Universidade Federal do Rio Grande do Sul
Centro de Biotecnologia
Biotecnologia - Ênfase em Bioinformática

Análise comparativa de conversores de dados tabulares em imagem para classificação de genes diferencialmente expressos em CNNs

Porto Alegre
2026

André Gustavo Dessoy Hübner

Análise comparativa de conversores de dados tabulares em imagem para classificação de genes diferencialmente expressos em CNNs

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de bacharel em Biotecnologia - Ênfase em Bioinformática do Centro de Biotecnologia do Estado do Rio Grande do Sul da Universidade Federal do Rio Grande do Sul.
Orientador: Prof. Dr. Marcio Dorn

Porto Alegre
2026

CIP - Catalogação na Publicação

Dessoy Hübner, André Gustavo
Análise comparativa de conversores de dados tabulares em imagem para classificação de genes diferencialmente expressos em CNNs / André Gustavo Dessoy Hübner. -- 2026.
79 f.
Orientador: Marcio Dorn.

Trabalho de conclusão de curso (Graduação) -- Universidade Federal do Rio Grande do Sul, Instituto de Biociências, Curso de Biotecnologia: Bioinformática, Porto Alegre, BR-RS, 2026.

1. dados tabulares. 2. redes neurais convolucionais. 3. expressão gênica diferencial. 4. RNA-seq. 5. conversão tabela-para-imagem. I. Dorn, Marcio, orient. II. Título.

Elaborada pelo Sistema de Geração Automática de Ficha Catalográfica da UFRGS com os dados fornecidos pelo(a) autor(a).

André Gustavo Dessoy Hübner

Análise comparativa de conversores de dados tabulares em imagem para classificação de genes diferencialmente expressos em CNNs

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de bacharel em Biotecnologia - Ênfase em Bioinformática do Centro de Biotecnologia do Estado do Rio Grande do Sul da Universidade Federal do Rio Grande do Sul.

Orientador: Prof. Dr. Marcio Dorn

Aprovado em: Porto Alegre, 9 de julho de 2026.

BANCA EXAMINADORA:

Dr. Marcio Dorn

Departamento de Informática Teórica

Dr. Bruno Iochins Grisci

Departamento de Informática Teórica

Dr. Anderson Rocha Tavares

Departamento de Informática Aplicada

AGRADECIMENTOS

Gostaria de agradecer ao meu orientador Prof. Dr. Marcio Dorn, que me apresentou ao tema das possíveis vantagens advindas da aplicação de CNNs em tabelas e cuja ajuda foi fundamental para delimitar o escopo do trabalho.

Também agradeço imensamente aos colegas e professores com quem convivi na graduação. A diversidade de pensamentos, focos e especializações com que tive contato tornaram essa jornada enriquecedora de uma forma única e me ajudaram a ser uma pessoa melhor.

Enfim, evidentemente sou eternamente grato à minha família, em particular a meus pais, Guido e Sueli, e aos meus irmãos, Ana e Daniel, sem cujo apoio tudo isso não seria possível e sequer teria começado.

“A loucura, objeto dos meus estudos, era até
agora uma ilha perdida no oceano da razão;
começo a suspeitar que é um continente”
Machado de Assis, *O alienista*

RESUMO

Este trabalho avaliou comparativamente conversores de dados tabulares em imagens para a classificação de genes diferencialmente expressos por redes neurais convolucionais. Para isso, foram utilizados dados públicos de RNA-seq de três projetos do TCGA, correspondentes aos cânceres LUAD, KIRC e PRAD. As matrizes de contagem foram processadas com etapas de filtragem, normalização e análise de expressão diferencial com DESeq2, permitindo a construção de tabelas gene-cêntricas contendo características como expressão média em amostras tumorais e normais, variância, dispersão, log2FC, valor-p e valor-p ajustado. A partir dessas tabelas, genes foram classificados em três categorias: regulados positivamente (UR), regulados negativamente (DR) ou sem diferença significativa (ND). Em seguida, as tabelas foram convertidas em imagens por três métodos tabela-para-imagem (T2I), DeepInsight, NCTD e HACNet, e classificadas com uma arquitetura ResNet-18 em validação cruzada estratificada com cinco folds. Como referência comparativa, também foram treinados modelos tradicionais de aprendizado de máquina diretamente sobre as tabelas originais. Os resultados indicaram que a conversão T2I é uma estratégia viável para esse tipo de tarefa, mas não superou de forma consistente os modelos tabulares tradicionais, que permaneceram como referências robustas para dados com poucas características. Entre os conversores, observou-se que a qualidade da representação visual depende fortemente da estabilidade espacial, da preservação das informações originais e da forma como a imagem é ajustada aos dados. As análises de enriquecimento funcional GO mostraram que conjuntos de genes classificados pelos modelos apresentaram processos biológicos coerentes com mecanismos associados ao câncer, incluindo resposta imune, comunicação celular e organização tecidual. Conclui-se que conversores T2I têm potencial exploratório para classificação de genes diferencialmente expressos, mas seu desempenho ainda depende da estabilidade e da adequação da representação visual gerada.

Palavras-chave: dados tabulares; redes neurais convolucionais; expressão gênica diferencial; RNA-seq; conversão tabela-para-imagem.

ABSTRACT

This work comparatively evaluated tabular-to-image converters for the classification of differentially expressed genes using convolutional neural networks. Public RNA-seq data from three TCGA projects, corresponding to LUAD, KIRC, and PRAD cancers, were used. Count matrices were processed through filtering, normalization, and differential expression analysis with DESeq2, allowing the construction of gene-centric tables containing features such as mean expression in tumor and normal samples, variance, dispersion, log2FC, p-value, and adjusted p-value. Based on these tables, genes were classified into three categories: upregulated (UR), downregulated (DR), or not significantly differentially expressed (ND). The resulting tables were then converted into images using three T2I methods, DeepInsight, NCTD, and HACNet, and classified with a ResNet-18 architecture under five-fold stratified cross-validation. As a comparative baseline, traditional machine learning models were also trained directly on the original tabular features. The results indicated that T2I conversion is a feasible strategy for this task, but it did not consistently outperform traditional tabular models, which remained strong references for low-dimensional feature sets. Among the converters, the quality of the visual representation was strongly dependent on spatial stability, preservation of the original information, and the way the image representation was fitted to the data. Gene Ontology functional enrichment analyses showed that gene sets classified by the models were associated with biological processes consistent with cancer-related mechanisms, including immune response, cell communication, and tissue organization. Overall, the findings suggest that T2I converters have exploratory potential for the classification of differentially expressed genes, but their performance still depends on the visual representation’s stability and adequacy.

Keywords: tabular data; convolutional neural networks; differential gene expression; RNA-seq; table-to-image conversion.

LISTA DE ILUSTRAÇÕES

Fórmula 1 – Cálculo do valor de pixel no HACNet a partir dos pesos de atenção.....33
Fórmula 2 – Definição dos pesos de atenção por Gumbel-Softmax no HACNet.....34
Fórmula 3 – Função de perda do HACNet combinando entropia cruzada e erro quadrático médio.....34
Figura 1 – Esquematização geral do projeto.....36
Figura 2 – Resumo da geração das imagens quanto ao seu mecanismo, propriedades e características.....42
Gráfico 1 – Gráfico de vulcão para as tabelas obtidas com o dataset PRAD.....54
Gráfico 2 – Gráfico MA para as tabelas obtidas com o dataset PRAD.....54
Figura 3 – Exemplos de imagens geradas para as classes ND, DR e UR com o dataset LUAD e conversor DeepInsight.....56
Figura 4 – Exemplos de imagens geradas para as classes ND, DR e UR com o dataset LUAD e conversor DeepInsight.....56
Figura 5 – Exemplos de imagens geradas para as classes DR, ND e UR com HACNet.....57
Figura 6 – 4 imagens geradas com DeepInsight e dataset PRAD para a mesma amostra em diferentes folds.....59
Figura 7 – 4 imagens geradas com DeepInsight usando ajuste geral e dataset PRAD para a mesma amostra em diferentes folds.....61
Gráfico 3 – Diagramas de barras horizontais para comparação dos F1-Score obtidos por todos os métodos empregados.....64
Gráfico 4 – Matrizes de confusão dos melhores modelos tradicional (RF) e baseado em conversores (HACNet) obtidos para os datasets.....66
Gráfico 5 – Matrizes de concordância comparando as predições dos melhores modelos tradicional (RF) e baseado em conversores (HACNet) obtidos para os datasets.....66
Gráfico 6 – Diagramação por barras horizontais das significâncias estatísticas de termos GO enriquecidos negativamente para a combinação NCTD-KIRC.....69

Gráfico 7 – Diagramação por barras horizontais das significâncias estatísticas de termos GO enriquecidos positivamente para a combinação HACNet-KIRC.....70

LISTA DE TABELAS

Tabela 1 – Rótulos dos genes filtrados após análise por DESeq2.....52
Tabela 2 – Desempenho médio dos conversores T2I com ResNet-18 considerando os 5 folds de cada combinação.....59
Tabela 3 – Desempenho médio das aplicações de ML usadas para classificar as tabelas originais dos datasets considerando cinco folds.....62
Tabela 4 – Significância estatística da diferença por valor-p e diferença média entre os resultados de acurácia para as diferentes combinações no dataset PRAD.....65

LISTA DE ABREVIATURAS E SIGLAS

CNN	Convolutional Neural Network (rede neural convolucional)
DEG	Differentially Expressed Gene (gene diferencialmente expresso)
DL	Deep Learning (aprendizado profundo)
DNN	Deep Neural Network (rede neural profunda)
DR	Downregulated gene (gene regulado negativamente)
GDC	Genomic Data Commons
HACNet	Hard Attention-based Converter combined with a Convolutional Neural Network (conversor baseado em atenção rígida combinado com uma rede neural c onvolucional)
KIRC	Kidney Renal Clear Cell Carcinoma (carcinoma de células claras do rim)
log_2FC	log_2 Fold Change (logaritmo na base 2 da razão de mudança)
LR	Logistic Regression (regressão logística)
LUAD	Lung Adenocarcinoma (adenocarcinoma de pulmão)
NCTD	Novel Algorithm for Convolving Tabular Data (novo algoritmo para convolução de dados tabulares)
ND	No Difference gene (gene sem expressão diferencial)
PCA	Principal Component Analysis (análise de componentes principais)
PRAD	Prostate Adenocarcinoma (adenocarcinoma de próstata)
RF	Random Forest (florestas de decisão aleatórias)
SVM	Support Vector Machines (máquinas de vetores de suporte)
TCGA	The Cancer Genome Atlas
t-SNE	t-distributed Stochastic Neighbor Embedding (incorporação estocástica de vizinhos t-distribuída)
XGBoost	Extreme Gradient Boosting (método de aprendizado baseado em árvores com reforço de gradiente)
UR	Upregulated gene (gene regulado positivamente)
XAI	Explainable Artificial Intelligence (inteligência artificial explicável)

SUMÁRIO

1 INTRODUÇÃO.....13
1.1 JUSTIFICATIVA.....15
1.2 OBJETIVO GERAL.....16
1.2.1 Objetivos específicos.....16
2 DESENVOLVIMENTO.....18
2.1 TRABALHOS RELACIONADOS.....18
2.1.1 Conversores tabela-para-imagem para dados tabulares.....18
2.1.2 Aprendizado de máquina e aprendizado profundo em análise de DEGs. .19
2.2 FUNDAMENTAÇÃO TEÓRICA.....21
2.2.1 Dados de RNA-seq e análise de DEGs.....21
2.2.2 Aprendizado de Máquina aplicado à expressão gênica.....23
2.2.3 Redes neurais convolucionais e dados tabulares.....26
2.2.4 Conversão de dados tabulares em imagens e suas metodologias.....28
2.3 MÉTODOS.....31
2.3.1 Delineamento geral do pipeline.....31
2.3.2 Obtenção e pré-processamento dos dados.....33
2.3.3 Análise de expressão diferencial com DESeq2 e construção das tabelas 35
2.3.4 Conversão de dados tabulares em imagens.....37
2.3.4.1 DeepInsight.....39
2.3.4.2 NCTD.....40
2.3.4.3 HACNet.....41
2.3.5 Treinamento dos modelos baseados em CNN.....42
2.3.6 Classificação com modelos tradicionais de aprendizado de máquina.....44
2.4 ANÁLISE E DISCUSSÃO.....49
2.4.1 Tabelas obtidas e visualização dos rótulos iniciais obtidos.....49
2.4.2 Imagens geradas pelos conversores.....52
2.4.3 Classificações por conversor-CNN.....55
2.4.4 Classificação das tabelas com modelos tradicionais de aprendizado de máquina.....58
2.4.5 Comparação entre pipelines conversor-CNN e modelos tradicionais.....60
2.4.6 Enriquecimento funcional GO e análise biológica dos genes classificados 65

3 CONCLUSÃO.....69
REFERÊNCIAS.....73
GLOSSÁRIO.....77

1 INTRODUÇÃO

Redes neurais convolucionais (CNNs) são uma classe de redes neurais profundas (DNNs) especializadas em lidar com tipos de dados homogêneos, ou seja, dados cuja localização espacial das características é relevante para o significado e cujas características estão semanticamente relacionadas, sendo tipicamente imagens (Le Cun *et al.*, 1989). Nos anos recentes, com a crescente abundância de dados de diversos tipos, incluindo os provenientes das diferentes ômicas, as CNNs tornaram-se preponderantes em aplicações de classificação de imagens, texto e áudio (Borisov *et al.*, 2024).

Por outro lado, muitas aplicações dependem de dados estruturados tabulares que, diferentemente de imagens, são compostos por características não necessariamente relacionadas semanticamente, sendo, portanto, dados heterogêneos. Para que se possam desenvolver aplicações de DNN com esse tipo de dado, são necessárias etapas envolvendo seleção e extração de características, um passo desnecessário em CNNs, que realizam esse processo automaticamente por meio da semanticidade do dado (Borisov *et al.*, 2024; Alenizy; Berri, 2025). Dificuldades como essa estão relacionadas à aplicação de DNNs em dados tabulares, os quais ainda se mostram menos competitivos em relação a estratégias tradicionais de Aprendizado de Máquina (ML) devido à sua maior complexidade, enquanto que modelos mais fáceis de serem aplicados, como os métodos de aprendizado baseados em árvores com reforço de gradiente (XGBoost) (Chen; Guestrin, 2016), apresentam excelente desempenho (Borisov *et al.*, 2024; Somepalli *et al.*, 2021).

Uma estratégia que vem se tornando cada vez mais relevante para reduzir essa lacuna é a conversão de dados tabulares em imagens passíveis de aplicação em CNNs, aproveitando suas qualidades intrínsecas para extração automática de características e o alto grau de desenvolvimento que essas arquiteturas atingiram em tarefas de classificação (Borisov *et al.*, 2024; Neto; Neto; Endo, 2023). Essas ferramentas computacionais podem ser chamadas de conversores tabela-para-imagem (T2I). De fato, várias estratégias foram recentemente desenvolvidas com esse objetivo, buscando criar imagens que representem uma alocação semântica

das características do dado original em um espaço bidimensional, permitindo explorar as vantagens das CNNs (Sharma *et al.*, 2019; Bazgir *et al.*, 2020; Zhu *et al.*, 2021a; Neto; Neto; Endo, 2023; Matsuda *et al.*, 2024; Borisov *et al.*, 2024; Alenizy; Berri, 2025).

Para que a utilização de uma estratégia de ML seja bem-sucedida, mais importante do que o modelo em si é a qualidade do conjunto de dados utilizado. No contexto de dados de expressão gênica de RNA-seq, a inspeção de qualidade inclui a verificação de presença de artefatos de adaptadores, a remoção de reads de baixa qualidade e, em geral, a garantia de valores normalizados dos níveis de transcritos de cada amostra, de forma a eliminar possíveis vieses (Conesa *et al.*, 2016). Através do uso de linhagens de câncer do The Cancer Genome Atlas (TCGA) (Weinstein *et al.*, 2013) obtidas de bibliotecas como o TCGABiolinks (Colaprico *et al.*, 2016), é possível obter amostras tratadas com esses pré-processamentos por meio de pipelines padronizados do Genomic Data Commons (GDC) (Grossman *et al.*, 2016).

Por outro lado, dados públicos de grande escala, embora padronizados, ainda exigem etapas adicionais de curadoria e transformação para que possam ser utilizados em tarefas de classificação de DEGs fundamentados em aprendizado profundo (DL). No contexto deste trabalho, os dados de RNA-seq são tratados inicialmente a partir das matrizes de contagem gênicas obtidas por meio do TCGABiolinks, sobre as quais são aplicadas etapas como filtragem de genes de baixa expressão, normalização e análise de expressão diferencial. A partir dos resultados obtidos com DESeq2, torna-se possível construir uma representação tabular centrada em genes, combinando informações de expressão média, variação entre grupos, estatísticas de expressão diferencial e valores de significância ajustados. Essa estrutura permite que cada gene seja representado como uma observação com múltiplas features, adequada para posterior classificação supervisionada por CNN.

Assim, este trabalho propõe um pipeline comparativo composto por três etapas principais: construção de tabelas de características gênicas a partir de dados públicos de RNA-seq de câncer, conversão dessas tabelas em imagens por diferentes métodos T2I e treinamento de modelos baseados em CNN para a classificação dos genes em UR, DR ou não significativos. A comparação entre os conversores considera não apenas o desempenho preditivo obtido pelo classificador,

mas também aspectos relacionados à preservação da informação, estabilidade da representação gerada e potencial de interpretação dos resultados. Essa abordagem busca avaliar se métodos originalmente desenvolvidos para adaptar dados tabulares ao domínio de imagens podem ser úteis em um problema biologicamente relevante e ainda relativamente pouco explorado: a classificação de genes diferencialmente expressos a partir de features derivadas de análises de RNA-seq.

1.1 Justificativa

CNNs têm desempenhado muito bem em diversos tipos de dados homogêneos, como imagens, vídeos e áudios, há bastante tempo. Apesar disso, sua aplicação a dados heterogêneos tabulares através de métodos de conversão em imagem ainda é relativamente recente. Embora os métodos desenvolvidos para realizar essa codificação sejam promissores, suas metodologias utilizam princípios distintos, como projeção em espaços bidimensionais, reorganização geométrica de atributos, preservação de vizinhança entre características ou aprendizado conjunto da representação visual. Assim, ainda há pouca informação consolidada a respeito de quais estratégias são mais adequadas para diferentes tipos de dados, especialmente em contextos da Biotecnologia e, particularmente, classificação de DEGs com dados de RNA-seq.

Essa lacuna é particularmente relevante em dados de expressão gênica, uma vez que dados de RNA-seq possuem alta dimensionalidade, grande variabilidade biológica e forte dependência de etapas de pré-processamento, normalização e definição estatística de expressão diferencial. A identificação de genes diferencialmente expressos é uma tarefa central em estudos de câncer, pois contribui para a caracterização molecular de tumores, identificação de potenciais biomarcadores e compreensão de vias biológicas associadas à doença. Nesse sentido, investigar novas estratégias computacionais para representar e classificar genes a partir de características derivadas de análises de RNA-seq pode contribuir tanto para a área de aprendizado de máquina aplicado a dados tabulares quanto para aplicações em bioinformática.

Os três algoritmos que serão analisados neste estudo estão entre os que

apresentaram melhores resultados em análises comparativas e testes de referência recentes: DeepInsight (Sharma *et al.*, 2019) representa uma estratégia baseada na projeção das características em um espaço bidimensional; NCTD (Alenizy; Berri, 2025) explora transformações geométricas para induzir relações espaciais entre atributos; e HACNet (Matsuda *et al.*, 2024) propõe uma abordagem aprendida e mais diretamente associada à interpretabilidade da imagem gerada. A comparação desses métodos em um mesmo pipeline permite avaliar se diferenças na construção das imagens resultam em diferenças relevantes no desempenho de classificação de genes.

No entanto, até o momento, esses métodos não foram amplamente aplicados especificamente à análise de genes diferencialmente expressos. Ao meu conhecimento, apenas um estudo explorou de forma próxima esse enfoque, utilizando uma conversão simples por redimensionamento de matriz antes da classificação por CNN. Dessa forma, a aplicação de conversores T2I mais elaborados ao contexto de classificação de genes regulados positivamente, regulados negativamente ou não diferencialmente expressos configura uma área ainda subexplorada. Essa limitação reforça a necessidade de uma investigação comparativa mais aprofundada, considerando não apenas o desempenho preditivo dos modelos, mas também o potencial de interpretação das representações geradas e sua adequação ao contexto biológico.

1.2 Objetivo geral

O objetivo geral consiste em analisar comparativamente alguns dos principais algoritmos desenvolvidos nos últimos anos para a conversão de dados tabulares em imagens, avaliando sua eficiência para a posterior classificação de genes diferencialmente expressos (DEGs) em conjuntos de dados de câncer.

1.2.1 Objetivos específicos

1. Aprofundamento sobre as principais técnicas de conversão de dados tabulares em imagem e principais arquiteturas de CNN utilizadas nesse contexto;
2. Seleção e engenharia de características para a construção de um formato tabular a ser usado como entrada para os conversores;

3. Desenvolvimento de um pipeline de processamento de CNN aplicado às imagens geradas por cada conversor que possibilite sua análise comparativa quanto à identificação de DEGs.

2 DESENVOLVIMENTO

2.1 Trabalhos relacionados

2.1.1 Conversores tabela-para-imagem para dados tabulares

Estudos como os de (Neto; Neto; Endo, 2023) e (Alenizy; Berri, 2025) compararam o desempenho de métodos T2I usando arquiteturas de CNN em relação a aplicações de ML tradicionais. O primeiro verificou pouca diferença entre uma versão aprimorada da alternativa que desempenhou melhor e um modelo XGBoost, enquanto o segundo apresentou o novo algoritmo para convolução de dados tabulares (NCTD), que superou diversos modelos tradicionais de ML em diferentes conjuntos de dados. Ambos os estudos revelam o potencial de CNNs com dados tabulares, ao mesmo tempo que destacam a importância do tipo de dado nos resultados obtidos, tendo isto sido determinante nos diferentes desempenhos em (Alenizy; Berri, 2025).

DeepInsight apresentou, em sua descrição original, acurácias entre 88% e 99% para 5 datasets diferentes, sendo o resultado de 99% na classificação de 10 tipos de câncer utilizando dados oriundos do TCGA (Sharma *et al.*, 2019), o que o configura como particularmente relevante neste contexto. Por outro lado, esse resultado não foi congruente com outras análises posteriores como em (Alenizy; Berri, 2025), que teve resultados bastante inferiores para esse método. HACNet, também em dados do TCGA, obteve uma acurácia de 99,88% em uma classificação de 5 tipos de câncer no seu artigo original (Matsuda *et al.*, 2024), mas esta é, até onde se sabe, a única vez que sua performance foi analisada. NCTD também foi aplicado em um contexto de classificação de tipos de câncer, porém com dados do banco de dados de câncer de mama de Wisconsin, ocasião em que sua acurácia foi de 99,28%.

Mesmo sendo uma estratégia relativamente bastante simples, os autores do NCTD reportaram resultados muito expressivos nas suas comparações de performance com outras técnicas, atingindo valores de acurácia maiores que todos os outros conversores testados em 8 de 10 datasets e, talvez mais surpreendentemente ainda, em 7 de 10 datasets em relação a outros modelos de

ML, incluindo métodos ensemble baseados em árvores como XGBoost, que muitos consideram estado-da-arte para datasets tabulares (Alenizy; Berri, 2025). Com isso, NCTD se torna uma alternativa muito interessante para qualquer aplicação e análise de DEGs não é uma exceção. Como desvantagem, os autores mencionam sua tendência de gerar imagens muito grandes, sendo isto crítico para datasets com muitas features uma vez que o tamanho da imagem final é fixado em 2N, em que N é o número de features.

O uso de um conversor evidentemente tem como objetivo criar imagens aptas a serem classificadas por uma CNN com as melhores métricas possíveis. Contudo, um outro aspecto que cada vez é mais importante é a capacidade de interpretar os principais fatores envolvidos na tomada de decisão do modelo – a transparência –, a própria explicabilidade da qualidade dos resultados e outros critérios que reduzam o problema de “caixa-preta” desses modelos. Questões como essas estão relacionadas à área de XAI, e têm como algumas técnicas importantes a construção de mapas de saliência (Simonyan; Vedaldi; Zisserman, 2014), Grad-CAM (Selvaraju *et al.*, 2020) ou importância de atributos por permutação (Fisher; Rudin; Dominici, 2019).

No contexto de conversores T2I, a maioria dos métodos desenvolvidos não leva em conta esses critérios na geração das imagens, tornando os poucos conversores que o fazem particularmente interessantes (Ma; Zhang, 2019; Matsuda *et al.*, 2024; Gómez-Martínez *et al.*, 2024; Lee *et al.*, 2025). HACNet surge nesse contexto com uma abordagem que buscou melhorar sobre algoritmos anteriores não só introduzindo um mecanismo de seleção de características relevantes, mas principalmente gerando imagens interpretáveis cuja construção é aprendida junto com o classificador, visando maximizar a performance do modelo. Essa estratégia atingiu tanto uma melhora significativa de interpretabilidade como bons resultados de acurácia nos testes de referência, evidenciando, assim, seu potencial (Matsuda *et al.*, 2024).

2.1.2 Aprendizado de máquina e aprendizado profundo em análise de DEGs

Análise de genes diferencialmente expressos (DEGs) é uma das técnicas mais usadas para a análise de dados de RNA-seq no contexto das ômicas e da

Bioinformática, sendo os pipelines associados de tremenda importância para a medicina de precisão e descoberta de medicamentos tendo como alvo linhagens de cânceres, por exemplo (Rosati *et al.*, 2024). Esse tipo de análise é tradicionalmente feito em bibliotecas na linguagem R como edgeR (Robinson; McCarthy; Smyth, 2010) e DESeq2 (Love; Huber; Anders, 2014), que utilizam testes estatísticos para computar as diferenças de expressão entre amostras comparáveis. Após a identificação dos DEGs, pode-se então realizar uma análise de enriquecimento funcional para melhor entender os mecanismos moleculares e vias a respeito da condição analisada, assim como validar os achados ao conhecimento consolidado (Rosati *et al.*, 2024).

Nesse contexto, estratégias de ML e, mais especificamente, aprendizado supervisionado, também demonstraram ser uma alternativa interessante e de grande potencial, como por exemplo em (Wenric; Shemirani, 2018), em que estratégias baseadas em ML performaram melhor que análise de DEGs com DESeq2 para a seleção de features em diversos datasets, ou no estudo de (Feng *et al.*, 2024), no qual um modelo de Aprendizado Profundo (DL) para classificação de DEGs baseado em atenção (Vaswani *et al.*, 2023) atingiu valores de métricas como acurácia, precisão e F1-score comparáveis ou melhores em relação a outras técnicas de ML. A revisão de (Alharbi; Vakanski, 2023) elabora sobre diversas técnicas de ML elaboradas para a classificação de cânceres e análise de expressão gênica com base em dados de RNA-seq nos últimos anos, incluindo algumas baseadas em CNNs.

No entanto, especificamente sobre DEGs, ainda há pouca informação consolidada a respeito das melhores estratégias e modelos de ML a utilizar, especialmente porque DEGnext (Kakati *et al.*, 2022) e TabDEG (Feng *et al.*, 2024) são algumas das únicas abordagens recentes que visam à classificação de genes UR e DR, sendo que a primeira é a única dessas abordagens, até onde se sabe, que utiliza conversão em imagem e classificação por CNN. Esta técnica também usa uma conversão T2I extremamente simples através de uma mera chamada à função reshape do Numpy (Harris *et al.*, 2020), para a qual não existem precursores comparáveis e que não é interpretável. Ou seja, a aplicação da ampla gama de conversores T2I disponíveis para o contexto de classificação de DEGs é altamente inexplorada, constituindo uma lacuna do conhecimento.

2.2 Fundamentação teórica

2.2.1 Dados de RNA-seq e análise de DEGs

A análise computacional de dados de expressão gênica tem se tornado uma etapa central em estudos de bioinformática, especialmente no contexto de doenças complexas, como o câncer. Dados de RNA-seq permitem quantificar a abundância de transcritos em diferentes condições biológicas, possibilitando a comparação entre grupos amostrais, como tecidos tumorais e tecidos normais. A partir dessas comparações, é possível identificar genes diferencialmente expressos, isto é, genes cuja expressão apresenta alteração estatisticamente significativa entre duas ou mais condições experimentais. Esses genes podem indicar processos biológicos alterados, vias moleculares associadas à doença e potenciais biomarcadores diagnósticos, prognósticos ou terapêuticos (Rosati *et al.*, 2024).

Em estudos de câncer, a identificação de genes diferencialmente expressos é particularmente relevante porque a tumorigênese envolve alterações profundas na regulação gênica, no ciclo celular, na resposta imune, no metabolismo e em mecanismos de proliferação e sobrevivência celular. Assim, a comparação entre amostras tumorais e não tumorais pode revelar genes regulados positivamente, quando apresentam maior expressão no grupo tumoral, ou regulados negativamente, quando apresentam menor expressão nesse grupo. A interpretação desses padrões, no entanto, depende de métodos estatísticos adequados, uma vez que dados de RNA-seq são caracterizados por alta dimensionalidade, dispersão variável entre genes e diferenças de profundidade de sequenciamento entre amostras.

Uma das abordagens mais utilizadas para análise de expressão diferencial em RNA-seq é o DESeq2 (Love; Huber; Anders, 2014), que foi desenvolvido justamente para lidar com esses desafios. Esse método parte de matrizes de contagens, nas quais as linhas representam genes e as colunas representam amostras, e modela os dados considerando características próprias de experimentos de sequenciamento. Entre seus principais componentes estão a normalização por fatores de tamanho, a estimação da dispersão e a realização de testes estatísticos

para avaliar se as diferenças observadas entre condições são compatíveis com expressão diferencial. Dessa forma, o DESeq2 permite obter medidas importantes como log_2FC, valor-p, representando a significância estatística da mudança, e valores-p ajustados por múltiplos testes estatísticos. Esses valores são frequentemente utilizados para classificar genes como diferencialmente expressos ou não diferencialmente expressos.

O primeiro grande problema abordado pelo DESeq2 é o fato de diferentes amostras terem diferentes profundidades de sequenciamento, porém isso não implica que os genes das amostras com mais reads estão verdadeiramente superexpressos. Para corrigir isso utiliza-se de fatores de tamanho amostra-específicos, os quais atuam como divisores para cada contagem da respectiva amostra. Em seguida, estima-se as dispersões de cada gene, sendo estas importantes para evitar classificações erradas ao tornar mais rigorosa a classificação de diferencialmente expresso para um gene alta variabilidade. A construção dessas dispersões primeiro leva em conta um cálculo independente para os genes, e em seguida aplica uma correção considerando a dispersão de outros genes de força de expressão semelhante, utilizando uma curva empírica de encolhimento de bayes, com base em um trabalho anterior (Wu; Wang; Wu, 2013). Essa estratégia é descrita como muito importante para evitar falsos positivos advindos de ruídos, especialmente em casos com um limitado número de réplicas biológicas. Ainda, o fato do DESeq2 modelar os genes seguindo uma distribuição binomial negativa está intrinsecamente ligado a isso, uma vez que isso justamente implica que a variância não é tomada como constante, ao contrário de aplicações de modelos paramétricos que seguem outras distribuições, como a gaussiana.

Essas medidas são então usadas para estimar os valores de log_2FC para cada gene em amostras de tumores em relação a amostras saudáveis, assim como os testes estatísticos com o teste de Wald. Contudo, como análises deste tipo normalmente levam em consideração milhares de genes, até mesmo valores-p baixos implicam muitos falsos positivos por acaso. Para aliviar esse problema é empregada a correção de múltiplos testes de Benjamini e Hochberg (Benjamini; Hochberg, 1995), que gera os padjs para cada gene.

Também é importante mencionar que existe um passo de filtragem automático chamado de filtragem independente, em que genes com contagens

abaixo da média de contagens normalizada para todos os genes são removidos da análise para ajudar na qualidade do ajuste de Benjamini-Hochberg sem prejuízo, uma vez que eles possuem pouca informação para uma identificação confiável de expressão diferencial. Outra filtragem também realizada por padrão envolve o uso da distância de Cook (Cook, 1977) para identificar outliers, técnica que pode remover genes da análise caso não haja réplicas suficientes.

Por outro lado, a confiabilidade dos resultados obtidos pelos diversos métodos de análise de DEGs também jaz sobre a qualidade das contagens obtidas. Nesse sentido, os esforços empreendidos pelo TCGA (Weinstein *et al.*, 2013) têm sido fundamentais no estabelecimento da importância da genômica de câncer e no desenvolvimento de estudos de bioinformática, incluindo análise de DEGs, por meio da coleção, caracterização e análise de amostras de câncer de mais de 20.000 pacientes e 33 diferentes tipos de tumores. Já para a disponibilização efetiva desses dados, o GDC é o principal atuante, uma vez que desenvolveu fluxos padronizados de harmonização para os dados advindos de diferentes técnicas. No caso de dados de RNA-seq, os reads submetidos pelo sequenciamento são alinhados ao genoma de referência utilizando o fluxo STAR 2-Pass Genome (Dobin *et al.*, 2013), gerando arquivos BAM de reads alinhados. Em seguida, a quantificação da expressão gênica é realizada pelo fluxo STAR - Counts, que usa esses alinhamentos para produzir dados de perfil transcritômico em nível gênico. Como saída, o GDC disponibiliza contagens brutas por gene, além de medidas transformadas de expressão, como FPKM, FPKM-UQ e TPM, anotadas com símbolo gênico e biotipo. Fluxos padronizados como esses permitem uma maior comparabilidade entre amostras e projetos.

2.2.2 Aprendizado de Máquina aplicado à expressão gênica

Apesar da importância dos métodos estatísticos tradicionais (Kalantari-Dehaghi; Ghohabi-Esfahani; Emadi-Baygi, 2025), abordagens baseadas em ML também têm sido exploradas para a análise de dados transcritômicos. Nesse contexto, estudos de DL que focaram na análise de DEGs usaram tabelas representando valores de expressão em cada amostra por gene (Kakati *et al.*, 2022; Feng *et al.*, 2024). Contudo, de forma a compactar as tabelas e possivelmente

acelerar os treinamentos, os genes e as amostras podem ser representados por características extraídas, sumarizadas ou ambos, advindas dos dados de expressão. Em problemas envolvendo genes diferencialmente expressos, por exemplo, características como média de expressão em amostras tumorais, média de expressão em amostras normais, variância, $\log_2$FC e significância estatística, estes últimos dois derivados de análises estatísticas anteriores, podem ser usadas para treinar modelos capazes de distinguir genes regulados positivamente, regulados negativamente ou sem diferença significativa.

Apesar do avanço recente de modelos de aprendizado profundo para dados tabulares, métodos baseados em árvores de decisão ainda ocupam uma posição de destaque nesse tipo de problema. Entre eles, Random Forest (RF) combina múltiplas árvores treinadas sobre subconjuntos aleatórios dos dados e das características, reduzindo a variância e aumentando a robustez do modelo em relação a uma única árvore de decisão (Breiman, 2001). Já o XGBoost se baseia em árvores de decisão treinadas de forma sequencial por reforço de gradiente, de modo que cada nova árvore busca corrigir erros cometidos pelas anteriores, incorporando ainda regularização e otimizações computacionais que tornam o método eficiente e escalável (Chen; Guestrin, 2016).

Essa vantagem de métodos ensemble baseados em árvores é particularmente relevante em dados tabulares porque esses modelos não dependem de uma estrutura espacial ou semântica pré-definida entre as características. Ao contrário de CNNs, que exploram padrões locais em dimensões regulares da imagem, modelos como RF e XGBoost lidam naturalmente com relações não lineares, interações entre variáveis, distribuições heterogêneas e características pouco informativas. Estudos comparativos recentes indicam que, mesmo diante do desenvolvimento de arquiteturas neurais especializadas para tabelas, métodos baseados em árvores frequentemente permanecem competitivos ou superiores em conjuntos tabulares de tamanho pequeno a médio, além de exigirem menor custo computacional e menos etapas de ajuste fino (Grinsztajn; Oyallon; Varoquaux, 2022; Borisov *et al.*, 2024).

Em aplicações supervisionadas de ML, a avaliação do desempenho depende da separação entre dados utilizados no treinamento e dados utilizados apenas para teste, de modo a estimar a capacidade de generalização do modelo para observações não vistas. Uma alternativa comum à divisão única entre treino e teste é a validação cruzada em k partições, na qual o modelo é treinado e avaliado repetidamente em diferentes subconjuntos dos dados, produzindo uma estimativa menos dependente de uma única partição aleatória (Kohavi, 1995). Em problemas de classificação com classes desbalanceadas, a validação cruzada estratificada é especialmente útil, pois busca preservar em cada fold proporções de classes semelhantes às do conjunto original, reduzindo o risco de avaliações enviesadas por partições pouco representativas. Essa estratégia é relevante em tarefas multiclasse, como a classificação de genes em UR, DR e ND, nas quais diferenças na distribuição das classes podem afetar diretamente as métricas de desempenho.

Além da estratégia de divisão dos dados, a escolha das métricas de avaliação é uma etapa essencial para interpretar o desempenho de modelos de aprendizado de máquina. A acurácia representa a proporção total de classificações corretas em relação ao número total de observações avaliadas, sendo uma métrica simples e intuitiva. No entanto, ela pode ser insuficiente em conjuntos de dados desbalanceados, pois um modelo pode obter alta acurácia ao favorecer a classe majoritária, mesmo apresentando baixo desempenho nas classes menos frequentes. Por esse motivo, em problemas multiclasse e potencialmente desbalanceados, é comum complementar a acurácia com métricas calculadas separadamente por classe.

Entre essas métricas, precisão, revocação e F1-score são particularmente importantes. A precisão indica, entre as observações classificadas como pertencentes a determinada classe, quantas realmente pertencem a ela. A revocação, ou sensibilidade, indica, entre todas as observações reais daquela classe, quantas foram corretamente identificadas pelo modelo. O F1-score corresponde à média harmônica entre precisão e revocação, sendo útil quando se deseja equilibrar falsos positivos e falsos negativos.

Dados tabulares, entretanto, apresentam desafios específicos para modelos de CNN. Devido à ausência de organização espacial natural, a utilização de CNNs se torna imprática, impedindo o uso de arquiteturas bem estabelecidas para com

suas vantagens naturais, como a exploração de padrões locais por meio de filtros compartilhados (LeCun; Bengio; Hinton, 2015) e as possíveis melhoras de eficiência de classificação resultantes disto. Como consequência, uma etapa importante para o uso de CNNs em dados tabulares é a criação de uma representação visual artificial, na qual as características originais sejam organizadas em uma matriz bidimensional que possa ser interpretada como imagem; os algoritmos que realizam este processo podem ser chamados de conversores T2I (Sharma *et al.*, 2019; Bazgir *et al.*, 2020; Zhu *et al.*, 2021b; Alenizy; Berri, 2025).

Assim, o bom desempenho de modelos como XGBoost estabelece uma referência comparativa relevante para avaliar se a representação visual gerada por conversores T2I realmente permite explorar vantagens das CNNs. Dessa forma, a análise de métodos de conversão tabular-para-imagem deve considerar não apenas a capacidade das CNNs de extrair padrões espaciais, mas também o fato de que métodos ensemble baseados em árvores já possuem forte adequação indutiva ao formato tabular original.

2.2.3 Redes neurais convolucionais e dados tabulares

CNNs são arquiteturas de DL originalmente desenvolvidas para lidar com dados organizados em grades, como imagens. Diferentemente de redes densamente conectadas, nas quais cada neurônio de uma camada pode estar conectado a todos os neurônios da camada anterior, as CNNs utilizam filtros convolucionais que percorrem pequenas regiões da entrada. Esses filtros compartilham os mesmos pesos ao longo da imagem, permitindo que o modelo detecte padrões locais independentemente da posição em que ocorrem. Essa propriedade é particularmente útil em imagens, nas quais estruturas visuais próximas tendem a apresentar relação espacial e semântica (LeCun; Bengio; Hinton, 2015).

O funcionamento das CNNs depende, portanto, de duas premissas principais: a existência de padrões locais relevantes e a possibilidade de reutilizar filtros em diferentes regiões da entrada. Em uma imagem convencional, pixels vizinhos geralmente estão relacionados, formando bordas, texturas, formas e estruturas

progressivamente mais complexas. Dessa forma, camadas convolucionais iniciais podem aprender padrões simples, enquanto camadas mais profundas combinam essas informações para representar aspectos mais abstratos da imagem. Além disso, o compartilhamento de pesos torna as CNNs mais eficientes em número de parâmetros em comparação com redes totalmente conectadas aplicadas diretamente a entradas de alta dimensionalidade.

No entanto, essas vantagens não se transferem automaticamente para dados tabulares. Em uma tabela, cada coluna representa uma variável distinta, e a ordem das colunas nem sempre possui significado espacial ou biológico. Duas características posicionadas lado a lado em uma matriz tabular podem não apresentar relação semântica direta, enquanto características distantes podem estar fortemente associadas. Assim, aplicar uma CNN diretamente a uma tabela organizada de forma arbitrária pode limitar a utilidade dos filtros convolucionais, pois a vizinhança explorada pelo modelo não necessariamente corresponde a relações reais entre as variáveis.

Nesse contexto, os conversores de dados tabulares em imagens buscam criar uma representação bidimensional intermediária que torne o uso de CNNs mais adequado. A ideia central é reorganizar as características tabulares em uma matriz de pixels de modo que relações relevantes entre variáveis sejam, sempre que possível, refletidas por proximidade espacial ou por padrões estruturados na imagem gerada. Com isso, espera-se que os filtros convolucionais possam explorar dependências locais de maneira mais informativa do que ocorreria em uma disposição arbitrária das colunas.

Para dados de expressão gênica, essa questão é especialmente relevante. Embora os genes e as características derivadas de análises estatísticas possuam significado biológico próprio, eles não estão naturalmente organizados como pixels de uma imagem. Assim, ao representar genes ou amostras por meio de matrizes bidimensionais, torna-se necessário avaliar se o conversor utilizado preserva informações relevantes para a classificação. Nesse sentido, métodos como DeepInsight, NCTD e HACNet diferem não apenas na forma como produzem imagens, mas também no modo como tentam tornar essas imagens úteis para o aprendizado convolucional.

Outro ponto a ser entendido no contexto de CNNs é o fato de a tecnologia ser

relativamente antiga quando comparada a arquiteturas mais novas como transformers, mas que a partir de 2012, com o desempenho da arquitetura conhecida como AlexNet (LeCun; Bengio; Hinton, 2015), teve sua capacidade demonstrada através do treinamento de redes profundas treinadas com grandes volumes de dados e uso de GPUs, superando abordagens tradicionais de visão computacional em tarefas de classificação.

Especialmente desde essa época, diversas inovações foram feitas em subsequentes arquiteturas de CNNs e DL como um todo, incluindo a adoção de dropouts, convoluções 1 x 1, blocos repetidos, modularização, normalização por lotes, entre outras. Nesse contexto, em 2015, (He *et al.*, 2015) apresentaram uma arquitetura que incorporava melhorias dos anos anteriores, mas estava fortemente fundamentada em conexões residuais, isto é, ligações na passagem de gradientes entre camadas distantes que facilitam em muito este fluxo, particularmente em redes muito profundas. Este tipo de arquitetura ficou conhecido como a família ResNet, e é uma das redes mais revolucionárias ao apresentar ganhos de performance em redes cada vez mais profundas sem overfitting em grande medida devido às conexões residuais.

Existem vários tipos de ResNet, como ResNet18, ResNet34, ResNet50, ResNet101 e ResNet152. A ResNet18 é uma das versões mais compactas dessa família, sendo composta por uma camada convolucional inicial 7x7, seguida por blocos residuais organizados em diferentes estágios, camadas de agregação espacial e uma camada final totalmente conectada para classificação. Diferentemente de versões mais profundas, como ResNet50 ou ResNet152, a ResNet18 utiliza blocos residuais básicos, geralmente compostos por duas camadas convolucionais 3 x 3. Essa característica reduz sua complexidade computacional em comparação com versões mais profundas, mantendo, ao mesmo tempo, a principal vantagem conceitual das ResNets: o aprendizado residual.

2.2.4 Conversão de dados tabulares em imagens e suas metodologias

Conversores T2I buscam transformar cada observação tabular em uma

imagem, preservando, sempre que possível, relações relevantes entre características. A premissa central é que, se características relacionadas forem posicionadas próximas entre si na imagem gerada, uma CNN poderá explorar dependências locais e aprender padrões discriminativos de forma mais eficiente. A forma como essa organização espacial é construída varia entre os métodos; (Alenizy; Berri, 2025) apresenta algumas divisões possíveis entre a grande quantidade de conversores construídos recentemente. Destas, as metodologias mais importantes no contexto deste trabalho são as fundamentadas em redução de dimensionalidade (Sharma *et al.*, 2019; Bazgir *et al.*, 2020; Castillo-Cara *et al.*, 2023), classificação ponta-a-ponta (Matsuda *et al.*, 2024; Gómez-Martínez *et al.*, 2024) e transformações matemáticas (Alenizy; Berri, 2025).

DeepInsight (Sharma *et al.*, 2019) é um exemplo de conversor baseado na projeção das características em um espaço bidimensional. Nesse tipo de abordagem, cada variável original é posicionada em uma coordenada da imagem a partir de métodos de redução de dimensionalidade, como t-SNE (Maaten; Hinton, 2008) ou PCA (Hotelling, 1933). Após essa etapa, os valores das características de cada observação são mapeados para os pixels correspondentes, gerando uma imagem que representa a amostra a ser classificada.

A grande vantagem dessa estratégia é tentar preservar relações de vizinhança entre características através da clusterização realizada pelo algoritmo base, embora sua estabilidade possa depender do método de projeção utilizado, dados empregados no ajuste e da parametrização correta, especialmente em termos da perplexidade para t-SNE, por exemplo. Um outro ponto que pode gerar dificuldades ao usar o DeepInsight inadvertidamente é o fato de ele poder mapear mais de uma feature ao mesmo pixel, especialmente caso o tamanho de imagem utilizada não seja grande o suficiente em relação ao número de features, o que pode afetar significativamente sua capacidade de predição.

O NCTD (Alenizy; Berri, 2025), por sua vez, utiliza uma lógica diferente, baseada na transformação direta dos valores tabulares em intensidades de pixels e na aplicação de operações espaciais para gerar uma imagem estruturada. Após a normalização, cada linha da tabela tem seus valores transformados em valores de intensidade de pixel, efetivamente transformando-a em uma imagem em tons de cinza. Cada linha é então duplicada N vezes, onde N é o número de características,

formando uma matriz N x N que, por fim, sofre rotações progressivas à esquerda, introduzindo variação espacial e dependência entre pixels vizinhos. Para potencializar esse efeito, a imagem final possui dimensão 2N x 2N, sendo pela replicação da imagem N x N em cada quadrante.

HACNet (Matsuda *et al.*, 2024) representa outra categoria de método, na qual a conversão T2I e a classificação são integradas em um processo de aprendizado conjunto. Em vez de gerar imagens por uma regra fixa previamente definida, o método busca aprender uma representação visual interpretável com base na semelhança a imagens-modelo fornecidas e ao mesmo tempo útil para a tarefa de classificação, incorporando mecanismos de atenção e seleção de características. Essa característica torna o HACNet especialmente relevante em discussões sobre interpretabilidade, pois a construção da imagem fornece um resultado final cuja semelhança ao respectivo modelo está diretamente ligada à eficiência de classificação. Além disso, pode fornecer indícios sobre quais variáveis foram consideradas mais importantes pelo modelo através da análise de quais features foram mapeadas para cada pixel, fator que está relacionado à importância de cada feature.

Especificamente, HACNet incorpora pesos de atenção α para cada pixel considerando um vetor de features x conforme a seguinte fórmula:

$$P_{h,w}(x) = \langle \alpha^{h,w}, x \rangle = \sum_{k=1}^{d} \alpha_k^{h,w} x_k \quad (1)$$

em que $x \in R^d$ representa o vetor de características de entrada, d é o número de características, $P_{h,w}(x)$ é o valor do pixel localizado na posição (h,w), e $\alpha_{h,w,k}$ representa o peso de atenção atribuído à k-ésima característica para esse pixel.

Ou seja, o valor de cada pixel é o produto escalar do vetor de features e o vetor de pesos de atenção para determinada altura h e largura w. Contudo, α não é diretamente treinado nesse processo, mas sim parâmetros de distribuição π, seguindo um processo conhecido como Gumbel-Softmax, de acordo com a seguinte equação:

$$\alpha_k^{h,w} = \frac{\exp\left(\frac{\log\left(\frac{\pi_k^{h,w}}{1-\pi_k^{h,w}}\right) + g_k^{h,w}}{\tau}\right)}{\sum_{j=1}^{d} \exp\left(\frac{\log\left(\frac{\pi_j^{h,w}}{1-\pi_j^{h,w}}\right) + g_j^{h,w}}{\tau}\right)} \quad (2)$$

Em que τ é a temperatura controlada ao longo do treinamento, e os valores g são provenientes de uma distribuição uniforme. Como resultado de um processo de diminuição da temperatura τ realizado ao longo do treinamento, α gradualmente torna-se mais semelhante a um vetor one-hot para cada posição.

O processo de treinamento do HACNet visa diminuir uma função de perda caracterizada da seguinte forma:

$$l(x, y, y') = -\sum_{j=1}^{c} y'_j \log y_j + \frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} \left(P_{h,w}(x) - T_{h,w}^{class(y)}\right)^2 \quad (3)$$

Em que y são os rótulos corretos originais, ȳ os rótulos preditos, c é o número de classes, λ é o coeficiente de regularização para a imagem e T é a imagem modelo para a respectiva classe. Essa fórmula se dá essencialmente pela junção da perda de entropia cruzada da classificação por CNN à perda por erro quadrático médio em relação à imagem modelo, tornando possível o aprendizado da classificação e da correção das imagens ao mesmo tempo. Isso ocorre em fases separadas de treinamento e “retreinamento”, nas quais o treinamento conjunto ocorre apenas na primeira fase, enquanto que a segunda fase utiliza apenas o componente de entropia cruzada. Na transição dessas fases, força-se os vetores α a virarem one-hot, sendo o valor de 1 atribuído à feature que foi treinada a receber mais atenção até então, passo que garante a identificação de cada pixel com uma feature apenas, mas uma feature pode estar associada a mais de um pixel.

2.3 Métodos

2.3.1 Delineamento geral do pipeline

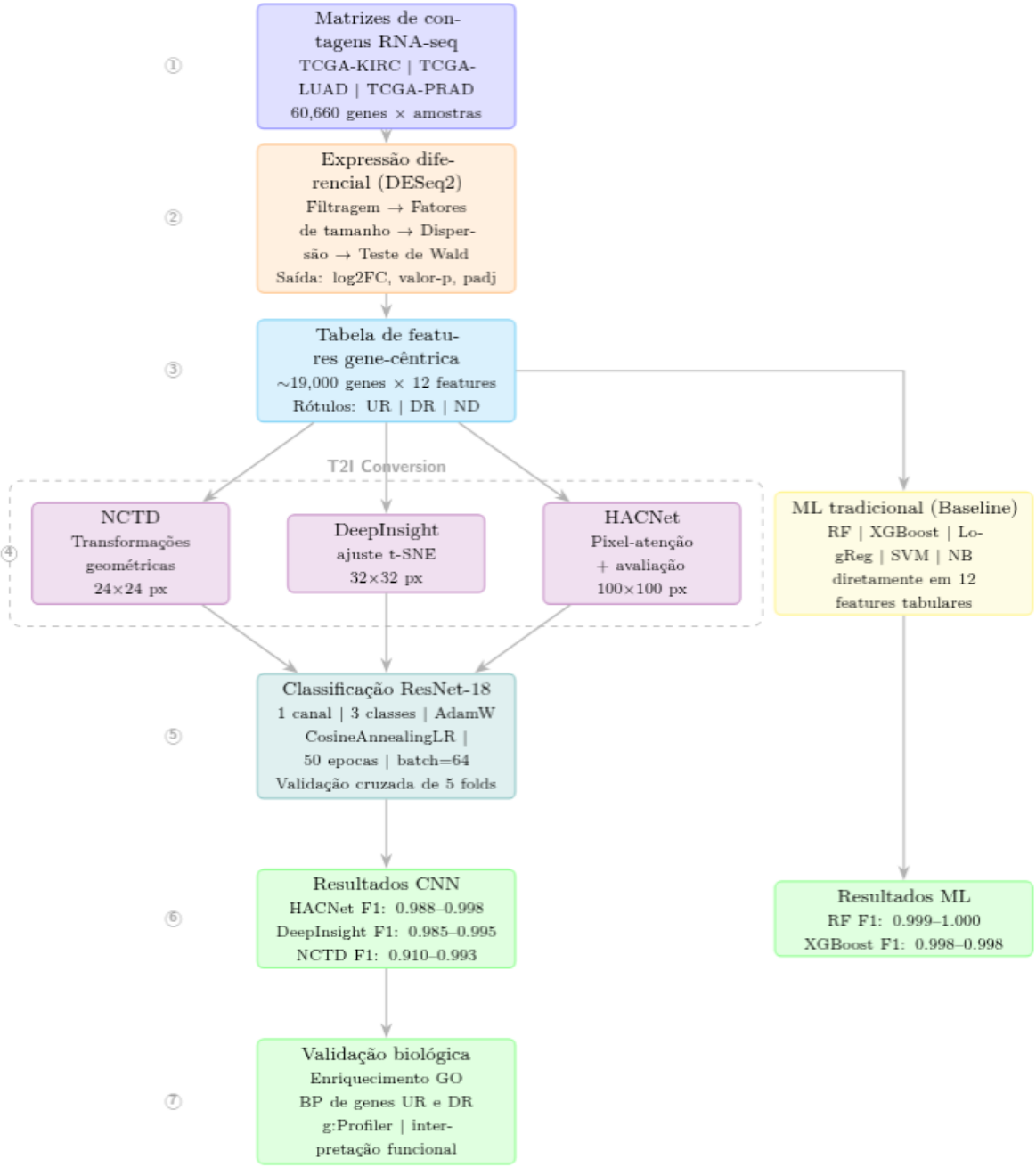
O delineamento geral do projeto é apresentado na Figura 1. Resumindo o projeto de uma forma geral, primeiramente foram obtidas as matrizes de contagens de RNA-seq dos datasets em questão, sendo estes submetidos a processamentos

relevantes para posterior análise de DEGs por DESeq2. Então, foram construídas tabelas gene-cêntricas constituídas por medidas importantes como features para a classificação de DEGs obtidas dos resultados do DESeq2 ou de sumarizações estatísticas destes e das contagens gênicas. A partir dos 60660 genes nas matrizes iniciais de cada dataset, 19304, 18603 e 17533 genes finais foram obtidos para LUAD, KIRC e PRAD, respectivamente, nas tabelas finais.

Então, essas tabelas foram, observando as particularidades de cada conversor, transformadas em imagens, gerando 9 combinações de conjuntos de imagens e conversor ao total. Em seguida, esses conjuntos foram submetidos como entrada para uma rede avaliadora ResNet18 adaptada, buscando obter os rótulos originais corretos de UR, DR ou ND. Paralelamente, as tabelas originais foram usadas para classificação em outros modelos de ML, como RF e XGBoost. Concluídas as classificações, os resultados desses dois pipelines de classificação foram analisados comparativamente entre si tendo em vista métricas de performance e interpretabilidade.

Por último, foi feita uma etapa de validação biológica das classificações de genes por meio de enriquecimento funcional GO, que teve como foco explicitar as vias e mecanismos celulares com os quais os genes expressos diferencialmente detectados estão envolvidos, visando expor as suas significâncias quando comparadas a processos celulares conhecidos das linhagens de câncer em questão.

Figura 1 - Esquematização geral do projeto



Fonte: elaboração própria

2.3.2 Obtenção e pré-processamento dos dados

Este trabalho foi desenvolvido como um estudo computacional comparativo voltado à avaliação de diferentes estratégias de conversão de dados tabulares em imagens para posterior classificação de genes diferencialmente expressos por redes neurais convolucionais. O pipeline geral foi dividido em quatro etapas principais: obtenção e curadoria de dados públicos de RNA-seq de câncer, construção de

tabelas centradas em genes a partir de análises de expressão diferencial, conversão dessas tabelas em imagens por diferentes algoritmos T2I e classificação dos genes por modelos baseados em CNN. Além disso, modelos tradicionais de aprendizado de máquina foram treinados diretamente sobre as tabelas originais, servindo como referência para a avaliação do benefício da etapa de conversão em imagem.

Foram utilizados dados públicos de RNA-seq provenientes de projetos do TCGA, pré-processados e harmonizados por meio do GDC como descrito anteriormente. Os projetos selecionados foram TCGA-LUAD, correspondente ao adenocarcinoma pulmonar, TCGA-PRAD, correspondente ao adenocarcinoma de próstata, e TCGA-KIRC, correspondente ao carcinoma renal de células claras. A escolha desses conjuntos de dados buscou contemplar diferentes tipos tumorais com disponibilidade de suficientes amostras tumorais e amostras normais, permitindo a realização de análises de expressão diferencial entre dois grupos biológicos principais.

A obtenção dos dados foi realizada no ambiente R por meio do pacote TCGAbiolinks. Para cada projeto, foi construída uma consulta ao GDC utilizando a categoria “Transcriptome Profiling”, o tipo de dado “Gene Expression Quantification” e o fluxo de trabalho “STAR - Counts”. Após o download dos arquivos, os dados foram preparados com a função `GDCprepare`, e a matriz de contagens gênicas foi extraída a partir do ensaio “unstranded”. Dessa forma, as linhas da matriz inicial corresponderam aos genes e as colunas às amostras de RNA-seq disponíveis para cada projeto.

Após a obtenção das matrizes de contagem, foram selecionadas apenas as amostras correspondentes aos grupos tumorais e normais. A classificação do tipo de amostra foi realizada a partir dos códigos presentes nos identificadores TCGA, considerando o código “01” como amostra tumoral primária e o código “11” como amostra normal sólida. Amostras não pertencentes a esses dois grupos foram removidas da análise. Em seguida, foi criado um fator experimental indicando a condição biológica de cada amostra, com os níveis “Tumor” e “Normal”, utilizado posteriormente na análise de expressão diferencial.

Antes da análise com DESeq2, foi aplicada uma filtragem inicial de genes de baixa expressão. Para isso, foram calculados valores de contagens por milhão, ou CPM, com o pacote edgeR. Foram mantidos apenas os genes que apresentaram

CPM maior que 1 em pelo menos 10% das amostras do respectivo conjunto de dados. Essa etapa teve como objetivo remover genes com expressão muito baixa ou ausente na maior parte das amostras, reduzindo ruído, custo computacional e a quantidade de testes estatísticos realizados em genes com pouca informação para inferência de expressão diferencial.

2.3.3 Análise de expressão diferencial com DESeq2 e construção das tabelas

A análise de expressão diferencial foi realizada com o pacote DESeq2. Para cada conjunto de dados, foi construído um objeto `DESeqDataSet` a partir da matriz de contagens filtrada e de uma tabela de metadados contendo a condição de cada amostra. O modelo estatístico utilizado teve como desenho experimental a fórmula `~ condition`, comparando amostras tumorais e normais. Em seguida, a função `DESeq` foi aplicada para estimar fatores de tamanho, dispersões gênicas e ajustar o modelo de expressão diferencial. Os resultados foram extraídos com a função `results`, obtendo, para cada gene, valores como $\log_2$FC, valor-p e valor-p ajustado – que é resultado dos testes múltiplos de Benjamin-Hochberg, como citado anteriormente.

Também levou-se em conta a possibilidade dos cálculos realizados pelo DESeq2 marcarem genes pela filtragem independente ou pela distância de Cook. No primeiro tipo de caso, os valores de padj (valor-p ajustado) nas linhas dos respectivos genes são marcados como NA, ou seja, nulos, enquanto que no segundo, tanto padj quanto pvalue (valor-p original) são marcados como NA. Portanto, antes da obtenção de mais características dos genes, em cada dataset os genes que apresentassem esses valores nulos foram removidos, realizando-se cálculos subsequentes apenas nas amostras restantes.

Os rótulos de classificação foram definidos a partir dos resultados da análise de expressão diferencial. Genes com valor-p ajustado inferior a 0,05 e $\log_2$FC positivo acima de 1, ou seja, que tiveram expressão ao menos 2 vezes maior em amostras tumorais, foram classificados como genes regulados positivamente, ou UR. Genes com valor-p ajustado inferior ao mesmo limiar e $\log_2$FC negativo abaixo de -1 foram classificados como genes regulados negativamente, ou DR. Os demais genes foram classificados como não diferencialmente expressos, ou ND.

A partir dos resultados do DESeq2 e das matrizes de contagens normalizadas, foram construídas tabelas centradas em genes. Diferentemente de abordagens nas quais cada observação representa uma amostra ou paciente, neste trabalho cada observação correspondeu a um gene. Para cada gene, foram calculadas features numéricas que buscavam sintetizar informações de expressão ou demonstrar características estatísticas diferenciais entre as classes alvo que possam ser importantes para a classificação. As features incluídas foram: média de expressão nas amostras tumorais, média de expressão nas amostras normais, log2FC, dispersão estimada pelo DESeq2, variância da expressão normalizada transformada por log2, porcentagem de amostras com expressão detectável, valor-p, valor-p ajustado, $\log_2$FC absoluto, média de expressão considerando os dois tipos de amostra, razão de expressão (entre os dois tipos de amostra) e o coeficiente de variação. A coluna de rótulo foi criada seguindo os critérios explicados no parágrafo anterior e mantida separadamente das variáveis preditoras. Portanto, as tabelas finais têm 12 features.

As médias de expressão em amostras tumorais e normais foram calculadas a partir das contagens normalizadas pelo DESeq2, considerando separadamente os grupos de amostras tumorais e não tumorais. A variância gênica foi estimada sobre os valores de expressão transformados em logaritmo base 2 após a adição de uma pseudocontagem, de modo a reduzir a influência de valores extremos e atenuar a assimetria das distribuições. A porcentagem de amostras expressas foi obtida pela proporção de amostras em que cada gene apresentou expressão acima do limiar mínimo de 1 contagem por milhão.

Além dessas métricas, foram extraídas algumas outras características diretamente relacionadas à análise de expressão diferencial. A média base de expressão foi obtida a partir dos resultados do DESeq2, enquanto o log2FC absoluto foi calculado como o valor absoluto da coluna de log2FC. A razão de expressão foi definida como a média de expressão nas amostras tumorais dividida pela média de expressão nas amostras normais. Por fim, o coeficiente de variação foi calculado pela divisão do desvio-padrão gênico, definido como a raiz quadrada da variância gênica, pela média das contagens normalizadas.

Sabe-se que o significado dessas características representa certa redundância em relação às características anteriores, uma vez que, por exemplo, o

valor absoluto de log_2FC está diretamente ligado ao valor que considera o lado da variação, assim como a razão de expressão é diretamente formada pelas expressões médias em amostras tumorais e normais, e o coeficiente de variabilidade é outra medida de dispersão – embora não considere a modelagem binomial negativa que o DESeq2 aplica às contagens gênicas. O intuito de adicionar essas features é justamente testar se elas podem ser úteis ao processo de aprendizagem mesmo considerando sua potencial redundância, ou se as features anteriores realmente são as mais importantes assim como se poderia imaginar.

É importante mencionar neste contexto que o uso de features diretamente envolvidas na formação dos rótulos originais constitui um vazamento de dados do alvo, sendo, portanto, um fator limitante quanto à relevância do aprendizado futuro dos modelos de ML a serem utilizados. Ao disponibilizar informações que o próprio DESeq2 disponibiliza nos seus cálculos estatísticos, espera-se que os modelos a serem utilizados aprendam um processo semelhante ao DESeq2 e não relações biológicas e transcriptômicas cruciais para a classificação de DEGs. Escolheu-se essa alternativa muito em função do foco do trabalho em aferir a viabilidade de abordagens T2I e sua comparabilidade com relação a métodos estabelecidos de ML no que tange datasets não necessariamente muito grandes ou complexos.

Ao final dessa etapa, cada projeto do TCGA resultou em uma tabela na qual as linhas representam genes, as colunas representam características numéricas e cada gene possui uma classe de expressão diferencial associada. Como etapa de controle de qualidade das tabelas, foram removidos ou tratados valores ausentes provenientes de filtragens estatísticas, baixa informação ou remoção automática de genes durante a análise diferencial. Também foram removidas colunas não numéricas que não deveriam ser utilizadas como entrada dos modelos, como identificadores de genes ou símbolos gênicos, preservando-os apenas para rastreamento posterior e interpretação biológica. As tabelas finais foram exportadas em formato CSV e utilizadas como entrada para as etapas de conversão em imagem e classificação.

2.3.4 Conversão de dados tabulares em imagens

Para avaliar a utilidade da conversão de dados tabulares em imagens, foram

testados três conversores T2I: DeepInsight, NCTD e HACNet. Esses métodos foram escolhidos por apresentarem resultados muito bons em trabalhos prévios, especialmente considerando datasets de RNA-seq e aplicações envolvendo câncer. Alguns desses principais trabalhos foram informados na seção de trabalhos relacionados. Muitas dessas aplicações usaram dados semelhantes aos deste trabalho, mas sem ter a abordagem gene-cêntrica visando classificar DEGs abordada aqui, além do fato de os conversores representarem princípios metodológicos bastante diferentes e com focos, vantagens e desvantagens diferentes.

DeepInsight baseia-se na projeção das características em um espaço bidimensional, constituindo uma metodologia relativamente simples e rápida para aplicar em diversas amostras; NCTD utiliza transformações geométricas determinísticas para reorganizar valores tabulares em matrizes de pixels, sendo ainda mais simples e rápido para datasets com poucas features, mas não sendo escalável para dimensionalidades muito maiores; e HACNet propõe uma conversão aprendida em conjunto com o classificador, incorporando mecanismos de atenção e maior potencial de interpretabilidade tanto na análise imediata da imagem quanto no mapeamento para features relevantes, tendo como desvantagem, no entanto, um processo de geração de imagens muito mais complexo e inflexível.

Um passo importante antes de aplicar as metodologias necessárias para cada conversor é realizar as divisões dos conjuntos de dados, uma vez que para a análise comparativa ter alguma validade é necessário que tenhamos certeza que os resultados apresentados não sejam devido a uma favorável alocação dos dados de treinamento e teste para o conversor em questão. Dessa forma, foi criado um script em Python para realizar a divisão de 5 folds de dados de treinamento de teste, com auxílio da função StratifiedKFold da biblioteca Scikit-learn (Pedregosa *et al.*, 2011). Esse script foi executado para cada dataset, gerando os 5 folds divididos em 80% amostras de treinamento e 20% amostras de teste para cada um deles, e garantindo reprodutibilidade ao executar no estado aleatório de número 42. Esse número de seed foi usado semelhantemente em todas as funções que contêm parametrização de estados aleatórios usadas posteriormente.

A menos que seja especificada uma versão em específico, todos os conversores foram utilizados na versão 3.14.3 do Python e com os pacotes das

bibliotecas mais recentes possíveis para a versão em questão. Uma comparação resumida dos tipos de imagens geradas está disponível na imagem 1 abaixo. Um ponto interessante mostrado aqui é que o DeepInsight por padrão gera imagens com 3 canais, mesmo que neste caso todos os canais tenham sido idênticos. Esses casos foram convertidos em 1 canal em tom de cinza, antes das classificações por CNN.

Figura 2 - resumo da geração das imagens quanto ao seu mecanismo, propriedades e características

Abordagem	Tipo de Conversão	Resolução	Profundidade de Cor	Exemplo
NCTD	Transformação Direta e Expansão	24 x 24	8 bits/pixel	
DeepInsight	Redução de Dimensionalidade (t-SNE/PCA)	32 x 32	24 bits/pixel	
HACNet	Template / IA Explicável	100 x 100	8 bits/pixel	

Fonte: elaboração própria.

2.3.4.1 DeepInsight

Para aplicar a conversão de imagens do DeepInsight, utilizou-se a biblioteca pyDeepInsight, uma utilidade criada pelos autores do DeepInsight que, embora não garanta os mesmos resultados do código MATLAB original, foi atualizado nos anos recentes e possui bom suporte para uso como API. O pyDeepInsight cria o modelo de conversão T2I através da função ImageTransformer, que aceita parametrização do algoritmo de redução de dimensionalidade, estratégia de discretização e tamanho final da imagem. Para o primeiro, escolheu-se o t-SNE com 2 componentes e perplexidade 3, tendo em vista que o dataset alvo tem um baixo número de features, manteve-se a discretização padrão de bins e escolheu-se um tamanho de imagem de 32 x 32, sendo o valor resultante de 1024 pixels julgado suficiente para evitar

possíveis colisões de features alocadas ao mesmo pixel em número significativo.

O processo de geração das imagens pelo DeepInsight usou o mesmo script mencionado anteriormente, expandindo-o com os passos necessários. Iterando-se pelos folds aplicando a função fit_transform do ImageTransformer, foi possível adaptar o algoritmo do DeepInsight separadamente paras os dados de treinamento de cada fold e, ao mesmo tempo, gerar as imagens. O script também compreendeu uma lógica de criação de metadados em formato .csv e a criação de diretórios de forma a armazená-los, assim como as imagens geradas. Dessa forma, para cada imagem gerada, esta era armazenada no caminho seguindo a lógica {nome_do_conversor}_{nome_do_dataset}_IMGS/{número_do_fold}/{nome_da_classe}, sendo os respectivos metadados salvos na pasta base da combinação conversor – dataset. Esta organização teve o intuito de facilitar a localização das imagens considerando as etapas de treinamento posteriores, e foi seguida para as conversões com NCTD e HACNet.

2.3.4.2 NCTD

O uso do conversor NCTD apresentou uma dificuldade em particular, uma vez que os autores não disponibilizaram o código fonte no artigo original e, apesar deles mencionarem que sob requisição seria disponibilizado, não houve resposta aos e-mails enviados. Todavia, devido à sua relativa simplicidade e detalhamento necessário disponível, foi possível implementá-lo seguindo fielmente a metodologia original (Alenizy; Berri, 2025). Nesse processo também foram implementadas as funções fit, transform e fit_transform, que permitem o ajuste aos dados e transformação da mesma forma como ocorre com o pyDeepInsight e funções auxiliares clássicas de ML do Scikit-learn. Isso permitiu que a geração das imagens com NCTD para cada dataset fosse otimizada em um script juntamente com o DeepInsight, facilitando o uso da mesma forma de organização das imagens e dos metadados. Visto que o tamanho das imagens geradas pelo NCTD é fixo na forma 2Nx2N, em que N é o número de features, o tamanho das imagens geradas foi de 24x24.

2.3.4.3 HACNet

Para o HACNet, primeiramente foi necessário cloná-lo de seu repositório original e realizar a atualização de seus pacotes e versão-alvo de Python a uma versão recente. Com o auxílio do gerenciador de pacotes uv, criou-se um ambiente virtual para a execução de Python 3.13.13, ao qual foram adicionados os pacotes necessários para a execução do projeto. Verificando-se a funcionalidade correta do sistema, partiu-se para sua aplicação.

Neste passo a conversão em imagem foi realizada de forma integrada ao treinamento do modelo, uma vez que, diferentemente do DeepInsight e do NCTD, o HACNet não é apenas um conversor, mas um mecanismo de aprendizado de representação em imagem e classificação simultânea, como explicado anteriormente, de forma a produzir imagens que sejam simultaneamente úteis para a classificação e semelhantes a imagens-modelo associadas às classes.

Nesse contexto, como o sistema original oferece um sistema de treinamento parametrizável através de arquivos, foi possível adaptá-lo para considerar os datasets relevantes deste trabalho sem alteração ao algoritmo de treinamento e classificação em si. Isso foi feito essencialmente por meio da mudança do laço de repetição padrão para considerar os 5 folds para cada dataset gerados anteriormente, a inclusão das tabelas e dos metadados de folds na pasta de datasets e a adição de código relevante para considerá-los assim como os datasets originais.

Para cada conjunto de dados, as imagens-modelo associadas às classes DR, UR e ND foram as de letras do alfabeto disponibilizadas por padrão com o projeto. Portanto, o modelo foi treinado para criar imagens representando a letra A para DR, B para UR e C para ND, além de considerar a perda da classificação. Isso é possível através do componente de erro quadrático entre os pixels da imagem-modelo e a imagem obtida presente na função de perda do HACNet e visualizável na fórmula 3. Os parâmetros utilizados foram os padrões com exceção do número de divisões, que foi de 5 para considerar os exatos 5 folds obtidos anteriormente. Entre os valores de parâmetros destacáveis estão os números de iterações de treinamento e retreinamento de 8000 e 2000, respectivamente, o que foi constatado como o mínimo para um notável aprendizado das imagens (Matsuda *et al.*, 2024), coeficiente

λ de valor 10, assim como usado nas tentativas com melhores criações das imagens no artigo original, e um tamanho de imagem-alvo de 100x100 – relativamente bem maior que os tamanhos obtidos com o NCTD e o DeepInsight. Ademais, o uso da rede de classificação padrão – que é a ResNet18 – é a mesma usada para as classificações na etapa seguinte como será descrito, o que permite maior especificidade às imagens geradas anteriormente com o mesmo modelo de avaliador.

Ainda, foi necessário criar um script para extração das representações em imagens de cada amostra em virtude de o HACNet por padrão armazenar apenas algumas durante o processo de treinamento para demonstração, juntamente dos arquivos .pth representando os diferentes estados das redes atoras e críticas ao final do treinamento. A descrição do processo realizado aqui foge do contexto, porém basta dizer que a partir disso foi possível gerar as respectivas imagens para cada dataset seguindo a mesma organização de dataset, fold e classe.

2.3.5 Treinamento dos modelos baseados em CNN

O treinamento e a classificação das imagens foram realizados com uma arquitetura baseada em ResNet18 modificada para as particularidades das entradas e implementada em PyTorch. Como as imagens geradas pelos conversores foram tratadas como imagens em tons de cinza, a primeira camada convolucional da arquitetura foi modificada para receber uma única dimensão de canal. Além disso, a camada totalmente conectada final foi substituída para produzir três saídas, correspondentes às classes UR, DR e ND. Os pesos da rede foram inicializados sem pré-treinamento em ImageNet, uma vez que as imagens utilizadas neste trabalho não correspondem a imagens naturais e possuem significado estrutural distinto.

O treinamento da CNN foi realizado separadamente para cada combinação de conjunto de dados, conversor e fold de validação cruzada, usando as divisões criadas anteriormente na proporção 80% treinamento e 20% teste de forma estratificada, mantendo as proporções de classes originais em cada fold. Para cada um deles, o modelo foi treinado sobre o subconjunto de treino e avaliado sobre o

subconjunto de teste correspondente. O processo de treinamento envolveu um tamanho de lote de 64 e 50 épocas de treinamento. A função de perda utilizada foi a entropia cruzada categórica, adequada para classificação multiclasse, enquanto que o otimizador escolhido foi o AdamW, com uma taxa de decaimento de pesos de 0,0001 e taxa de aprendizado de 0,001. A taxa de aprendizado ainda esteve sujeita a um agendador de decaimento de cosseno, de forma a lentamente diminuí-la de acordo com a época e tornar o aprendizado mais rápido das primeiras épocas e mais refinado nas últimas.

Durante o pré-processamento das imagens para a CNN, foram aplicadas transformações compatíveis com a natureza artificial das imagens geradas. As imagens foram convertidas para tensores e então normalizadas para uma escala de ponto flutuante entre -1 e 1, de forma a não tornar a rede enviesada a valores numéricos mais altos nem a sentidos positivos. Transformações de aumento de dados frequentemente empregadas em imagens naturais, como espelhamento horizontal ou vertical, foram explicitamente evitadas, pois poderiam alterar a estrutura espacial produzida pelos conversores e modificar o significado relativo das posições dos pixels. Dessa forma, buscou-se preservar a correspondência entre características tabulares e posições da imagem codificadas pelos conversores.

O processo de treinamento envolveu várias funções auxiliares e facilitadores do PyTorch. Foram criados dataloaders de treinamento, validação e teste, que permitiram a estruturação dessas etapas em torno do número de lotes especificados. No processo de treinamento, para cada época o modelo foi configurado ao modo de treinamento, e então testado em um conjunto de validação consistindo de 15% do conjunto de treinamento original. Ou seja, a divisão final foi 68/12/20 para as alocações de treinamento, validação e teste. Ainda, a forma de avaliação foi idêntica para os conjuntos de validação e teste – o modelo foi configurado para modo de avaliação e as métricas de avaliação acurácia, F1-score, precisão e revocação foram computadas utilizando funções disponibilizadas pelo Scikit-learn. Os resultados da validação serviram para salvar o melhor modelo após cada época com base na acurácia medida, permitindo que ele fosse carregado posteriormente para o teste.

Especificamente para o HACNet, foi feita uma adaptação para que os pesos do ResNet18 fossem carregados dos anteriormente obtidos na conversão ponta-a-

ponta dessa técnica. Dessa forma, o avaliador usado aqui poderia aprender a especificidade treinada, retendo a capacidade das imagens geradas pelo HACNet de serem classificadas corretamente. Por outro lado, também se estruturou o script de treinamento e classificação de forma que ele fosse parametrizável para fácil reutilização através da adição de parâmetros que permitem especificar o dataset para execução e o caminho dos pesos do HACNet utilizados, entre outras utilidades. Nos treinamentos finais, o carregamento se deu por utilizar os pesos critic.pth obtidos do HACNet, inserindo-os na mesma pasta base das imagens.

Para cada fold de cada alvo de treinamento, os resultados das classificações, incluindo rótulo verdadeiro, rótulo obtido e probabilidades de classificação foram armazenados em arquivos .csv. As melhores previsões considerando todos os folds também foram armazenadas como um resultado geral, assim como arquivos sumário em json que armazenam dados de treinamento de todas as 50 épocas. Para auxiliar na interpretação visual dos resultados alcançados, também foram armazenadas tabelas no formato de matrizes de confusão e de concordância como importante ferramentas de análise comparativa resultados obtidos de diferentes estratégias de classificação.

Por fim, após a extração dos diversos dados sobre os resultados, foram computados testes estatísticos de Wilcoxon (Wilcoxon, 1992) sobre as diferentes combinações de métodos empregados, visando aferir a significância estatística da diferença entre as métricas de performance desses casos, em especial a acurácia. Isso foi feito com ajuda da função Wilcoxin disponibilizada pela biblioteca SciPy (Virtanen *et al.*, 2020).

2.3.6 Classificação com modelos tradicionais de aprendizado de máquina

Além dos modelos baseados em CNN treinados sobre as imagens geradas pelos conversores, foram avaliados modelos tradicionais de aprendizado de máquina treinados diretamente sobre as tabelas de características gênicas. Essa etapa teve como objetivo estabelecer uma fonte tabular de comparação com a abordagem de conversão T2I seguida de classificação convolucional. Dessa forma, foi possível avaliar se a transformação das features em imagens trouxe ganho preditivo em relação ao uso direto da representação tabular original.

A classificação tradicional foi realizada separadamente para cada um dos três conjuntos de dados utilizados neste trabalho: LUAD, KIRC e PRAD. As tabelas de entrada corresponderam aos arquivos finais de features gênicas, nos quais cada linha representa um gene e cada coluna representa uma característica numérica derivada da análise de expressão diferencial ou das contagens normalizadas. A coluna de rótulo foi utilizada como variável alvo, enquanto as demais colunas numéricas foram utilizadas como variáveis preditoras. Assim como nas etapas de conversão em imagem e treinamento das CNNs, foram excluídos identificadores não numéricos e informações que não deveriam ser utilizadas diretamente como entrada dos modelos.

Foram avaliados cinco classificadores tradicionais: LR, XGBoost, Random Forest, SVM e Naive Bayes gaussiano. LR foi incluída como modelo linear de referência para classificação multiclasse, enquanto que RF e XGBoost foram incluídos por serem métodos ensemble baseados em árvores de decisão, frequentemente competitivos em problemas tabulares. A máquina de vetores de suporte foi utilizada com kernel radial, permitindo a modelagem de fronteiras de decisão não lineares. Por fim, o Naive Bayes gaussiano foi incluído como modelo probabilístico simples, servindo como comparação adicional em relação aos modelos mais complexos.

Todos os modelos foram implementados em Python utilizando bibliotecas do ecossistema Scikit-learn, exceto o XGBoost que foi implementado com a biblioteca homônima. Para garantir reprodutibilidade, foi utilizado o estado aleatório 42 nos modelos e na geração das divisões de validação cruzada sempre que aplicável. A avaliação foi realizada com validação cruzada estratificada em cinco folds, preservando aproximadamente a proporção das classes DR, UR e ND em cada partição. Em cada fold, o modelo foi ajustado sobre o subconjunto de treinamento e avaliado sobre o subconjunto de teste correspondente, de forma que cada gene fosse classificado uma vez em uma predição fora do conjunto de treinamento.

Os hiperparâmetros foram mantidos fixos ao longo dos datasets, buscando uma comparação padronizada entre modelos e conjuntos de dados. A regressão logística foi treinada com parâmetro de regularização C igual a 1,0 e limite máximo de 2000 iterações. O XGBoost foi configurado para classificação multiclasse probabilística, utilizando 300 estimadores, taxa de aprendizado de 0,05,

profundidade máxima 5, min_child_weight igual a 2, subsample e colsample_bytree iguais a 0,8, regularização L2 igual a 1,0, métrica de avaliação mlogloss e método de construção de árvores hist. O Random Forest foi treinado com 300 árvores e mínimo de duas amostras por folha. A máquina de vetores de suporte utilizou kernel RBF, C igual a 2,0, gamma definido como scale e estimativa de probabilidades ativada. O Naive Bayes gaussiano foi utilizado com var_smoothing igual a 10^{-9}. Por outro lado, regressão logística, máquina de vetores de suporte e Naive Bayes gaussiano receberam também padronização por StandardScaler, de modo que as variáveis passassem a ter média zero e variância unitária antes do ajuste dos modelos.

Diferentemente do treinamento das CNNs, os modelos tradicionais não utilizaram uma divisão adicional de validação interna, pois não houve treinamento iterativo por épocas nem seleção de melhor época de treinamento. Assim, a comparação foi realizada a partir das predições obtidas nos subconjuntos de teste de cada fold. Para cada modelo e dataset, foram armazenadas as predições de cada gene, as probabilidades associadas a cada classe, as métricas por fold e a matriz de confusão. Também foram geradas tabelas comparativas, uma das quais representa os resultados das métricas dos modelos ordenadas de acordo com o F1-score macro, e outras que apresentam os resultados dos testes de diferença estatística Wilcoxin entre métodos. Esses testes foram computados da mesma forma como descrito anteriormente.

O desempenho dos modelos foi avaliado com as mesmas métricas gerais utilizadas na análise das CNNs: acurácia, precisão macro, revocação macro e F1-score macro. A escolha por métricas macro teve como objetivo atribuir o mesmo peso a cada classe, independentemente de sua frequência no conjunto de dados. Essa decisão é particularmente importante em uma tarefa como a classificação de genes diferencialmente expressos, na qual o desempenho sobre genes regulados positivamente e negativamente pode ser mascarado caso sejam utilizadas apenas métricas globais influenciadas pela classe majoritária. As matrizes de confusão foram utilizadas para analisar os principais tipos de erro cometidos pelos modelos, especialmente confusões entre genes UR, DR e ND.

Ao final dessa etapa, os resultados obtidos pelos modelos tradicionais foram comparados aos resultados das CNNs treinadas sobre imagens geradas por DeepInsight, NCTD e HACNet. Essa comparação permitiu discutir se a etapa de

conversão T2I contribuiu efetivamente para a classificação dos genes ou se os padrões discriminativos presentes nas 12 features originais já eram suficientemente capturados pelos modelos tabulares. Dessa forma, os modelos tradicionais atuaram como referência metodológica essencial para interpretar os resultados dos conversores e avaliar a real vantagem da representação em imagem no contexto deste trabalho.

Além do desempenho preditivo, os conversores foram comparados quanto a fatores como estabilidade da representação gerada, preservação da informação tabular, interpretabilidade da imagem e facilidade de relacionamento entre pixels e características originais. No caso de métodos determinísticos, como o NCTD, avaliou-se a consistência da geração das imagens a partir da mesma entrada. No caso de métodos baseados em projeção, como o DeepInsight, considerou-se a sensibilidade da representação aos parâmetros de redução de dimensionalidade e à discretização das coordenadas na grade de pixels. Para o HACNet, considerou-se o potencial de interpretação da atenção aprendida e da relação entre imagens geradas e imagens-modelo.

Por fim, os resultados foram analisados separadamente para cada tipo de câncer e também de forma comparativa entre os três conjuntos de dados. Essa organização permitiu observar se o desempenho dos conversores variou conforme o conjunto de dados, a distribuição das classes ou a estrutura das características gênicas. A comparação entre CNNs treinadas sobre imagens e modelos tradicionais treinados sobre as tabelas originais foi utilizada para discutir se a representação em imagem contribuiu efetivamente para a classificação de genes diferencialmente expressos, ou se os modelos tabulares foram suficientes para capturar os padrões presentes nas características construídas.

2.3.7 Análise de enriquecimento funcional GO

Após a etapa de classificação dos genes pelas CNNs, foi realizada uma análise de enriquecimento funcional com o objetivo de avaliar se os conjuntos de genes classificados como regulados positivamente ou negativamente pelas técnicas T2I apresentavam coerência biológica. Essa etapa não teve como objetivo validar individualmente cada predição, mas sim verificar se os grupos de genes resultantes

dos modelos estavam associados a processos biológicos esperados ou interpretáveis no contexto dos tipos tumorais analisados.

A análise foi realizada separadamente para cada combinação entre dataset e conversor, utilizando os arquivos finais de predição gerados pelas CNNs. Para cada arquivo, os genes foram separados de acordo com a classe predita pelo modelo, formando dois conjuntos principais: genes classificados como UR e genes classificados como DR. Genes classificados como ND não foram submetidos à análise de enriquecimento, pois essa classe representa principalmente genes sem alteração estatística relevante entre os grupos tumoral e normal.

Como os arquivos de predição armazenavam os identificadores das linhas classificadas, foi necessário mapear esses índices de volta aos respectivos identificadores gênicos presentes nas tabelas originais de features. Para isso, os identificadores foram recuperados a partir da tabela correspondente de cada dataset, removendo-se sufixos de versão dos identificadores Ensembl quando presentes. O conjunto de background utilizado em cada análise correspondeu aos genes presentes na tabela de features do respectivo dataset, isto é, aos genes que permaneceram após as etapas de filtragem, análise diferencial e construção da tabela gene-cêntrica. Essa escolha buscou evitar o uso de um universo de referência mais amplo que o conjunto efetivamente disponível para classificação. Esse procedimento segue a lógica implementada no script de enriquecimento, que mapeia as predições de volta aos genes da tabela de features e define o background a partir desses mesmos genes .

O enriquecimento funcional foi realizado com o pacote g:Profiler (Reimand *et al.*, 2016), considerando o organismo *Homo sapiens* e restringindo a análise aos termos de processos biológicos da Gene Ontology (GO:BP) (The Gene Ontology Consortium, 2026). Para cada conjunto de genes regulados positivamente ou negativamente, foram retidos os termos enriquecidos com limiar de significância de 0,05, utilizando correção por FDR para múltiplos testes (Benjamini; Hochberg, 1995). Os resultados de cada análise foram exportados em arquivos CSV, contendo os termos enriquecidos, valores de significância e demais informações retornadas pelo g:Profiler.

Por fim, para facilitar a interpretação visual dos resultados, foram gerados gráficos de barras horizontais contendo os termos mais significativos de cada

análise. Os termos foram ordenados pelo valor-p ajustado e representados pela transformação $-\log_{10}$ do valor de significância, permitindo destacar os processos biológicos com maior evidência estatística. Quando necessário, nomes de termos muito longos foram abreviados para melhorar a legibilidade das figuras. Dessa forma, a análise de enriquecimento funcional serviu como uma etapa complementar de interpretação biológica dos resultados obtidos pelos conversores T2I e pelas CNNs, permitindo avaliar se os genes classificados em cada direção de regulação preservavam sinais funcionais compatíveis com os processos moleculares esperados em câncer.

2.4 Análise e discussão

2.4.1 Tabelas obtidas e visualização dos rótulos iniciais obtidos

As estruturas das tabelas gene-cêntricas obtidas para cada dataset podem ser resumidas de acordo com a Tabela 1.

Tabela 1 - rótulos dos genes filtrados após análise por DESeq2

Projeto	Genes	Features	DR	UR	ND
TCGA-LUAD	19.304	12	1.904	3.623	13.777
TCGA-KIRC	18.603	12	1.846	3.899	12.858
TCGA-PRAD	17.533	12	1.363	1.159	15.011

Fonte: elaboração própria com base nas tabelas originais

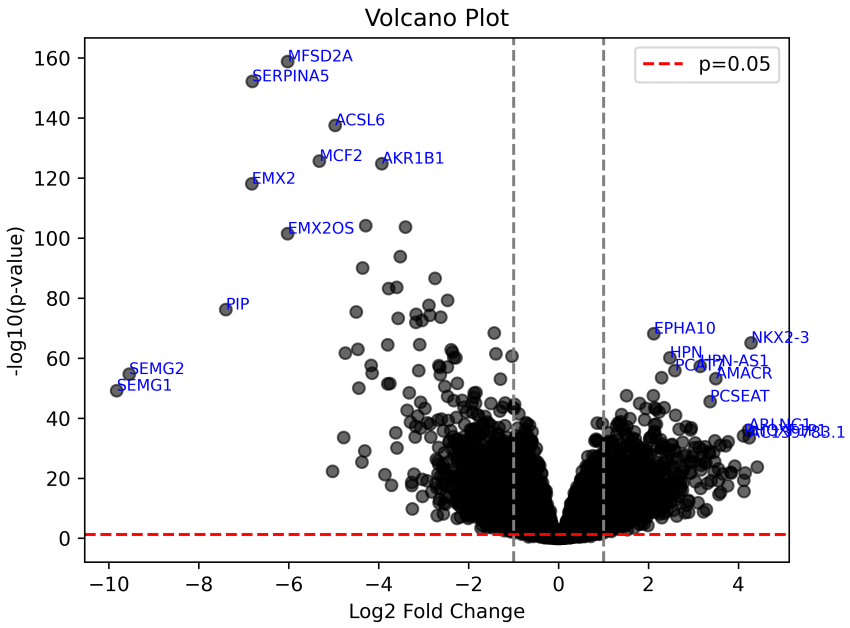
É possível perceber que a maioria das classificações obtidas com DESeq2 foram de genes não significativos, seguindo com variações para as proporções de UR e DR entre datasets, mas no geral havendo um maior número de URs. Significativamente, também se nota que todos os datasets tiveram a maioria dos seus genes filtrados, reduzindo a quantidade de amostras gênicas analisadas a níveis abaixo de 20.000. Isso ocorreu devido à etapa de filtragem por CPM de mais de 1 para pelo menos 10% das células sequenciadas, que pode-se concluir então ter concluído seu objetivo ao limitar o estudo a uma quantidade boa de genes mas não esmagadora e que sejam contributivas à análise, sem prejudicar os testes estatísticos devido a valores extremos de contagem.

Para LUAD e KIRC, a quantidade de genes filtrados corresponde inteiramente a essa etapa. No entanto, para PRAD, identificou-se que a matriz de resultados do

DESeq2 havia assinalado valores NA para p-valor e padj, significando que algumas amostras foram tiradas da análise devido ao critério de distância de Cook. Como esse critério tem como intuito justamente eliminar genes que podem prejudicar a análise por baixo número de réplicas, a geração das tabelas finais nesse caso ainda contou com a exclusão dos respectivos genes, que representaram 558 das 43.127 filtragens feitas para esse dataset.

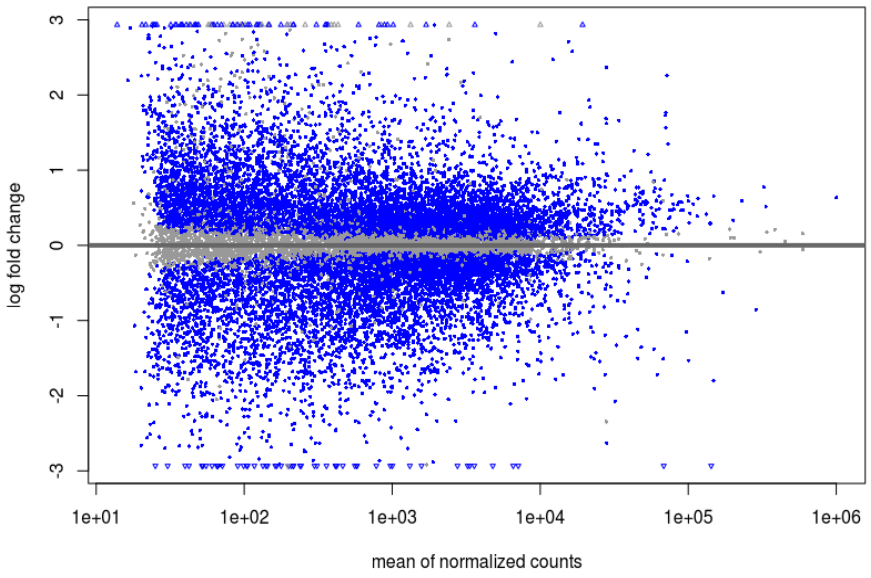
As características das rotulações por DESeq2 foram analisadas visualmente por meio de gráficos de vulcão (*volcano plots*) e MA (*MA plots*). Os gráficos de vulcão consistem de esquematizações $\log_2$FC por -$\log_{10}$p-valor, representando, portanto, uma das principais formas de visualizar as duas principais variáveis envolvidas na rotulação de DEGs por DESeq2. De forma semelhante, os gráficos MA mostram a média de contagens normalizadas por $\log_2$FC, permitindo observar as variações entre tipos de amostras tendo em conta o nível de expressão global desses genes. No contexto deste trabalho, os gráficos de vulcão foram construídos por meio da diagramação das variáveis relevantes provenientes das tabelas com funções da biblioteca matplotlib (Hunter, 2007), enquanto que os gŕaficos MA foram obtidos de uma função padrão do DESeq2.

Gráfico 1 - gráfico de vulcão para as tabelas obtidas com o dataset PRAD



Fonte: elaboração própria com o auxílio da ferramenta matplotlib

Gráfico 2 - gráfico MA para as tabelas obtidas com o dataset PRAD



Fonte: elaboração própria com o auxílio da ferramenta DESeq2

Em geral, as tabelas consistem de poucos rótulos diferencialmente expressos. Porém, as visualizações construídas permitiram identificar vários genes cuja

combinação de log_2FC torna-os muito relevantes para aferir sua relevância quanto à patologia em questão. Isso é especialmente evidente no Gráfico 1, o qual apresentou uma quantidade surpreendente de genes com expressão muito reprimida em amostras de câncer com uma significância estatística igualmente grande. Nesse gráfico, os 10 genes com maior valor absoluto log_2FC * -log_{10}p-valor para cada sentido de variação foram marcados. Em suma, as tabelas geradas apresentaram uma estrutura condizente com o esperado e apresentaram genes com diferentes níveis de certeza para a classificação aos rótulos-alvo.

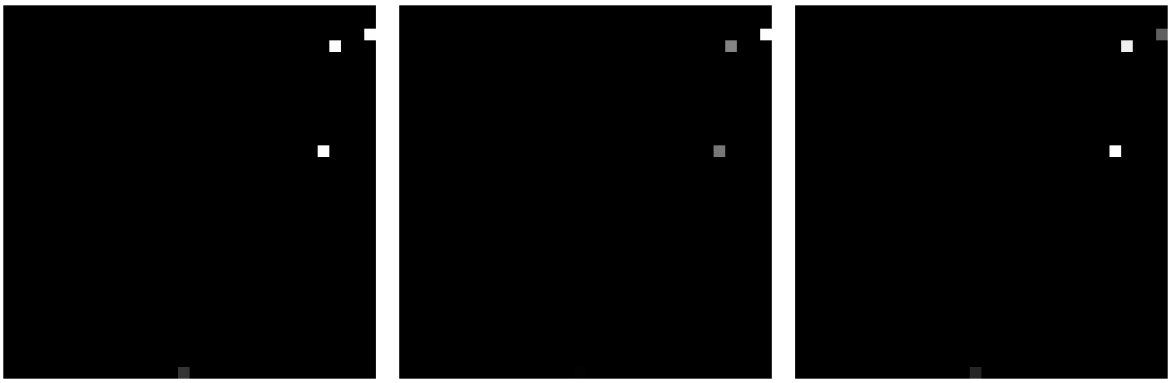
2.4.2 Imagens geradas pelos conversores

As imagens adquiridas com cada conversor apresentaram características bastante distintas, as quais podem ser observadas sumariamente na Figura 2. Essas particularidades estão de acordo com o padrão de imagens reportadas por outros trabalhos, então foram interpretadas como sinais de conversões bem-sucedidas.

Para o DeepInsight, as imagens apresentaram o padrão de pixels isolados em um fundo predominantemente preto como esperado. Também foi possível observar a existência de um certo grau de clusterização dos pixels nem que seja de um cluster claramente identificável, o que mostra que a escolha de t-SNE com perplexidade pode ter sido uma escolha boa o suficiente, embora seja difícil estimar se foi o melhor possível.

O mecanismo de atuação do DeepInsight pode incentivar uma certa análise visual das imagens para identificar padrões de features diferencialmente expressas através da diferença de intensidade dos pixels para diferentes classes. Ao fazer isso, identificou-se que as posições dos pixels realmente são estáveis quanto a suas posições, mas amostras de uma mesma classe podem variar significativamente quanto a suas intensidades, tornando muito difícil estabelecer padrões críveis meramente pelo visual. Nessa mesma linha, as imagens da Figura 3, representativas de diferentes classes, mostram padrões diferentes que suscitam uma análise diferencial.

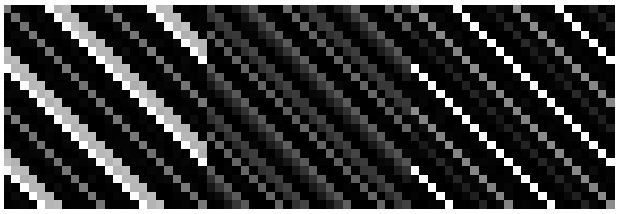
Figura 3 - exemplos de imagens geradas para as classes ND, DR e UR com o dataset LUAD e conversor DeepInsight



Fonte: elaboração própria com a API pyDeepInsight

Para o NCTD, as imagens geradas tiveram o padrão característico de diagonais em virtude das operações de duplicação e rotação feitas nesse conversor. Pode-se dizer que a interpretabilidade dessas imagens é ainda mais difícil que a das imagens geradas com DeepInsight, uma vez que o padrão é exatamente o mesmo, apenas com intensidades de píxeis diferentes.

Figura 4 – exemplos de imagens geradas para as classes ND, DR e UR com o dataset KIRC e conversor NCTD



Fonte: elaboração própria

Já com o HACNet, identificou-se que as imagens estavam corretamente sendo construídas de acordo com os modelos, sendo DR mapeado para “A”, UR para “B” e ND para “C”. Contudo, elas tiveram muita dificuldade em representá-los sem ruído, uma vez que na maioria das imagens são visíveis os formatos de outras letras. A Figura 5 ilustra esse problema, ao mostrar que, embora as três imagens pertençam a classes diferentes, todas foram mapeadas para a letra “C”, com ruídos de outras letras. Esse foi um padrão recorrente nas imagens geradas, e ao mesmo passo que as imagens UR e DR dificilmente eram modeladas precisamente, as

imagens ND eram mapeadas para a classe correta com muito menos ruído. O fato dos datasets estarem fortemente desequilibrados no sentido da classe ND provavelmente é um dos principais responsáveis por tamanha influência da letra “C” em todas as imagens.

Mesmo que esses casos sejam particularmente ruins quanto à interpretabilidade da diferença de classes, eles demonstram um padrão preocupante e que poderia inclusive danificar o poder de avaliação em si ao usar essas imagens posteriormente para classificação. Contudo, de fato eles não são de todo inesperados uma vez que (Matsuda *et al.*, 2024) já havia posto a dificuldade do HACNet em datasets com baixo número de features em razão do limitado poder de expressão de só poder assinalar uma única feature para cada imagem. Nesse contexto, eles também haviam chegado a imagens não próximas dos modelos em um dataset com 20 features, enquanto que os usados aqui são de 12. Isso fornece grande evidência da incompletude do HACNet para gerar imagens interpretáveis em datasets de baixa dimensionalidade de feature.

Por outro lado, isso não elimina a possibilidade de gerar imagens altamente interpretáveis no contexto de análise de DEGs. Uma abordagem utilizando um maior número de features, como é o caso (Kakati *et al.*, 2022), poderia possivelmente aproveitar essa maior dimensionalidade para gerar imagens mais interpretáveis, como já foi reportado pelos próprios autores do HACNet em datasets de TCGA. O processo aplicado aqui serviu, portanto, para expor ao mesmo tempo o potencial de técnicas que usam atenção para a geração de imagens interpretáveis, assim como limitações ainda presentes no HACNet.

Figura 5 - exemplos de imagens geradas para as classes DR, ND e UR com HACNet



Fonte: geração de pixels pelo ator HACNet

2.4.3 Classificações por conversor-CNN

Os desempenhos obtidos na etapa de treinamento e classificação das imagens convertidas – medidos por medidas padronizadas como acurácia, precisão, revocação e F1-score – estão disponíveis na Tabela 2. Notoriamente, as classificações no geral obtiveram desempenhos muito bons ou quase-perfeitos, exceto para as imagens geradas com DeepInsight, que obtiveram resultados parecidos ao de um classificador aleatório em termos de acurácia, mas consideravelmente pior se levados em conta F1-score e as medidas correlatas de precisão e revocação

Esse fato de início gerou muita dúvida quanto aos motivos que poderiam tê-lo causado, suscitando explorações que incluíram a análise dos padrões de imagens gerados para os diferentes folds com DeepInsight, onde se percebeu que imagens entre folds tinham uma relevante diferença quanto ao posicionamento, o que é exposto na Figura 7. Já a Figura 8 mostra as mesmas amostras também em 4 folds diferentes, mas utilizando apenas um ajuste com todas as amostras da tabela de features de LUAD, ao invés das 5 feitas separadas para cada divisão treinamento-teste.

Analisando comparativamente os dois conjuntos de imagem, fica claro que o ajuste geral aprende um posicionamento das posições de features e seus clusters que é reproduzido entre as amostras, enquanto que a abordagem original, utilizando ajustes separados para cada fold, gerava posicionamentos distintos, geralmente posicionando o mesmo cluster em cantos diferentes da imagem, além de parecer ser estocástico quanto ao posicionamento relativo das features. De fato, t-SNE é um método altamente estocástico, sendo o uso de um mesmo estado aleatório não suficiente para gerar as mesmas distribuições.

Tabela 2 - Desempenho médio e desvio-padrão dos conversores T2I com ResNet-18 considerando os 5 folds de cada combinação.

Conversor	Dataset	Acurácia	Precisão	Revocação	F1-Score
HACNet	KIRC	0.9988 ± 0.000	0.9989 ± 0.000	0.9924 ± 0.000	0.9984 ± 0.000
HACNet	LUAD	0.9965 ± 0.009	0.9972 ± 0.001	0.9980 ± 0.001	0.9948 ± 0.001
HACNet	PRAD	0.9951 ± 0.001	0.9891 ± 0.008	0.9865 ± 0.005	0.9877 ± 0.003
DeepInsight	KIRC	0.5400 ± 0.243	0.3810 ± 0.151	0.3827 ± 0.049	0.2843 ± 0.070
DeepInsight	LUAD	0.4063 ± 0.406	0.3873 ± 0.387	0.3851 ± 0.385	0.2407 ± 0.240
DeepInsight	PRAD	0.5717 ± 0.388	0.3507 ± 0.161	0.3275 ± 0.098	0.2605 ± 0.172
NCTD	KIRC	0.9960 ± 0.004	0.9929 ± 0.011	0.9952 ± 0.003	0.9939 ± 0.006
NCTD	LUAD	0.9898 ± 0.011	0.9937 ± 0.004	0.9765 ± 0.030	0.9845 ± 0.018
NCTD	PRAD	0.9640 ± 0.043	0.9340 ± 0.114	0.9226 ± 0.129	0.9114 ± 0.104
DeepInsight_full	KIRC	0.9928 ± 0.008	0.9940 ± 0.003	0.9871 ± 0.016	0.9904 ± 0.009
DeepInsight_full	LUAD	0.9968 ± 0.001	0.9951 ± 0.002	0.9957 ± 0.002	0.9954 ± 0.002
DeepInsight_full	PRAD	0.9953 ± 0.003	0.9955 ± 0.003	0.9805 ± 0.019	0.9876 ± 0.009

Figura 6 - 4 imagens geradas com DeepInsight e dataset PRAD para a mesma amostra em diferentes folds.



Fonte: elaboração própria com a API pyDeepInsight

Estima-se, portanto, que essa diferença entre folds demonstra uma falha do sistema de ajuste empregado ao DeepInsight – cada fold é treinado com um arranjo espacial estocástico, para então ele ser testado em outro. Por isso, foram feitas novas execuções com DeepInsight e conversor, considerando um ajuste global aos dados de treinamento ao invés de ajustes por folds. Os resultados desses treinamentos também estão na Tabela 2. Assim, nota-se que com esta metodologia diferente, os resultados com DeepInsight passaram a performar de maneira competitiva ou melhor que os outros conversores em todos os datasets, constituindo uma etapa essencial no projeto.

É preciso ter em consideração, no entanto, que o fato do ajuste ter sido

realizado com base em todas as amostras do dataset nessas conversões configura um vazamento de informação, pois a representação usada pelo classificador foi parcialmente definida com base nos próprios dados posteriormente empregados na avaliação. Dessa forma, embora os rótulos das amostras de teste não tenham sido utilizados diretamente, a distribuição dessas observações participou da construção da projeção bidimensional e, consequentemente, da organização espacial das features na imagem. Assim, a alta performance observada com essa metodologia deve ser levada com cautela, pois pode ser que não seja totalmente reproduzível ao aplicar o modelo a amostras não vistas.

Por outro lado, examinando as performances das combinações conversor-dataset como um todo, os desempenhos são notáveis e, em alguns casos até surpreendentes. O NCTD é um conversor relativamente novo, e os resultados apresentados em seu trabalho original são muito promissores, uma vez que competem com ou superam modelos estratégias com RF e XGBoost. Ademais, os resultados com HACNet parecem ter sido os melhores no geral, o que contrasta com o fato do padrão de imagens gerados por este método não terem se assemelhado de forma significativa. Isso provavelmente advém do fato da limitação de expressão do dataset em tornar as imagens interpretáveis a humanos não significa sua incapacidade para gerar valores de atenção relevantes para classificação para uma CNN robusta como a ResNet18.

Outro ponto interessante é que as execuções com as imagens geradas por HACNet foram capazes de aprender os padrões com força mesmo sem carregar os pesos oriundos do processo ponta-a-ponta do HACNet. Isso é importante pois significa que os valores de pixels gerados pelo conversor foram capazes de ser generalizados para a classificação correta sem dependência da rede original, um fator considerável tendo em conta a natureza de conversão e avaliação acoplada do HACNet. Os resultados das execuções sem carregamento prévio de pesos conseguiram performances muito semelhantes às execuções contando com esse passo, motivo pelo qual elas não foram incorporadas à Tabela 2.

Figura 7 - 4 imagens geradas com DeepInsight usando ajuste geral e dataset PRAD para a mesma amostra em diferentes folds.



Fonte: elaboração própria com a API pyDeepInsight

2.4.4 Classificação das tabelas com modelos tradicionais de aprendizado de máquina

Os resultados obtidos pelos modelos tradicionais de ML treinados diretamente sobre as tabelas originais demonstram que a representação tabular construída a partir das features gênicas contém informação altamente discriminativa para a classificação dos genes em DR, UR e ND. Conforme apresentado na Tabela 3, Random Forest e XGBoost atingiram desempenho quase perfeito nos três conjuntos de dados, com valores de F1-score macro superiores a 0,997 em praticamente todos os casos. O Random Forest apresentou o melhor desempenho geral, atingindo F1-score de 0,9996 em KIRC, 0,9999 em LUAD e 1,0000 em PRAD. O XGBoost também apresentou resultados muito próximos, confirmando a forte adequação de modelos baseados em árvores ao formato tabular deste problema.

A regressão logística e a SVM também apresentaram desempenho elevado, ainda que levemente inferior aos modelos ensemble. Esse resultado indica que a separação entre as classes é, em grande parte, bem definida pelas variáveis construídas, uma vez que até modelos relativamente mais simples foram capazes de aproximar as fronteiras de decisão entre genes regulados positivamente, regulados negativamente e não diferencialmente expressos. Por outro lado, o Naive Bayes apresentou desempenho consideravelmente inferior, especialmente em KIRC e PRAD, com F1-score macro de 0,6452 e 0,6000, respectivamente. Esse comportamento é coerente com as limitações desse modelo, que assume independência condicional entre as variáveis e distribuições gaussianas por classe, pressupostos pouco compatíveis com features fortemente relacionadas entre si,

como log2FC, log2FC absoluto, razão de expressão, médias de expressão e valores de significância estatística.

Tabela 3 - Desempenho médio e desvio-padrão dos modelos de ML usados para classificar as tabelas originais dos datasets considerando 5 folds.

Dataset	Modelo	Acurácia	Precisão	Revocação	F1-Score
KIRC	Random Forest	0.9998 ± 0.000	0.9995 ± 0.000	0.9998 ± 0.000	0.9996 ± 0.000
KIRC	XGBoost	0.9997 ± 0.000	0.9997 ± 0.000	0.9994 ± 0.001	0.9995 ± 0.000
KIRC	Regressão logística	0.9978 ± 0.001	0.9986 ± 0.000	0.9942 ± 0.002	0.9965 ± 0.001
KIRC	SVM	0.9943 ± 0.001	0.9947 ± 0.000	0.9868 ± 0.004	0.9907 ± 0.002
KIRC	Naive Bayes	0.6194 ± 0.012	0.6627 ± 0.014	0.8127 ± 0.005	0.6452 ± 0.014
LUAD	Random Forest	0.9999 ± 0.000	0.9998 ± 0.000	1.0000 ± 0.000	0.9999 ± 0.000
LUAD	XGBoost	0.9992 ± 0.000	0.9990 ± 0.000	0.9987 ± 0.000	0.9988 ± 0.000
LUAD	Regressão logística	0.9977 ± 0.000	0.9982 ± 0.000	0.9950 ± 0.000	0.9966 ± 0.000
LUAD	SVM	0.9942 ± 0.001	0.9943 ± 0.001	0.9894 ± 0.002	0.9918 ± 0.001
LUAD	Naive Bayes	0.8047 ± 0.014	0.7214 ± 0.012	0.9018 ± 0.008	0.7696 ± 0.013
PRAD	Random Forest	1.0000 ± 0	1.0000 ± 0	1.0000 ± 0	1.0000 ± 0
PRAD	XGBoost	0.9990 ± 0.000	0.9968 ± 0.002	0.9987 ± 0.001	0.9977 ± 0.001
PRAD	Regressão logística	0.9981 ± 0.001	0.9984 ± 0.001	0.9922 ± 0.004	0.9952 ± 0.002
PRAD	SVM	0.9953 ± 0.001	0.9936 ± 0.003	0.9831 ± 0.005	0.9883 ± 0.003
PRAD	Naive Bayes	0.7028 ± 0.020	0.5632 ± 0.007	0.8798 ± 0.009	0.6000 ± 0.013

Fonte: elaboração própria

Nesse contexto, os modelos tradicionais funcionam como uma referência fundamental para os conversores T2I. Como as tabelas originais já permitem desempenho quase perfeito, a conversão em imagem só pode ser considerada vantajosa se preservar essa informação discriminativa sem introduzir instabilidade, perda de informação ou redução substancial de interpretabilidade. Portanto, a comparação com Random Forest, XGBoost, regressão logística, SVM e Naive Bayes permite avaliar se a etapa de conversão T2I adiciona algum benefício real ao pipeline ou se, no caso deste conjunto de features, apenas reorganiza uma informação que já é facilmente explorada por modelos tabulares.

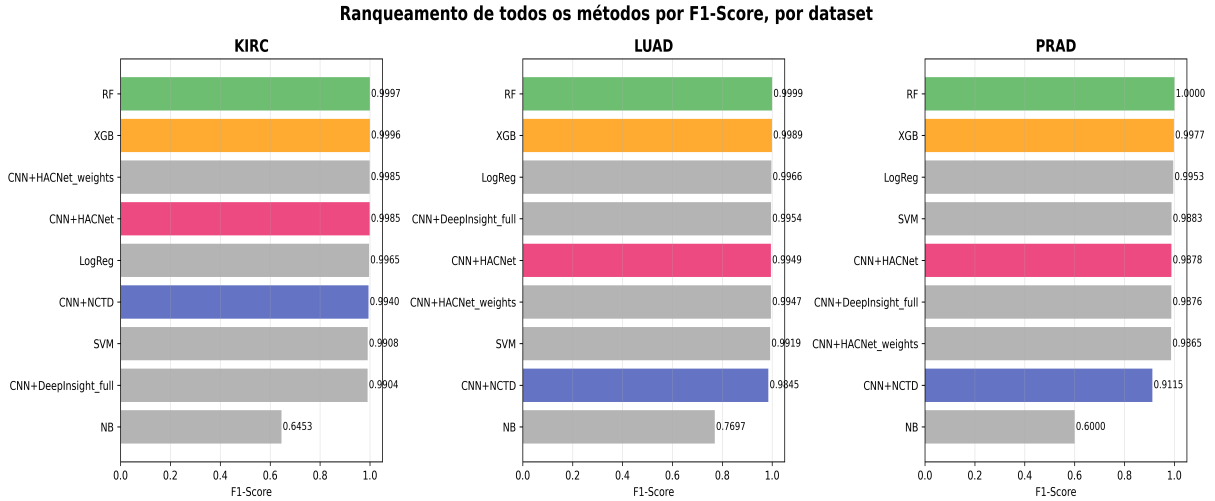
2.4.5 Comparação entre pipelines conversor-CNN e modelos tradicionais

A comparação entre os resultados dos modelos tradicionais e dos pipelines conversor-CNN pode ser resumida pelo gráfico 3, mostrando que a conversão em imagem não trouxe uma vantagem preditiva clara em relação ao uso direto das tabelas. Os melhores modelos tradicionais, especialmente Random Forest e XGBoost, apresentaram desempenho igual ou superior ao dos melhores conversores em praticamente todos os conjuntos de dados. Esse resultado reforça a hipótese discutida na fundamentação teórica de que métodos baseados em árvores ainda possuem forte adequação indutiva a dados tabulares, especialmente quando o número de features é pequeno e as variáveis carregam diretamente a informação usada para definir os rótulos.

Entre os conversores avaliados em condições sem ajuste global contaminado, HACNet e NCTD apresentaram os melhores desempenhos. O HACNet atingiu F1-score de 0,9984 em KIRC, 0,9948 em LUAD e 0,9877 em PRAD, enquanto o NCTD atingiu 0,9939 em KIRC, 0,9845 em LUAD e 0,9114 em PRAD. Esses valores demonstram que ambos os métodos foram capazes de preservar grande parte da informação discriminativa das tabelas originais. No entanto, o NCTD apresentou queda mais acentuada em PRAD, sugerindo que sua transformação geométrica direta pode ser menos robusta em alguns contextos, especialmente quando a distribuição das classes é mais desbalanceada ou quando a separação entre UR e DR é menos numerosa em relação à classe ND.

Ainda sobre NCTD, o patamar de classificação alcançado aqui não chegou ao potencial demonstrado em (Alenizy; Berri, 2025), onde um pipeline usando esse conversor obteve desempenhos melhores que todos os outros modelos de ML, incluindo XGBoost, em 7 de 10 datasets. Pelo contrário, XGBoost superou o NCTD em todos os datasets. Ou seja, pelo menos para o contexto deste estudo, o NCTD não teve a vantagem significativa que se poderia pensar ter.

Gráfico 3 - diagramas de barras horizontais para comparação dos F1-Score obtidos por todos os métodos empregados.



Fonte: medidas de performances obtidas pelos respectivos métodos e diagramação por matplotlib

É importante ter em conta, contudo, que mesmo que tenham sido observados padrões de desempenhos melhores em HACNet em relação aos outros conversores, por exemplo, este dado não foi confirmado por testes estatísticos de diferença Wilcoxon. Ainda que o poder estatístico desses testes seja limitado neste trabalho em função do baixo número de folds e execuções, o maior valor-p entre diferenças de conversores foi de 0,3125, o que traz dúvidas no que diz respeito à relevância dessas diferenças. De fato, as pequenas diferenças observadas neste estudo para a maioria das combinações urge um maior número de comparações em datasets diferentes e maiores.

Embora na prática nenhuma diferença entre um par de métodos utilizados tenha atingido um patamar significativo de 0,05, os dados presentes na Tabela 4 revelam outros padrões interessantes. Excluindo-se os pares que compreendem apenas abordagens T2I, os valores-p foram consistentemente baixos, alcançando, na maioria das vezes, valores de 0,0625 que são o limite estatístico para o teste de Wilcoxin considerando 5 folds. Esse último fato novamente demonstra a cautela com que devem ser levados os resultados, mas permite inferir que algumas diferenças realmente foram mais significativas que outras, como é o caso para todas as combinações envolvendo Naive Bayes e todas as combinações conversor-modelo ensemble baseado em árvore (RF e XGBoost). Também é possível ver que até mesmo HACNet apresentou uma diferença média negativa em relação a todos os modelos tradicionais exceto Naive Bayes, com significância estatística no limite possível, o que reforça a afirmação de que a abordagem T2I não apresentou

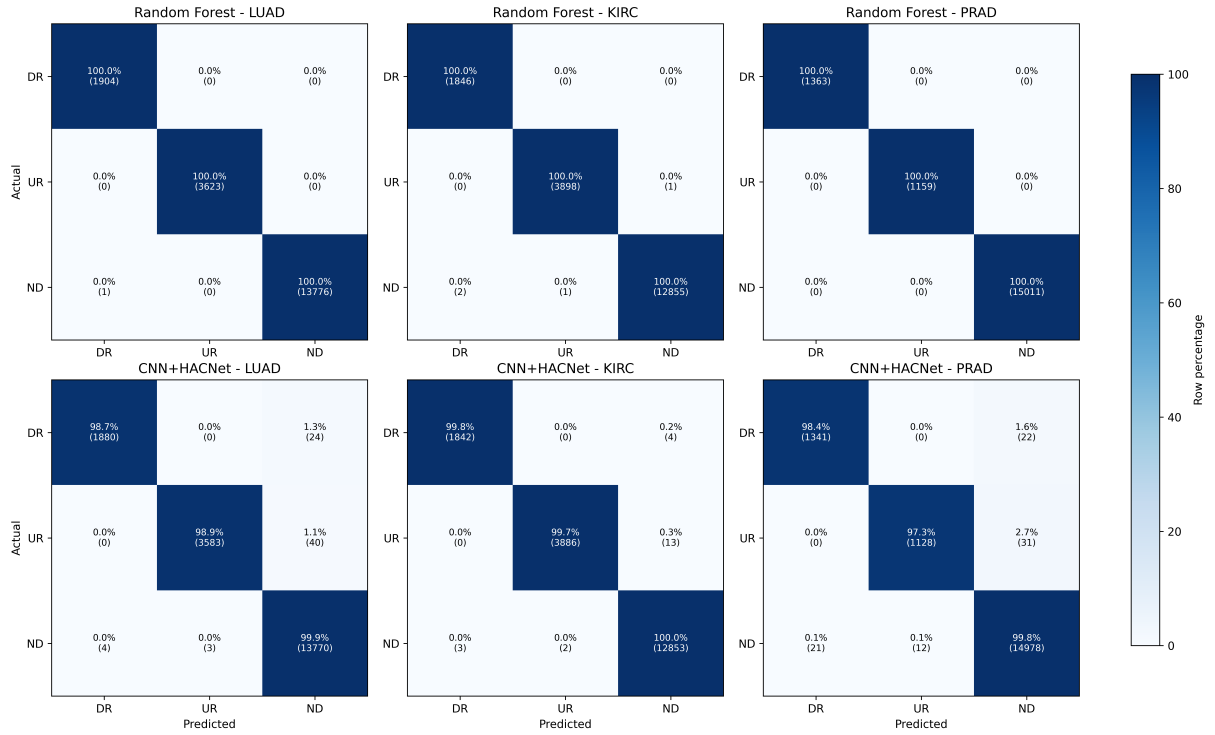
vantagens significativas em relação a métodos estabelecidos no que tange às medidas de performance neste trabalho.

Tabela 4 – Significância estatística da diferença por valor-p e diferença média entre os resultados de acurácia para as diferentes combinações no dataset PRAD.

Par de métodos	Valor-p	Diferença média
DeepInsight x NCTD	0.4375	0.0324
DeepInsight x HACNet	0.8125	0.0010
DeepInsight x Regressão logística	0.6250	-0.0019
DeepInsight x Naive Bayes	0.0625	0.2933
DeepInsight x Random Forest	0.0625	-0.0038
DeepInsight x Support Vector Machine	0.6250	0.0008
DeepInsight x XGBoost	0.0625	-0.0029
NCTD x HACNet	0.3125	-0.0313
NCTD x Regressão logística	0.1250	-0.0343
NCTD x Naive Bayes	0.0625	0.2609
NCTD x Random Forest	0.0625	-0.0362
NCTD x Support Vector Machine	0.3750	-0.0315
NCTD x XGBoost	0.0625	-0.0353
HACNet x Regressão logística	0.0625	-0.0030
HACNet x Naive Bayes	0.0625	0.2922
HACNet x Random Forest	0.0625	-0.0049
HACNet x Support Vector Machine	0.8125	-0.0002
HACNet x XGBoost	0.0625	-0.0039

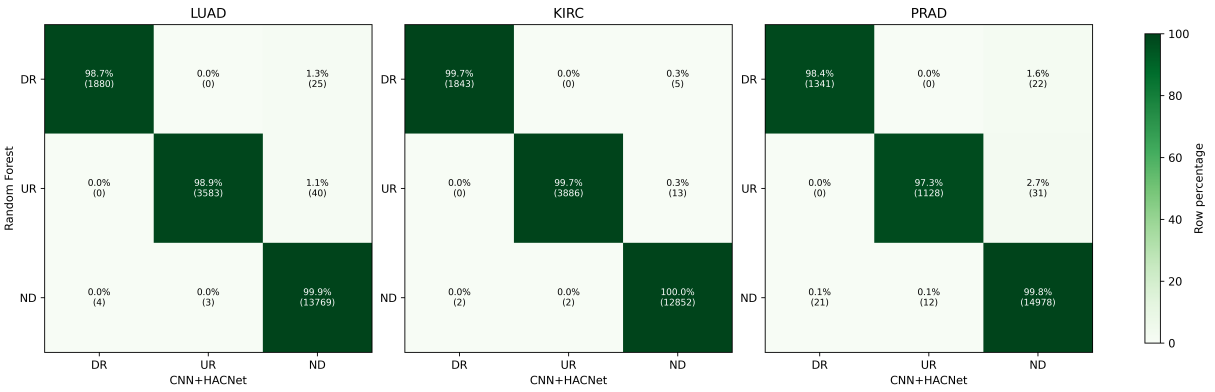
Fonte: elaboração própria

Gráfico 4 - matrizes de confusão dos melhores modelos tradicional (RF) e baseado em conversores (HACNet) obtidos para os datasets.



Fonte: medidas de performances obtidas pelos respectivos métodos e diagramação por matplotlib

Gráfico 5 – matrizes de concordância comparando as predições dos melhores modelos tradicional (RF) e baseado em conversores (HACNet) obtidos para os datasets



Fonte: medidas de performances obtidas pelos respectivos métodos e diagramação por matplotlib

Por outro lado, do ponto de vista de interpretabilidade, os modelos tradicionais também apresentam uma vantagem prática importante. Como operam diretamente sobre as features originais, suas decisões podem ser relacionadas de forma imediata a variáveis como log2FC, valor-p ajustado, média de expressão tumoral, média de expressão normal e dispersão. Mesmo que Random Forest e XGBoost não sejam modelos completamente transparentes, suas importâncias de variáveis e

métodos posteriores de explicabilidade, como SHAP ou permutação de features, podem ser interpretados diretamente no espaço tabular original. Já no caso das CNNs, a explicação precisa passar por uma representação intermediária em imagem, o que torna necessário reconstruir a relação entre regiões da imagem, pixels e features originais.

Todavia, ainda que modelos de ensemble baseados em árvores continuem sendo considerados por muitos como o estado-da-arte para os casos analisados, matrizes de confusão como as expostas no gráfico 4 e as matrizes de concordância do gráfico 5 indicam que as diferenças de performance podem ser consideradas pequenas em determinadas situações. No contexto de DEGs, em que geralmente se busca os genes mais relacionados com a patologia em questão, e é possível que aqueles que não são classificados corretamente estão próximos da fronteira de decisão entre classes, algumas unidades a mais de erros como nas matrizes do HACNet podem ser aceitáveis dependendo da contrapartida.

Nesse contexto, aplicações T2I como o HACNet se diferenciam pelo potencial de gerar imagens interpretáveis, além de permitirem mais facilmente a construção de mapeamentos entre cada pixel da imagem e a feature responsável pelo seu valor. Especificamente, imagens interpretáveis através do mapeamento a certos modelos podem ser importantes ao fornecer mais uma camada de auditabilidade visual, o que pode ser particularmente relevante em alguns contextos regulatórios, por exemplo. Dessa forma, o HACNet apresenta uma abordagem diferente dos outros conversores que é muito promissora ao combinar classificação, representação espacial aprendida e mapeamento pixel-feature.

Nesse aspecto, os outros conversores apresentaram limitações distintas. O NCTD possui a vantagem de ser determinístico e estável, mas suas imagens são formadas por padrões diagonais artificiais, derivados da duplicação e rotação das features. Assim, embora seja fácil saber como a imagem foi gerada, a proximidade espacial entre pixels não possui necessariamente significado biológico ou estatístico. O DeepInsight, por sua vez, tem a vantagem conceitual de tentar aproximar features semelhantes no espaço bidimensional, mas sua dependência de métodos estocásticos de redução de dimensionalidade dificulta a estabilidade entre folds e torna sua aplicação sensível à forma como o ajuste é realizado.

Dessa forma, a análise conjunta indica que, para as tabelas construídas neste

trabalho, os modelos tabulares tradicionais foram a alternativa mais eficiente, simples e interpretável. Os conversores T2I mostraram que é possível preservar a informação necessária para a classificação em imagens artificiais, mas não demonstraram superioridade preditiva em relação aos modelos tradicionais. A contribuição principal da análise dos conversores, portanto, está menos em superar os modelos tabulares e mais em revelar como diferentes estratégias de representação visual lidam com estabilidade, preservação de informação e explicabilidade em um problema de classificação de DEGs.

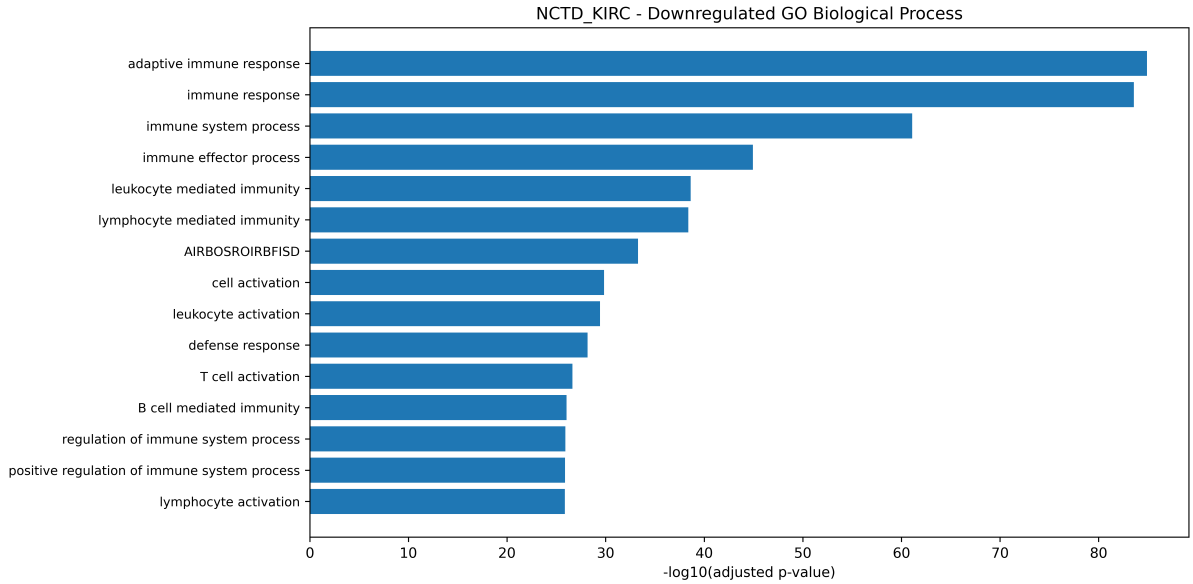
Por outro enfoque, é importante lembrar neste contexto de métricas de performance em geral muito positivas que as tabelas utilizadas para classificação em todos os casos contêm os mesmos dados que foram usados para computar os rótulos. Logo, resultados muito bons e quase perfeitos em muitos casos devem ser interpretados como sinais de solidez básicos dessas abordagens, e não como fontes para estabelecer que os métodos empregados neste trabalho aprenderam relações biológicas e transcriptômicas relevantes e são muito eficientes em classificação de DEGs. Pelo contrário, eles estão aprendendo uma lógica de classificação muito semelhante àquela presente no DESeq2. Logo, mesmo que HACNet e XGBoost tenham apresentado resultados muito bons no contexto deste trabalho, isso não significa que a utilização dessas técnicas em outros datasets terá desempenhos semelhantes.

2.4.6 Enriquecimento funcional GO e análise biológica dos genes classificados

Os resultados mais consistentes de enriquecimento funcional GO, considerando todos os datasets, foram observados para KIRC. Nos genes classificados como downregulated, tanto nos resultados derivados do DeepInsight com ajuste global quanto nos resultados de HACNet e NCTD, houve forte enriquecimento de processos relacionados à resposta imune, incluindo resposta imune adaptativa, processo do sistema imune, ativação de leucócitos, imunidade mediada por linfócitos e resposta de defesa. A convergência desses termos entre diferentes classificadores sugere que esse conjunto de genes possui um sinal funcional robusto, não dependente exclusivamente de um único método de conversão. Para os genes classificados como UR em KIRC, destacaram-se termos

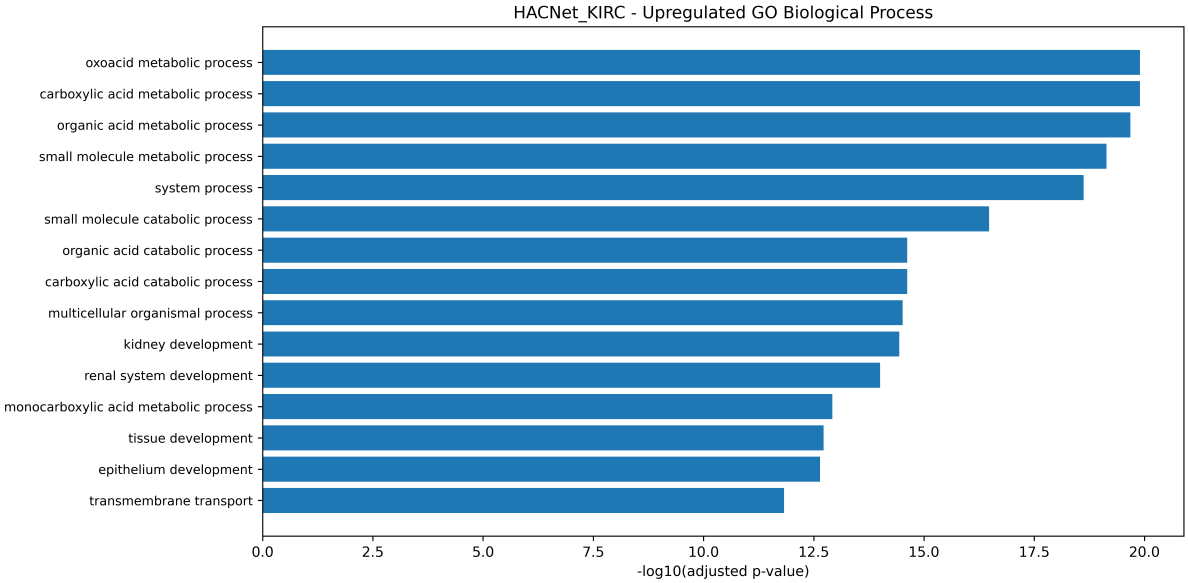
relacionados a metabolismo de oxoácidos, metabolismo de ácidos carboxílicos, metabolismo de pequenas moléculas e desenvolvimento renal, resultados coerentes com a origem tecidual do carcinoma renal e com a importância de alterações metabólicas em tumores (Hanahan; Weinberg, 2011).

Gráfico 5 - diagramação por barras horizontais das significâncias estatísticas de termos GO enriquecidos negativamente para a combinação NCTD-KIRC



Fonte: dados obtidos com g:Profiler e diagramação por matplotlib

Gráfico 6 - diagramação por barras horizontais das significâncias estatísticas de termos GO enriquecidos positivamente para a combinação HACNet-KIRC



Fonte: dados obtidos com g:Profiler e diagramação por matplotlib

Em LUAD, os genes classificados como UR apresentaram enriquecimento de processos ligados a motilidade celular, desenvolvimento de estruturas anatômicas, comunicação celular, desenvolvimento da vasculatura e locomoção, novamente de forma consistente. Esses termos são biologicamente relevantes no contexto tumoral, uma vez que processos como motilidade, comunicação celular e desenvolvimento vascular estão associados à invasão, progressão tumoral e angiogênese (Hanahan; Weinberg, 2011). Já os genes classificados como DR em LUAD apresentaram enriquecimento para resposta imune adaptativa, resposta imune mediada por imunoglobulina, imunidade mediada por células B e também termos relacionados à divisão nuclear e ciclo celular meiótico. Essa combinação indica que os conjuntos classificados capturaram módulos funcionais distintos, associados tanto à resposta imune quanto a processos de divisão celular.

Nos genes classificados como regulados positivamente em PRAD, os três conversores apresentaram enriquecimento expressivo e bastante convergente para processos amplos associados a desenvolvimento e organização multicelular, incluindo processos de organismo multicelular, desenvolvimento de estrutura anatômica, processo de desenvolvimento, diferenciação celular, desenvolvimento de sistema e morfogênese de estrutura anatômica. Além desses termos gerais, também apareceram de forma consistente processos relacionados a contração muscular, sistema muscular, circulação sanguínea, comunicação celular, sinalização,

neurogênese e migração celular. No contexto de PRAD, esse padrão pode ser interpretado como compatível com alterações em processos de diferenciação, remodelamento tecidual, comunicação entre células e organização funcional do tecido tumoral.

Nos genes classificados como DR, por outro lado, os três conversores apresentaram um conjunto menor de termos enriquecidos, mas igualmente com alta coerência interna. Os principais termos estiveram relacionados à regulação da divisão mitótica, separação de cromátides irmãs, segregação cromossômica, checkpoint do fuso mitótico, transição metáfase/anáfase e divisão nuclear. Esse padrão foi observado de forma muito semelhante para DeepInsight, HACNet e NCTD, embora com diferenças no número total de termos enriquecidos e na significância estatística. A convergência desses processos indica que os genes classificados como downregulated em PRAD carregam um sinal funcional associado ao controle do ciclo celular e da segregação cromossômica, processos diretamente relevantes para a biologia tumoral.

De maneira geral, a análise de enriquecimento GO valida biologicamente as rotulagens feitas após RNA-seq e explicita que os modelos de classificação foram capazes de classificar DEGs relevantes no contexto das amostras de câncer usadas. No entanto, para obter validações ainda mais robustas no nível de genes, uma estratégia adicional que poderia ser usada é comparar genes corretamente classificados, incorretamente classificados e localizados próximos às fronteiras de decisão dos modelos. Isso poderia ajudar a identificar se os erros de classificação estão concentrados em genes funcionalmente ambíguos, genes com baixa evidência estatística ou genes associados a processos biológicos mais complexos.

3 CONCLUSÃO

De modo geral, o objetivo de comparar conversores T2I para a classificação de genes diferencialmente expressos foi alcançado, uma vez que foi possível implementar um pipeline completo envolvendo construção de tabelas gene-cêntricas, geração de imagens por diferentes métodos, treinamento de CNNs e comparação com modelos tradicionais de aprendizado de máquina. Essa estrutura permitiu observar que a eficiência dos conversores não depende apenas da métrica final de classificação, mas também da forma como cada método organiza as features no espaço da imagem, da estabilidade dessa organização entre folds e da possibilidade de relacionar os padrões aprendidos pela CNN às características biológicas e estatísticas originais.

Tendo em vista o foco do projeto, a abordagem utilizada aqui teve sucesso ao identificar particularidades, vantagens e desvantagens no que diz respeito aos conversores e classificação de DEGs. Os resultados fortes obtidos com NCTD, como acurácias de 0.9960, 0.9898 e 0.9640 para KIRC, LUAD e PRAD, demonstram que até mesmo metodologias simples e determinísticas que codifiquem os valores dos dados tabulares originais como intensidades de pixels com transformações mínimas podem ser bem-sucedidas para classificação com CNN.

As dificuldades que as imagens de DeepInsight tiveram ao serem geradas com ajustes diferentes por fold ilustram os cuidados que se deve tomar ao usar métodos baseados em técnicas de redução de dimensionalidade altamente estocásticos como t-SNE, e suscita mais comparações tendo em conta as diferentes técnicas que podem ser usadas com o DeepInsight, tal como PCA e UMAP. Por exemplo, será que a teórica menor estocasticidade do PCA poderia evitar esse problema? O uso de diferentes valores de perplexidade também pode ser explorado.

Por outro lado, HACNet mostrou ser a estratégia de conversão mais relevante para os datasets abordados, uma vez que não apenas teve, no geral, as melhores performances com respeito aos outros conversores, mas também porque estabelece uma via de interpretabilidade das classificações por meio visual e de mapeamento direto dos pixels da imagem a certas features. As imagens obtidas com HACNet geralmente não eram representativas das imagens-modelo estabelecidas em função

do baixo número de features, sendo isto um ponto de atenção ao usá-lo com datasets de baixa dimensionalidade e caracterizando um grande ponto de melhora a ser desenvolvido para futuros avanços.

No geral, as aplicações T2I comparadas neste trabalho tiveram bons resultados quando analisadas isoladamente, mas a comparação com outros modelos de ML, em especial RF e XGBoost, demonstraram o porquê de a aplicação de fluxos como esses ainda constituir uma metodologia auxiliar e experimental para classificações. Para que essa área avance de patamar, é necessário que as relações espaciais desenvolvidas pelos conversores sejam significativas de forma a alcançar performances comparáveis ou superiores às de modelos ensemble baseados em árvores e que também ofereçam alternativas de interpretabilidade. Como o HACNet foi o conversor que mais chegou perto desse patamar, ele foi considerado o mais promissor aqui.

É importante, no entanto, relembrar o mecanismo que permitiu a aprendizagem dos rótulos neste trabalho. Embora as boas performances obtidas demonstrem a solidez funcional das abordagens estudadas no que diz respeito ao seu propósito básico – classificar genes –, isso não significa que essas aplicações aprenderam relações biológicas novas a partir das quais seria possível classificar cada padrão de gene em UR, DR e ND. Na verdade, a estruturação final estabelecida para as tabelas posteriormente aplicadas a conversores e classificadores significa que uma fatia muito grande do poder preditivo neste contexto reside em justamente aprender os fatores considerados pelo DESeq2 para rotulação de DEGs. Isso é evidente ao se lembrar que as features constituídas nas tabelas são sumarizações obtidas diretamente ou indiretamente dessa ferramenta.

Não obstante, essa abordagem não invalida os resultados concretizados, mas os coloca no contexto de uma análise comparativa de conversores T2I tendo em vista outros métodos de ML, contexto no qual foi importante para focar nas vantagens e desvantagens de cada metodologia sem complicar demasiadamente o processo experimental. Ainda, o trabalho realizado em (Kakati *et al.*, 2022) foi semelhante no sentido que usou rótulos criados por DESeq2 para julgar classificações por CNN. Os dados de entrada nesse contexto foram os próprios perfis de expressão dos genes para cada célula do dataset, que podem ser considerados como indiretamente relacionados às features usadas aqui, uma vez

que elas são essencialmente diferentes métricas estatísticas das matrizes de contagens.

No que diz respeito às contribuições prestadas por este projeto, destacam-se o avanço no entendimento de conversores novos pouco estudados como NCTD e HACNet, bem como suas vantagens e desvantagens, e rotas de melhoria. Considerando o NCTD em particular, foi fornecida uma fonte de comparação menos otimista em relação a sua aplicação anterior, enquanto que para o HACNet foram apontadas importantes fragilidades que devem ser endereçadas para sua bem-sucedida aplicação futura. Além disso, todos os conversores comparados já haviam sido utilizados em pipelines de classificação de dados de RNA-seq, mas não necessariamente na aplicação de classificação de DEGs, o que já não é mais verdade.

Tendo em vista o grande potencial de interpretabilidade em métodos T2I como o HACNet, faltou neste projeto realizar uma análise de importância de features deste conversor, aproveitando a sua capacidade de relacionar pixels das imagens-alvo a uma única feature. Análises futuras que levem em consideração estimativas de importância de features e métodos de interpretabilidade visuais, modelo-agnósticos ou não, são essenciais para desvendar a verdadeira aplicabilidade de métodos T2I como importantes fontes alternativas de confiabilidade e auditabilidade em aplicações como análise de DEGs, ou meras ferramentas auxiliares para predição. Constituem, portanto, uma importante área de desenvolvimento para o futuro.

Outra direção natural para novos estudos seria ampliar o conjunto de conversores avaliados, incluindo métodos desenvolvidos especificamente para limitações observadas neste trabalho. O LM-IGTD (Gómez-Martínez *et al.*, 2024), por exemplo, foi proposto para dados tabulares de baixa dimensionalidade e tipos mistos, combinando geração estocástica de features com uma versão modificada do IGTD, além de manter uma relação interpretável entre features originais e regiões da imagem. De forma complementar, métodos como Table2Image (Lee *et al.*, 2025) buscam criar representações visuais mais realistas e diversas, incorporando relações estatísticas entre variáveis, como multicolinearidade, e estratégias de interpretabilidade combinando explicações no espaço tabular e no espaço da imagem. Esses métodos poderiam ser especialmente relevantes para tabelas gene-

cêntricas com poucas features, como as deste trabalho, nas quais a simples projeção ou reorganização geométrica pode não explorar todo o potencial da conversão T2I.

Também seria importante avaliar formulações alternativas da própria tarefa de classificação de DEGs. Uma possibilidade seria treinar modelos sem utilizar diretamente features que participam da definição dos rótulos, como padj, valor-p e $\log_2$FC, mantendo apenas características derivadas das contagens normalizadas. Outra alternativa seria construir representações com maior dimensionalidade, incluindo perfis de expressão por amostra, estatísticas por subgrupos clínicos, anotações funcionais, informações de localização genômica ou medidas de coexpressão. Isso permitiria testar se os conversores T2I são capazes de capturar padrões biológicos mais complexos, e não apenas reproduzir fronteiras estatísticas já definidas pelo DESeq2.

REFERÊNCIAS

ALENIZY, H. A.; BERRI, J. Transforming tabular data into images via enhanced spatial relationships for CNN processing. **Scientific Reports**, [*s. l.*], v. 15, n. 1, p. 17004, 2025.

ALHARBI, F.; VAKANSKI, A. Machine Learning Methods for Cancer Classification Using Gene Expression Data: A Review. **Bioengineering**, [*s. l.*], v. 10, n. 2, p. 173, 2023.

BAZGIR, O. *et al.* Representation of features as images with neighborhood dependencies for compatibility with convolutional neural networks. **Nature Communications**, [*s. l.*], v. 11, n. 1, p. 4391, 2020.

BENJAMINI, Y.; HOCHBERG, Y. Controlling The False Discovery Rate - A Practical And Powerful Approach To Multiple Testing. **J. Royal Statist. Soc., Series B**, [*s. l.*], v. 57, p. 289–300, 1995.

BORISOV, V. *et al.* Deep Neural Networks and Tabular Data: A Survey. **IEEE Transactions on Neural Networks and Learning Systems**, [*s. l.*], v. 35, n. 6, p. 7499–7519, 2024.

BREIMAN, L. Random Forests. **Machine Learning**, [*s. l.*], v. 45, n. 1, p. 5–32, 2001.

CASTILLO-CARA, M. *et al.* TINTO: Converting Tidy Data into image for classification with 2-Dimensional Convolutional Neural Networks. **SoftwareX**, [*s. l.*], v. 22, p. 101391, 2023.

CHEN, T.; GUESTRIN, C. XGBoost: A Scalable Tree Boosting System. *In*: XGBOOST, 2016. **Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. [*S. l.*]: [*s. d.*], 2016. p. 785–794. Disponível em: http://arxiv.org/abs/1603.02754. Acesso em: 24 jun. 2026.

COLAPRICO, A. *et al.* TCGAbiolinks: an R/Bioconductor package for integrative analysis of TCGA data. **Nucleic Acids Research**, [*s. l.*], v. 44, n. 8, p. e71, 2016.

CONESA, A. *et al.* A survey of best practices for RNA-seq data analysis. **Genome Biology**, [*s. l.*], v. 17, p. 13, 2016.

COOK, R. D. Detection of Influential Observation in Linear Regression. **Technometrics**, [*s. l.*], v. 19, n. 1, p. 15–18, 1977.

DOBIN, A. *et al.* STAR: ultrafast universal RNA-seq aligner. **Bioinformatics**, [*s. l.*], v. 29, n. 1, p. 15–21, 2013.

FENG, S. *et al.* TabDEG: Classifying differentially expressed genes from RNA-seq data based on feature extraction and deep learning framework. **PLOS ONE**, [*s. l.*], v. 19, n. 7, p. e0305857, 2024.

FISHER, A.; RUDIN, C.; DOMINICI, F. All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. **Journal of machine learning research : JMLR**, [*s. l.*], v. 20, p. 177, 2019.

GÓMEZ-MARTÍNEZ, V. *et al.* **LM-IGTD: a 2D image generator for low-dimensional and mixed-type tabular data to leverage the potential of convolutional neural networks**. [*S. l.*]: arXiv, 2024. Disponível em: http://arxiv.org/abs/2406.14566. Acesso em: 22 jun. 2026.

GRINSZTAJN, L.; OYALLON, E.; VAROQUAUX, G. **Why do tree-based models still outperform deep learning on tabular data?**. [*S. l.*]: arXiv, 2022. Disponível em: http://arxiv.org/abs/2207.08815. Acesso em: 24 jun. 2026.

GROSSMAN, R. L. *et al.* Toward a Shared Vision for Cancer Genomic Data. **New England Journal of Medicine**, [*s. l.*], v. 375, n. 12, p. 1109–1112, 2016.

HANAHAN, D.; WEINBERG, R. A. Hallmarks of cancer: the next generation. **Cell**, [*s. l.*], v. 144, n. 5, p. 646–674, 2011.

HARRIS, C. R. *et al.* Array programming with NumPy. **Nature**, [*s. l.*], v. 585, n. 7825, p. 357–362, 2020.

HE, K. *et al.* **Deep Residual Learning for Image Recognition**. [*S. l.*]: arXiv, 2015. Disponível em: http://arxiv.org/abs/1512.03385. Acesso em: 26 jun. 2026.

HOTELLING, H. Analysis of a complex of statistical variables into principal components. **Journal of Educational Psychology**, US, v. 24, n. 6, p. 417–441, 1933.

HUNTER, J. D. Matplotlib: A 2D Graphics Environment. **Computing in Science & Engineering**, [*s. l.*], v. 9, n. 3, p. 90–95, 2007.

KAKATI, T. *et al.* DEGnext: classification of differentially expressed genes from RNA-seq data using a convolutional neural network with transfer learning. **BMC Bioinformatics**, [*s. l.*], v. 23, n. 1, p. 17, 2022.

KALANTARI-DEHAGHI, M.; GHOHABI-ESFAHANI, N.; EMADI-BAYGI, M. From bulk RNA sequencing to spatial transcriptomics: a comparative review of differential gene expression analysis methods. **Human Genomics**, [*s. l.*], v. 20, n. 1, p. 9, 2025.

KOHAVI, R. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. **IJCAI International Joint Conference on Artificial Intelligence**, [*s. l.*], v. 2, p. 1137–1143, 1995.

LE CUN, Y. *et al.* Handwritten digit recognition with a back-propagation network. 1989, Cambridge, MA, USA. **Proceedings of the 3rd International Conference on Neural Information Processing Systems**. Cambridge, MA, USA: MIT Press, 1989. p. 396–404. Disponível em: Acesso em: 1 fev. 2026.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, [*s. l.*], v. 521, n. 7553, p. 436–444, 2015.

LEE, Seungeun *et al.* **Table2Image: Interpretable Tabular Data Classification with Realistic Image Transformations**. [*S. l.*]: arXiv, 2025. Disponível em: http://arxiv.org/abs/2412.06265. Acesso em: 22 jun. 2026.

LOVE, M. I.; HUBER, W.; ANDERS, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. **Genome Biology**, [*s. l.*], v. 15, n. 12, p. 550, 2014.

MA, S.; ZHANG, Z. **OmicsMapNet: Transforming omics data to take advantage of Deep Convolutional Neural Network for discovery**. [*S. l.*]: arXiv, 2019. Disponível em: http://arxiv.org/abs/1804.05283. Acesso em: 22 jun. 2026.

MAATEN, L. van der; HINTON, G. Visualizing Data using t-SNE. **Journal of Machine Learning Research**, [*s. l.*], v. 9, n. 86, p. 2579–2605, 2008.

MATSUDA, T. *et al.* HACNet: End-to-end learning of interpretable table-to-image converter and convolutional neural network. **Knowledge-Based Systems**, [*s. l.*], v. 284, p. 111293, 2024.

NETO, L. M.; NETO, S. R. da S.; ENDO, P. T. A comparative analysis of converters of tabular data into image for the classification of Arboviruses using Convolutional Neural Networks. **PLOS ONE**, [*s. l.*], v. 18, n. 12, p. e0295598, 2023.

PEDREGOSA, F. *et al.* Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, [*s. l.*], v. 12, n. 85, p. 2825–2830, 2011.

REIMAND, J. *et al.* g:Profiler—a web server for functional interpretation of gene lists (2016 update). **Nucleic Acids Research**, [*s. l.*], v. 44, n. Web Server issue, p. W83–W89, 2016.

ROBINSON, M. D.; MCCARTHY, D. J.; SMYTH, G. K. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. **Bioinformatics**, [*s. l.*], v. 26, n. 1, p. 139–140, 2010.

ROSATI, D. *et al.* Differential gene expression analysis pipelines and bioinformatic tools for the identification of specific biomarkers: A review. **Computational and Structural Biotechnology Journal**, [*s. l.*], v. 23, p. 1154–1168, 2024.

SELVARAJU, R. R. *et al.* Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. **International Journal of Computer Vision**, [*s. l.*], v. 128, n. 2, p. 336–359, 2020.

SHARMA, A. *et al.* DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture. **Scientific Reports**, [*s. l.*], v. 9, n. 1, p. 11399, 2019.

SIMONYAN, K.; VEDALDI, A.; ZISSERMAN, A. **Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps**. [*S. l.*]: arXiv, 2014. Disponível em: http://arxiv.org/abs/1312.6034. Acesso em: 22 jun. 2026.

SOMEPALLI, G. *et al.* **SAINT: Improved Neural Networks for Tabular Data via Row Attention and Contrastive Pre-Training**. [*S. l.*]: arXiv, 2021. Disponível em: http://arxiv.org/abs/2106.01342. Acesso em: 2 fev. 2026.

THE GENE ONTOLOGY CONSORTIUM. The Gene Ontology knowledgebase in 2026. **Nucleic Acids Research**, [*s. l.*], v. 54, n. D1, p. D1779–D1792, 2026.

VASWANI, A. *et al.* **Attention Is All You Need**. [*S. l.*]: arXiv, 2023. Disponível em: http://arxiv.org/abs/1706.03762. Acesso em: 31 jan. 2026.

VIRTANEN, P. *et al.* SciPy 1.0: fundamental algorithms for scientific computing in Python. **Nature Methods**, [*s. l.*], v. 17, n. 3, p. 261–272, 2020.

WEINSTEIN, J. N. *et al.* The Cancer Genome Atlas Pan-Cancer analysis project. **Nature Genetics**, [*s. l.*], v. 45, n. 10, p. 1113–1120, 2013.

WENRIC, S.; SHEMIRANI, R. Using Supervised Learning Methods for Gene Selection in RNA-Seq Case-Control Studies. **Frontiers in Genetics**, [*s. l.*], v. 9, 2018. Disponível em: https://www.frontiersin.org/journals/genetics/articles/10.3389/fgene.2018.00297/full. Acesso em: 22 jun. 2026.

WILCOXON, F. Individual Comparisons by Ranking Methods. New York, NY, Springer Series in Statistics, p. 196–202, 1992.

WU, H.; WANG, C.; WU, Z. A new shrinkage estimator for dispersion improves differential expression detection in RNA-seq data. **Biostatistics**, Oxford, England, v. 14, n. 2, p. 232–243, 2013.

ZHU, Y. *et al.* Converting tabular data into images for deep learning with convolutional neural networks. **Scientific Reports**, [*s. l.*], v. 11, n. 1, p. 11325, 2021a.

ZHU, Y. *et al.* Converting tabular data into images for deep learning with convolutional neural networks. **Scientific Reports**, [*s. l.*], v. 11, n. 1, p. 11325, 2021b.

GLOSSÁRIO

Background – conjunto de genes utilizado como universo de referência em análises de enriquecimento funcional.

Conversor T2I – método que transforma dados tabulares em uma representação em imagem para uso por redes neurais convolucionais.

Dataset – conjunto de dados utilizado como entrada em uma análise computacional.

Enriquecimento funcional – análise que testa se uma lista de genes está associada a processos biológicos, funções ou vias em frequência maior que a esperada ao acaso.

Feature – variável numérica usada como entrada por modelos de aprendizado de máquina.

Fold – uma das partições criadas durante a validação cruzada.

Matriz de confusão – tabela que compara classes verdadeiras e classes preditas por um classificador.

Modelo tabular – modelo treinado diretamente sobre uma tabela de características, sem conversão prévia para imagem.

Pipeline – sequência organizada de etapas computacionais aplicadas aos dados.

Reads – fragmento de RNA sequenciado por RNA-seq que pode ser alinhado com genes

Rótulo – classe atribuída a cada observação no treinamento ou na avaliação de um modelo.

Validação cruzada – estratégia de avaliação em que os dados são divididos em múltiplas partições para treinamento e teste repetidos.