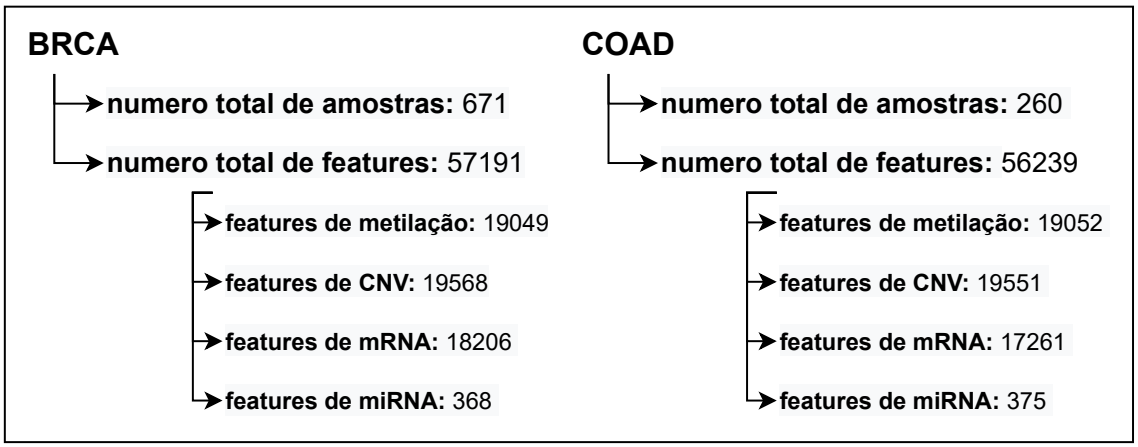
nar as diferenças entre ômicas devido às diferentes escalas. A normalização por Z-score consiste em transformar os dados, removendo a média e escalando para a variância da unidade, conforme a equação abaixo

$$Z = \frac{X - U}{S} \tag{4.1}$$

onde o X é a amosta, U é a média das amostras de treinamento ou zero e S é o desvio padrão das amostras de treinamento ou um. Essa normalização é necessária quando dados possuem escalas muito discrepantes. Portanto, buscando evitar problemas com o treinamento dos modelos de AM neste trabalho, escolhemos aplicar essa normalização em todos os dados de entrada de cada uma das estratégias de integração de dados ômicos.

A Figura 4.1 apresenta um resumo da quantidade de amostras e de atributos contida nos dados utilizados.

Figura 4.1 – Sumarização do número de amostras e atributos por tipo de câncer



Fonte: O Autor

Os dados que escolhemos para esse trabalho foram selecionados por alguns motivos: (i) a origem, pois foram dados originalmente gerados pelo projeto *The Cancer Genome Atlas* (TCGA), referência na área de genômica do câncer; (ii) o pré-processamento efetuado, o qual conforme foi explicado anteriormente, foi bastante minucioso e claramente descrito pelos autores; (iii) a disponibilidade de anotação clínica por amostra, a qual incluía o subtipo tumoral, dentre outras informações. Estas informações a respeito de subtipo tumoral foram usadas no treinamento dos modelos, sendo consideradas as classes verdadeiras (isto é, rótulo real) neste trabalho. Portanto, além de pré-processados, os dados de câncer de mama (BRCA) e câncer de cólon (COAD) já vinham com as classificações do TCGA como referência, facilitando muito o uso desses dados no treinamento de modelos de AM.

A Tabela 4.1 mostra a distribuição de exemplos por classe para cada tipo de câncer analisado no trabalho. Podemos ver que existe um grande desbalanceamento de dados, principalmente nos dados de COAD. Conforme será detalhado posteriormente, este desbalanceamento serviu como inspiração para os experimentos com os dados de COAD sem o subtipo de câncer POLE, para avaliarmos o impacto do desbalanceamento desses dados.

Tabela 4.1 – Numero de instâncias de cada classes dos dados utilizados no trabalho

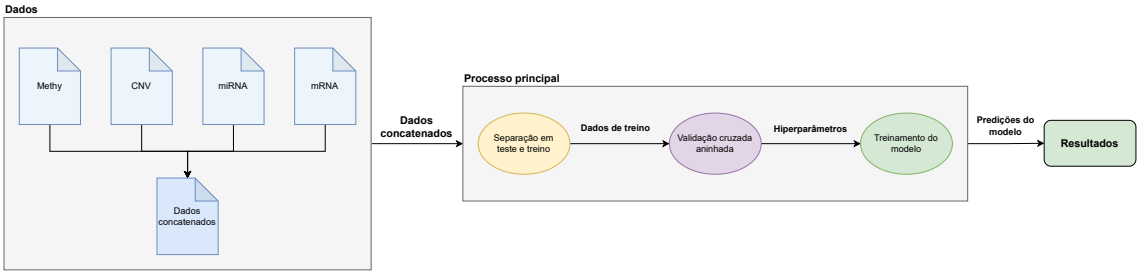
BRCA		COAD	
Subtipo de câncer	**Número de instâncias**	**Subtipo de câncer**	**Número de instâncias**
LumA	353	CIN	174
LumB	132	MSI	48
Basal	113	GS	34
Her2	42	POLE	4
Normal	31		

4.2 Aplicação das estratégias de integração de dados

A primeira estratégia de integração de dados desenvolvida e aplicada foi a estratégia de estágio inicial, que consiste na concatenação dos dados das quatro ômicas diferentes para cada um dos dois tipos de cânceres selecionados. Com isso, construímos um conjunto de dados concatenados com um total de 671 instâncias com 57191 atributos para o câncer BRCA e 260 instâncias com 56239 atributos para o COAD. Depois da construção dos dados de entrada, inicia-se o processo principal da pipeline de execução, que consiste uma divisão dos dados em treinamento e teste, na validação cruzada aninhada que otimiza os hiperparâmetros e seleciona os melhores modelos, e com o treinamento do modelo final para análise nos dados de teste. Estas etapas referentes ao processo principal são comuns a todas as estratégias de integração de dados e serão detalhadas na Seção 4.3. Na Figura 4.2 podemos ver com mais clareza o processo principal da pipeline de execução e os dados.

Na estratégia de interação de estágio intermediário, o processo geral continua bastante semelhante, entretanto, a execução de alguns métodos de integração de dados ômicos foram incluídos no início da pipeline. A partir do resultado da execução desses métodos, retiramos os dados de entrada para o pipeline de execução principal de aprendizado de máquina seguindo os passos que serão descritos na Seção 4.3. Os dois algoritmos que mais se destacaram em trabalhos anteriores foram o NEMO e o CIMLR. Ambos usam diferentes estratégias para construir relações de similaridade para integrar seus dados, o

Figura 4.2 – Pipeline da estratégia de estágio inicial



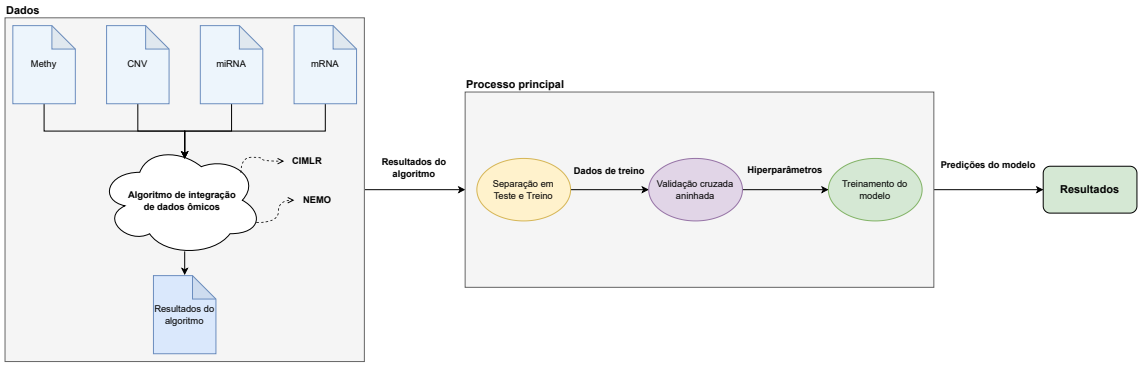
Fonte: O Autor

primeiro por meio de grafos e o segundo por meio de combinações de Kernels. Ambos apresentaram bons desempenhos e uma fácil aplicabilidade, comparado com os demais algoritmos. Na Figura 4.3, podemos observar que a grande mudança no pipeline está no início da execução, que passa por um processo a mais de tratamento de dados com os algoritmos de transformação intermediária.

Nesta estratégia, para cada resultado obtido com um algoritmo de transformação utilizado, foi gerado um ou mais conjunto de dados para treinamento dos modelos de AM. No caso do CIMLR, foram construídos os seguintes conjuntos de dados: (i) *clusters* de cada instância concatenados com seus respectivos centros; (ii) apenas *clusters* de cada instância; (iii) Os *f-values*, os quais são os resultados de uma rede de difusão executada no algoritmo; (iv) Os *y-data* que se refere, exclusivamente, a dados de resultados do k-means executado; (v) todos esses dados anteriores concatenados. No caso do NEMO, ele apenas retorna os *clusters* e uma matriz de similaridade, onde apenas utilizamos os *clusters* devido à complexidade de utilizar a matriz de similaridade para treinamento dos modelos. Para escolha dos hiperparâmetros desses algoritmos visamos seguir o padrão de execução proposto pelos autores dos algoritmos. Conforme o número de ômicas usadas ou número de subtipos de câncer existentes, um valor para o parâmetro K ou número de *clusters* era escolhido. Os parâmetros de entrada utilizados no algoritmo NEMO e CIMLR foram os padrões recomendados para sua execução de cada um deles, *i.e.,* para o NEMO, uma lista com as quatro ômicas (CNV, mRNA, miRNA e metilação de DNA), numero de *clusteres* igual ao número de subtipos de câncer existentes nos dados (no total 5 para BRCA e 4 para COAD) considerando os 50 vizinhos mais próximos. Para o CIMLR foi apenas o número de *clusteres* igual ao número de subtipos de câncer existentes, apesar do algoritmo CIMLR possuir já uma função para estimar o número ideal de *clusteres* ela não foi usada.

Por fim, na estratégia de estágio final, embora o processo principal seja o mesmo

Figura 4.3 – Pipeline da estratégia de estágio intermediário



Fonte: O Autor

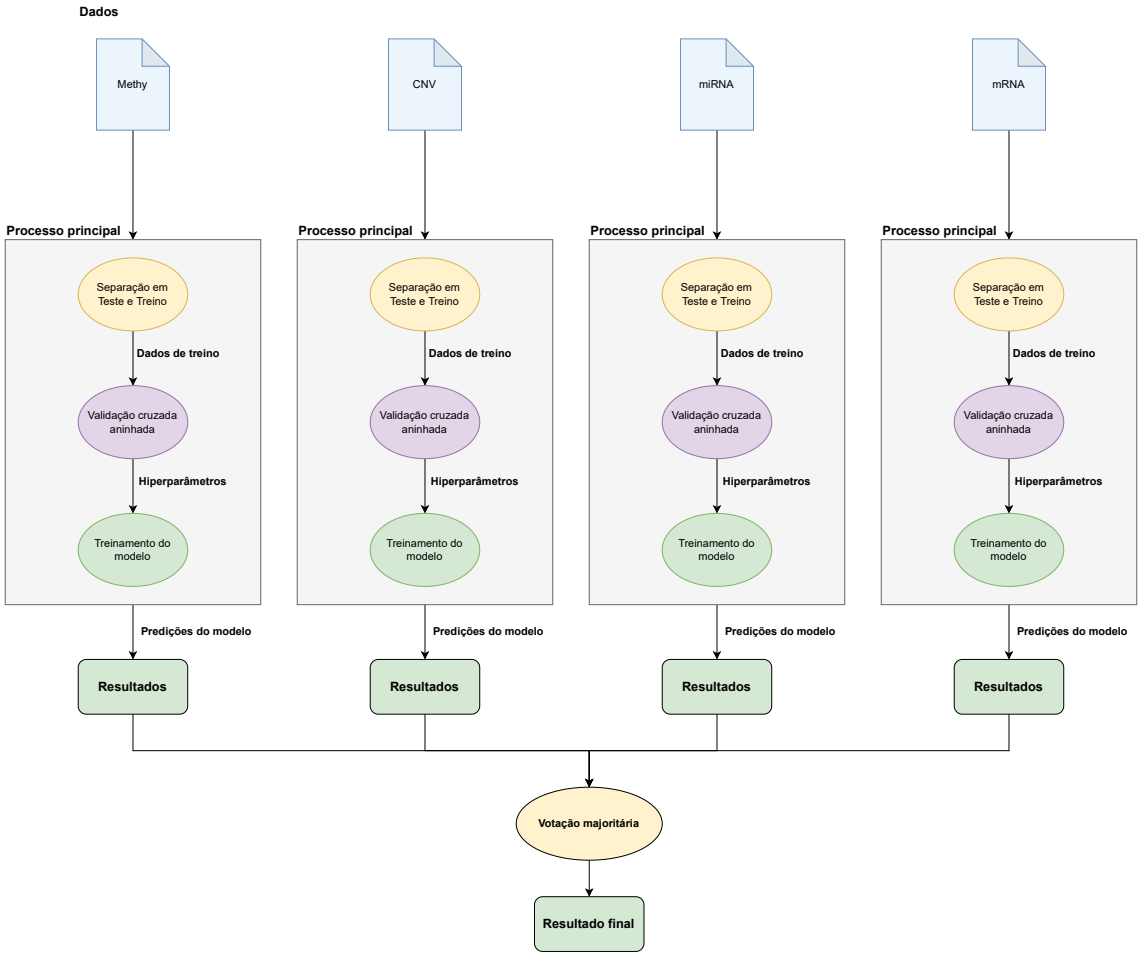
(a ser descrito na Seção 4.3), os dados de entrada para o treinamento não são concatenados nem extraídos da execução de algoritmos intermediários. Nesta estratégia, os dados de cada tipo de ômica são utilizados separadamente para treinamento de N modelos (no nosso caso, $N = 4$), gerando resultados individuais. Ao fim da execução, após obter cada resultado separadamente par cada ômica, executamos o método de votação majoritária entre os resultados das ômicas para assim combinar as classificações obtendo um resultado final. Essa execução segue exatamente os passos ilustrados na Figura 4.4.

4.3 Desenvolvimento do processo principal

Além de detalhes relacionados à construção dos dados de treinamento por meio das três estratégias de integração de dados descritas na seção anterior, é importante esclarecer os passos do processo principal de treinamento e validação dos modelos de AM. Conforme mostrado nas Figuras 4.2, 4.3, 4.4, estes passos são fundamentalmente os mesmos, independente da estratégia de integração de dados. O que muda é a natureza do conjunto de dados utilizado como entrada.

O processo principal é detalhado na Figura 4.5. Como pode ser observado, ele inicia com a divisão dos dados em conjunto de treinamento e teste na proporção 60% e 40%. O conjunto de treinamento será usado no passo seguinte, o qual consiste na aplicação da validação cruzada aninhada que visa otimizar os hiperparâmetros do modelo e avaliar o desempenho preditivo resultante. Com os melhores hiperparâmetros selecionados na validação cruzada aninhada, realizamos o treinamento de um modelo final, o qual é aplicado para predição dos dados de teste. Ao final do processo, extraímos as métricas de avaliação de desempenho definidas na Seção 2.2.2.2 tanto da validação cruzada quanto

Figura 4.4 – Pipeline da estratégia de estágio final

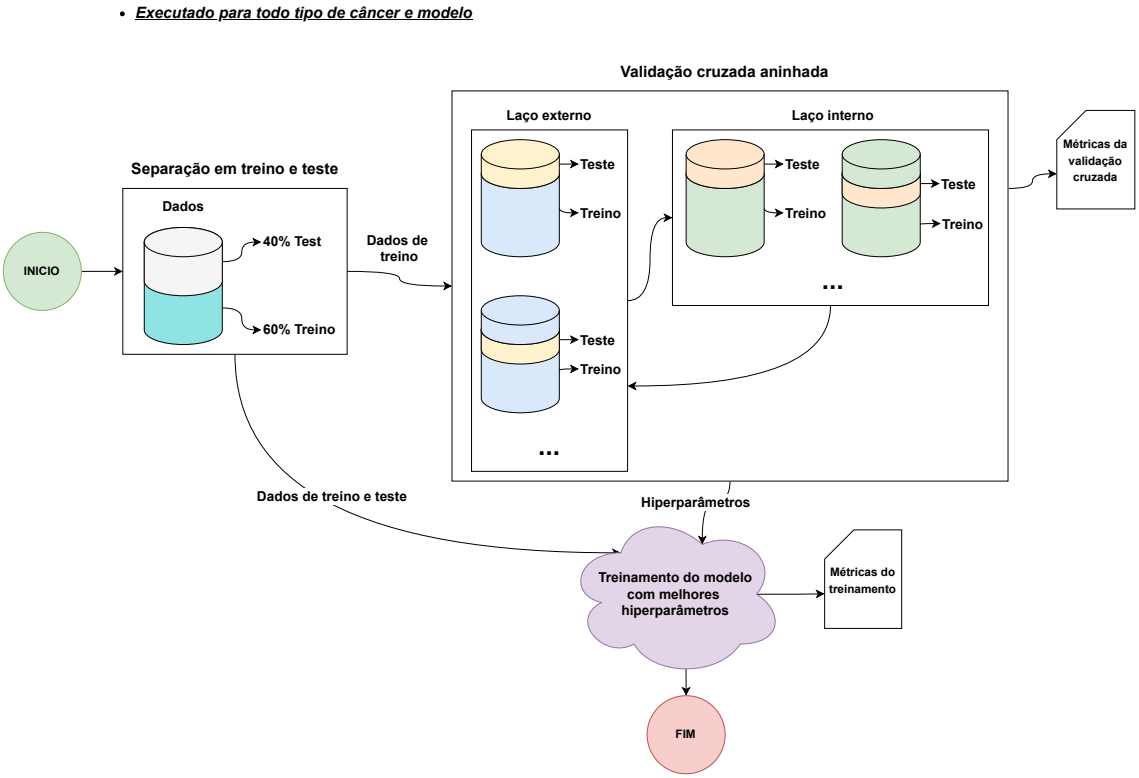


Fonte: O Autor

da predição nos dados do teste. As seções a seguir irão detalhar subetapas do processo principal.

A decisão pelo método de validação cruzada aninhada para treinamento e validação dos modelos se deu devido à maior confiabilidade deste método para avaliação robusta de algoritmos de AM, tentando evitar tanto o viés quanto o possível *overfitting* causado nas validações cruzadas simples (CAWLEY; TALBOT, 2010). Para implementação desse algoritmo utilizamos duas validações cruzadas estratificadas, variando a divisão dos dados de treino em algumas combinações. Junto a essas validações estratificadas, foi utilizada a técnica de validação cruzada com *Grid Search* para criar um modelo para cada combinação de hiperparâmetros especificados e aninhar as outras duas validações cruzadas estratificadas.

Figura 4.5 – Processo principal da pipeline de execução.



Fonte: O Autor

4.3.1 Seleção dos algoritmos de aprendizado supervisionado

Durante o desenvolvimento do trabalho, visamos explorar diferentes algoritmos de AM a fim de permitir analisar o efeito da escolha da estratégia de integração de dados com algoritmos contendo diferentes vieses de aprendizado. O algoritmo de árvores de decisão foi a primeira escolha feita. Avaliaremos o desempenho do algoritmo, analisando o quanto a instabilidade desse método pode causar perdas durante o processo de aprendizagem utilizando dados ômicos. Logo após a escolha de usar árvores de decisão, exploramos como se desempenharia um *ensemble* de árvores de decisão, quais melhorias o método de florestas aleatórias poderia trazer para a identificação de subtipos de tumor, e o quanto isso impacta no resultado e na escolha do método de integração de dados. Por último, escolhemos o modelo de SVM, tendo em vista que possui um viés de aprendizado bastante diferente dos demais, e é um algoritmo normalmente associado com uma alta precisão, portanto um bom método para incluir em uma análise comparativa.

4.3.2 Treinamento dos modelos e otimização de hiperparâmetros

Para todos os algoritmos de classificação selecionados para o trabalho, utilizamos a biblioteca *scikit-learn*, variando seus hiperparâmetros como mostrado na Tabela 4.2. A escolha dos hiperparâmetros dos algoritmos selecionados foi feita buscando explorar a relevância de hiperparâmetros importantes, tentando extrair bons resultados de predição.

Nos modelos de árvores de decisão e florestas aleatórias, variamos a profundidade máxima das árvores, e o número mínimo de amostras para separar nodos internos. No modelo SVM, sendo um modelo mais sensível à configuração de hiperparâmetros, variamos mais hiperparâmetros, sendo eles o tipo de Kernel, o parâmetro de regulação C e o fator gamma. Na Tabela 4.2 podemos ver cada um dos hiperparâmetros utilizados em cada classificador.

Tabela 4.2 – Tabela com hiperparâmetros aplicados aos modelos

Classificadores	**Parâmetros**
Árvores de Decisão (Decision Trees)	max_depth:[None, 3, 4, 5,6,7,8], min_samples_split: [4, 5, 6, 7, 8]
Florestas Aleatórias (Random Forests)	max_depth: [None, 3, 4, 5,6,7,8], min_samples_split: [4, 5, 6, 7, 8]
Maquina de vetores de suporte (SVM)	C: [0.1,1, 10, 100], gamma: [1,0.1,0.01,0.001], kernel: ['rbf', 'poly', 'linear']

A validação dos algoritmos foi realizada com a validação cruzada aninhada. Dado que este procedimento gera múltiplos modelos treinados a partir de variações na configuração dos hiperparâmetros, cada configuração associada a um desempenho preditivo, adotamos duas estratégias para determinar a melhor configuração possível de hiperparâmetros para gerar o modelo final, a ser aplicado nos dados de teste. São elas:

- **Hiperparâmetros com maior desempenho de validação**: dessa forma, a validação cruzada aninhada é executada normalmente, e a combinação de hiperparâmetros associada ao melhor desempenho médio do F1-Score macro entre as múltiplas iterações da validação cruzada é selecionada, independente da frequência com que ocorrem.
- **Hiperparâmetros que mais se repetem**: nesse caso, após a execução da validação cruzada aninhada, conta-se quantas vezes cada combinação de hiperparâmetros se repete entre os resultados do processo de otimização de hiperparâmetros. A configuração mais frequente entre estes resultados é escolhida para treinamento do modelo final.

Em ambas as estratégias, são avaliados 10 resultados de configuração de hiperparâmetros em razão do número de folds usados na validação cruzada aninhada.

Os valores de hiperparâmetros em colchetes na Tabela 4.2 são a lista de hiper-

parâmetros passados para a validação cruzada aninhada que executa efetuando todas as combinações de hiperparâmetros possíveis. Após execução, o resultado é uma lista das dez combinações de hiperparâmetros que obtiveram os melhores resultados, e assim segue para a estratégia de hiperparâmetros explicada anteriormente. Por fim, com os hiperparâmetros já selecionados, rodam-se os modelos novamente, assim evitando viés e *overfitting*. Na Figura 4.5 temos de forma mais ilustrativa todos os passos do processo principal de execução.

4.3.3 Tratamento do desbalanceamento de dados

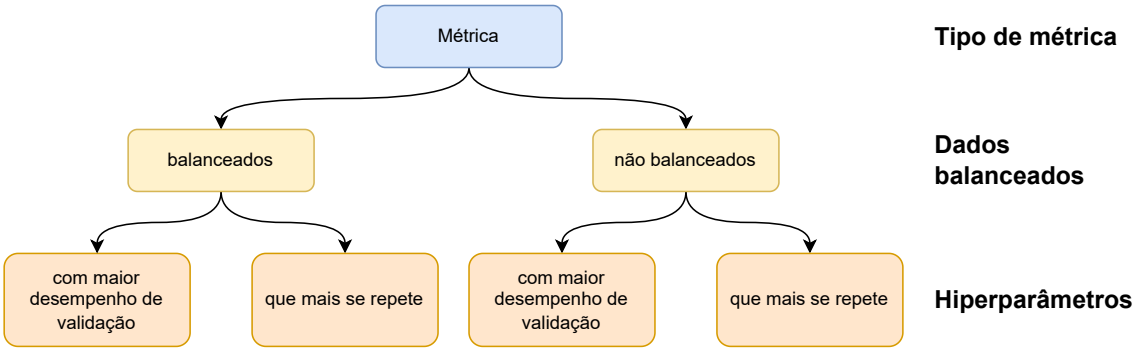
Além da variação na escolha de melhores hiperparâmetros, também fizemos testes aplicando e removendo o hiperparâmetro *class_weight* responsável por tratar o desbalanceamento dos dados. Esse hiperparâmetro *class_weight* trata-se de uma opção disponibilizada pela biblioteca Scikit-learn que quando atribuído o valor verdadeiro, aplica uma distribuição de pesos entre as classes com o intuito de balancear o aprendizado das instâncias, atribuindo um custo maior de erro a instâncias da classe minoritária. Esta estratégia é conhecida como aprendizado sensível a custo, e a escolha de aplicá-la foi feita, pois percebemos um alto grau de desbalanceamento nos dados. Adicionalmente, na tentativa de remover ao máximo esse desbalanceamento, executamos os experimentos removendo uma classe específica do conjunto de dados COAD, denominada POLE, que tinha muito poucas instâncias no conjunto de dados coletados.

4.3.4 Geração de métricas de desempenho

As métricas geradas no trabalho são divididas entre métricas de treinamento dos modelos e métricas da validação cruzada aninhada. Em cada uma dessas duas possíveis métricas, ainda existe a subdivisão nos dados balanceados (isto é, com tratamento do desbalanceamento de dados) e não balanceados (isto é, sem o tratamento), que diz respeito ao parâmetro de *class_weight* citado anteriormente. Além disso, também subdividimos entre as duas formas de escolher os melhores hiperparâmetros a partir da validação cruzada aninhada, isto é, hiperparâmetros com maior desempenho de validação ou hiperparâmetros que mais se repetem. Assim, a Figura 4.6 sumariza os cenários em que foram extraídas as métricas definida na Seção 2.2.2.2: precisão, sensibilidade, *f1-score*, coeficiente de

correlação de Matthews, acurácia e acurácia balanceada.

Figura 4.6 – Cenários experimentais utilizados na extração de métricas.



Fonte: O Autor

5 EXPERIMENTOS E RESULTADOS

Neste capítulo, com intuito de facilitar a apresentação dos resultados, separamos os experimentos executados durante o trabalho em diferentes seções, iniciando por uma separação entre experimentos com tratamento do desbalanceamento de classes e experimentos sem este tratamento. Assim, na Seção 5.1, comparamos os experimentos entre as três diferentes estratégias de integração de dados utilizando o balanceamento de dados com *class_weight* durante o aprendizado dos modelos. Já na Seção 5.2, fazemos as mesmas análises descritas na seção anterior, entretanto, focando na discussão de algumas anomalias identificadas nos experimentos sem o tratamento dos dados desbalanceados, *i.e.,* sem o hiperparâmetro *class_weight*. Por fim, na Seção 5.3, analisamos o impacto nos resultados a partir da remoção de um subtipo de tumor de cólon (COAD), o qual é pouco representado no conjunto de dados coletados, apresentando-se como uma classe minoritária.

Adicionalmente, nestas seções separaremos os experimentos entre as estratégias de integração de dados usadas, apresentando primeiro a estratégia de estágio inicial, posteriormente, a estratégia de estágio intermediário e, finalmente, a de estágio final. Além disso, todos os resultados seguem a mesma ordem de apresentação referente ao algoritmo de AM usado: (i) Árvores de decisão; (ii) Florestas aleatórias; e (iii) SVM. Na avaliação desses experimentos, identificaremos, principalmente, o impacto da escolha da estratégia de integração de dados, o desempenho dos modelos, a escolha da forma de selecionar os hiperparâmetros, e o quão prejudicial é o desbalanceamento de dados para o desempenho dos modelos nos nossos experimentos.

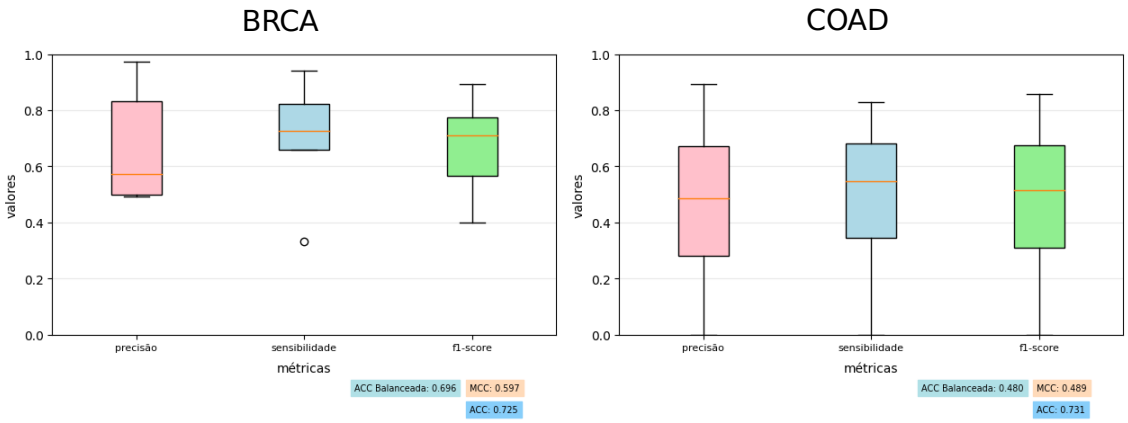
5.1 Análise de estratégias de integração com *class_weight* ativo

No início da análise de dados, executamos os experimentos sem qualquer tipo de tratamento de desbalanceamento de classes e notamos que o desbalanceamento presente nos dados seria um problema para avaliarmos justamente os prós e contras das estratégias de integração de dados. Portanto, para resolver esse problema, utilizamos o hiperparâmetro de *class_weight* que coloca pesos nas instâncias dos dados, assim dando um pouco mais de ênfase para as que estão em pequena quantidade.

5.1.1 Integração de estágio inicial

Aplicando a estratégia de estágio inicial e utilizando os hiperparâmetros que mais se repetem, a acurácia do modelo de árvores de decisão ficou acima de 72%. Conforme pode ser visto na Figura 5.1, temos as métricas gerais do experimento que obteve variações pequenas na sensibilidade e altas na precisão para os dados de BRCA. Algumas classes foram consideradas mais que outras, a causa disso é o desbalanceamento de classes, pois as duas classes com menores precisões (LumB e Normal) são as que estão em menor quantidade, notamos isso na Tabela 5.1. A questão dos dados desbalanceados se confirma quando analisamos os resultados dos experimentos do COAD. Neste cenário, as precisões e sensibilidades foram muito piores, com MCC e Acurácia balanceada abaixo de 50%.

Figura 5.1 – Métricas de desempenho geral do resultado da execução do modelo de árvores de decisão para a estratégia de integração de dados ômicos de estágio inicial com hiperparâmetros que mais se repetem

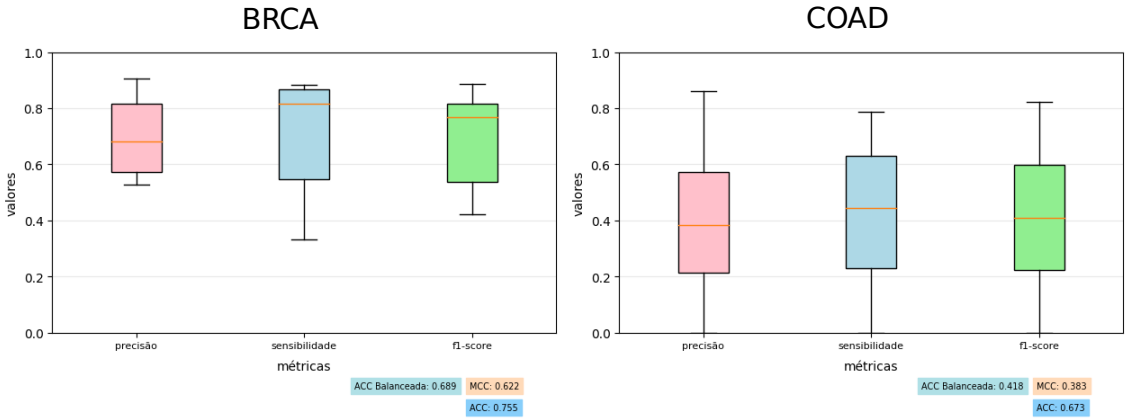


Fonte: O Autor

No caso dos experimentos utilizando os hiperparâmetros com maior desempenho de validação, mostrados na Figura 5.2, vemos um ganho tanto na acurácia quanto na precisão dos experimentos com os dados de BRCA. A precisão mostrou menor variação, o MCC ficou acima dos 60% e as acurácias foram igualmente melhores. Entretanto, os testes com os dados de COAD foram piores, mostrando que o modelo teve muita dificuldade para classificar corretamente as instâncias com os dados de COAD. Novamente, a principal suspeita é o desbalanceamento das classes.

Avaliando mais detalhadamente, na Tabela 5.2 nenhuma instância da classe POLE nos dados de COAD foi classificada corretamente, e as instâncias de CIN tiveram precisões acima de 85%. Isso indica que devido à pequena quantidade de dados das demais

Figura 5.2 – Métricas de desempenho geral do resultado da execução do modelo de árvores de decisão para estratégia de integração de dados de estágio inicial com hiperparâmetros com maior desempenho de validação



Fonte: O Autor

Tabela 5.1 – Precisão e sensibilidade do experimento com modelo de árvores de decisão na estratégia de estágio inicial com balanceamento de dados e hiperparâmetros que mais se repetem

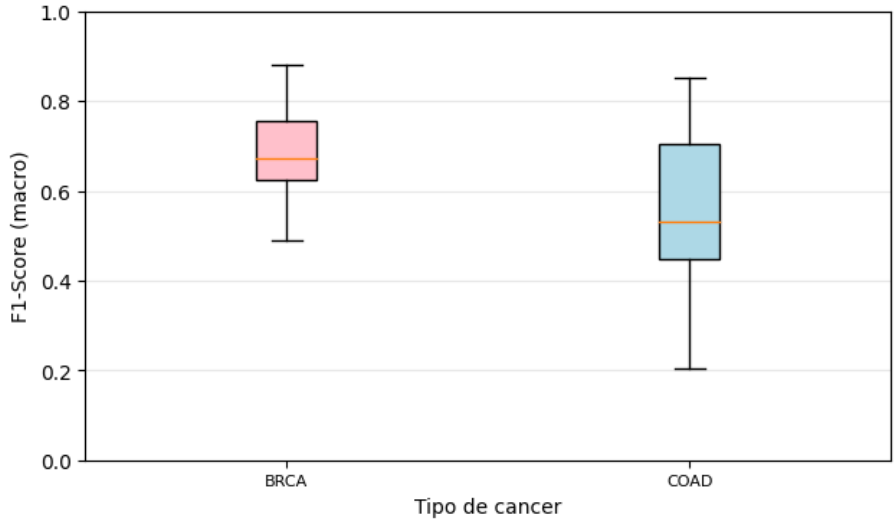
BRCA			COAD		
Classes	Precisão	Sensibilidade	Classes	Precisão	Sensibilidade
Basal	0.97	0.82	CIN	0.89	0.82
Her2	0.57	0.94	GS	0.37	0.46
LumA	0.83	0.72	MSI	0.6	0.63
LumB	0.49	0.66	POLE	0.0	0.0
Normal	0.50	0.4			

classes, o algoritmo apenas conseguiu aprender alguns padrões para um subtipo de câncer. Esse fenômeno não acontece com os dados de BRCA, pois nos dois experimentos para árvores de decisão, no mínimo duas classes apresentavam precisões maiores que 80%. Por fim, temos o gráfico da Figura 5.3 ilustrando os *test scores* da validação cruzada aninhada, reforçando como os dados de BRCA foram melhor avaliados pelo método de árvores de decisão na estratégia de estágio inicial.

Tabela 5.2 – Precisão e sensibilidade do experimento com modelo de árvores de decisão na estratégia de estágio inicial com balanceamento de dados e hiperparâmetros com maior desempenho de validação

BRCA			COAD		
Classes	Precisão	Sensibilidade	Classes	Precisão	Sensibilidade
Basal	0.90	0.86	CIN	0.86	0.78
Her2	0.68	0.88	GS	0.28	0.30
LumA	0.81	0.81	MSI	0.47	0.57
LumB	0.52	0.54	POLE	0.0	0.0
Normal	0.57	0.33			

Figura 5.3 – *Test scores* dos hiperparâmetros da Validação cruzada aninhada para o experimento com árvores de decisão na estratégia de estágio inicial



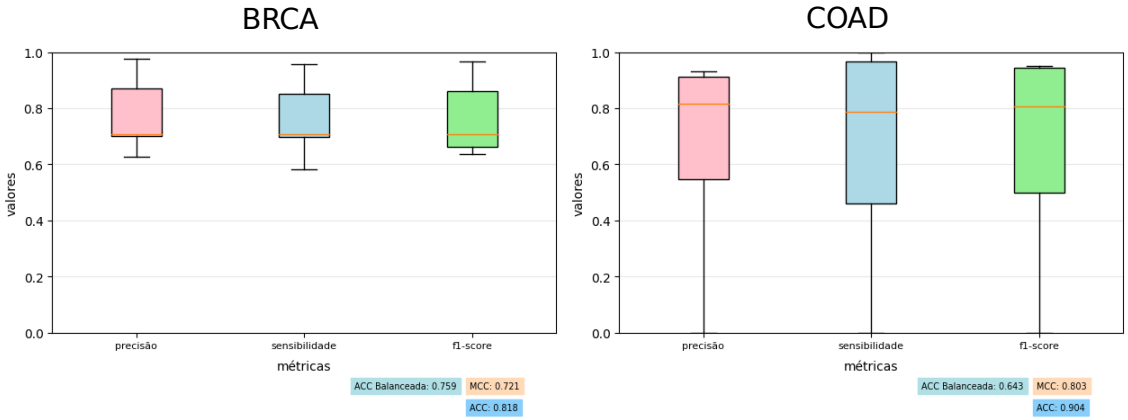
Fonte: O Autor

Os experimentos para o modelo de florestas aleatórias foram surpreendentes, pois esse modelo teve os melhores resultados utilizando a estratégia de integração de estágio inicial. Acreditamos que isso se deve à sua abordagem *ensemble* na geração do modelo. Nos resultados dos experimentos com os hiperparâmetros que mais se repetem (Figura 5.4), os dados de BRCA tiveram ótimos resultados. A precisão e sensibilidade possuem pequena variabilidade e a maioria dos valores está acima de 70%. Entretanto, o destaque fica para os resultados com os dados de COAD, com a maior acurácia e MCC entre todos os experimentos realizados no escopo deste trabalho. Apesar da grande variação tanto da precisão quanto da sensibilidade nos dados de COAD, as duas classes com maior quantidade de instâncias obtiveram resultados acima de 90% em ambas as métricas.

Utilizando os hiperparâmetros com maior desempenho de validação, cujos resultados são mostrados na Figura 5.5, os resultados continuaram ótimos, mas os testes com dados de COAD tiveram uma perda significativa. Acreditamos que isso acontece, pois o algoritmo acerta muitas classificações para um caso de teste isolado e como o número de instâncias de algumas classes é muito baixa, não existe outra combinação que seja mais fácil de ser predita para o algoritmo. Olhando com mais detalhes as predições realizadas, aparentemente o algoritmo passou a sempre classificar uma das classes em alguns casos. Dessa forma, anomalias são geradas, como o caso da classe GS que possui precisão de 100% mas sensibilidade de 30%, indicando que o algoritmo possui confiança quando retorna esta classe para uma dada instância, entretanto, ele possui muitos falsos negativos.

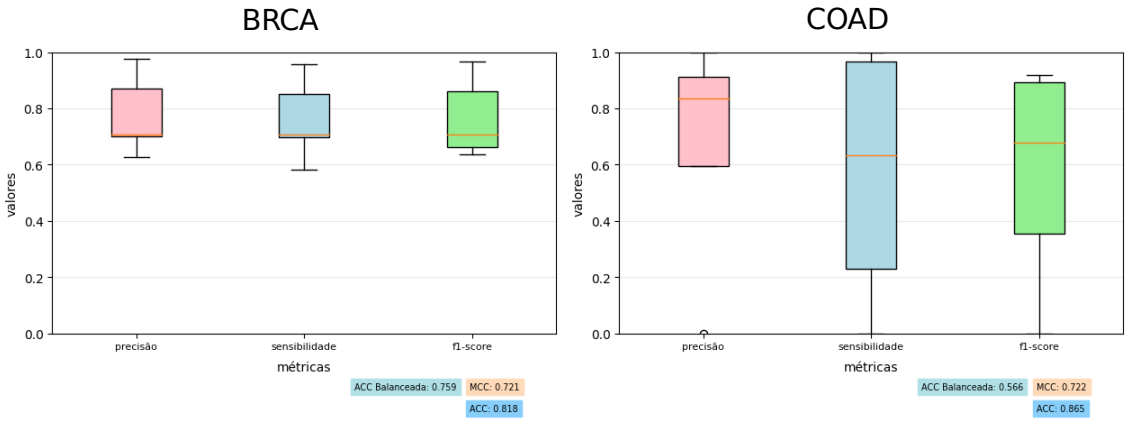
No gráfico da Figura 5.6 observamos os valores de *test score* da validação cru-

Figura 5.4 – Métricas de desempenho do resultado da execução do modelo de florestas aleatórias para estratégia de integração de dados de estágio inicial com hiperparâmetros que mais se repetem



Fonte: O Autor

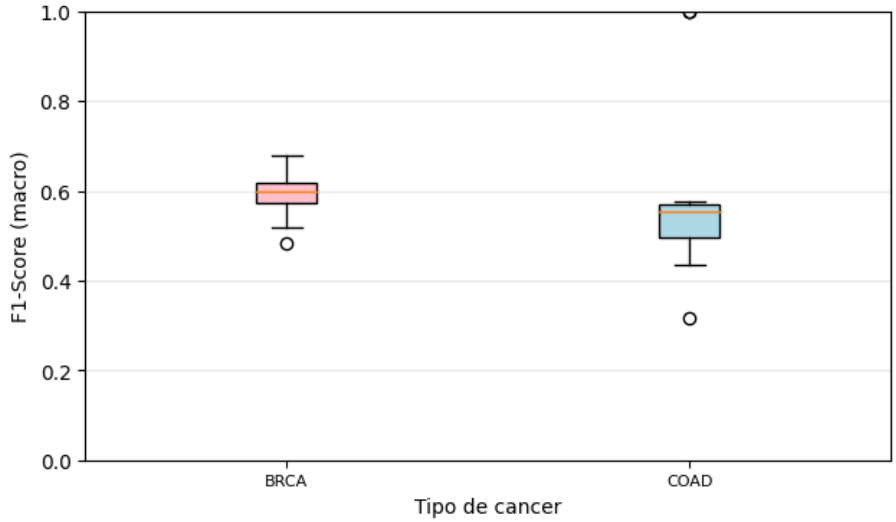
Figura 5.5 – Métricas de desempenho do resultado da execução do modelo de florestas aleatórias para estratégia de integração de dados de estágio inicial com hiperparâmetros com maior desempenho de validação



Fonte: O Autor

zada aninhada para o experimento de florestas aleatórias. Nessa imagem, apesar dos dois *outliers*, percebemos que os dados de BRCA obtiveram ótimas pontuações com pouca variabilidade. Devido ao grande desbalanceamento que existe nos dados de COAD, é nítido que o algoritmo tinha maior facilidade para lidar com os dados de BRCA. Notamos que em razão do grande desbalanceamento de dados, os modelos possuem instabilidade na predição de instâncias para os dados de COAD. De fato, observa-se um *outlier* com F1-score de 100%.

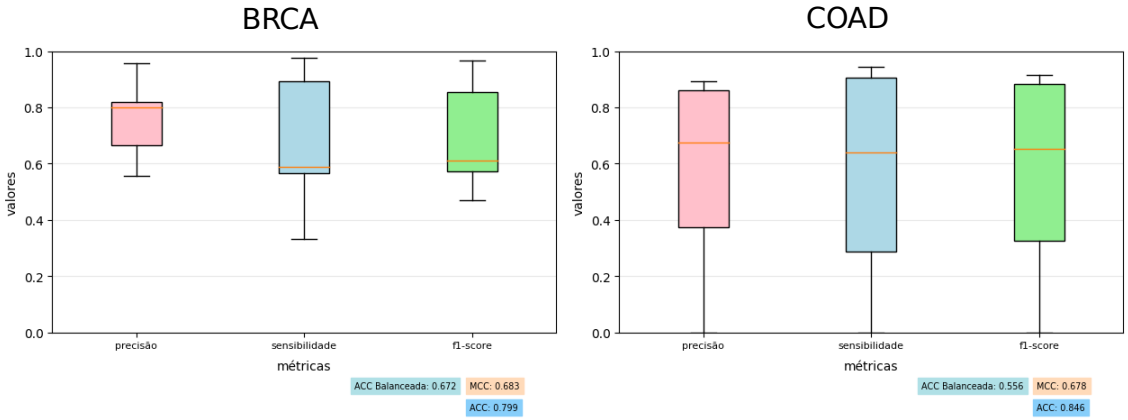
Figura 5.6 – *Test scores* dos hiperparâmetros da Validação cruzada aninhada para o experimento com florestas aleatórias na estratégia de estágio inicial



Fonte: O Autor

No caso do SVM, notamos algumas similaridades nos resultados. Os experimentos utilizando os hiperparâmetros que mais se repetem e utilizando os com maior desempenho de validação foram os mesmos, os quais são sumarizados na Figura 5.7. Apesar do modelo ter melhores resultados com dados do COAD, a precisão e sensibilidade possuem uma grande variabilidade, denotando novamente uma dificuldade do algoritmo a aprender algumas classes menos frequentes. Novamente, se observarmos a Tabela 5.3 nota-se que algumas classes (GS e POLE) do banco de dados COAD com resultados muito baixos, enquanto outras tiveram resultados muito altos.

Figura 5.7 – Métricas de SVM para estratégia de integração de dados de estágio inicial

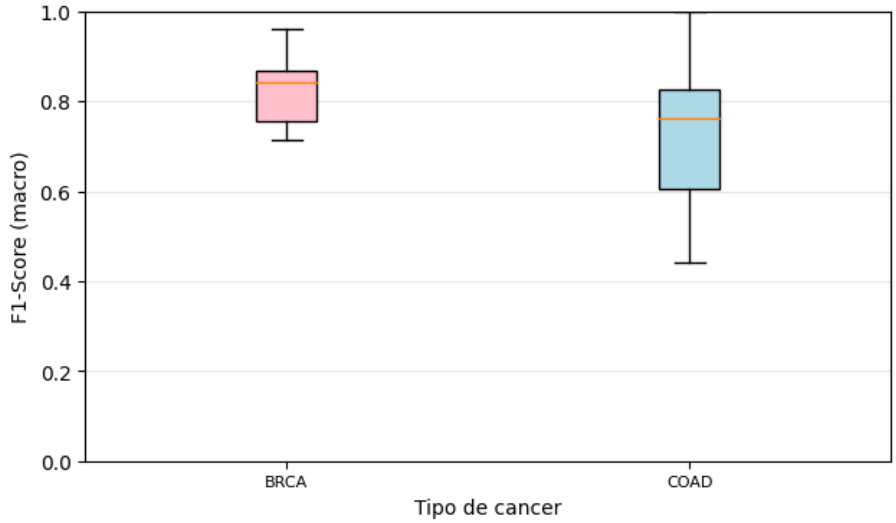


Fonte: O Autor

Enfim, analisando a imagem 5.8, vemos que nos dados de BRCA a mediana do *F1-score* ficou acima de 80% assim como nos testes com florestas aleatórias. O COAD teve

mediana próxima a 80%. Entretanto, diferentemente dos testes com florestas aleatórias, o SVM não gerou *outliers*, denotando maior confiabilidade nos resultados do algoritmo.

Figura 5.8 – *Test scores* dos hiperparâmetros da Validação cruzada aninhada para o experimento com SVM na estratégia de estágio inicial



Fonte: O Autor

5.1.2 Integração de estágio intermediário

Na estratégia de estágio intermediário, alguns experimentos não obtiveram resultados relevantes para análise devido a valores de métricas de desempenho muito baixos. Portanto, o foco aqui será analisar apenas alguns conjuntos que possuem resultados relevantes. Os conjuntos de dados analisados do algoritmo CIMLR são: (i) Conjunto de *f-values* que são os resultados de uma rede de difusão executada no algoritmo; (ii) Conjunto com *y-data* que se refere, exclusivamente, a dados de resultados do k-means executado; (iii) Por fim, união com todos os conjuntos de dados ou *atributos* selecionados concatenados. No caso do NEMO, seu retorno é apenas os *clusters*, portanto, focaremos apenas

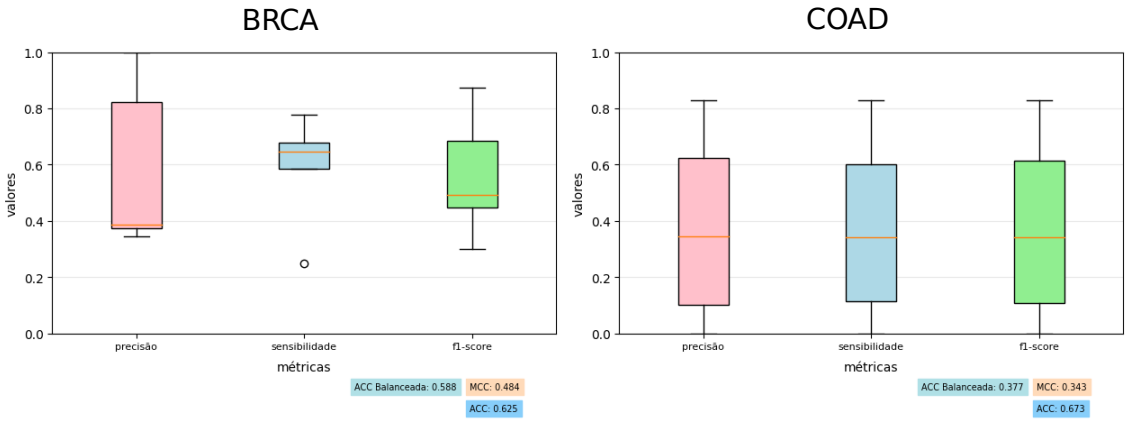
Tabela 5.3 – Precisão e sensibilidade do experimento com modelo SVM com dados COAD na estratégia de estágio inicial com balanceamento de dados

Classes	Precisão	Sensibilidade
CIN	0.89	0.94
GS	0.5	0.38
MSI	0.85	0.89
POLE	0.0	0.0

nas análises usando seu resultado do processo de agrupamento. Os resultados dos testes com hiperparâmetros com maior desempenho de validação não serão apresentados nessa seção, pois seus resultados em todos os casos foram bem piores que com hiperparâmetros que mais se repetem.

Devido ao grande desbalanceamento de classes presente nos dados, tanto o CIMLR quanto o NEMO não tiveram desempenhos ótimos. No caso do CIMLR, um modelo matemático complexo é usado para integrar dados multi-ômicos, o que pode dificultar a interpretação dos resultados. Se os dados estiverem desbalanceados, pode ser mais difícil entender por que o modelo está efetuando certas previsões e como está lidando com a classe minoritária, caracterizando uma perda de interpretabilidade no algoritmo. O NEMO possui um problema parecido, mas como o algoritmo NEMO é baseado em uma abordagem de rede, que pode ser sensível a ruídos, se os dados estiverem desbalanceados, o NEMO pode aprender padrões que se aplicam apenas à classe majoritária, causando problemas de *overfitting*. Além dessas particularidades de cada algoritmo, existem problemas clássicos causados pelo desbalanceamento de dados em algoritmos de AM (*e.g.,* predições incorretas, avaliação não confiável) que prejudicam ainda mais os resultados.

Figura 5.9 – Métricas de desempenho para árvores de decisão para estratégia de integração de dados de estágio intermediário com conjunto de dados com todos os e hiperparâmetros que mais se repetem

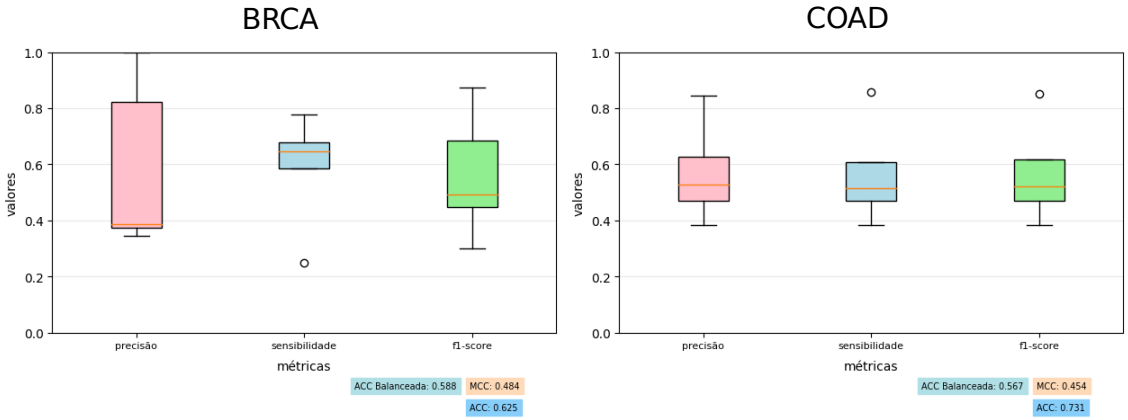


Fonte: O Autor

Utilizando CIMLR com conjunto de todos os dados concatenados, vemos novamente o desbalanceamento de dados reduzindo os valores das métricas. Nitidamente, ao observarmos a Figura 5.9, os resultados nos dados de COAD foram piores devido ao maior desbalanceamento. Outra intrigante observação é que a variação da sensibilidade nos dados do câncer BRCA é bem pequena, mas a variação da precisão é grande, logo pode existir uma certa tendência a falsos positivos nos dados de BRCA.

Dentre os experimentos com os conjuntos de dados citados anteriormente, o conjunto de *f-values* foi o que obteve os melhores resultados no geral. Os *f-values* são resultados de um processo de difusão na rede usada como base do algoritmo, e tem como o objetivo aprender representações dos nós do grafo que capturam as informações, portanto, usá-los como dados de treinamento foi extremamente benéfico. Na Figura 5.10 vemos os resultados do experimento do treinamento do modelo de árvores de decisão usando o conjunto de dados com *f-values*.

Figura 5.10 – Métricas de desempenho para árvores de decisão para estratégia de integração de dados de estágio intermediário com conjunto de dados *f-values* e hiperparâmetros que mais se repetem



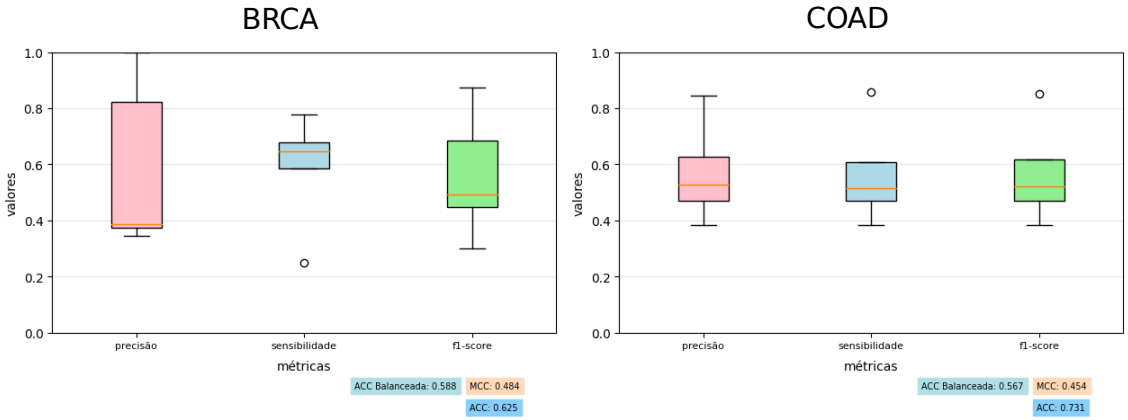
Fonte: O Autor

Outro conjunto de dados que obteve resultados bons na integração de dados de estágio intermediário foi o derivado do uso do *y-data*, que são, basicamente, os dados dos resultados do algoritmo K-means executado no algoritmo CIMLR. No experimento utilizando as árvores de decisão, sumarizados na Figura 5.11, vemos que esse conjunto de dados trouxe resultados bem próximos, tanto para dados de BRCA quanto para dados de COAD.

O modelo de florestas aleatórias, novamente, foi o vencedor de desempenho. A melhor combinação foi o conjunto de dados com todos os atributos concatenados submetidos ao aprendizado através do algoritmo de florestas aleatórias. Na Figura 5.12, vemos que com os dados de COAD a acurácia chegou a 85%, o que é surpreendente, considerando que o melhor resultado de todo o trabalho obteve acurácia de 90%. Entretanto, vemos uma grande variação na métrica de sensibilidade, o qual foi o grande problema dessa execução.

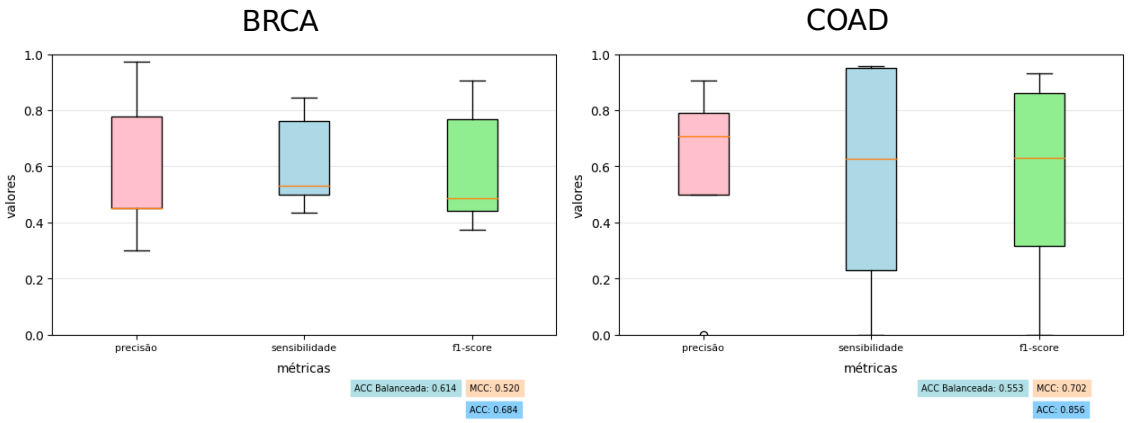
Com o conjunto de dados usando *f-values* os resultados para florestas aleatórias não poderiam ser muito diferentes dos descritos no parágrafo acima. Na Figura 5.13,

Figura 5.11 – Métricas de desempenho para árvores de decisão para estratégia de integração de dados de estágio intermediário com conjunto de dados *y-data* e hiperparâmetros que mais se repetem



Fonte: O Autor

Figura 5.12 – Métricas de desempenho para florestas aleatórias para estratégia de integração de dados de estágio intermediário com conjunto com todos os atributos e hiperparâmetros que mais se repetem



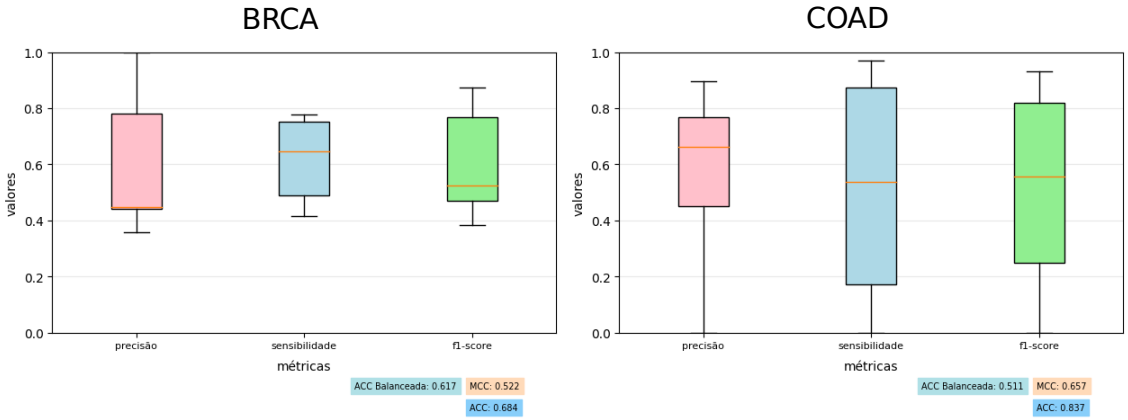
Fonte: O Autor

podemos ver os resultados da execução do experimento utilizando o modelo de florestas aleatórias com o conjunto de dados de *f-values*. Percebemos padrões bem semelhantes de desempenho para todas as métricas, com os dados de acurácia balanceada significativamente mais altos para os dados de COAD.

Apenas para fins de comparação com os outros experimentos, na Figura 5.14 temos o modelo de árvores aleatórias executado com o conjunto de dados que usam o *y-data*. Nesse experimento ainda vemos uma grande variação de todas as métricas, o que aconteceu com bastante frequência na estratégia de integração de dados de estágio intermediário.

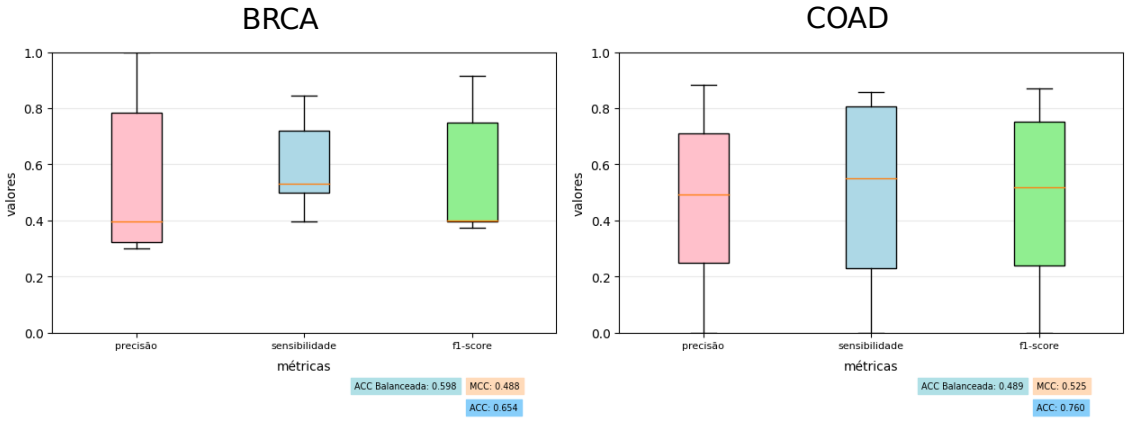
O modelo de SVM foi o modelo que obteve as menores variações de métricas

Figura 5.13 – Métricas de desempenho para florestas aleatórias para estratégia de integração de dados de estágio intermediário com conjunto de dados *f-values* e hiperparâmetros que mais se repetem



Fonte: O Autor

Figura 5.14 – Métricas de desempenho para florestas aleatórias para estratégia de integração de dados de estágio intermediário com conjunto de dados *y-data* e hiperparâmetros que mais se repetem

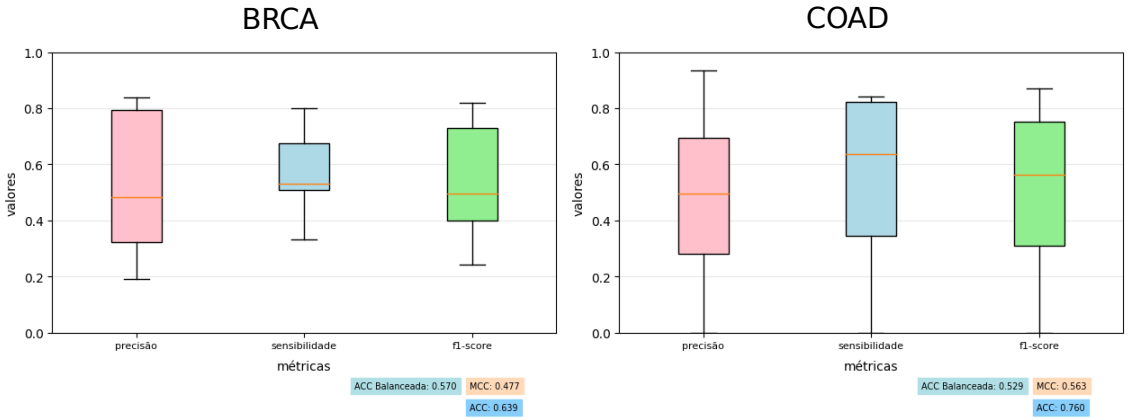


Fonte: O Autor

dentre os demais nos experimentos com a estratégia de estágio intermediário. Entretanto, não foi o algoritmo que teve os melhores resultados de acurácia. Observamos na Figura 5.15, o resultado dos experimentos usando o conjunto de dados com todos os atributos. O SVM obteve seu melhor desempenho na estratégia de estágio intermediário, com o conjunto de dados baseado nos *f-values* do algoritmo CIMLR. Podemos notar na Figura 5.16 uma variação muito pequena nas métricas com dados de COAD, comparado com os demais modelos. O SVM também teve os melhores valores de mediana na precisão e sensibilidade em dados de COAD quando comparado a todos os outros modelos.

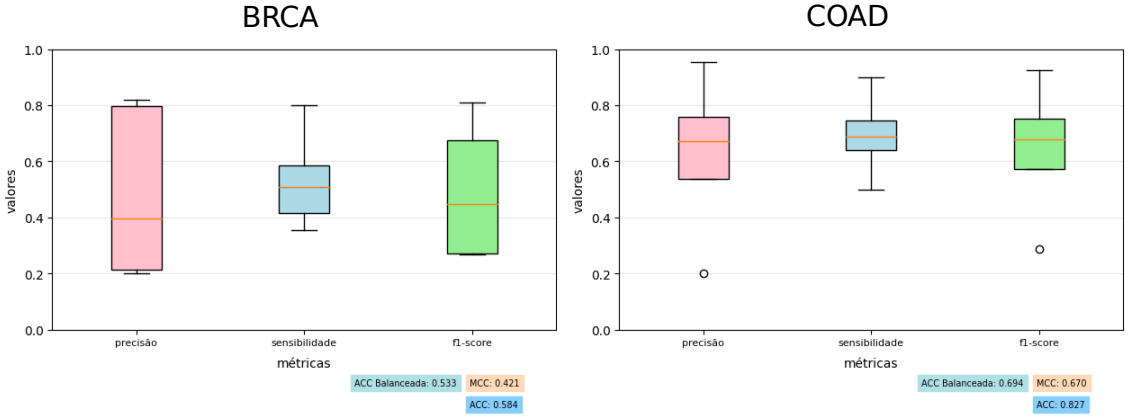
O conjunto de dados que menos se destacou em combinação com o SVM foi o conjunto com o *y-data*. Seus resultados não chegaram a 70% de acurácia para os dados

Figura 5.15 – Métricas de desempenho para SVM para estratégia de integração de dados de estágio intermediário com conjunto com todos os atributos e hiperparâmetros que mais se repetem



Fonte: O Autor

Figura 5.16 – Métricas de desempenho para SVM para estratégia de integração de dados de estágio intermediário com conjunto de dados *f-values* e hiperparâmetros que mais se repetem

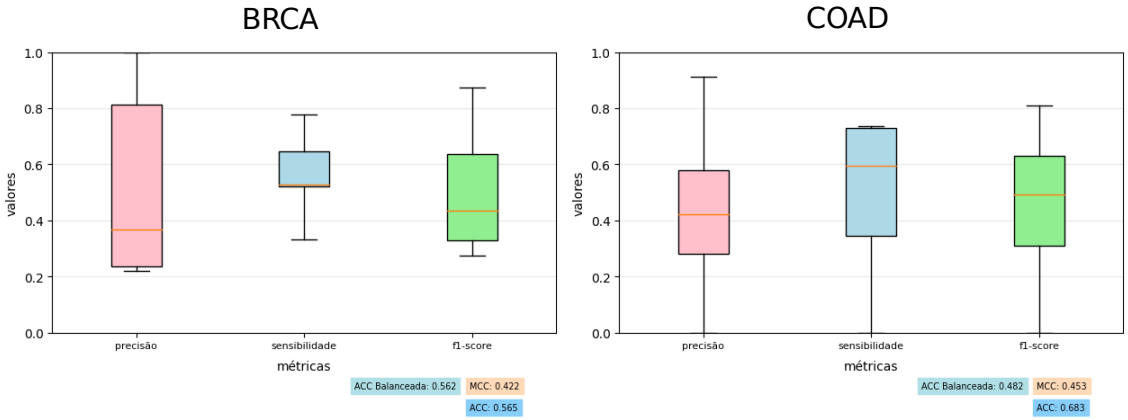


Fonte: O Autor

de COAD, e sua precisão e sensibilidade voltaram a ter variações muito grandes. Estes resultados são sumarizados na Figura 5.17.

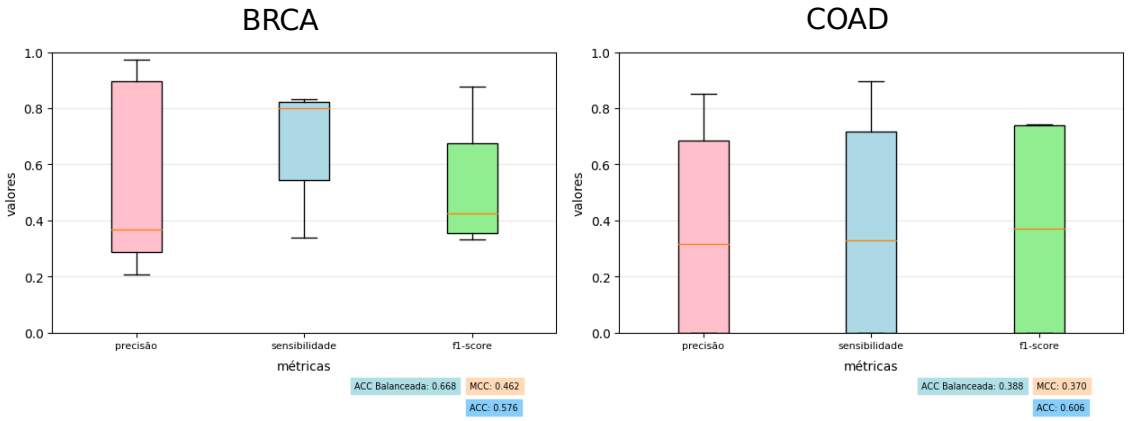
No caso do algoritmo de integração NEMO, como dito antes, o único conjunto de dados é o resultado da tarefa de agrupamento. Para os dois tipos de câncer, BRCA e COAD, os resultados de todos os três modelos foram muito similares, tendo mudanças quase imperceptíveis entre diferentes algoritmos. Portanto, na Figura 5.18 vemos os dados do experimento utilizando dados de COAD, com conjunto de dados da clusterização do NEMO apenas do experimento com o modelo de SVM.

Figura 5.17 – Métricas de desempenho para SVM para estratégia de integração de dados de estágio intermediário com conjunto de dados *y-data* e hiperparâmetros que mais se repetem



Fonte: O Autor

Figura 5.18 – Métricas de desempenho para SVM para estratégia de integração de dados de estágio intermediário com *clusters* do algoritmo NEMO e hiperparâmetros que mais se repetem



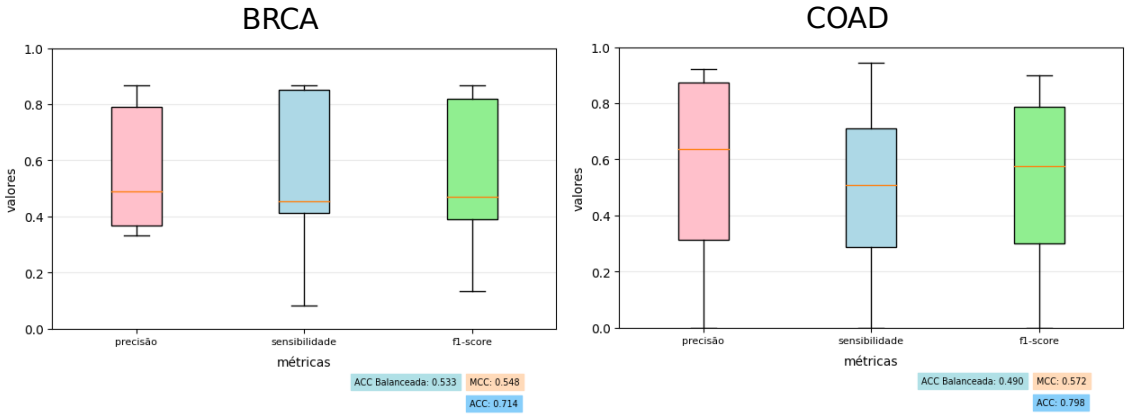
Fonte: O Autor

5.1.3 Integração de estágio final

A integração de dados de estágio final, como dito anteriormente, combina os dados no final do processo, por meio de alguma função ou método que agrega as predições de diferentes modelos. No caso deste trabalho essa função é uma votação majoritária, *i.e.,* a classe a mais votada dentre todas as predições é retornada como predição final.

Portanto, na Figura 5.19 observamos as métricas referentes ao resultado da votação majoritária do método de árvores de decisão, executado utilizando a seleção dos hiperparâmetros que mais repetem. Podemos notar que são resultados medianos, com alta variação da sensibilidade e precisão, apesar de uma boa acurácia. Assim podemos dizer que o algoritmo teve certa dificuldade no aprendizado.

Figura 5.19 – Métricas de desempenho para árvores de decisão para estratégia de integração de dados de estágio final com hiperparâmetros que mais se repetem (votação majoritária)

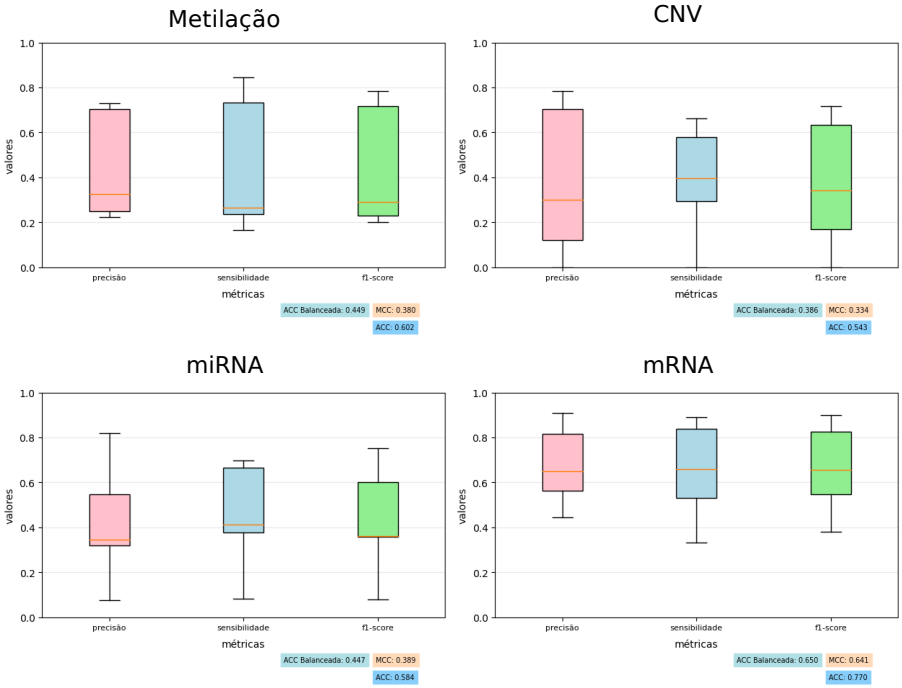


Fonte: O Autor

Para avaliarmos com mais precisão quais ômicas tiveram maior impacto nesses resultados da Figura 5.19, podemos analisar as Figuras 5.20 e 5.21 que mostram métricas de cada ômica separadamente para BRCA e COAD, respectivamente.

Podemos notar que para o BRCA (Figura 5.20), os dados de mRNA obtiveram os melhores resultados dentre as demais ômicas. Assim, podemos dizer que o modelo de árvores de decisão aprendeu a identificar alguns subtipos de câncer pelos dados de

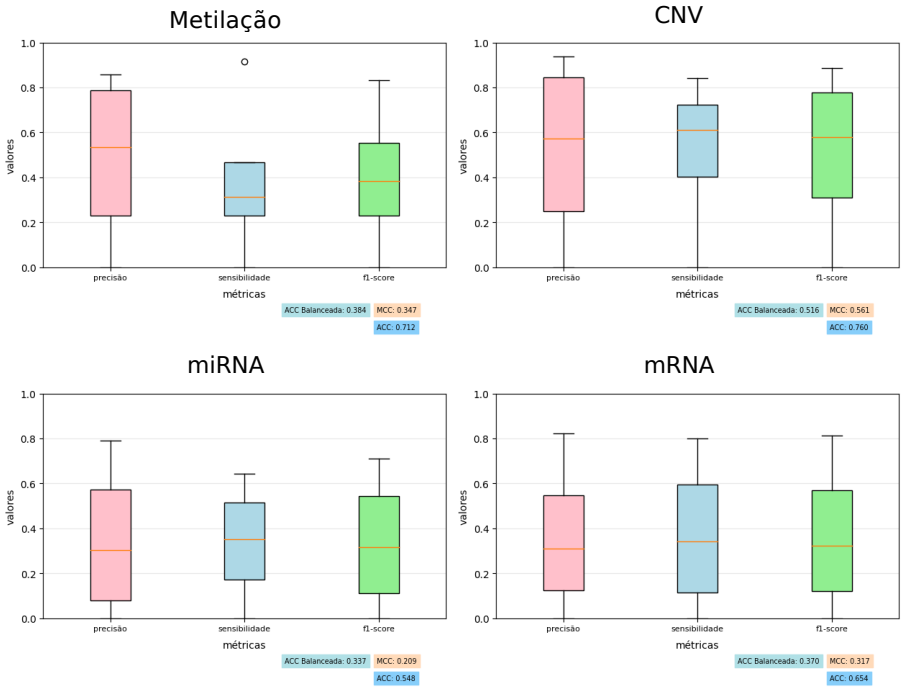
Figura 5.20 – Métricas de desempenho para árvores de decisão para estratégia de integração de dados de estágio final com hiperparâmetros que mais se repetem separadas por ômicas para BRCA



Fonte: O Autor

mRNA. Entretanto, observamos uma grande dificuldade com a ômica de metilação de DNA, com o modelo demonstrando variações muito grandes para as métricas analisadas e, de uma forma geral, medianas baixas.

Figura 5.21 – Métricas de desempenho para árvores de decisão para estratégia de integração de dados de estágio final com hiperparâmetros que mais se repetem separadas por ômicas para COAD



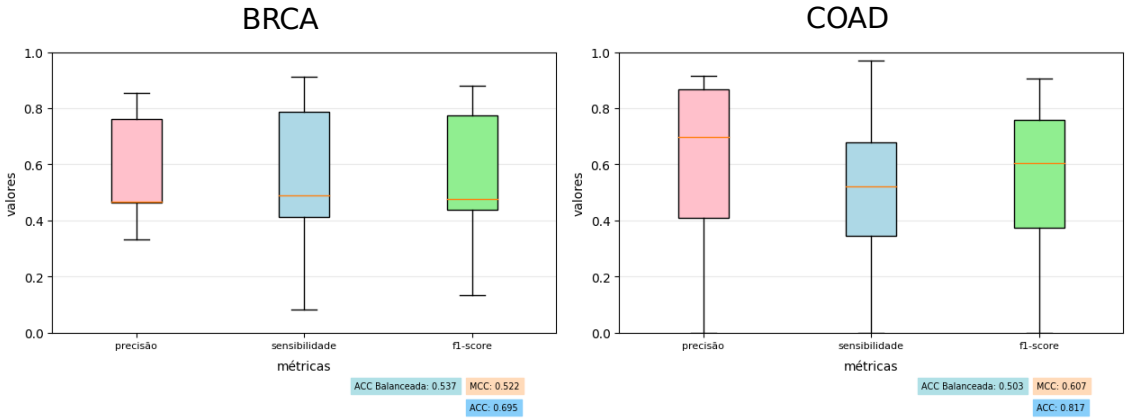
Fonte: O Autor

No caso dos dados de COAD (Figura 5.21), percebe-se que todas as métricas analisadas para os quatro tipos de ômicas empregadas obtiveram altas variações, reforçando dificuldades no aprendizado do modelo para dados com alto desbalanceamento entre classes. Entretanto, diferentemente do BRCA, os melhores resultados ficaram com os dados de CNV, apesar de ainda estarem longe de serem ótimos.

Para os testes utilizando hiperparâmentros selecionados a partir dos maiores desempenhos, as análises não mudam muito, exceto que no geral os testes para COAD tiveram um ganho considerável, enquanto para dados de BRCA houveram algumas perdas. Entretanto, mRNA continua tendo melhores resultados para BRCA, assim como o CNV continua sendo o melhor conjunto de atributos para o COAD. Na Figura 5.22, apresentamos o resultado da votação majoritária com hiperparâmetros com maior desempenho de validação e podemos ver com mais detalhes os ganhos e as perdas mencionados.

Avaliando os dados de validação cruzada aninhada na Figura 5.23, podemos confirmar alguns pontos como a dificuldade do modelo de identificar alguns subtipos de câncer. Podemos notar essa característica quando vemos a ômica CNV com a melhor medi-

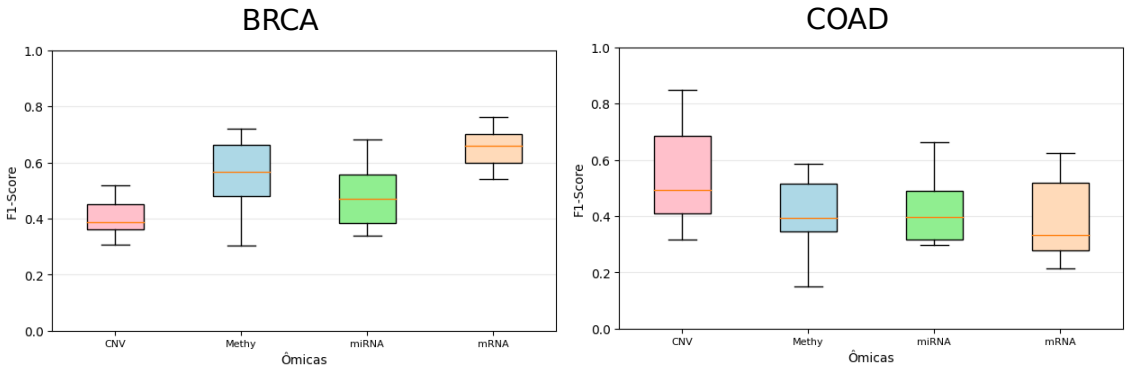
Figura 5.22 – Métricas de desempenho para árvores de decisão para estratégia de integração de dados de estágio final com hiperparâmetros com maior desempenho de validação (votação majoritária)



Fonte: O Autor

ana de *F1-Score* mas com uma grande variação de dados, tendo resultados com 30% de precisão e outros com quase 70%.

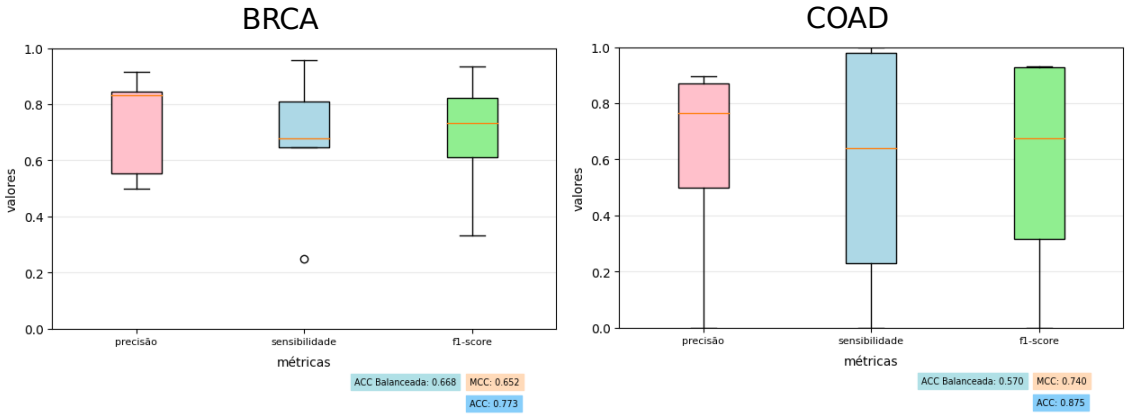
Figura 5.23 – *Test scores* dos hiperparâmetros da Validação cruzada aninhada para o experimento com Árvores de decisão na estratégia de estágio final



Fonte: O Autor

Os experimentos com o modelo de florestas aleatórias, como esperado, tiveram resultados bem melhores que os de árvores de decisão. Todas as métricas melhoraram e apresentaram resultados mais consistentes e confiáveis. Na Figura 5.24 observamos a votação majoritária do modelo de florestas aleatórias para hiperparâmentros que mais se repetem. Em comparação com os resultados da estratégia de estágio inicial, os resultados foram piores. Porém, comparando com os resultados já mostrados de integração de estágio original, os resultados foram bons quando comparados aos testes de árvores de decisão.

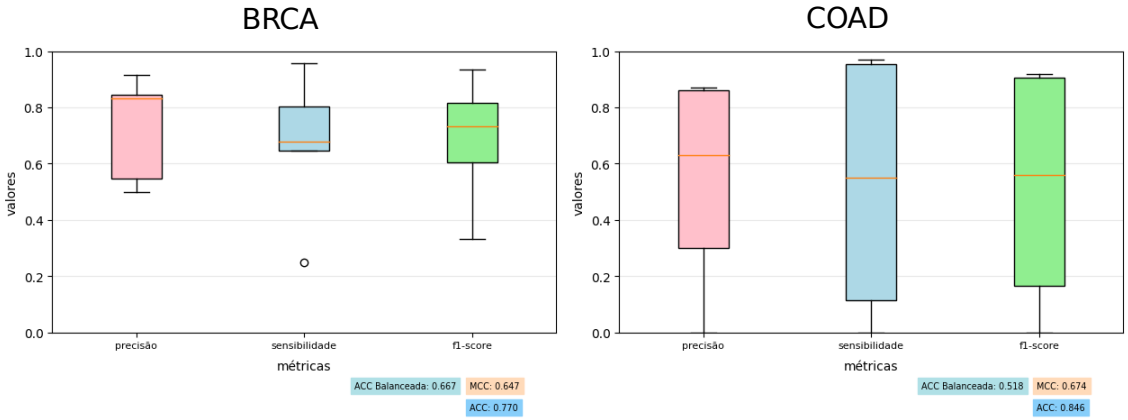
Figura 5.24 – Métricas de desempenho para florestas aleatórias para estratégia de integração de dados de estágio final com hiperparâmetros que mais se repetem (votação majoritária)



Fonte: O Autor

Utilizando os hiperparâmetros com maior desempenho de validação (Figura 5.25), os resultados foram piores, pois o modelo aprendeu a prever apenas a classe com mais quantidade de instâncias nos dados de treinamento. O resultado é compreensível, tendo em vista que uma estratégia que visa maximizar desempenho tenderá a, indiretamente, favorecer as classes mais representadas nos dados de treinamento. Analisando as predições feitas pelos modelos, observou-se que nos outros subtipos de câncer o modelo cometia muitos erros de predição.

Figura 5.25 – Métricas de desempenho para florestas aleatórias para estratégia de integração de dados de estágio final com hiperparâmetros com maior desempenho de validação (votação majoritária)

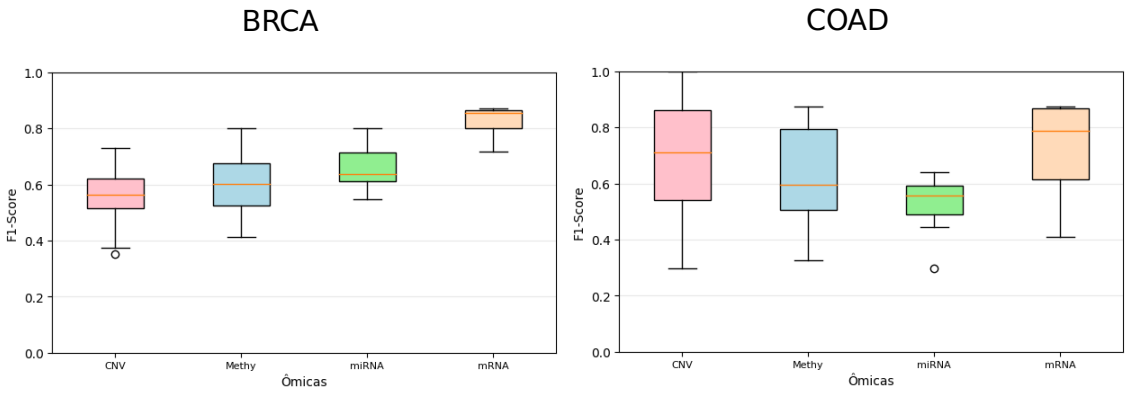


Fonte: O Autor

Analisando com mais detalhes nas métricas da validação cruzada aninhada mostradas na Figura 5.26, vemos que os dados de mRNA foram os que mais beneficiaram os bons resultados da estratégia para esse modelo. A mediana do *F1-Score* ficou acima

de 75% em mRNA, tanto para dados de BRCA quanto para COAD. Adicionalmente, a pequena variação observada mostra que os resultados foram bem consistentes.

Figura 5.26 – *Test scores* dos hiperparâmetros da Validação cruzada aninhada para o experimento com Florestas aleatórias na estratégia de estágio final



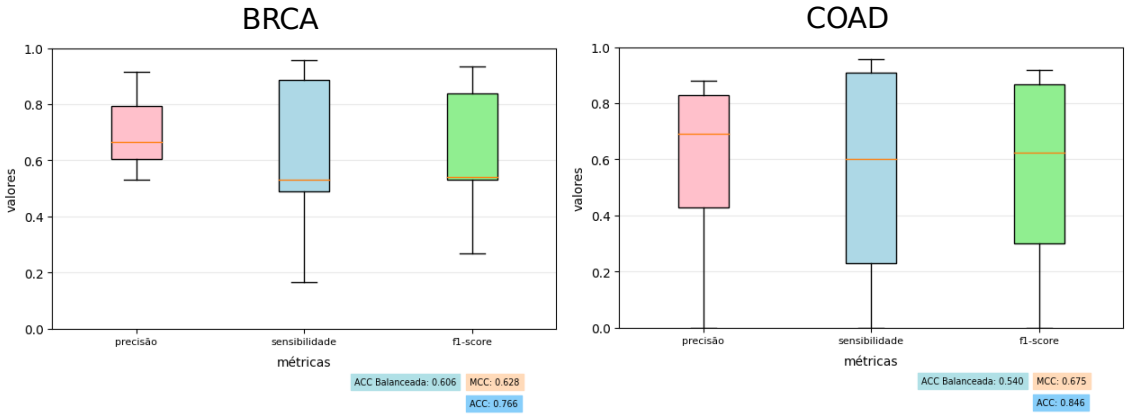
Fonte: O Autor

O SVM foi, novamente, superado pelo modelo de florestas aleatórias. Percebemos na Figura 5.27 que mesmo com bons resultados, se compararmos os resultados deste modelo com o modelo de florestas aleatórias no mesmo tipo de experimento (Figura 5.24, os resultados foram bem inferiores.

O mesmo podemos dizer para o modelo SVM usando hiperparâmetros com maior desempenho de validação. Os resultados foram bastante semelhantes aos obtidos com os hiperparâmetros selecionados a partir da análise da frequência. Apesar dos resultados serem muito próximos do mesmo experimento realizado com florestas aleatórias, ainda assim, existe uma menor variação de valores para os testes com o modelo de florestas aleatórias. Vemos que a variação dos valores é bem maior nos testes com SVM (Figura 5.28) que nos testes com florestas aleatórias (Figura 5.25).

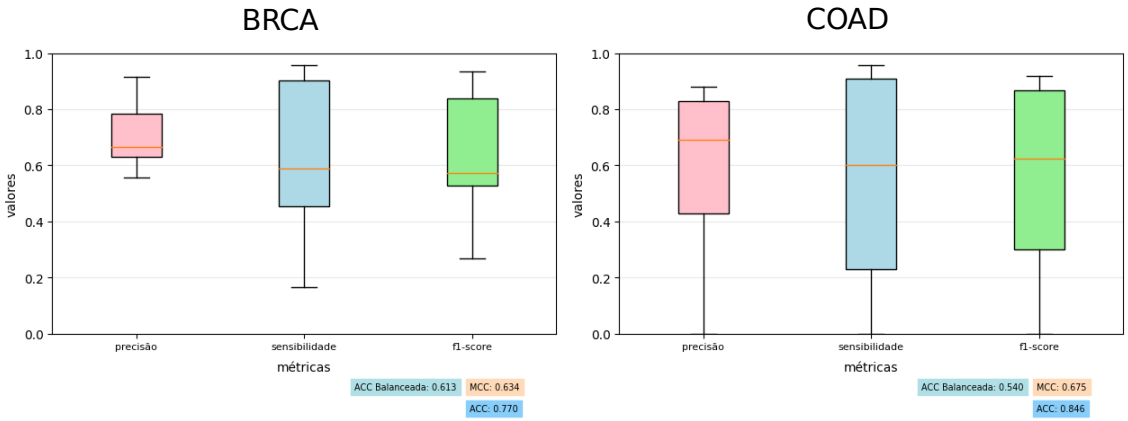
O modelo de florestas aleatórias foi mais assertivo e aprendeu melhor como classificar os subtipos de câncer corretamente, quando comparado com o SVM. Entretanto, os resultados individuais de cada ômica utilizando o SVM foram, em alguns casos, melhores que os resultados obtidos com os modelos de florestas aleatórias. Na Figura 5.29 temos o gráfico dos resultados da execução da validação cruzada aninhada para reforçar os pontos mostrados.

Figura 5.27 – Métricas de desempenho para SVM para estratégia de integração de dados de estágio final com hiperparâmetros que mais se repetem (votação majoritária)



Fonte: O Autor

Figura 5.28 – Métricas de desempenho para SVM para estratégia de integração de dados de estágio final com hiperparâmetros com maior desempenho de validação (votação majoritária)

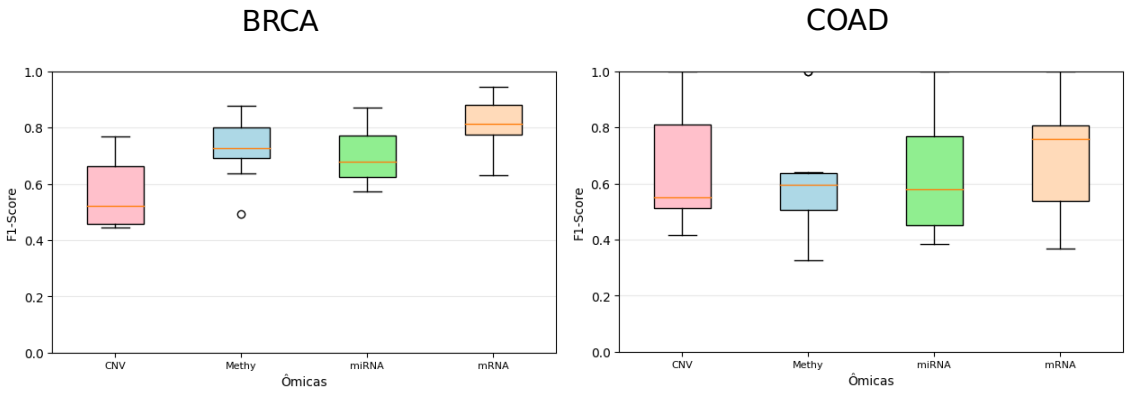


Fonte: O Autor

5.2 Análise de estratégias de integração sem parâmetro *class_weight*

Nos experimentos sem o tratamento do desbalanceamento dos dados por meio da opção de *class_weight*, os resultados foram piores comparados aos experimentos com este tratamento, mostrados na seção anterior. Portanto, nessa seção vamos nos deter apenas na análise dos experimentos que obtiveram resultados melhores ou com resultados anômalos. Durante a análise, notamos que o desbalanceamento de dados prejudica muito mais estratégias que executam mais passos de processamento nas ômicas, representadas neste trabalho pelas estratégias de estágio final e intermediário.

Figura 5.29 – *Test scores* dos hiperparâmetros da Validação cruzada aninhada para o experimento com SVM na estratégia de estágio final



Fonte: O Autor

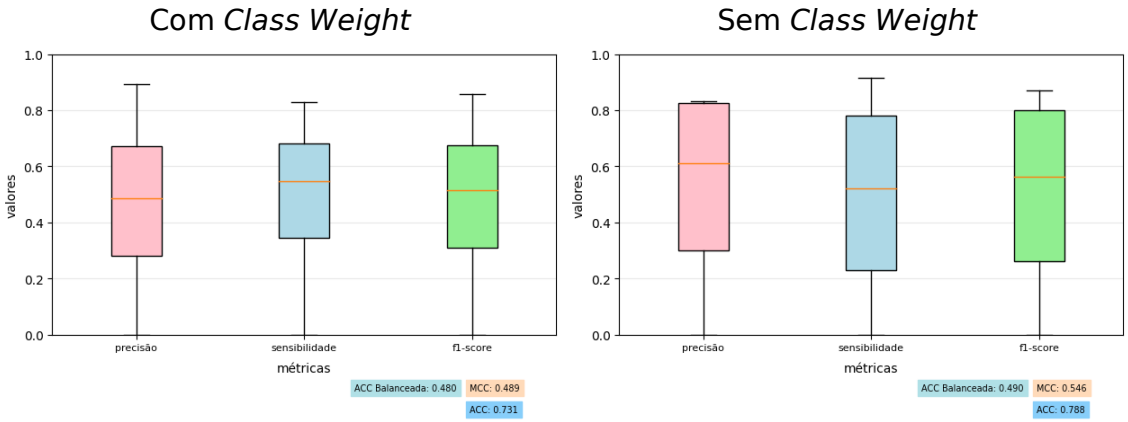
5.2.1 Integração de estágio inicial

Na integração de estágio inicial, algumas anomalias foram observadas, principalmente na resposta dos experimentos com modelo de árvores de decisão e florestas aleatórias. Na Figura 5.30, vemos que os resultados dos experimentos sem o tratamento do desbalanceamento de dados para o modelo de árvores de decisão usando dados de COAD obtiveram resultados ligeiramente melhores que com balanceamento. A principal análise que podemos fazer com os gráficos é que com dados balanceados, as medianas da precisão e sensibilidade são mais baixas, logo existe a possibilidade de maior tendência a falsos negativos e falsos positivos. O balanceamento deve ter causado maiores dificuldades para o modelo ao demandar mais para aprender classes menos representadas nos dados. Logo, ele acabou não aprendendo bem nenhuma das classes de subtipos de câncer.

Outra anomalia observada foi como o modelo de florestas aleatória lidou com os dados sem balanceamento para o câncer BRCA. O modelo teve uma dificuldade tremenda em aprender. Se observarmos na Figura 5.31, vemos a tamanha variação que existe na sensibilidade ao executar o modelo sem balanceamento. Como dito antes, essa grande variação na sensibilidade indica uma dificuldade de classificar algumas classes especificamente. A causa dessa grande variação é a tentativa do algoritmo a se adaptar a classes com pequena quantidade de dados sem sucesso. Assim como no modelo de árvores de decisão, ele acaba aprendendo muito bem apenas as classes com mais instâncias.

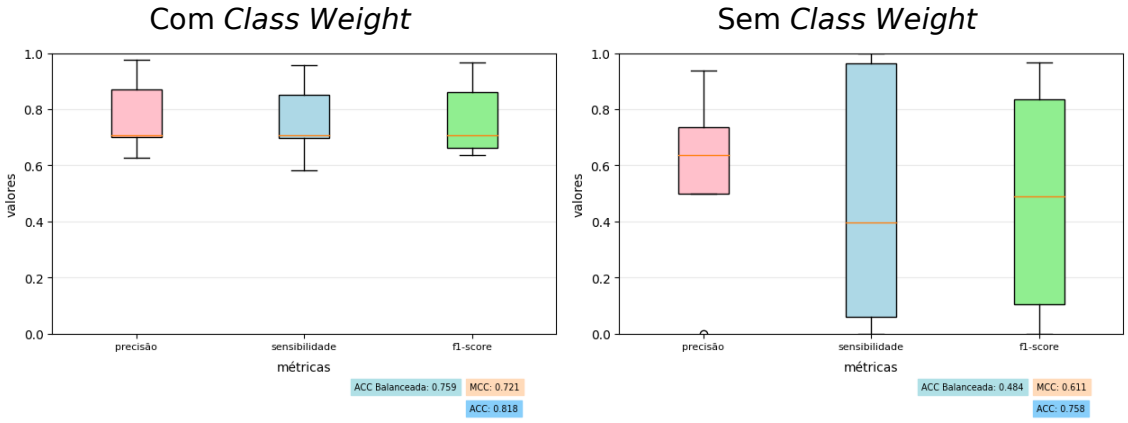
Os resultados com o modelo SVM quase não mudaram; de fato, em alguns casos ficaram idênticos. Portanto, podemos dizer que o balanceamento dos dados usando o

Figura 5.30 – Anomalia no experimento com modelo de árvores de decisão para estratégia de estágio inicial utilizando dados de COAD sem parâmetro *class_weight*



Fonte: O Autor

Figura 5.31 – Anomalia no experimento com modelo de florestas aleatórias para estratégia de estágio inicial utilizando dados de BRCA sem parâmetro *class_weight*



Fonte: O Autor

class_weight não causou alteração nos resultados usando o modelo SVM. Essa não alteração do SVM é a maior das anomalias, pois SVMs são conhecidos por sua sensibilidade a dados desbalanceados (AWAD et al., 2015).

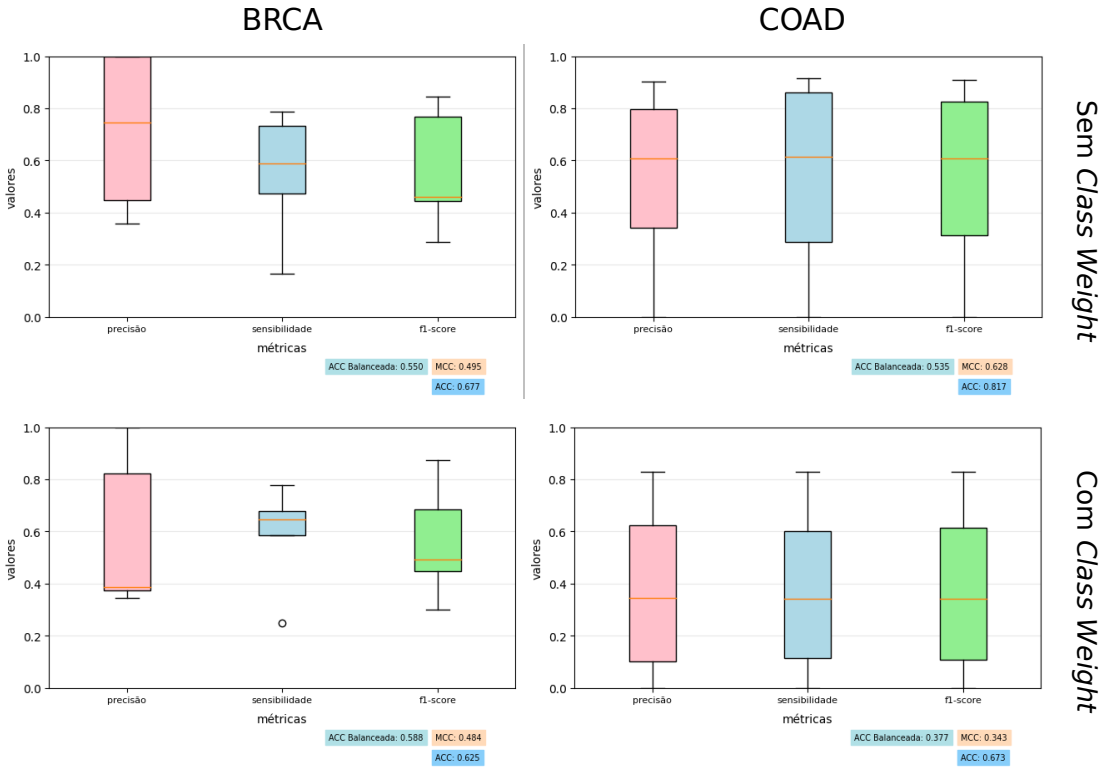
5.2.2 Integração de estágio intermediário

Na estratégia de estágio intermediário, como existe um passo de processamento feito por outros algoritmos (CIMLR e NEMO), novamente o desbalanceamento se acentua devido ao maior número de processamentos necessários sobre os dados. Entretanto, alguns casos anômalos mostraram resultados ainda melhores que os experimentos com tratamento do desbalanceamento dos dados.Todos esses casos aconteceram pela mesma

causa: mais ênfase no aprendizado de classes que estão em maior quantidade nos dados, logo, resultando em mais acertos.

No modelo de árvores de decisão, houveram algumas melhoras em outras estratégias de integração; assim, nessa não seria diferente. Podemos notar na Figura 5.32 que quando não existe a tentativa de balancear as classes, a variação das métricas é sempre maior. Entretanto, a mediana dos resultados é maior quando não estão balanceadas. Como explicado antes, isso acontece pois existem mais instâncias de algumas classes e sem balanceamento essas instâncias são favorecidas, culminando em maior acerto geral.

Figura 5.32 – Anomalia no experimento com modelo de árvores de decisão para estratégia de estágio intermediário com conjunto de dados com todos os atributos sem parâmetro *class_weight* e hiperparâmetros que mais se repetem

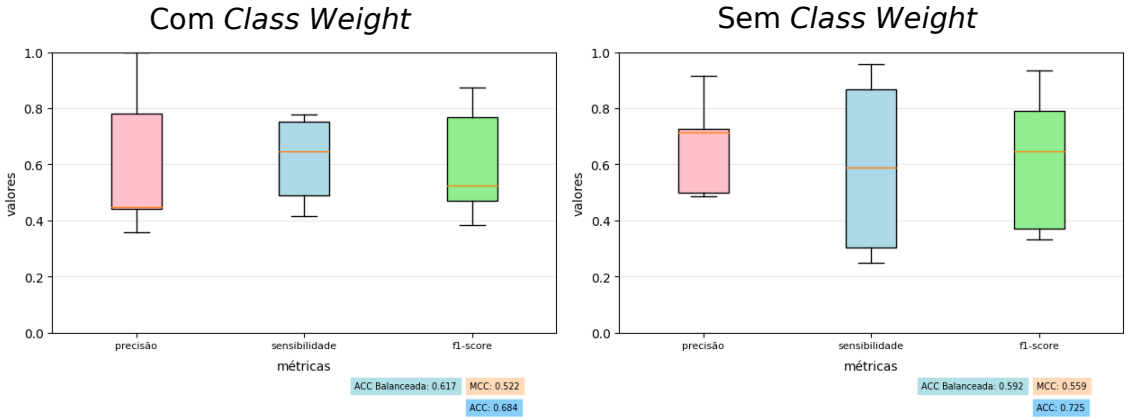


Fonte: O Autor

No caso das florestas aleatórias, temos apenas dois casos de melhora significativa, quando fazemos os testes com o conjunto de dados *f-values* e com o conjunto com *y-data* para os dados de BRCA. Na Figura 5.33, temos o caso do conjunto de *f-values* que, curiosamente, quando não possuía tratamento do desbalanceamento de dados, mostrou uma precisão com menor variação. A explicação é, novamente, a ênfase dada às classes com maior quantidade de instâncias.

Observamos o mesmo comportamento do parágrafo anterior com os dados *y-data*.

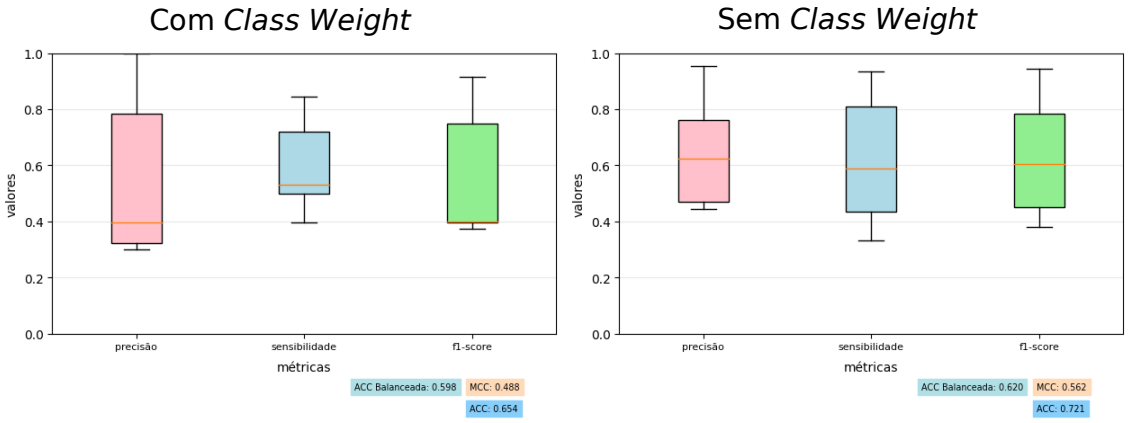
Figura 5.33 – Anomalia no experimento com modelo de florestas aleatórias para estratégia de estágio intermediário com conjunto de dados *f-values* para dados de BRCA sem parâmetro *class_weight* e hiperparâmetros que mais se repetem



Fonte: O Autor

A precisão melhora, devido à ênfase nas classes mais representadas, e a sensibilidade tende a apresentar maior variabilidade, conforme pode ser visto na Figura 5.34.

Figura 5.34 – Anomalia no experimento com modelo de florestas aleatórias para estratégia de estágio intermediário com conjunto de dados *y-data* para dados de BRCA sem parâmetro *class_weight* e hiperparâmetros que mais se repetem



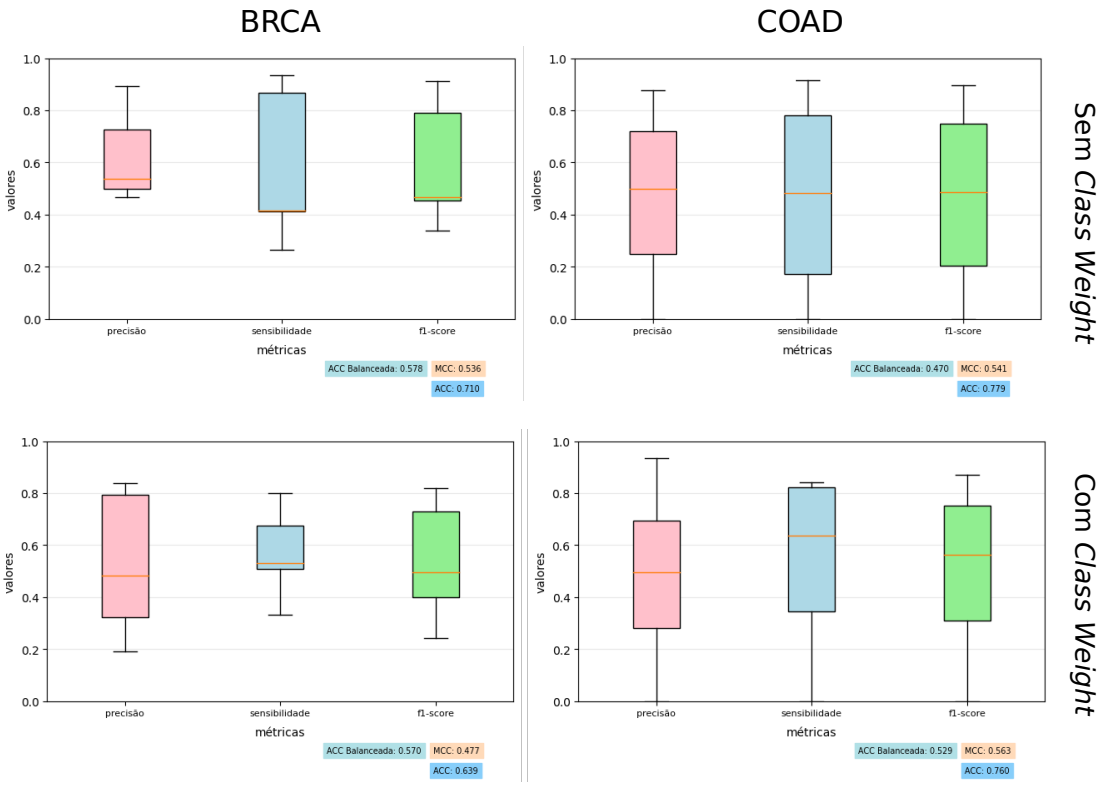
Fonte: O Autor

O SVM também teve casos com melhoras para conjunto com todos os atributos concatenadas. Observamos na Figura 5.35 que o comportamento foi muito parecido com o das florestas aleatórias, só que ainda mais acentuado.

Por fim, temos o NEMO, nos casos usando dados de COAD ele se saiu melhor com dados desbalanceados. O motivo é o mesmo, é mais fácil para os modelos aprenderem classes com maior quantidade de informações. Como exemplo apresentamos apenas os resultados do SVM, pois os resultados dos algoritmos foram muito próximos, com diferenças de 0,001%. Portanto, na Figura 5.36 temos a comparação com e sem *class_weight*,

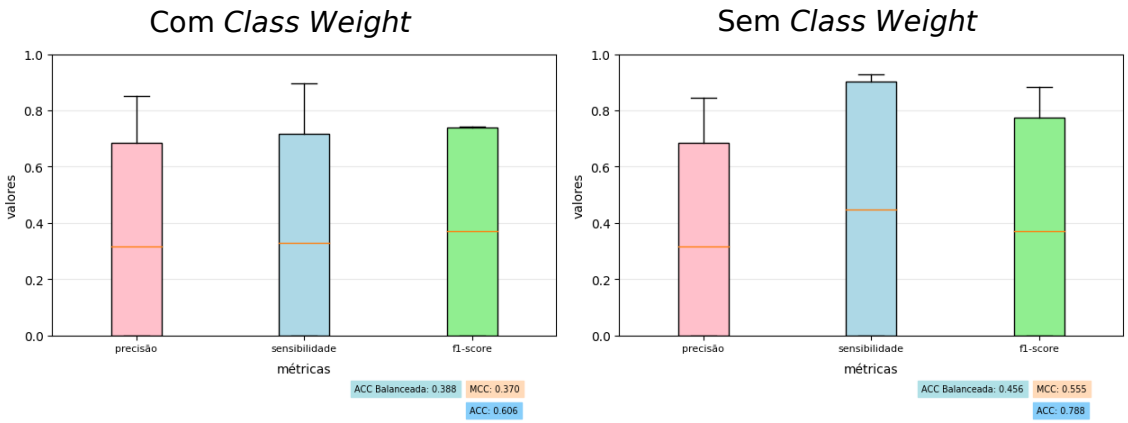
i.e., com e sem balanceamento de dados.

Figura 5.35 – Anomalia no experimento com modelo de SVM para estratégia de estágio intermediário com conjunto de dados com todos os atributos sem parâmetro *class_weight* e hiperparâmetros que mais se repetem



Fonte: O Autor

Figura 5.36 – Anomalia no experimento com modelo de SVM para estratégia de estágio intermediário com *clusters* do algoritmo NEMO sem parâmetro *class_weight* e hiperparâmetros que mais se repetem

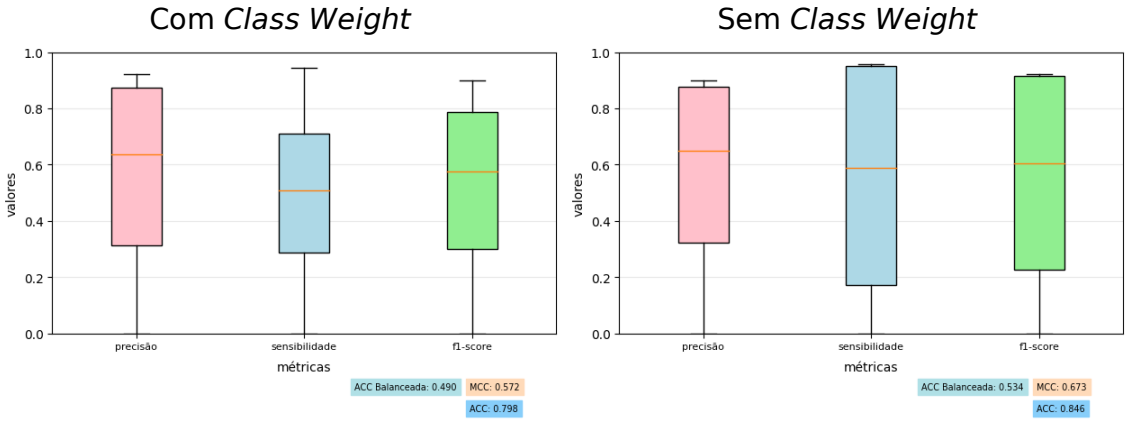


Fonte: O Autor

5.2.3 Integração de estágio final

As mesmas anomalias encontradas na integração de estágio inicial podem ser observadas nos experimentos com integração de estágio final, porém com algumas mudanças. No caso da anomalia dos experimentos com COAD usando modelo de árvores de decisão, a mudança foi que os dados não balanceados aumentaram muito o número de falsos positivos e de falsos negativos. Entretanto, se observarmos a Figura 5.37, a mediana das duas métricas ficaram muito próximas, mostrando nitidamente, que o modelo aprendeu muito bem as instâncias pertencentes às classes majoritárias e não deu a mesma ênfase às demais. Assim, observou-se a maior variação nas métricas para dados não balanceados. Novamente, quando o *class_weight* estava ativo, o modelo tentou aprender as outras instâncias, assim prejudicando seu aprendizado das instâncias das classes majoritárias.

Figura 5.37 – Anomalia no experimento com modelo de árvores de decisão para estratégia de estágio final utilizando dados de COAD sem parâmetro *class_weight*



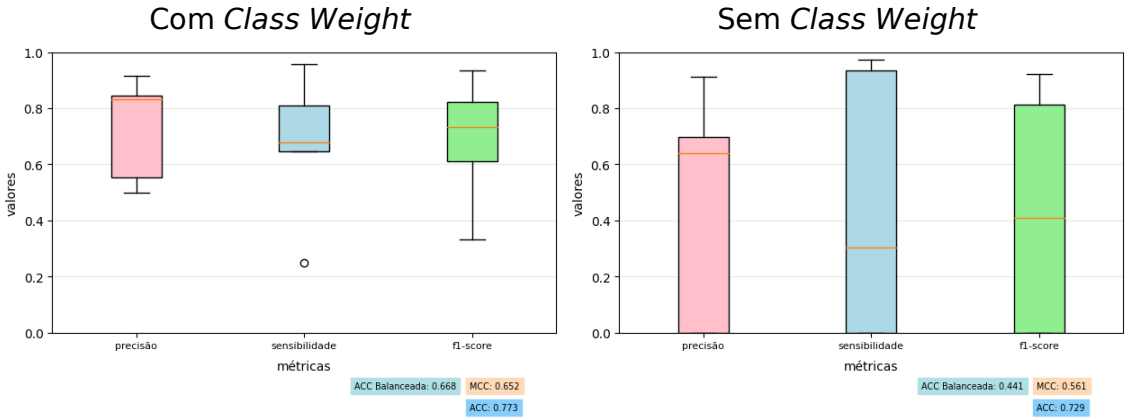
Fonte: O Autor

O caso das florestas aleatórias, ficou ainda mais nítido a grande variação dos valores das métricas utilizando a estratégia de estágio final. Os resultados são sumarizados na Figura 5.38, o comportamento apresentado na figura aponta que os resultados são menos confiáveis sem o parâmetro *class_weight*.

O modelo SVM, assim como aconteceu na integração de estágio inicial, não teve grandes mudanças. Portanto, os algoritmos que foram mais sensíveis a esses balanceamentos, foram os algoritmos de árvores de decisão e florestas aleatórias. Por fim, nota-se que a estratégia de estágio final sofre mais com o desbalanceamento de dados, pois executa individualmente cada ômica o que acaba acentuando os problemas de desbalancea-

mento.

Figura 5.38 – Anomalia no experimento com modelo de florestas aleatórias para estratégia de estágio final utilizando dados de BRCA sem parâmetro *class_weight*



Fonte: O Autor

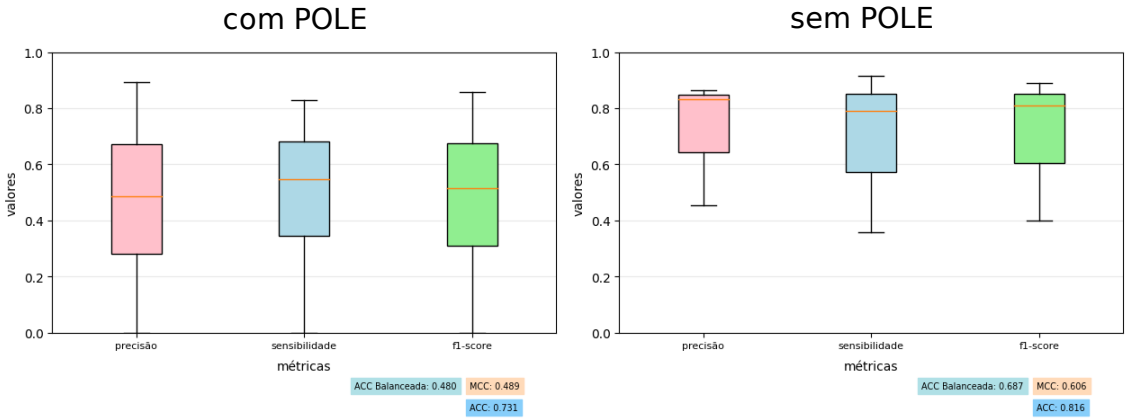
5.3 Experimentos com COAD sem a classe POLE

Os resultados com os dados de COAD melhoraram muito sem a classe POLE (classe minoritária). As variações de precisão e sensibilidade reduziram drasticamente na grande maioria dos casos. Com isso, as suspeitas com o desbalanceamento dos dados se confirmaram e talvez por isso não obtivemos resultados tão bons nos experimentos anteriores para este conjunto de dados. Em todos os casos, a acurácia balanceada melhorou, mas em algumas situações MCC e acurácia normal pioraram um pouco. Na Figura 5.39, temos os gráficos da estratégia de estágio inicial para o modelo de árvores de decisão com e sem a classe POLE, ambos lado a lado. É visível a melhora na variação das métricas, mostrando que o modelo teve muito mais facilidade para identificar as classes do problema.

Para todos os modelos avaliados neste conjunto de experimentos, os resultados tiveram mesmo efeito, com o processo de aprendizado tornando-se aparentemente mais fácil e com as métricas apresentando menos variações. Entretanto, algumas coisas curiosas aconteceram, principalmente no caso do modelo de florestas aleatórias.

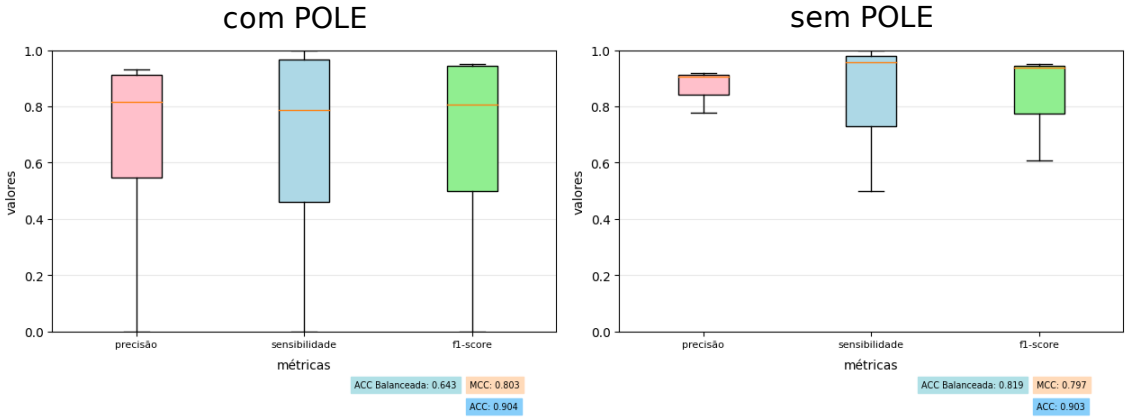
Na Figura 5.40, referente ao modelo de florestas aleatórias, podemos ver uma grande melhora na acurácia balanceada, uma alteração sutil na acurácia normal, mas uma queda no MCC. O MCC, como dito anteriormente, é uma medida da qualidade da classificação e essa queda no seu valor pode ter relação com a variação da métrica de sensibili-

Figura 5.39 – Comparação de execução da estratégia de estágio inicial com modelo de árvores de decisão com e sem as instâncias POLE dos dados de câncer COAD



Fonte: O Autor

Figura 5.40 – Comparação de execução da estratégia de estágio inicial com modelo de florestas aleatórias com e sem as instâncias POLE dos dados de câncer COAD



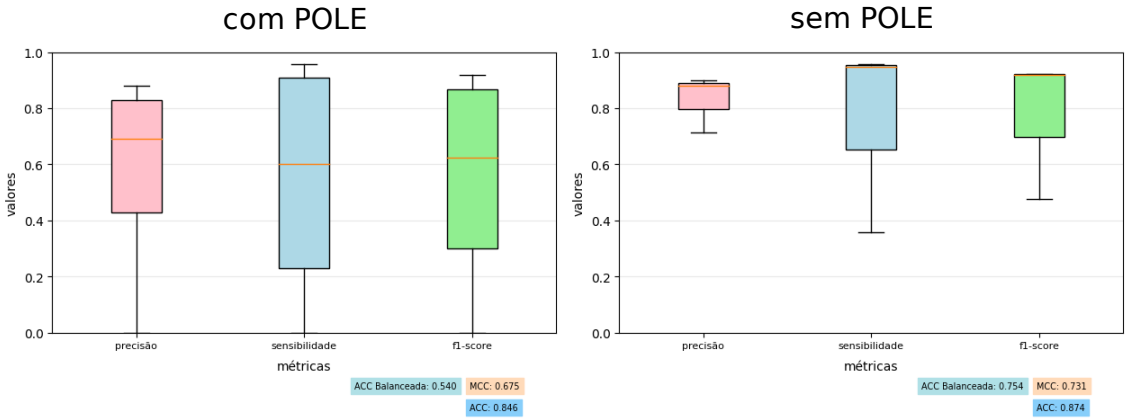
Fonte: O Autor

dade que continua grande.

Outro ponto que podemos observar é que as estratégias que mais se beneficiaram com essa remoção de instâncias foram as estratégias de estágio final e inicial. A estratégia de estágio final teve os maiores ganhos, ultrapassando a estratégia de estágio inicial em alguns casos. Na Figura 5.41 temos um dos modelos que surpreenderam com os ganhos nesse experimento na estratégia de estágio final. A surpresa aconteceu, pois nesse experimento com o modelo SVM, a estratégia de estágio final conseguiu ultrapassar os valores das métricas da estratégia de estágio inicial. Apenas para comparação do que foi mencionado, na Figura 5.42 temos lado a lado as métricas da execução com o modelo SVM para estratégia de estágio final e inicial.

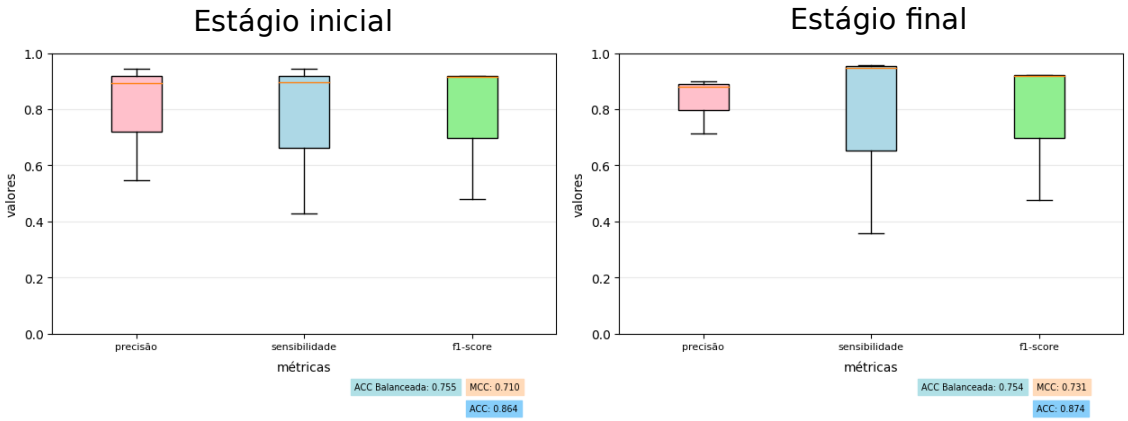
Finalmente, podemos notar o grande impacto que existe nos resultados dos expe-

Figura 5.41 – Comparação de execução da estratégia de estágio final com modelo de SVM com e sem as instâncias POLE dos dados de câncer COAD



Fonte: O Autor

Figura 5.42 – Comparação de execução da estratégia de estágio final com estratégia de estágio inicial usando modelo de SVM sem as instâncias POLE dos dados de câncer COAD



Fonte: O Autor

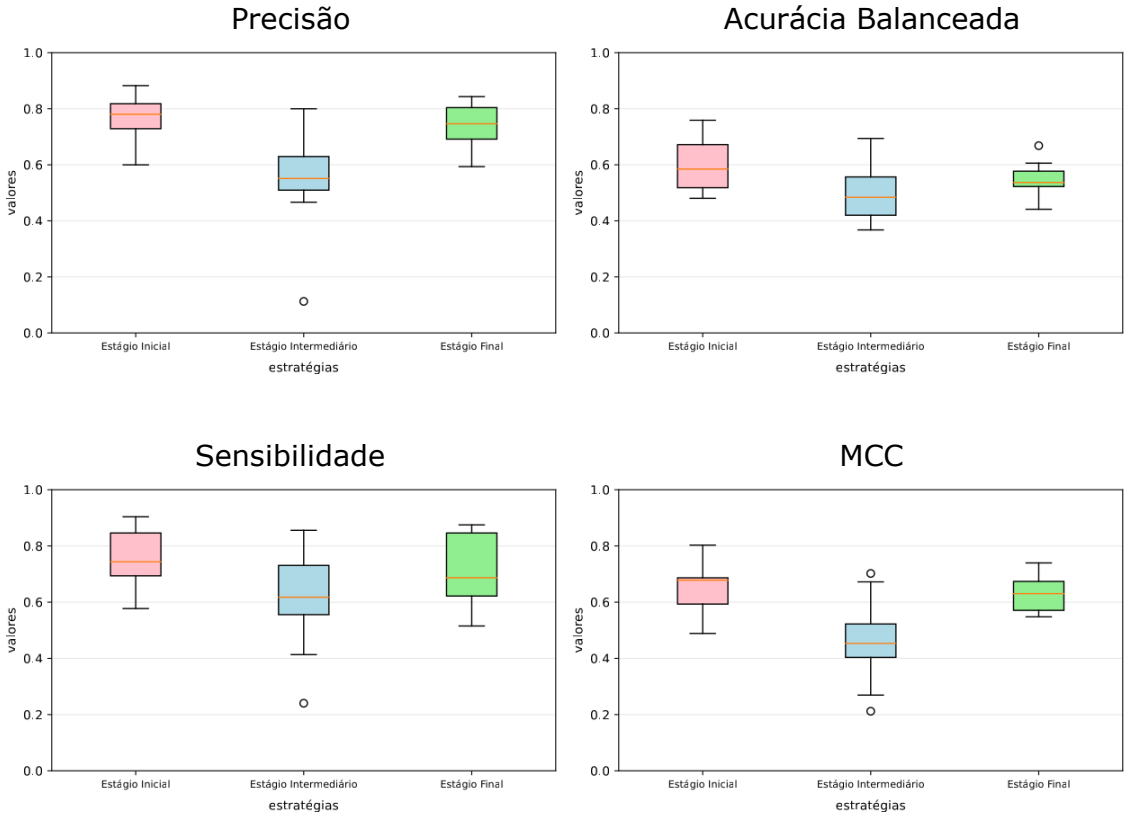
rimentos devido ao desbalanceamento dos dados. Com a simples remoção de um tipo de classe dos dados de treinamento, a qual representava uma classe minoritária e com muito poucas instâncias disponíveis, os resultados tiveram um crescimento médio de acurácia balanceada de 18% na estratégia de estágio final e 19% na estratégia de estágio inicial.

5.4 Comparativo entre estratégias

Na Figura 5.43 podemos visualizar um sumário das quatro principais métricas retiradas dos experimentos desse trabalho (Precisão, Sensibilidade, Acurácia Balanceada e MCC), comparando as estratégias lado a lado. Como mencionado anteriormente, a estratégia de estágio inicial foi a que obteve os melhores resultados por diversos fatores já

destacados. Entretanto, através da análise destes resultados, é importante notarmos como a escolha da estratégia impacta nos resultados das métricas, e consequentemente, dos experimentos. A estratégia de estágio intermediário foi a mais impactada pelos problemas citados nesse capítulo. Podemos observar que em todos os casos envolvendo esta estratégia de integração existem variações maiores no desempenho e um maior numero de *outliers* comparado com as demais estratégias.

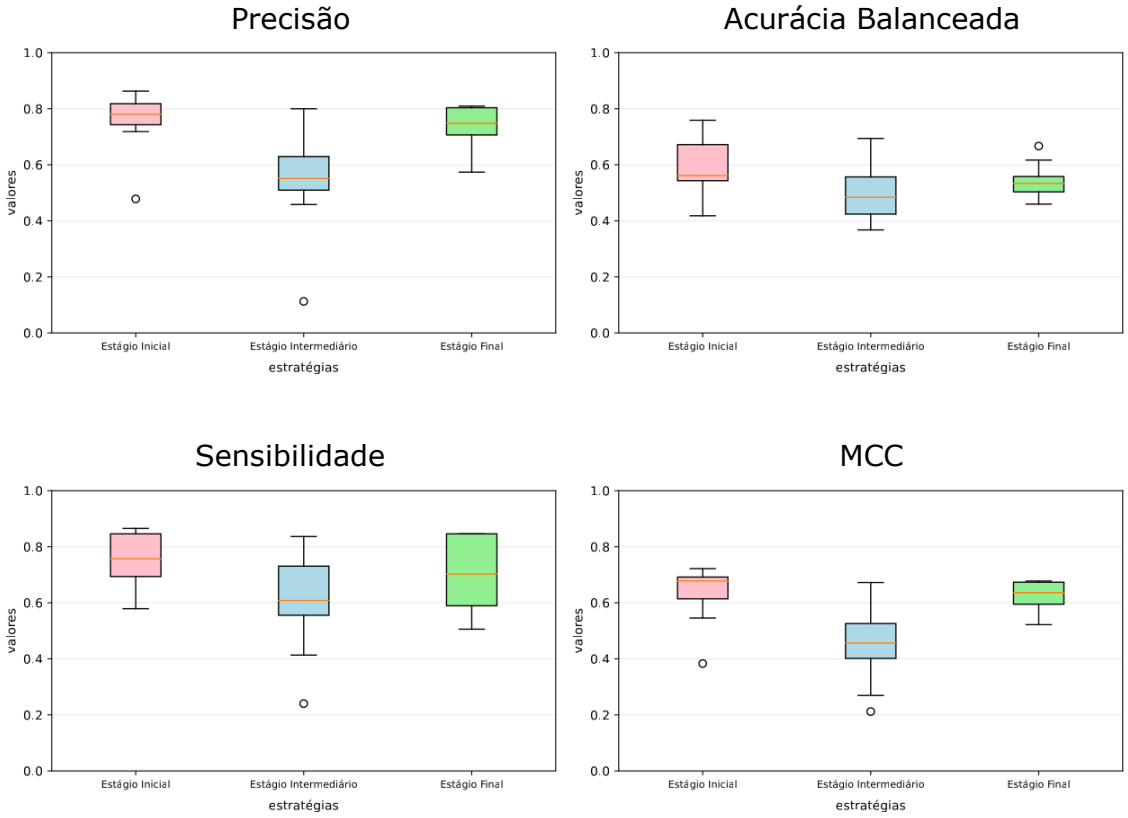
Figura 5.43 – Comparativo de métricas entre estratégias de integração de dados ômicos utilizando hiperparâmetros que mais se repetem com resultados de BRCA e COAD



Fonte: O Autor

Podemos extrair as mesmas conclusões observando a Figura 5.44, que faz a comparação entre as estratégias, porém utilizando a seleção dos hiperparâmetros com maior desempenho de validação. Pode-se notar que os resultados utilizando os hiperparâmetros que mais se repetem foram melhores.

Figura 5.44 – Comparativo de métricas entre estratégias de integração de dados ômicos utilizando hiperparâmetros com maior desempenho de validação com resultados de BRCA e COAD



Fonte: O Autor

Por fim, sumarizamos os melhores resultados de cada análise comparativa realizadas no trabalho:

- **Melhor estratégia de integração de dados ômicos**: Estratégia de estágio inicial
- **Melhor algoritmo de integração de dados ômicos**: CIMLR (*Cancer Integration via Multi-kernel Learning*)
- **Melhor estratégia de seleção de hiperparâmetros**: Hiperparâmetros que mais se repetem
- **Modelo de AM com melhor desempenho geral**: Florestas aleatórias

6 CONCLUSÃO

Com base nos resultados obtidos neste estudo, podemos concluir que a escolha da estratégia de integração de dados ômicos é um fator crítico que influencia significativamente a qualidade das análises realizadas pelos modelos de AM. A relação entre a escolha da estratégia e os ganhos nas métricas de análises depende do objetivo que se busca. A estratégia de estágio inicial é boa para identificar padrões ou relacionamentos entre os dados que abrangem várias ômicas diferentes. Dessa forma, a estratégia de estágio inicial atingiu os melhores resultados nos experimentos. Esse bom desempenho, aliado à fácil aplicabilidade da técnica, a torna uma boa candidata em geral.

No caso da estratégia de estágio intermediário, sua qualidade está em ajudar a reduzir a complexidade da análise dos dados pelos modelos, apesar de ser mais difícil de aplicar por depender muito do algoritmo utilizado para a transformação intermediária dos dados. Na estratégia de estágio final, sua qualidade está em identificar padrões ou relacionamentos específicos em uma única ômica. Como neste trabalho foram analisados dados de múltiplas ômicas diferentes com suas relações para classificar subtipos de câncer, faz sentido que os melhores resultados tenham sido obtidos com os experimentos da estratégia de estágio inicial.

Em relação à qualidade do aprendizado dos modelos, nota-se que eles têm mais dificuldade em aprender utilizando as estratégias que passam por mais processamentos, ou seja, estratégias de estágio final e intermediário. Quanto mais processos, execuções e algoritmos diferentes a estratégia aplica, maiores são as chances de erros se acumularem. No caso da estratégia de estágio inicial, a única coisa que ela faz é concatenar os dados. Ainda que isto possa introduzir algum prejuízo em razão da maior dimensionalidade dos dados, a chance de erros aparenta ser bem menor do que nas demais estratégias.

Outro ponto importante a ser ressaltado são os ganhos e perdas ao trocarmos o tipo de algoritmo utilizado na análise dos dados. Por exemplo, quando trocamos do modelo de árvores de decisão para florestas aleatórias, estamos basicamente trocando para uma versão mais robusta de um mesmo modelo, logo, é perfeitamente normal que os resultados sejam bem melhores. Entretanto, ao trocar do modelo de florestas aleatórias para SVM, geralmente vemos uma queda no desempenho. Essa diminuição na capacidade preditiva do modelo acontece porque o SVM, apesar de bom com dados de alta dimensionalidade, se sai melhor com separações mais claras entre as classes, o que não é o caso dos tipos de dados que estamos analisando neste trabalho.

A principal dificuldade encontrada na execução desse trabalho foi o desbalanceamento dos dados e a aplicação dos algoritmos de integração de dados ômicos na estratégia de estágio intermediário. Buscamos diversas formas de contornar o desbalanceamento dos dados. Além disso, executamos experimentos para mostrar especificamente essa grande dificuldade, esse é o caso dos experimentos sem a instância POLE dos dados de COAD (Seção 5.3). Entretanto, no caso da aplicação dos algoritmos de integração, contornamos buscando utilizar algoritmos mais simples de serem aplicados, mas ainda assim foi uma limitação encontrada no trabalho. Essa dificuldade se deve, principalmente, ao fato destes algoritmos serem usados em contextos de forma muito mais específica, com modelos criados especificamente para alguns tipos de análises (como classificação de subtipos de tumor). Além disso, salienta-se que estes algoritmos são implementados com diferentes linguagens de programação, em diferentes versões, assumindo diferentes formatos de entrada, dificultando a sua integração em um pipeline experimental.

No que diz respeito a trabalhos futuros, acreditamos que alguns pontos poderiam ser explorados: (i) Adequar ou utilizar modelos mais específicos para os algoritmos de integração de dados como CIMLR e NEMO e assim extrair o máximo desses algoritmos; (ii) Executar a estratégia de estágio final com outros tipos de funções para combinação de predições e não apenas uma votação majoritária; (iii) Executar testes com algoritmos de integração de dados ômicos mais distintos, ou seja, utilizando estratégias diferentes disponíveis na literatura; (iv) Ampliar os experimentos para novos conjuntos de dados, explorando dados melhores balanceados para focar mais a análise nas estratégias de integração de dados.

Por fim, no desenvolvimento desse trabalho foram utilizados muitos conhecimentos adquiridos durante toda a graduação em Ciência da Computação. Alguns desses conhecimentos são a análise e tratamento de dados para AM, técnicas de construção de programas, planejamento e desenvolvimento de pipelines, bem como organização, tratamento e agrupamento de dados. Assim, o conhecimento adquirido durante os anos de graduação permitiu o desenvolvimento deste trabalho.

REFERÊNCIAS

ASADA, K. et al. Uncovering prognosis-related genes and pathways by multi-omics analysis in lung cancer. **Biomolecules**, MDPI, v. 10, n. 4, p. 524, 2020.

AWAD, M. et al. Support vector machines for classification. **Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers**, Springer, p. 39–66, 2015.

BLAGDEN, S. P.; WILLIS, A. E. The biological and therapeutic relevance of mrna translation in cancer. **Nature reviews Clinical oncology**, Nature Publishing Group, v. 8, n. 5, p. 280–291, 2011.

BUHMANN, M. D. **Radial Basis Functions: Theory and Implementations**. [S.l.]: Cambridge University Press, 2003. (Cambridge Monographs on Applied and Computational Mathematics).

CAI, Z. et al. Machine learning for multi-omics data integration in cancer. **iScience**, v. 25, n. 2, p. 103798, 2022. ISSN 2589-0042. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S2589004222000682>.

CAWLEY, G. C.; TALBOT, N. L. On over-fitting in model selection and subsequent selection bias in performance evaluation. **The Journal of Machine Learning Research**, JMLR. org, v. 11, p. 2079–2107, 2010.

DAS, R. S. P. M. Dna methylation and cancer. **Journal of Clinical Oncology**, Nov 2004. Available from Internet: <https://doi.org/10.1200/JCO.2004.07.151>. Accessed in: 25 set. 2022.

DUAN, R. et al. Evaluation and comparison of multi-omics data integration methods for cancer subtyping. **PLoS computational biology**, Public Library of Science San Francisco, CA USA, v. 17, n. 8, p. e1009224, 2021.

FAWCETT, T. ROC graphs: Notes and practical considerations for researchers. **Machine learning**, Citeseer, v. 31, n. 1, p. 1–38, 2004.

HASIN, Y.; SELDIN, M.; LUSIS, A. Multi-omics approaches to disease. **Genome Biology**, BioMed Central, v. 18, n. 1, p. 1–15, 2017.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. Random forests. In: **The elements of statistical learning**. [S.l.]: Springer, 2009. p. 587–604.

HE, H. et al. **Biostatistics, Data Mining and Computational Modeling**. Springer Dordrecht, 2016. Available from Internet: <https://link.springer.com/chapter/10.1007/978-94-017-7543-4_2#citeas>. Accessed in: 25 set. 2022.

KARCZEWSKI, K. J.; SNYDER, M. P. Integrative omics for health and disease. **Nature Reviews Genetics**, Nature Publishing Group UK London, v. 19, n. 5, p. 299–310, 2018.

KAVITHA, C. et al. Early-stage alzheimer's disease prediction using machine learning models. **Front Public Health**, Mar 2022. Available from Internet: <https://doi.org/10.3389/fpubh.2022.853294>.

MA, B. et al. Diagnostic classification of cancers using extreme gradient boosting algorithm and multi-omics data. **Computers in Biology and Medicine**, Elsevier, v. 121, p. 103761, 2020.

MAYORAZ, E.; ALPAYDIN, E. Support vector machines for multi-class classification. In: SPRINGER. **Engineering Applications of Bio-Inspired Artificial Neural Networks: International Work-Conference on Artificial and Natural Neural Networks, IWANN'99 Alicante, Spain, June 2–4, 1999 Proceedings, Volume II**. [S.l.], 2006. p. 833–842.

MENDOZA, M. R. **Exploring ensemble learning techniques to optimize the reverse engineering of gene regulatory networks**. Thesis (PhD) — Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil, March 2014.

OLIVIER, M. et al. The need for multi-omics biomarker signatures in precision medicine. **International Journal of Molecular Sciences**, v. 20, n. 19, 2019. ISSN 1422-0067. Available from Internet: <https://doi.org/10.3390/ijms20194781>. Accessed in: 11 out. 2022.

PENG, Y.; CROCE, C. M. The role of micrornas in human cancer. **Signal Transduction and Targeted Therapy**, Jan 2016. Available from Internet: <https://doi.org/10.1038/sigtrans.2015.4>. Accessed in: 25 set. 2022.

PICARD, M. et al. Integration strategies of multi-omics data for machine learning analysis. **Computational and Structural Biotechnology Journal**, v. 19, p. 3735–3746, 2021. ISSN 2001-0370. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S2001037021002683>.

RAJAMOHANA, S. et al. Analysis of classification algorithms for breast cancer prediction. In: SPRINGER. **Data Management, Analytics and Innovation: Proceedings of ICDMAI 2019, Volume 1**. [S.l.], 2020. p. 517–528.

RAPPOPORT, N.; SHAMIR, R. Multi-omic and multi-view clustering algorithms: review and cancer benchmark. **Nucleic acids research**, Oxford University Press, v. 46, n. 20, p. 10546–10562, 2018.

RAPPOPORT, N.; SHAMIR, R. Nemo: cancer subtyping by integration of partial multi-omic data. **Bioinformatics**, Oxford University Press, v. 35, n. 18, p. 3348–3356, 2019.

REEL, P. S. et al. Using machine learning approaches for multi-omics data analysis: A review. **Elsevier Biotechnology Advances**, Dec 2020. Available from Internet: <https://doi.org/10.1016/j.biotechadv.2021.107739>. Accessed in: 25 set. 2022.

REEL, P. S. et al. Using machine learning approaches for multi-omics data analysis: A review. **Biotechnology Advances**, v. 49, p. 107739, 2021. ISSN 0734-9750. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S0734975021000458>.

SALZBERG, S. L. Open questions: How many genes do we have? **BMC Biology**, BioMed Central, v. 16, n. 1, p. 1–3, 2018.

SANI, H. M.; LEI, C.; NEAGU, D. Computational complexity analysis of decision tree algorithms. In: SPRINGER. **Artificial Intelligence XXXV: 38th SGAI International**

Conference on Artificial Intelligence, AI 2018, Cambridge, UK, December 11–13, 2018, Proceedings 38. [S.l.], 2018. p. 191–197.

SHLIEN, A.; MALKIN, D. Copy number variations and cancer. **Genome Medicine**, Jun 2009. Available from Internet: <https://doi.org/10.1186/gm62>. Accessed in: 25 set. 2022.

SPICKER, J. S. et al. Integration of Clinical Chemistry, Expression, and Metabolite Data Leads to Better Toxicological Class Separation. **Toxicological Sciences**, v. 102, n. 2, p. 444–454, 01 2008. ISSN 1096-6080. Available from Internet: <https://doi.org/10.1093/toxsci/kfn001>.

TOMCZAK, K.; CZERWIŃSKA, P.; WIZNEROWICZ, M. Review the cancer genome atlas (tcga): an immeasurable source of knowledge. **Contemporary Oncology**, Termedia, v. 2015, n. 1, p. 68–77, 2015.

VARMA, S.; SIMON, R. Bias in error estimation when using cross-validation for model selection. **BMC Bioinformatics**, BioMed Central, v. 7, n. 1, p. 1–8, 2006.

WANG, B. et al. Similarity network fusion for aggregating data types on a genomic scale. **Nature Methods**, v. 11, n. 3, p. 333–337, Mar 2014. ISSN 1548-7105. Available from Internet: <https://doi.org/10.1038/nmeth.2810>.

YAO, Q.; CHEN, Y.; ZHOU, X. The roles of microrna s in epigenetic regulation. **Current Opinion in Chemical Biology**, v. 51, p. 11–17, 2019. ISSN 1367-5931. Chemical Genetics and Epigenetics • Molecular Imaging. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S1367593118301868>. Accessed in: 11 out. 2022.

ZARREI, M. et al. A copy number variation map of the human genome. **Nature Reviews Genetics**, Feb 2015. Available from Internet: <https://doi.org/10.1038/nrg3871>. Accessed in: 25 set. 2022.

ZITNIK, M. et al. Machine learning for integrating data in biology and medicine: Principles, practice, and opportunities. **Information Fusion**, Elsevier, v. 50, p. 71–91, 2019.