UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO

Thamires Lopes das Mercês

Comparação de Técnicas de Redução de Dimensionalidade Aplicadas à Clusterização de Dados do Censo da Educação Superior

Recife

2025

Thamires Lopes das Mercês

Comparação de Técnicas de Redução de Dimensionalidade Aplicadas à Clusterização de Dados do Censo da Educação Superior

Monografia apresentada ao Curso de Bacharelado em Sistemas de Informação da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação.

Universidade Federal Rural de Pernambuco – UFRPE

Departamento de Estatística e Informática

Curso de Bacharelado em Sistemas de Informação

Orientador: Gabriel Alves de Albuquerque Júnior

Recife

2025

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema Integrado de Bibliotecas da UFRPE
Bibliotecária Suely Manzi – CRB/4 - 809

M554c Mercês, Thamires Lopes das
Comparação de técnicas de redução de dimensionalidade aplicadas à clusterização de dados do censo da educação superior / Thamires Lopes das Mercês. – 2025.
76 f.: il.

Orientador: Gabriel Alves de Albuquerque Júnior.
Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) – Universidade Federal Rural de Pernambuco, Departamento de Estatística e Informática, Recife, BR-PE, 2025.
Inclui bibliografia.

1. Educação – Aspectos demográficos 2. Censos 3. Ensino superior 4. Mineração de dados 5. Dimensionalidade 6. Aglomeração I. Albuquerque Júnior, Gabriel Alves de, orient. II. Título

CDD 020

THAMIRES LOPES DAS MERCÊS

COMPARAÇÃO DE TÉCNICAS DE REDUÇÃO DE DIMENSIONALIDADE APLICADAS À CLUSTERIZAÇÃO DE DADOS DO CENSO DA EDUCAÇÃO SUPERIOR

Monografia apresentada ao Curso de Bacharelado em Sistemas de Informação da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação.

Aprovada em: 08 de agosto de 2025.

BANCA EXAMINADORA

Gabriel Alves de Albuquerque Júnior
Departamento de Estatística e Informática
Universidade Federal Rural de Pernambuco

Roberta Macêdo Marques Gouveia
Departamento de Estatística e Informática
Universidade Federal Rural de Pernambuco

Emanuel Marques Queiroga
Diretoria de Tecnologia da Informação
Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense

Agradecimentos

Gostaria de agradecer a todos que participaram de alguma forma dessa minha longa caminhada e a Deus por me permitir estar aqui.

Primeiramente, gostaria de agradecer a minha família, meus pais Nadja Suely Lopes das Mercês e Edvaldo Julião das Mercês, por todo o apoio e amor que me deram durante toda a minha vida, sem eles eu não estaria aqui hoje. Ao meu irmão, Thales Lopes das Mercês, por ser o exemplo e suporte que eu sempre precisei.

Meu muito obrigada a Jesulen Vicente da Silva, por ser um amigo tão presente e que me ajudou tanto desde o começo deste curso. Meu agradecimento a Ana Clara Mazza Feitosa, por ter me dado todo o suporte no meu início de carreira e por ter se tornado uma grande amiga. Agradeço também a Tiago Sousa Carvalho, por todo o apoio e por não ter me deixado desistir.

Agradeço a todos os meus amigos e colegas de faculdade, que fizeram essa caminhada ser mais leve e me ensinaram muito no decorrer do caminho. Agradecimento especial para Guilherme Carneiro de Farias, por tanta parceria em tantos trabalhos de grupo e estudos, e por ter se tornado um amigo que espero levar para o restante da vida.

Meu agradecimento também a Gabriel Cezário Ramos dos Santos, Carlos Rogner de Oliveira Júnior, Felipe Burégio Viana e Andreza Dantas Layme Pifano de Moura, por todos os momentos necessários de descontração e risadas.

Agradeço também à Universidade Federal Rural de Pernambuco e a todo o corpo docente que faz parte do curso de Bacharelado em Sistemas de Informação. Também deixo aqui meu agradecimento ao meu professor e orientador Gabriel Alves de Albuquerque Júnior, que me guiou durante o período deste projeto.

Resumo

A grande quantidade de informações coletadas em censos da educação e avaliações nacionais demanda métodos eficientes para extração de conhecimento, permitindo identificar padrões e tendências relevantes. Nesse contexto, a clusterização se destaca como uma ótima técnica para segmentar e interpretar grandes volumes de dados educacionais, sendo o K-Means um dos algoritmos mais utilizados devido à sua simplicidade e eficiência. No entanto, quando aplicado a conjuntos de dados de alta dimensionalidade, seu desempenho pode ser comprometido, tornando necessário o uso de técnicas de redução de dimensionalidade como Principal Component Analysis (PCA), t-Distributed Stochastic Neighbor Embedding (t-SNE) e Uniform Manifold Approximation and Projection (UMAP). Este trabalho investiga o impacto dessas técnicas na qualidade dos agrupamentos gerados pelo K-Means em uma base de dados composta pela junção dos Microdados do Censo da Educação Superior de 2022 e os indicadores de qualidade educacional Conceito Enade e CPC. A análise é realizada utilizando o índice de silhueta como métrica de avaliação e comparando o tempo de execução de cada método. Com dois componentes, o PCA superou o t-SNE e o UMAP na maioria dos testes. Com três componentes, o PCA teve melhor desempenho que o t-SNE em todos os testes, mas ficou equilibrado com o UMAP, onde foi superior em cinco dos nove cenários. Observou-se, ainda, que a quantidade de clusters teve influência relevante nos resultados, especialmente no desempenho crescente do UMAP à medida que se aumentava o número de clusters. O UMAP e o t-SNE mostraram resultados equilibrados com dois componentes. Porém, com três componentes, o UMAP se mostrou melhor em todos os cenários. Além disso, o PCA foi a técnica mais rápida em todos os cenários avaliados, superando tanto o t-SNE quanto o UMAP em termos de tempo de execução.

Palavras-chave: Clusterização, K-Means, Redução de Dimensionalidade, PCA, t-SNE, UMAP, Censo da Educação Superior, Mineração de dados educacionais.

Abstract

The large amount of information collected in education censuses and national assessments demands efficient methods for knowledge extraction, allowing the identification of relevant patterns and trends. In this context, clustering stands out as a great technique to segment and interpret large volumes of educational data, with K-Means being one of the most widely used algorithms due to its simplicity and efficiency. However, when applied to high-dimensional datasets, its performance can be compromised, making it necessary to use dimensionality reduction techniques such as Principal Component Analysis (PCA), t-Distributed Stochastic Neighbor Embedding (t-SNE), and Uniform Manifold Approximation and Projection (UMAP). This work investigates the impact of these techniques on the quality of clusters generated by K-Means in a database composed of the merger of the 2022 Higher Education Census Microdata and the educational quality indicators Enade Score and CPC. The analysis is conducted using the silhouette index as an evaluation metric and comparing the execution time of each method. With two components, PCA outperformed t-SNE and UMAP in most tests. With three components, PCA performed better than t-SNE in all tests, but was on par with UMAP, outperforming it in five of the nine scenarios. It was also observed that the quantity of clusters had a relevant influence on the results, especially on UMAP's increasing performance as the number of clusters increased. UMAP and t-SNE showed balanced results with two components. However, with three components, UMAP performed better in all scenarios. Furthermore, PCA was the fastest technique in all evaluated scenarios, outperforming both t-SNE and UMAP in terms of execution time.

Keywords: Clustering, K-Means, Dimensionality Reduction, PCA, t-SNE, UMAP, Higher Education Census, Educational Data Mining.

Lista de Gráficos

Lista de tabelas

Sumário

1 Introdução

Com o avanço da tecnologia e a digitalização de processos, a quantidade de dados educacionais disponíveis cresce continuamente, fornecendo uma base rica para análises que podem contribuir para a formulação de políticas públicas e a melhoria do ensino. No entanto, a grande disponibilidade de dados não significa automaticamente melhores resultados, sendo essencial o desenvolvimento de técnicas para extrair conhecimento útil e interpretar essas informações de forma eficiente. No Brasil, apesar do aumento do interesse pela mineração de dados educacionais, ainda há a necessidade de pesquisas que explorem melhor esses dados para gerar insights que possam impactar positivamente o setor (FERREIRA; RODRIGUES; SOUZA, 2021).

Um dos principais desafios ao lidar com grandes volumes de dados é a chamada “maldição da dimensionalidade”, que pode prejudicar o desempenho computacional e impactar negativamente a qualidade dos resultados obtidos pelos algoritmos de Aprendizado de Máquina, que são comumente utilizados para a análise desses dados (GéRON, 2021). Para mitigar esse problema, são utilizados algoritmos de redução de dimensionalidade, que permitem transformar os dados originais em uma representação mais compacta, preservando suas principais características.

Diante desse cenário, este trabalho tem como objetivo analisar o impacto da aplicação de três métodos de redução de dimensionalidade no desempenho da clusterização com o algoritmo K-Means, um dos mais utilizados para análise exploratória de dados. Os métodos avaliados neste estudo são Principal Component Analysis (PCA), t-Distributed Stochastic Neighbor Embedding (t-SNE) e Uniform Manifold Approximation and Projection (UMAP). A investigação é realizada sobre uma base de dados composta pela junção dos Microdados do Censo da Educação Superior de 2022 e os indicadores de Conceito do Exame Nacional de Desempenho dos Estudantes (Enade) e Conceito Preliminar de Curso (CPC) dos anos 2019, 2021 e 2022, com avaliação baseada no índice

de silhueta e no tempo de execução dos algoritmos.

1.1 Objetivos

Este trabalho tem como objetivo analisar o impacto da aplicação de técnicas de redução de dimensionalidade na clusterização de dados utilizando o algoritmo K-means. Para isso, serão avaliadas as técnicas PCA, t-SNE e UMAP em termos de desempenho computacional e qualidade dos agrupamentos obtidos.

A fim de alcançar o objetivo geral desse trabalho, têm-se os seguintes objetivos específicos:

- Aplicar as técnicas PCA, t-SNE e UMAP em um conjunto de dados, reduzindo sua dimensionalidade antes da clusterização.
- Aplicar o algoritmo K-Means sobre os dados transformados por cada técnica de redução de dimensionalidade.
- Avaliar a qualidade dos agrupamentos gerados por meio do índice de silhueta.
- Comparar o tempo de execução das técnicas de redução de dimensionalidade, apresentando estatísticas descritivas como média, desvio padrão e intervalo de confiança de 95% para cada método.

2 Fundamentação teórica

A fundamentação teórica desta pesquisa apresenta os principais conceitos que sustentam o trabalho, abordando inicialmente a Mineração de Dados Educacionais, área voltada à extração de padrões e conhecimentos a partir de dados do contexto educacional. Em seguida, discute-se o Aprendizado de Máquina, com ênfase em seus diferentes tipos e aplicações. Na sequência, é detalhada a técnica de Clusterização, com destaque para o algoritmo K-Means, amplamente utilizado em análises exploratórias. Por fim, são explorados os métodos de Redução de Dimensionalidade, incluindo PCA, t-SNE e UMAP, que desempenham papel fundamental na simplificação dos dados e na melhoria da interpretação dos resultados.

2.1 Mineração de dados educacionais

Mineração de dados é o ato de extrair conhecimento de um determinado conjunto de dados, assim descobrindo padrões que podem ser usados para guiar futuras decisões (PAL; PAL, 2013). Outra definição utilizada também pode ser a seguinte: “Mineração de dados é um processo de análise multidisciplinar que se concentra em extrair e descobrir conhecimento útil a partir de dados e informação” (ALI et al., 2020, p. 2143). Assim, esse processo tem sido utilizado em diversas áreas, entre elas: vendas, bioinformática, etc (BAKER; ISOTANI; CARVALHO, 2011).

O contexto educacional não foge dessa tendência e, nesse cenário, surgiu um novo campo de pesquisa denominado “Mineração de Dados Educacionais” (do inglês, “Educational Data Mining”, ou EDM) (BAKER; ISOTANI; CARVALHO, 2011). “A EDM é definida como a área de pesquisa que tem como principal foco o desenvolvimento de métodos para explorar conjuntos de dados coletados em ambientes educacionais” (BAKER; ISOTANI; CARVALHO, 2011, p. 4). Segundo Baker, Isotani e Carvalho (2011, p. 11), a EDM “[...] possui grande

potencial para contribuir com a melhor compreensão dos processos de ensino, de aprendizagem e de motivação dos alunos tanto em ambientes individuais quanto em ambientes colaborativos de ensino".

Atualmente, as técnicas de mineração de dados têm sido amplamente aplicadas no contexto educacional, com diversos trabalhos desenvolvidos devido à enorme contribuição que produzem para analisar e melhorar a performance dos estudantes (PAL; PAL, 2013). Mas no Brasil, mesmo com o interesse na área tendo aumentado nos últimos anos, ainda são necessários avanços. É o que mostra a pesquisa de Ferreira, Rodrigues e Souza (2021). O levantamento feito em estudos publicados entre 2010 até junho de 2021 mostrou que mesmo com o interesse pelo campo tendo crescido nos últimos anos, ainda faltam pesquisas focadas na educação básica e Encceja (Exame Nacional para Certificação de Competências de Jovens e Adultos) (FERREIRA; RODRIGUES; SOUZA, 2021). Mas, ainda segundo os autores, o Brasil tem grande disponibilidade de dados, o que falta é uma melhora na utilização desses dados para, assim, ser possível a construção de políticas públicas educacionais (FERREIRA; RODRIGUES; SOUZA, 2021).

2.2 Aprendizado de Máquina

Existem várias definições para aprendizagem de máquina, mas a dada por Géron (2021) é uma simples, porém que define bem do que se trata: aprendizagem de máquina é a área da programação que possibilita os computadores aprenderem com os dados. Ainda segundo o mesmo autor, aplicar aprendizado de máquina é uma ótima opção para problemas complexos, para problemas que exigem muitas regras e ajustes finos, onde também é necessário adaptar-se a novos dados, análise de problemas complexos e grande quantidade de dados (GéRON, 2021).

2.2.1 Tipos de Aprendizado de Máquina

Existem alguns diferentes tipos de aprendizado de máquina. São 4 principais divisões de acordo com o tipo de aprendizado: supervisionado, não su-

pervisionado, semissupervisionado e aprendizado por reforço. No aprendizado supervisionado, os dados usados para treinar o algoritmo são rotulados; já no aprendizado não supervisionado ocorre o oposto, os dados não têm rótulos. Já o aprendizado semissupervisionado, como o nome já entrega, lida com dados rotulados e não rotulados. O aprendizado por reforço é diferente dos demais tipos, pois o treinamento é feito através de um sistema de recompensas; o agente (como é chamado nesse contexto) aprende ao executar ações e receber recompensas ou penalidades; cabe a esse agente escolher suas ações de forma a maximizar o número de recompensas (GéRON, 2021).

Outro tipo de definição usada para dividir os tipos de aprendizado é classificá-los de acordo com a sua capacidade de aprender de forma incremental ou não. O aprendizado em batch (por ciclo) é aquele em que o sistema não é capaz de aprender de forma incremental, ou seja, ele precisa ser treinado com todos os dados disponíveis e, para atualizar esse sistema com dados novos, é necessário treinar novamente com a junção dos dados antigos e dos novos. Já o aprendizado incremental, como o nome já diz, é aquele em que o sistema é capaz de aprender de forma incremental, ótima opção para sistemas que precisam se adaptar a mudanças rapidamente (GéRON, 2021).

Outra maneira de categorizar o aprendizado de máquina é dividi-lo de acordo com seu tipo de generalização: aprendizado baseado em instância ou aprendizado baseado em modelo. No aprendizado baseado em instância, o sistema aprende através da memorização dos exemplos e depois generaliza em novos dados. Já no aprendizado baseado em modelo, o sistema constrói um modelo através dos exemplos fornecidos e o utiliza para fazer predições (GéRON, 2021).

2.3 Clusterização

Clusterização é uma técnica de aprendizado de máquina não supervisionada onde seu objetivo é identificar instâncias semelhantes e agrupá-las em clusters. Essa técnica é ótima para diversos casos de uso, como: análise de dados, redução de dimensionalidade, sistemas de recomendação, segmenta-

ção de clientes, mecanismos de busca, segmentação de imagens, aprendizado semissupervisionado, entre outros (GéRON, 2021).

Como já falado, o objetivo da clusterização é dividir os itens em clusters. Porém, a definição de cluster depende do contexto e do algoritmo que está identificando esses clusters: alguns métodos buscam instâncias centradas em torno de um ponto específico, o centroide, enquanto outros buscam regiões contínuas de instâncias fortemente concentradas, entre outras várias formas (GéRON, 2021).

2.3.1 K-Means

O K-Means é um dos algoritmos mais utilizados quando se trata de clusterização. O objetivo desse algoritmo é identificar os clusters através dos seus centroides. Ele funciona da seguinte forma: começa atribuindo centroides de forma aleatória, depois rotula as instâncias, após isso os centroides são atualizados e esse processo se repete até que os centroides parem de se mover. Esse algoritmo não oscilará indefinidamente e chegará a uma conclusão em um número finito de etapas, isso acontece pois a distância quadrada média entre as instâncias e o centroide mais próximo só tem a possibilidade de diminuir a cada etapa. Porém, isso não é garantia de que o algoritmo irá convergir para a solução correta, esse resultado vai depender da inicialização correta do centroide. Para tentar mitigar esse problema, uma das estratégias adotadas pelo algoritmo implementado na biblioteca Scikit-Learn é rodar o algoritmo várias vezes com diferentes inicializações, o número de inicializações é controlado pelo parâmetro `n_init`. A biblioteca salva a melhor solução e essa melhor solução é encontrada através de uma métrica de desempenho: a inércia do modelo (que é a distância quadrada média entre cada instância e o centroide mais próximo) (GéRON, 2021).

Em termos de complexidade computacional, o K-Means é linear em relação ao número de instâncias, o número de clusters e o número de dimensões, porém isso só acontece se os dados têm uma estrutura de clusterização. Se isso não ocorrer, a complexidade pode aumentar exponencialmente com o número de instâncias (na pior das hipóteses), mas é algo que dificilmente acontece e,

assim, o K-Means costuma ser um dos algoritmos de clusterização mais rápidos (GéRON, 2021).

Um outro ponto importante sobre a clusterização através do algoritmo K-Means é a escolha do melhor valor para o número de clusters (normalmente chamado de K). Uma das formas de escolher o valor de K seria pelo método do cotovelo: que consiste em rodar o algoritmo com vários valores de K e plotar o gráfico com o valor de inércia para cada resultado. Para escolher o melhor valor de K, é necessário achar o ponto no gráfico (que tem formato de um braço) em que seja formado um "cotovelo": qualquer valor menor que esse K seria pior que escolhê-lo e qualquer valor maior não traria grandes melhorias (o valor da inércia diminui à medida que o valor de K cresce) (GéRON, 2021).

Porém a técnica de escolher o melhor valor de K através do método de cotovelo é um tanto rudimentar. Então, costuma-se usar uma outra abordagem (mesmo sendo mais custosa em termos computacionais): o índice de silhueta - que é o coeficiente médio de silhueta de todas as instâncias (GéRON, 2021).

O coeficiente de silhueta de uma instância é igual

$$a(b–a)/max(a,b),$$

> em que a é a distância média para as outras instâncias no mesmo cluster (ou seja, a distância média intracluster) e b é a média mais próxima da distância do cluster (ou seja, a distância média às instâncias do próximo cluster mais próximo, definida como aquela que minimiza b, excluindo o cluster da própria instância) (GéRON, 2021, p. 104).

Esse coeficiente varia de -1 a +1 (GéRON, 2021). Tendo o seguinte comportamento:

> Um coeficiente próximo a +1 significa que a instância está dentro de seu próprio cluster e distante de outros clusters, enquanto um coeficiente próximo a 0 significa que está próximo a uma fronteira de cluster e, por último, um coeficiente próximo a –1 significa que a

> instância pode ter sido atribuída ao cluster errado (GéRON, 2021, p. 104).

Outra forma de visualizar o índice de silhueta é plotando o seu gráfico: segundo (GéRON, 2021, p. 104) no diagrama de silhueta "[...] você plota o coeficiente de Silhouette de cada instância, classificado pelo cluster ao qual está atribuído e pelo valor do coeficiente". Os diagramas têm formato de faca, sua altura indica o número de instâncias que o cluster tem e a largura indica os coeficientes de silhueta classificados do cluster; quanto maior a largura, melhor. Já a linha pontilhada no gráfico é a indicação do coeficiente de silhueta médio (GéRON, 2021).

2.4 Redução de Dimensionalidade

Dados reais geralmente têm uma dimensionalidade alta (MAATEN et al., 2009) e com isso podem surgir problemas como a chamada "maldição da dimensionalidade". A chamada "maldição da dimensionalidade" caracteriza-se pelo crescimento exponencial do espaço de atributos e seus impactos, como o aumento do esforço computacional, desperdício de espaço e dificuldade de visualização (BELLMAN, 1957, apud VENKAT, 2018).

> Muitos problemas de aprendizado de máquina envolvem milhares ou até milhões de características para cada instância de treinamento. Todas essas características não somente tornam o treinamento extremamente lento, como também dificultam e muito encontrar uma boa solução, conforme veremos. Esse problema costuma ser chamado de a maldição da dimensionalidade (GéRON, 2021, p. 92).

Muitos algoritmos de clusterização sofrem com a maldição da dimensionalidade, pois não foram projetados para lidar com dados de alta dimensionalidade, o que compromete a eficácia dos agrupamentos (HINNEBURG; KEIM, 1999).

Para conseguir lidar bem com dados de alta dimensionalidade, é necessário que ela seja reduzida. Redução de dimensionalidade é o processo de transformar dados de alta dimensionalidade em uma representação que seja significativa, mas que tenha sua dimensionalidade reduzida (MAATEN et al., 2009). “Idealmente, a representação reduzida deve ter uma dimensionalidade que corresponda à dimensionalidade intrínseca dos dados” (MAATEN et al., 2009, p.1). É importante ressaltar que a redução de dimensionalidade causa perda de dados e deve ser utilizada com cautela, pois mesmo melhorando o tempo de treinamento do algoritmo que utiliza esses dados, pode ocasionar perda de desempenho do sistema (GéRON, 2021).

Existem dois tipos principais de abordagem de redução de dimensionalidade: projeção e aprendizado manifold. A projeção, como o nome já diz, é o ato de projetar as instâncias do conjunto de dados em um subespaço de menor dimensionalidade. Mas nem sempre esse tipo de abordagem é a melhor solução, já que o subespaço pode dar voltas (não ser linear). Para tipos de conjuntos assim, podemos usar o chamado “aprendizado manifold”(GéRON, 2021) . Mas antes de explicar o que é o aprendizado manifold, precisamos definir o que é um manifold: “[...] um manifold d-dimensional é uma parte de um espaço n-dimensional (sendo d < n) que se assemelha localmente a um hiperplano d-dimensional” (GéRON, 2021, p. 93). No aprendizado manifold, os algoritmos “funcionam modelando o manifold em que estão as instâncias de treinamento” (GéRON, 2021, p. 93). Essa abordagem se baseia na manifold assumption, que é descrita a seguir: “[...] também chamada de hipóteses múltiplas, que sustenta que a maioria dos conjuntos de dados de alta dimensão do mundo real fica próxima a um manifold de dimensões bem mais baixas.”

2.4.1 PCA

O algoritmo de redução de dimensionalidade mais popular é a Análise de Componentes Principais, (PCA, do inglês *Principal Component Analysis*) (GéRON, 2021). Trata-se de um método de projeção, onde o algoritmo “[...] identifica o hiperplano mais próximo dos dados e, em seguida, projeta os dados nele” (GéRON, 2021, p. 93).

> O PCA é um método popular usado em estudos exploratórios, com o objetivo de encontrar as direções de máxima variância em dados de alta dimensão e projetá-los em um novo subespaço para obter espaços de características de baixa dimensão, preservando a maior parte da variância. Os componentes principais do novo subespaço podem ser interpretados como as direções de máxima variância, tornando os novos eixos de características ortogonais entre si (HOZUMI et al., 2021, p.2, tradução nossa).

A Análise de Componentes Principais (PCA) é uma técnica estatística multivariada amplamente utilizada para reduzir a dimensionalidade de conjuntos de dados que contêm variáveis correlacionadas. Essa redução é feita por meio da transformação das variáveis originais em componentes ortogonais, os quais preservam a maior parte da variância presente nos dados (ABDI; WILLIAMS, 2010).

A decomposição da matriz de dados centrada $\mathbf{X} \in \mathbb{R}^{I \times J}$, por meio da decomposição em valores singulares, é representada por:

$$\mathbf{X} = \mathbf{P}\boldsymbol{\Delta}\mathbf{Q}^T \tag{2.1}$$

A variância total explicada por cada variável pode ser expressa pela inércia, dada pela soma dos quadrados dos elementos da j-ésima coluna:

$$\gamma_j^2 = \sum_i x_{i,j}^2 \tag{2.2}$$

A distância euclidiana entre uma observação i e o centro de gravidade g é definida por:

$$d_{i,g}^2 = \sum_j (x_{i,j} - g_j)^2 \tag{2.3}$$

No caso de dados centrados, essa equação se simplifica para:

$$d_{i,g}^2 = \sum_j x_{i,j}^2 \tag{2.4}$$

Os escores fatoriais, que representam as coordenadas dos dados no novo sistema de eixos principais, são obtidos por:

$$\mathbf{F} = \mathbf{P}\boldsymbol{\Delta} \tag{2.5}$$

Alternativamente, a relação entre escores fatoriais, autovetores e a matriz de dados pode ser expressa por:

$$\mathbf{F} = \mathbf{P}\boldsymbol{\Delta} = \mathbf{P}\boldsymbol{\Delta}\mathbf{Q}^\top\mathbf{Q} = \mathbf{X}\mathbf{Q} \tag{2.6}$$

A matriz original pode ser reconstituída a partir dos escores fatoriais por:

$$\mathbf{X} = \mathbf{F}\mathbf{Q}^\top \quad \text{com} \quad \mathbf{F}^\top\mathbf{F} = \boldsymbol{\Delta}^2 \quad \text{e} \quad \mathbf{Q}^\top\mathbf{Q} = \mathbf{I} \tag{2.7}$$

Para projetar uma nova observação $\mathbf{x}_{\text{sup}}$ no espaço de componentes principais, utiliza-se:

$$\mathbf{f}_{\text{sup}}^T = \mathbf{x}_{\text{sup}}^T\mathbf{Q} \tag{2.8}$$

A *contribuição* da observação i ao componente ℓ, que indica a importância relativa de i na definição do eixo ℓ, é dada por:

$$\text{ctr}_{i,\ell} = \frac{f_{i,\ell}^2}{\sum_i f_{i,\ell}^2} = \frac{f_{i,\ell}^2}{\lambda_\ell} \tag{2.9}$$

em que λ_ℓ é o autovalor associado ao componente ℓ.

Já o *cosseno ao quadrado* (ou qualidade de representação) de uma observação i no componente ℓ é expresso por:

$$\cos^2_{i,\ell} = \frac{f^2_{i,\ell}}{\sum\limits_{\ell} f^2_{i,\ell}} = \frac{f^2_{i,\ell}}{d^2_{i,g}} \tag{2.10}$$

em que $d^2_{i,g}$ representa a distância ao quadrado da observação i ao centro.

A qualidade do ajuste do modelo pode ser avaliada pela soma residual dos quadrados (modelo de efeito fixo), expressa como:

$$\text{RESS}_M = \|\mathbf{X} - \hat{\mathbf{X}}^{[M]}\|^2 = \text{trace}(\mathbf{E}^T\mathbf{E}) = I - \sum_{\ell=1}^{M} \lambda_\ell \tag{2.11}$$

No modelo de efeito aleatório, a métrica correspondente é:

$$\text{PRESS}_M = \|\mathbf{X} - \tilde{\mathbf{X}}^{[M]}\|^2 \tag{2.12}$$

Entre os critérios utilizados para a escolha do número de componentes, destacam-se o Q^2_ℓ:

$$Q^2_\ell = 1 - \frac{\text{PRESS}_\ell}{\text{RESS}_{\ell-1}} \tag{2.13}$$

e o teste de significância W_ℓ:

$$W_\ell = \frac{\text{PRESS}_{\ell-1} - \text{PRESS}_\ell}{\text{PRESS}_\ell} \cdot \frac{df_{\text{residual},\ell}}{df_\ell} \tag{2.14}$$

Um dos problemas do PCA é que pode haver perda de informações se o número de componentes principais selecionado seja inadequado. Outro problema é que, por ser um algoritmo linear, ele não se comporta bem com dados com características com relações não lineares (HOZUMI et al., 2021).

2.4.2 t-SNE

O t-SNE (do inglês *t-distributed Stochastic Neighbor Embedding*) é uma técnica de redução de dimensionalidade proposta por Maaten e Hinton (2008),

cujo objetivo é preservar as relações de vizinhança locais ao projetar dados de alta dimensão em um espaço de menor dimensão. É um método de redução de dimensionalidade "[...] não linear que consegue preservar a estrutura local e global dos dados" (HOZUMI et al., 2021, p.2, tradução nossa). Existem duas etapas principais nesse método:

> Primeiro, ele encontra uma distribuição de probabilidade do conjunto de dados de alta dimensão, onde pontos de dados semelhantes recebem maior probabilidade. Segundo, ele encontra uma distribuição de probabilidade semelhante no espaço de dimensão inferior, e a diferença entre as duas distribuições é minimizada (HOZUMI et al., 2021, p.2, tradução nossa).

A técnica t-SNE foi proposta por Maaten e Hinton (2008) como um método não linear de redução de dimensionalidade voltado à preservação das relações de vizinhança locais. O processo inicia-se com a definição de uma distribuição de probabilidade condicional que quantifica a similaridade entre pares de pontos no espaço original.

A probabilidade condicional de x_j ser um vizinho de x_i é dada por:

$$p_{j|i} = \frac{\exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma_i^2}\right)}{\sum_{k \neq i} \exp\left(-\frac{\|x_i - x_k\|^2}{2\sigma_i^2}\right)} \tag{2.15}$$

onde:

- $\|x_i - x_j\|^2$ representa a distância euclidiana ao quadrado entre os pontos x_i e x_j,
- σ_i é o desvio padrão da distribuição centrada em x_i, ajustado de forma adaptativa.

O parâmetro σ_i é determinado a partir de uma perplexidade desejada, que define uma medida da efetiva quantidade de vizinhos considerados. A perplexidade da distribuição P_i é definida como:

$$\text{Perp}(P_i) = 2^{H(P_i)} \tag{2.16}$$

sendo que $H(P_i)$ representa a entropia de Shannon, dada por:

$$H(P_i) = -\sum_j p_{j|i} \log_2 p_{j|i} \tag{2.17}$$

Com base nas probabilidades condicionais, define-se uma medida de similaridade simétrica entre os pontos x_i e x_j:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N} \tag{2.18}$$

em que N é o número total de amostras. Essa simetrização assegura que $p_{ij} = p_{ji}$, e que a matriz P seja uma distribuição de probabilidade conjunta válida.

No espaço reduzido, o algoritmo associa a cada ponto x_i uma representação y_i, e define uma nova distribuição de similaridade, Q, com pesos obtidos por:

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l}(1 + \|y_k - y_l\|^2)^{-1}} \tag{2.19}$$

A escolha de uma distribuição de cauda pesada (distribuição de Student com um grau de liberdade) no espaço reduzido evita o problema de ”crowding”, comum em técnicas de redução de dimensionalidade.

A qualidade do mapeamento é avaliada por meio da divergência de Kullback-Leibler (KL) entre as distribuições P e Q, definida como:

$$C = KL(P\|Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{2.20}$$

A minimização dessa função de custo é feita via descida do gradiente. O gradiente em relação à posição y_i no espaço reduzido é dado por:

$$\frac{\partial C}{\partial y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j)(1 + \|y_i - y_j\|^2)^{-1} \tag{2.21}$$

Esse gradiente indica a direção na qual os pontos devem ser movidos para reduzir a divergência entre as distribuições, aproximando a estrutura do espaço reduzido da do espaço original.

Por calcular a probabilidade computacional para cada par de amostras e ter hiperparâmetros que podem ser complicados de ajustar, o t-SNE tem como desvantagem sua complexidade computacional (HOZUMI et al., 2021).

2.4.3 UMAP

UMAP (do inglês *Uniform Manifold Approximation and Projection*) é um algoritmo de redução de dimensionalidade que se baseia "[...] em técnicas de aprendizado de variedades e ideias de análise de dados topológicos" (MCINNES, 2024, tradução nossa). É proposto por McInnes, Healy e Melville (2018) e fundamentado em princípios de geometria diferencial e teoria das categorias. Ele parte da hipótese de que os dados estão uniformemente distribuídos sobre uma variedade riemanniana (MCINNES; HEALY; MELVILLE, 2018).

O algoritmo UMAP é dividido em duas fases: em sua primeira fase, o algoritmo constrói uma representação topológica difusa (MCINNES, 2024), já sua segunda fase é focada em "[...] otimizar a representação de baixa dimensão para ter uma representação topológica difusa o mais próxima possível conforme medido pela entropia cruzada (MCINNES, 2024, tradução nossa)". O funcionamento do algoritmo é explicado de forma mais detalhada a seguir.

O algoritmo UMAP, proposto por McInnes, Healy e Melville (2018), é uma técnica de redução de dimensionalidade baseada em suposições da geometria riemanniana e da teoria das categorias. Ele busca preservar tanto a estrutura local quanto a global dos dados ao construir grafos de vizinhança e otimizá-los em um espaço de menor dimensão.

A primeira etapa do UMAP consiste em estimar a estrutura de vizinhança local no espaço original. Para isso, define-se, para cada ponto x_i, a menor dis-

tância positiva até seus k vizinhos mais próximos:

$$\rho_i = \min\{d(x_i, x_{ij}) \mid 1 \leq j \leq k,\ d(x_i, x_{ij}) > 0\} \tag{2.22}$$

Esse valor ρ_i representa a distância mínima significativa a partir de x_i, e serve como limite inferior para a aplicação da função exponencial.

Em seguida, ajusta-se um parâmetro de suavização σ_i específico para cada ponto, de forma que a soma das similaridades com seus vizinhos iguale $\log_2(k)$, garantindo uma distribuição local com perplexidade aproximadamente constante:

$$\sum_{j=1}^{k} \exp\left(-\frac{\max(0,\ d(x_i, x_{ij}) - \rho_i)}{\sigma_i}\right) = \log_2(k) \tag{2.23}$$

Com os parâmetros ρ_i e σ_i, calcula-se a similaridade direcionada entre os pontos x_i e x_j com a fórmula:

$$v_{j|i} = \exp\left(-\frac{\max(0,\ d(x_i, x_j) - \rho_i)}{\sigma_i}\right) \tag{2.24}$$

Nesta equação:

- $d(x_i, x_j)$ é a distância euclidiana entre x_i e x_j,
- ρ_i funciona como um deslocamento mínimo,
- σ_i ajusta a ”largura”da distribuição.

Para obter uma medida simétrica de afinidade entre os pares de pontos, utiliza-se a chamada média fuzzy:

$$w_{ij} = v_{j|i} + v_{i|j} - v_{j|i} \cdot v_{i|j} \tag{2.25}$$

Esse cálculo assegura que $w_{ij} = w_{ji}$, aproximando uma distribuição de probabilidade conjunta com base em contribuições direcionais.

No espaço de dimensão reduzida, UMAP define uma nova função de similaridade $\Phi(y_i, y_j)$ entre os pontos projetados y_i e y_j, dada por:

$$\Phi(y_i, y_j) = \left(1 + a\|y_i - y_j\|^{2b}\right)^{-1} \tag{2.26}$$

onde os parâmetros a e b são ajustados empiricamente para que a função se aproxime de uma distribuição de cauda longa, semelhante à utilizada no t-SNE.

Por fim, a função de custo do UMAP é formulada como uma divergência cruzada binária entre as distribuições w_{ij} (no espaço original) e $\Phi(y_i, y_j)$ (no espaço reduzido). A função a ser minimizada é:

$$C = \sum_{(i,j)\in S} w_{ij} \log\left(\frac{w_{ij}}{\Phi(y_i, y_j)}\right) + (1 - w_{ij}) \log\left(\frac{1 - w_{ij}}{1 - \Phi(y_i, y_j)}\right) \tag{2.27}$$

Essa divergência promove a proximidade dos pares com alta similaridade no espaço original e afasta aqueles com baixa afinidade, assegurando uma representação que preserva a estrutura da vizinhança dos dados.

Segundo descrito na documentação do próprio UMAP, o algoritmo resultante das etapas descritas anteriormente é rápido e escalável, como também é baseado em uma teoria matemática sólida (MCINNES, 2024).

As técnicas de redução de dimensionalidade PCA, t-SNE e UMAP apresentam abordagens distintas, mas complementares, sendo amplamente utilizadas na visualização e pré-processamento de dados de alta dimensão. O PCA, de natureza linear, transforma os dados originais em combinações ortogonais que explicam a variância total do conjunto, sendo mais indicado quando os padrões nos dados possuem estrutura linear ou quando se deseja interpretabilidade (ABDI; WILLIAMS, 2010). Por outro lado, o t-SNE adota uma estratégia não linear e probabilística que busca preservar relações locais, sendo particularmente eficaz na visualização de agrupamentos, ainda que com maior custo computacional e dificuldade de generalização para novos dados (MAATEN; HINTON, 2008). O UMAP, também não linear, baseia-se em fundamentos topoló-

gicos e diferencia-se por preservar tanto estruturas locais quanto globais, apresentando maior escalabilidade em comparação ao t-SNE (MCINNES; HEALY; MELVILLE, 2018). Embora compartilhem o objetivo comum de representar dados complexos em espaços reduzidos, essas técnicas variam quanto ao princípio matemático, fidelidade estrutural e aplicabilidade prática, sendo a escolha dependente do tipo de dado, objetivo da análise e restrições computacionais.

3 Trabalhos relacionados

A seção de Trabalhos Relacionados reúne estudos que oferecem subsídios teóricos e práticos para a presente pesquisa. Inicialmente, são apresentados trabalhos que exploram diferentes técnicas de redução de dimensionalidade, destacando suas aplicações e limitações. Em seguida, são discutidas pesquisas voltadas à clusterização, com foco principalmente no algoritmo K-Means, que foi o algoritmo escolhido para este trabalho. Por fim, são analisados estudos que combinam clusterização e redução de dimensionalidade, destacando de que forma os algoritmos de redução de dimensionalidade influenciam diretamente os resultados obtidos na clusterização.

3.1 Redução de dimensionalidade

O trabalho de Maaten et al. (2009) foca em uma revisão comparativa de diferentes tipos de métodos de redução de dimensionalidade. O objetivo dessa revisão foi comparar o algoritmo linear PCA com diversos algoritmos não lineares (12 no total). Esse trabalho foi motivado tendo em vista que algoritmos não lineares deveriam se sair melhor que algoritmos lineares em dados reais, se levado em consideração que é comum dados reais serem não lineares. Outros estudos mostram que isso é verdade em dados artificiais, porém há uma escassez de estudos com análise desse tipo em dados do mundo real. Os autores chegaram à conclusão de que a maioria das técnicas não lineares não se saíram melhor que o tradicional método linear PCA em datasets formados por dados reais.

Já o trabalho de Reddy et al. (2020) foca na utilização de redução de dimensionalidade, mais especificamente das técnicas PCA e LDA (Análise Discriminante Linear), a fim de analisar se essas técnicas melhoram ou não os resultados de alguns algoritmos de aprendizado de máquina. Foram utilizados 3 datasets da área de saúde nos seguintes algoritmos de classificação: árvore

de decisão, Naive Bayes, floresta aleatória e SVM (Support Vector Machine). Os resultados mostraram que a performance dos algoritmos de classificação, quando utilizados após redução de dimensionalidade, foi melhor com o PCA do que com o LDA. Porém, com datasets de tamanhos menores, utilizar as técnicas de redução de dimensionalidade mostrou um impacto negativo na performance dos classificadores, entre outras observações.

A pesquisa de Devassy e George (2020) focou em analisar o uso do método t-SNE para reduzir a dimensionalidade de dados espectrais de tinta. Os autores compararam o t-SNE com o PCA e chegaram a alguns resultados interessantes: o t-SNE performou melhor que o PCA em todos os testes, e os autores também informaram que a redução de dimensionalidade não linear funciona melhor em dados espectrais de tinta do que métodos lineares; porém, essa última afirmação precisaria ser estendida para mais métodos lineares e não lineares além dos que foram utilizados nesta pesquisa.

3.2 Clusterização

A clusterização é uma técnica bastante popular na área de mineração de dados e existem vários algoritmos com essa finalidade. O que faz com que muitos trabalhos tenham como objetivo desenvolver e avaliar maneiras de melhorar os resultados obtidos com esses algoritmos.

Um dos algoritmos mais populares de clusterização é o K-means e, sabendo disso, o trabalho de Saputra, Saputra e Oswari (2020) teve como objetivo avaliar o impacto da utilização de métricas de distância diferentes na determinação do número de clusters ideal (geralmente chamado de K). O estudo foca em dois métodos: o método de cotovelo (mede a coesão do cluster) e o método de silhueta (utiliza um coeficiente próprio que combina separação e coesão) que são técnicas bastante usadas para determinar o valor de K. Utilizando os dois métodos citados, o artigo focou em aplicar diferentes métricas de distância para avaliar se os resultados de K obtidos mudavam de acordo com a métrica escolhida. As métricas utilizadas foram: distância de Manhattan, distância Euclidiana e distância de Minkowski. Os autores concluem que a mudança de métricas não

foi impactante nos resultados obtidos. De 37 datasets utilizados nos testes, apenas um dataset teve um K diferente obtido ao se mudar a métrica de distância no método do cotovelo. Já utilizando o método da silhueta, foram 2 datasets. Todos esses 3 K's diferentes obtidos foram resultado da utilização da métrica de distância Manhattan. Outra conclusão a que os autores chegaram a partir de sua pesquisa foi que ambos os métodos (método do cotovelo e método da silhueta) são capazes de determinar o valor de K, porém o método do cotovelo apresenta algumas dificuldades na visualização desse número e, em alguns momentos (em 3 situações) não conseguiu obter o valor do número de clusters.

No trabalho de Yuan e Yang (2019) o foco foi analisar métodos de determinação ideal do valor de K (número de clusters). O trabalho utilizou o dataset Iris em seus testes e testou um total de 4 métodos: método do cotovelo, Gap Statistic, Coeficiente de Silhueta e Canopy. Os autores chegaram à conclusão de que, para datasets pequenos, todos os quatro métodos analisados são viáveis (todos eles obtiveram o resultado de K sendo 2). Além disso, também chegaram à conclusão de que, para datasets maiores e mais complexos, o método Canopy seria uma melhor opção em relação aos outros.

Já o trabalho de Ogbuabor e Ugwoke (2018) focou na aplicação de clusterização em dados da área de saúde e em avaliar dois diferentes algoritmos: K-means e DBSCAN. Segundo os autores, a escolha desses algoritmos se deu pela diferença entre os dois em relação à seleção do número de clusters: o K-means necessita que essa escolha seja feita previamente à execução do algoritmo, já o DBSCAN não precisa, sendo o próprio algoritmo quem encontra o número de clusters de acordo com o dataset. Outro ponto interessante sobre a escolha desses algoritmos é que, enquanto o K-means utiliza uma abordagem baseada no particionamento de clusters, o DBSCAN é um exemplo de método baseado em densidade. Os algoritmos foram avaliados utilizando a análise de silhueta, porém foram testadas diferentes métricas de distância. A melhor métrica para o K-means foi a Euclidiana, já para o DBSCAN foi a Manhattan. Analisando os resultados obtidos, os autores chegaram à conclusão de que obter os melhores clusters depende dos parâmetros passados para cada algoritmo.Os autores também apontaram que ambos os algoritmos analisados têm forte coesão intra-

cluster e separação entre clusters, mas em tempo de execução e precisão de agrupamento, o algoritmo K-means obteve melhores resultados.

Ainda sobre a importância da escolha do número de clusters para utilização do algoritmo K-means, o trabalho de Wella et al. (2023) procurou utilizar diferentes métricas para assim definir o valor que seria utilizado como K ideal para sua análise da qualidade dos serviços oferecidos por concessionárias. As métricas utilizadas foram o método do cotovelo, método da silhueta, Índice Davies–Bouldin (DBI) e o Índice Calinski-Harabasz. Ao analisar os resultados dessas métricas, os autores concluíram que o melhor valor de clusters seria 3, e assim conseguiram dividir os dados em: boa performance (good), muito boa performance (very good) e performance ruim (not good).

Já o trabalho de Khan et al. (2024) focou em melhorar o método de estatística de Gap para obter um melhor número de clusters para utilizar junto do método K-means. Com o método modificado, os autores testaram em 3 diferentes datasets contra diferentes outros tipos de método de escolha de número de clusters: como o método tradicional de estatística de Gap, método do cotovelo, método da silhueta, Índice Davies–Bouldin (DBI) e o Índice Calinski-Harabasz. Nos diferentes testes, o método de estatística de Gap modificado obteve resultados melhores tanto em eficiência como em acurácia quando comparado aos outros métodos citados.

A clusterização também pode ser utilizada para o aumento do volume de amostras, como acontece no trabalho de RODRIGUES (2024), onde tinha o objetivo de prever a evasão de cursos de graduação com técnicas de aprendizado de máquina supervisionado e não supervisionado. Isso foi feito através da "execução de agrupamentos de cursos de graduação com o objetivo de gerar, de forma não supervisionada, grupos coesos de cursos com perfis semelhantes para o aumento do volume de mostras de estudantes evadidos" (RODRIGUES, 2024, p. 59). Os resultados do trabalho mostram que é possível agrupar estudantes de cursos de graduação com características semelhantes para prever a evasão, visto que modelos binários obtiveram métricas superiores a 80% — chegando a 90% em alguns casos — em termos de acurácia, precisão e *recall*, enquanto os modelos multiclasse apresentaram desempenho próximo de 75%

nessas mesmas métricas.

3.3 Clusterização aliada à redução de dimensionalidade

Já a pesquisa de Allaoui, Kherfi e Cheriet (2020) teve como objetivo avaliar se a utilização do método de redução de dimensionalidade UMAP causaria impactos positivos sendo utilizado como preparação dos dados antes da aplicação de algoritmos de clusterização. Foram testados 5 datasets de imagens e foram avaliados os seguintes algoritmos de clusterização: K-means, Aglomerativo, HDBSCAN e GMM. As métricas utilizadas foram acurácia e Informação Mútua Normalizada (Normalized Mutual Information). Ao comparar os resultados das clusterizações antes de usar o UMAP e após usar o método de redução de dimensionalidade, os autores perceberam melhora em ambas as métricas utilizadas, como também uma melhora no tempo de execução dos algoritmos de clusterização.

O artigo de Hozumi et al. (2021, p. 2) tem como objetivo "[...] explorar métodos computacionais eficientes para a análise filogenética do SARS-CoV-2 em um grande volume de sequências do genoma do SARS-CoV-2". E para isso os autores focaram em testar métodos de redução de dimensionalidade (PCA, t-SNE e UMAP) antes da aplicação do algoritmo de clusterização K-means. Foram comparados os resultados antes da utilização dos métodos de redução de dimensionalidade e após a utilização desses métodos. Para ser possível avaliar a precisão do K-means, os autores reformularam os problemas de classificação supervisionada com rótulos em problemas de agrupamento K-means. Os autores chegaram à conclusão de que "[...] o UMAP é o algoritmo mais eficiente, robusto, confiável e preciso" (HOZUMI et al., 2021, p.2).

O trabalho de Baligodugula e Amsaad (2025) teve como objetivo comparar diversos algoritmos de clusterização (K-means, DBSCAN e Agrupamento Espectral) em conjuntos de dados de alta dimensionalidade, inclusive o impacto de técnicas de redução de dimensionalidade no desempenho da clusterização. Para isso, foram utilizadas diversas métricas: internas (Coeficiente de Silhueta, Índice de Davies-Bouldin e Índice Calinski-Harabasz), externas (Índice Rand

Ajustado e Informação Mútua Normalizada) e métricas computacionais (tempo de treino e uso de memória). Os resultados desse estudo mostraram que a redução de dimensionalidade melhorou significativamente a performance de todos os algoritmos (independente do conjunto de dados utilizado). E sobre quais das técnicas de redução de dimensionalidade se apresenta como a melhor: “O UMAP supera consistentemente outras técnicas, provavelmente devido à sua capacidade de preservar tanto a estrutura local quanto a global” (BALIGODUGULA; AMSAAD, 2025).

4 Ferramentas e Método

A seção de Ferramentas e Método descreve os recursos utilizados e o processo seguido para o desenvolvimento da pesquisa. Inicialmente, são apresentadas a linguagem de programação e as bibliotecas empregadas na implementação dos experimentos. Em seguida, detalham-se os dados analisados, bem como as etapas de tratamento, que englobam a seleção, o pré-processamento e a transformação dos dados. Posteriormente, são discutidos os métodos de análise aplicados para alcançar os objetivos propostos. Por fim, são expostas as principais limitações enfrentadas ao longo do trabalho, de modo a contextualizar os resultados obtidos.

4.1 Linguagem e bibliotecas

A linguagem de programação utilizada no trabalho foi Python na versão 3.12.3. Para manipulação e tratamento dos dados, foi utilizada a biblioteca Pandas na versão 2.2.2 e NumPy na versão 1.26.4. Para visualização dos dados, foram utilizadas as bibliotecas Matplotlib na versão 3.9.1 e Yellowbrick na versão 1.5. A Scikit-learn na versão 1.5.1 foi utilizada para manipulação e tratamento dos dados, como também para a utilização dos algoritmos de redução de dimensionalidade PCA e t-SNE. A biblioteca UMAP na versão 0.5.7 foi usada para redução de dimensionalidade utilizando o algoritmo UMAP.

4.2 Dados

Para a análise dos métodos de redução de dimensionalidade escolhidos - PCA, T-SNE e UMAP - foram utilizados dados relacionados à Educação Superior do Brasil fornecidos pelo INEP. Foram utilizados os Microdados do Censo de Educação Superior do ano de 2022, como também os indicadores de qualidade da educação superior Conceito Enade e CPC dos anos 2019, 2021 e 2022. A escolha dos anos dessas bases de dados deveu-se ao fato de serem

os mais recentes disponíveis no início do projeto. “O Conceito Enade é um indicador de qualidade que avalia os cursos por intermédio dos desempenhos dos estudantes no Enade. Seu cálculo e sua divulgação ocorrem anualmente para os cursos com pelo menos dois estudantes concluintes participantes do exame” (Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP), 2020a). Já o CPC é definido da seguinte forma:

> O CPC é um indicador de qualidade que avalia os cursos de graduação. Seu cálculo e sua divulgação ocorrem no ano seguinte ao da realização do Enade, com base na avaliação de desempenho de estudantes, no valor agregado pelo processo formativo e em insumos referentes às condições de oferta – corpo docente, infraestrutura e recursos didático-pedagógicos –, conforme metodologia aprovada pela Comissão Nacional de Avaliação da Educação Superior (Conaes). (Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP), 2020b)

Os Microdados do Censo de Educação Superior do ano de 2022 são compostos por características dos cursos e de suas respectivas instituições de ensino. Algumas das características utilizadas nesse estudo foram: nome da área geral, conforme adaptação da Classificação Internacional Normalizada da Educação Cine/Unesco, tipo de grau acadêmico, quantidade de vagas (diurnas, noturnas, remanescentes, etc.), quantidade de inscritos (diurno, noturno, etc.), quantidade de matrículas (femininas, masculinas, parda, amarela, preta, etc.), quantidade de alunos que recebem apoio social, quantidade de concluintes (que recebem algum tipo de apoio social, que terminaram o ensino médio em escola privada, que terminaram o ensino médio em escola pública, etc.), entre outras.

Em relação aos indicadores Conceito Enade e CPC, foram utilizados os campos de código de curso (para unir as informações) e os resultados de cada indicador. Os metadados correspondentes aos dados utilizados encontram-se no Apêndice A.

4.3 Tratamento dos dados

Para a leitura, manipulação e tratamento dos dados, foram utilizadas as seguintes bibliotecas: pandas, numpy, unidecode, re e sklearn.

4.3.1 Seleção dos dados

Os dados utilizados neste estudo foram extraídos de três fontes principais:

- Microdados do Censo da Educação Superior de 2022: contém informações sobre cursos e instituições de ensino superior, incluindo código do curso, nome, área, unidade federativa, número de ingressantes, matrículas e concluintes, entre outras variáveis.
- Indicadores CPC (Conceito Preliminar de Curso): foram utilizados os dados dos anos de 2019, 2021 e 2022, sendo que, quando um mesmo curso possuía notas em 2019 e 2022, optou-se por considerar apenas a nota de 2022.
- Indicadores Conceito Enade: dados também referentes aos anos de 2019, 2021 e 2022, aplicando a mesma lógica de escolha dos valores mais recentes quando havia duplicidade.

Além disso, os dados foram filtrados para incluir apenas cursos da rede pública e na modalidade presencial, excluindo instituições privadas e cursos a distância/especiais.

4.3.2 Pré-Processamento

Nesta etapa, os dados passaram por limpeza, padronização e fusão. O tratamento inicial envolveu:

- Remoção de dados nulos ou inconsistentes.
- Ajuste de nomenclaturas para padronização.

- Fusão dos três conjuntos de dados em um único dataset, utilizando o código do curso como chave de junção.

4.3.3 Transformação dos dados

Após a unificação dos datasets, foram aplicadas transformações para garantir a qualidade da análise:

- Os outliers inferiores e superiores foram identificados através do intervalo interquartil (IQR), sendo substituídos pelos respectivos limites inferior ou superior da amostra. Esse tratamento foi necessário para evitar a formação de clusters apenas com os outliers, uma vez que o objetivo da clusterização nesse trabalho não é identificar outliers, mas permitir a comparação de diferentes entidades com características semelhantes.
- Remoção de colunas com variância zero.
- Padronização dos dados numéricos utilizando o StandardScaler do sklearn, garantindo que todas as variáveis estivessem na mesma escala.
- Codificação das variáveis categóricas por meio do OneHotEncoder do sklearn.
- Análise de correlação e redundância entre as colunas, resultando na remoção de variáveis com informações duplicadas, como:
  - Colunas segmentadas por idade, pois as mesmas informações já estavam disponíveis em outras variáveis mais gerais.
  - Colunas altamente correlacionadas, como, por exemplo, a quantidade total de concluintes, que já era representada por colunas específicas de gênero.

4.4 Métodos de análise

De modo geral, para decidir a melhor quantidade de clusters para os testes de redução de dimensionalidade, foi utilizado o método do cotovelo. Já para

a avaliação da clusterização após a redução de dimensionalidade, foi utilizado o índice de silhueta.

Além disso, todos os algoritmos de redução de dimensionalidade foram utilizados com seus parâmetros padrão, exceto pelo número de componentes, conforme implementações disponíveis nas bibliotecas utilizadas. Dessa forma, não foram realizadas etapas adicionais de configuração ou ajuste específico para nenhum dos métodos. Essa abordagem buscou manter a uniformidade no processo de comparação e avaliar o desempenho de cada técnica em suas configurações mais comuns, garantindo uma análise objetiva dos resultados obtidos na clusterização com o K-Means.

4.5 Limitações

Algumas limitações do trabalho são as seguintes:

- Base de dados utilizada: todos os resultados e análises descritas nesse trabalho se referem a um único conjunto de dados, ou seja, em outros datasets os resultados podem ser completamente diferentes.
- Hiperparâmetros: a escolha dos hiperparâmetros pode afetar os resultados, então opções diferentes podem afetar a comparação entre as técnicas.
- Escolha do algoritmo de clusterização: o presente trabalho foca somente no K-Means, outros algoritmos poderiam ser testados para avaliar se os resultados se mantêm consistentes em diferentes abordagens.
- Uso do índice de silhueta como única métrica de avaliação: A qualidade dos agrupamentos foi avaliada apenas pelo índice de silhueta. Outras métricas poderiam fornecer uma visão mais completa da qualidade dos clusters formados.
- Limitações computacionais: O tempo de execução dos algoritmos foi medido em um ambiente específico (hardware utilizado no estudo). Em ou-

tras configurações de hardware, o desempenho relativo entre os métodos pode ser diferente.

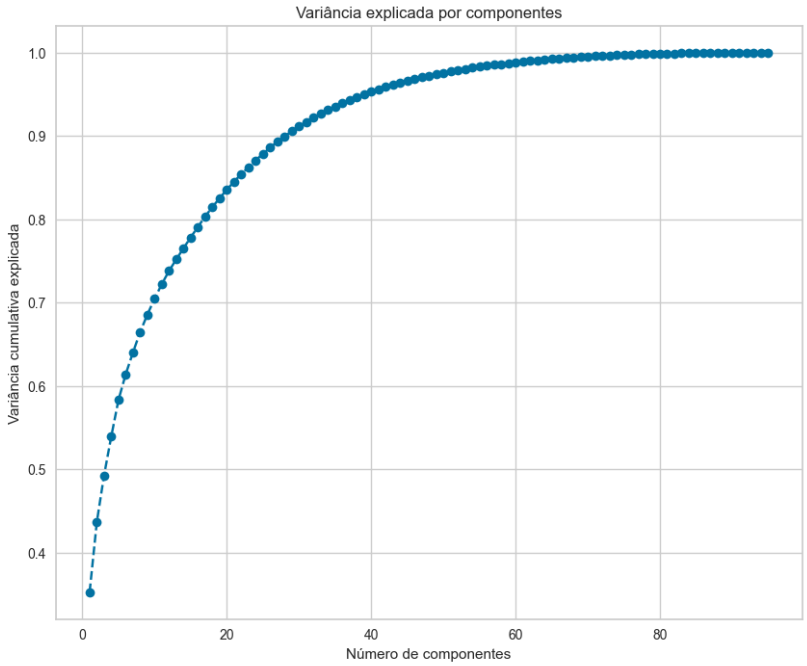
5 Resultados

A seção de Resultados apresenta as análises realizadas a partir da aplicação dos métodos de redução de dimensionalidade e da clusterização. Inicialmente, são discutidos os desempenhos individuais do PCA, t-SNE e UMAP, ressaltando suas particularidades. Em seguida, são realizadas comparações entre os métodos, considerando a qualidade dos agrupamentos obtidos e as diferenças observadas. Também é conduzida uma análise de tempo, que evidencia a eficiência computacional de cada técnica. Por fim, são discutidos os resultados alcançados, destacando implicações e pontos relevantes para a interpretação dos achados.

5.1 PCA

Na análise individual da redução de dimensionalidade com o PCA, a variância cumulativa explicada foi examinada primeiro.

Gráfico 1 – Variância cumulativa explicada.

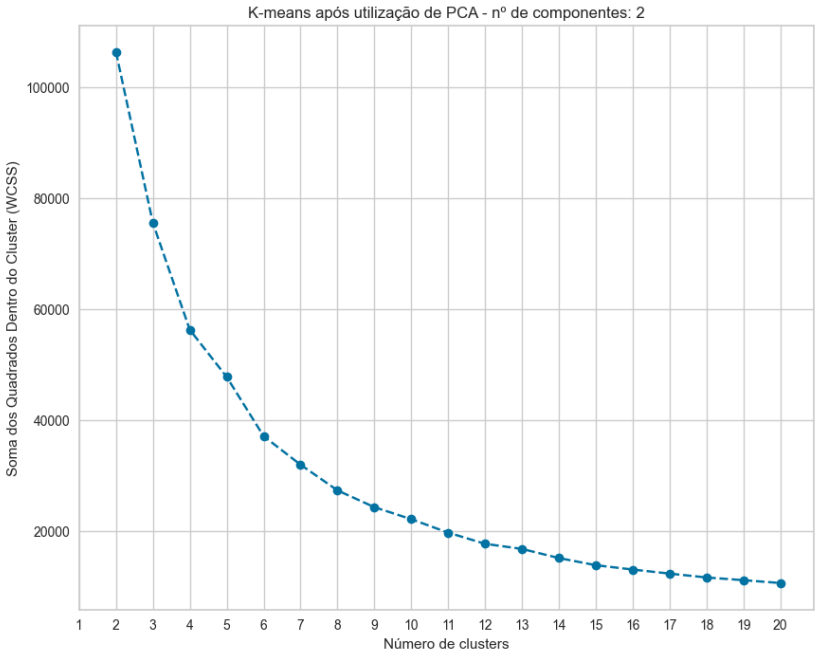


Fonte: autoria própria (2025).

O Gráfico 1 mostra o quanto da variância é preservada (valores no eixo y) em relação à quantidade de componentes (eixo x). Considera-se uma boa variância aquela de, no mínimo, 80%, o que indica que seriam necessários pelo menos 20 componentes ao se utilizar o PCA. Com o objetivo de garantir a comparabilidade dos resultados, foram considerados 2 e 3 componentes, mantendo a mesma configuração adotada nos demais métodos de redução de dimensionalidade.

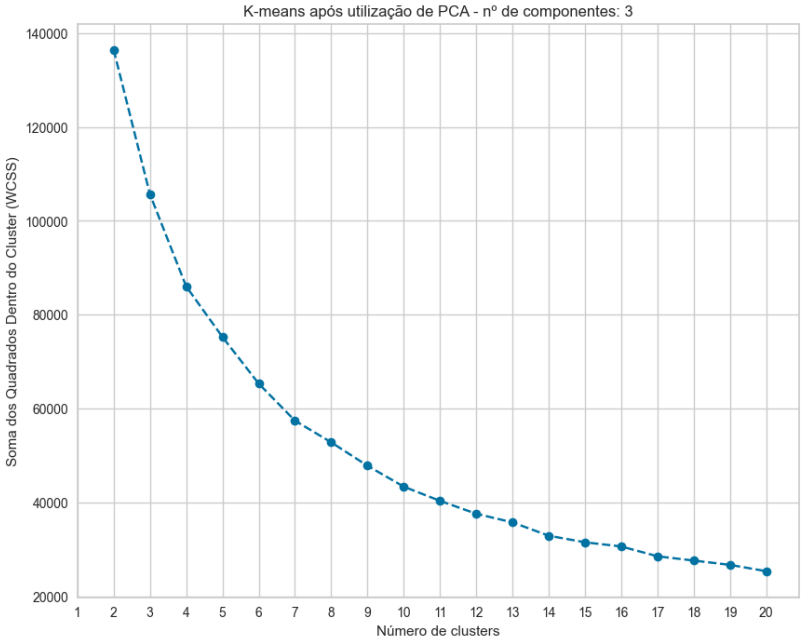
Em seguida, foram realizadas análises utilizando o método do cotovelo. A redução de dimensionalidade com PCA foi aplicada para 2 e 3 componentes e, para cada quantidade de componentes, utilizou-se o método do cotovelo, com o número de clusters variando de 2 a 20. O resultado encontra-se no Gráfico 2 e Gráfico 3. É importante destacar que o método do cotovelo foi utilizado apenas como uma estimativa da quantidade de clusters, não sendo seguido estritamente.

Gráfico 2 – Método do cotovelo para 2 componentes



Fonte: autoria própria (2025).

Gráfico 3 – Método do cotovelo para 3 componentes

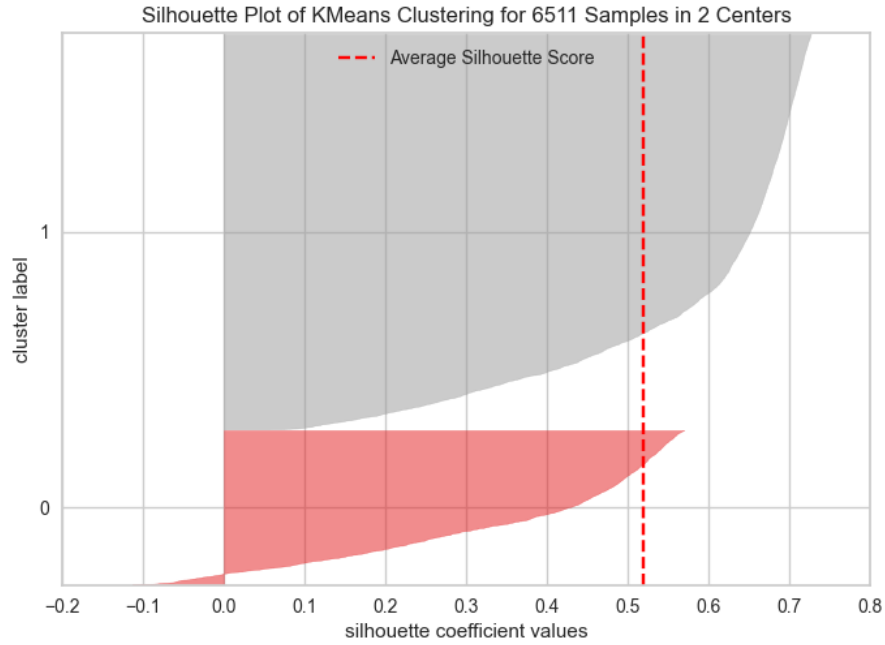


Fonte: autoria própria (2025).

A maior parte das análises com diferentes números de componentes indicou que a melhor quantidade de clusters estaria entre 3 e 8. No entanto, para ampliar a quantidade de dados disponíveis para análise — e também porque, em alguns testes, foi difícil identificar com precisão o ponto do “cotovelo” em certos gráficos — optou-se por expandir um pouco o intervalo de clusters na análise de silhueta, utilizando quantidades de 2 a 10 clusters.

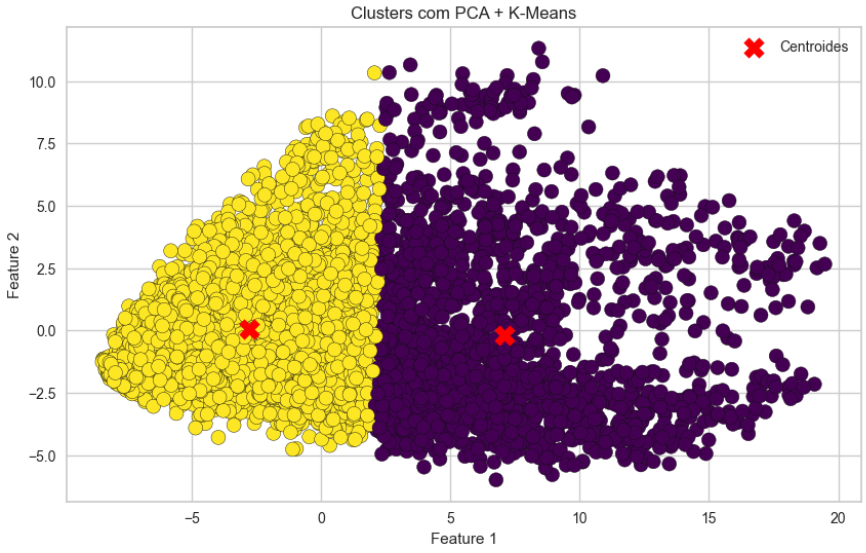
Na análise da aplicação do PCA, o melhor resultado do índice de silhueta foi 0,52, obtido com a configuração de apenas 2 componentes e 2 clusters, como mostra o Gráfico 4. Podemos ver a visualização de clusters nessa configuração no Gráfico 5. À medida que o número de clusters aumentava, independentemente da quantidade de componentes, o índice de silhueta diminuía. Também ficou evidente que, neste conjunto de dados, aumentar o número de componentes no PCA não resultava em uma melhora do índice — na verdade, ocorreu o oposto: quanto mais componentes, menor o índice. Com 3 componentes, por exemplo, o melhor valor obtido foi 0,47, também com a utilização de apenas 2 clusters, como mostrado no Gráfico 6. A visualização dessa configuração pode ser vista em Gráfico 7

Gráfico 4 – Índice de silhueta para KMeans com 2 clusters pós redução de dimensionalidade utilizando PCA com 2 componentes



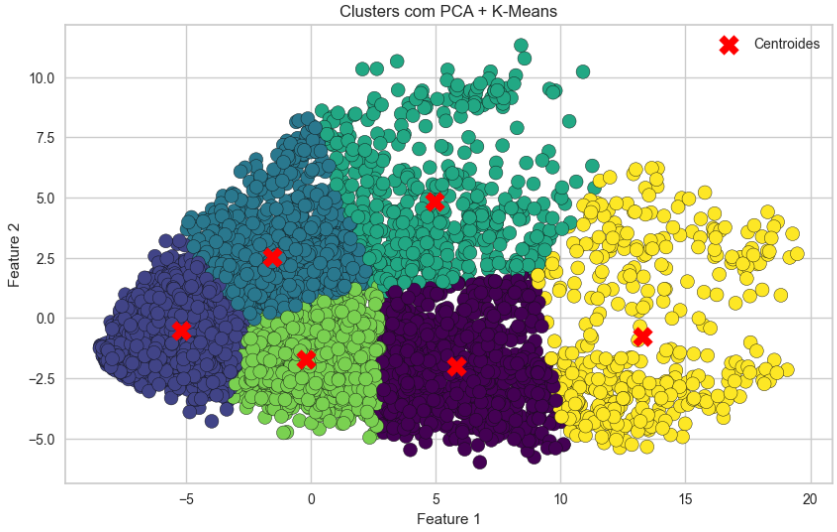
Fonte: autoria própria (2025).

Gráfico 5 – Visualização de clusters com utilização de KMeans com 2 clusters + PCA com 2 componentes



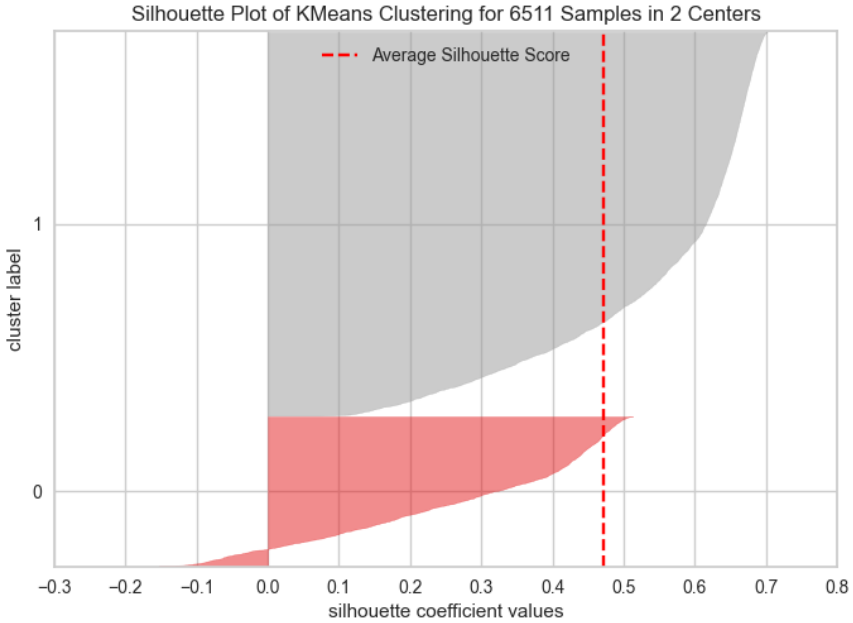
Fonte: autoria própria (2025).

Gráfico 6 – Visualização de clusters com utilização de KMeans com 6 clusters + PCA com 2 componentes



Fonte: autoria própria (2025).

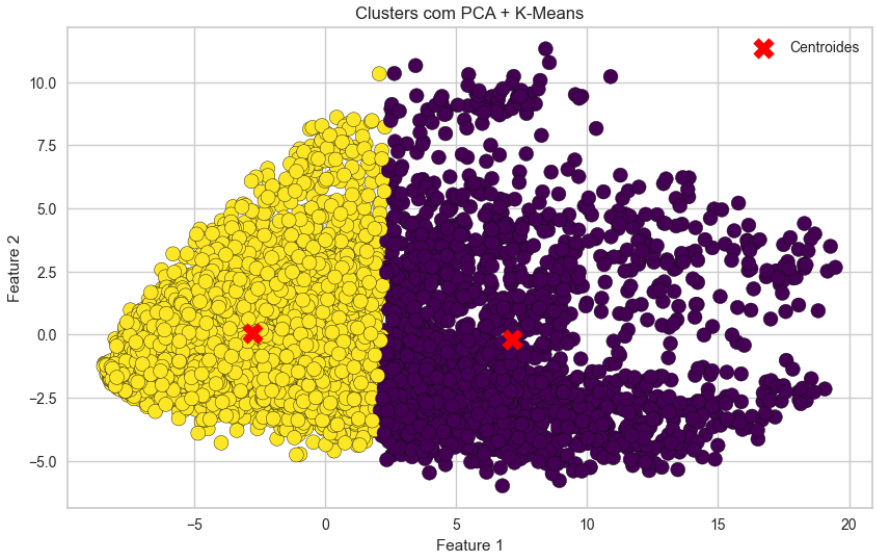
Gráfico 7 – Índice de silhueta para KMeans pós redução de dimensionalidade utilizando PCA com 3 componentes



Fonte: autoria própria (2025).

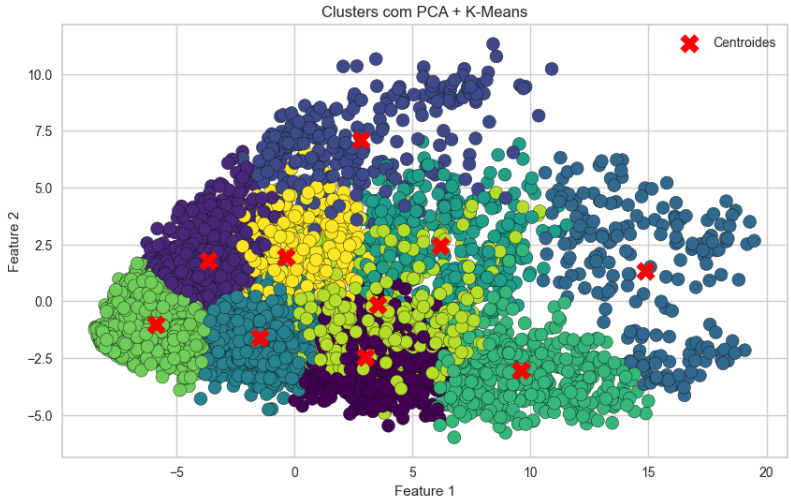
Abaixo, no Gráfico 8 e Gráfico 9, se encontram alguns outros exemplos de resultados da clusterização após redução de dimensionalidade com PCA.

Gráfico 8 – Visualização de clusters com utilização de KMeans com 2 clusters + PCA com 3 componentes



Fonte: autoria própria (2025).

Gráfico 9 – Visualização de clusters com utilização de KMeans com 10 clusters + PCA com 3 componentes



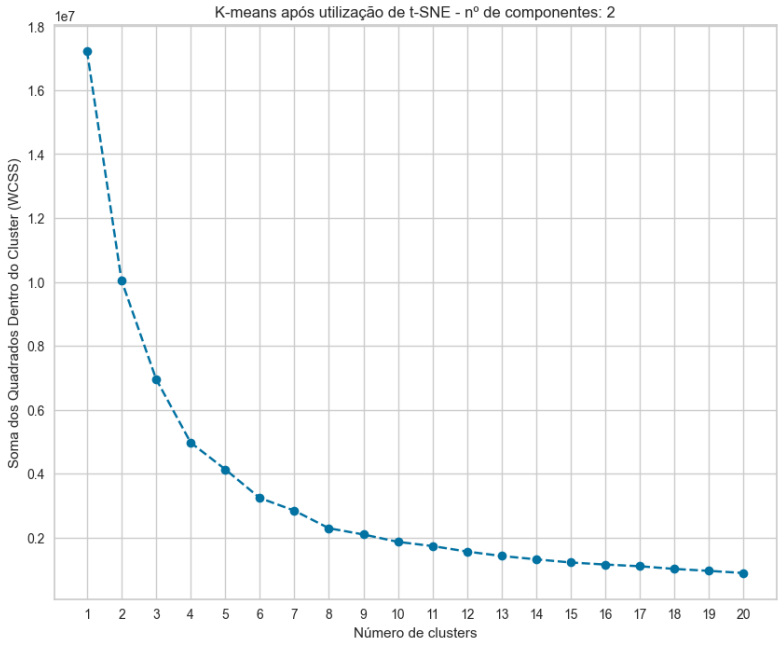
Fonte: autoria própria (2025).

5.2 t-SNE

Na análise do método t-SNE, foi realizada uma quantidade menor de testes devido às características do algoritmo e ao seu tempo de execução. O algoritmo padrão para o cálculo do gradiente do t-SNE na biblioteca Scikit-Learn, Barnes-Hut, permite reduzir a dimensionalidade de um dataset utilizando até 3 componentes. Acima disso, seria necessário utilizar outro algoritmo, o Exact. No entanto, ao tentar usar esse algoritmo para reduzir a dimensionalidade para 4 componentes, o tempo de execução ultrapassou significativamente o esperado — após mais de uma hora, o processo ainda não havia sido concluído. Por isso, optou-se por testar o t-SNE apenas com o algoritmo padrão, utilizando 2 e 3 componentes.

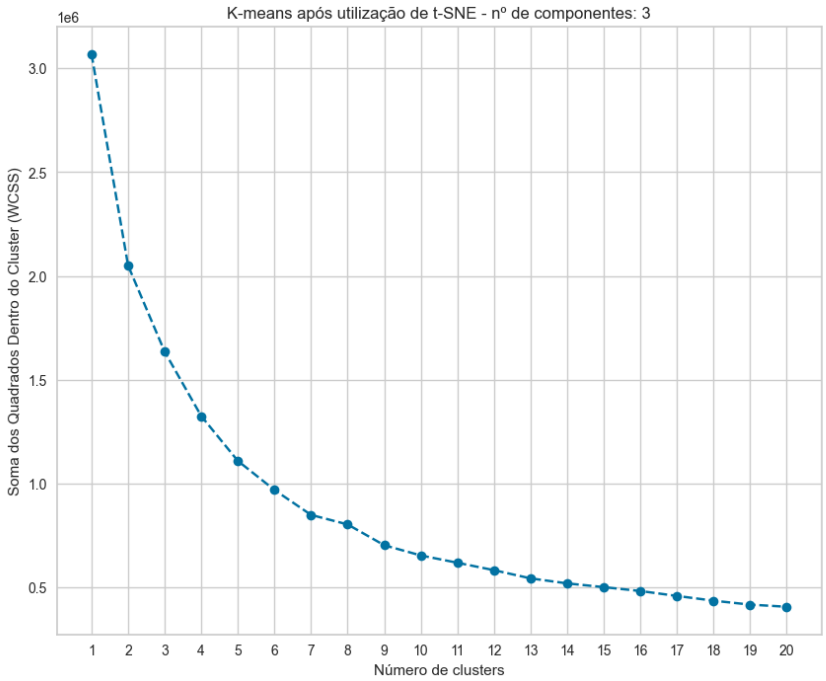
Assim como na análise com PCA, foi utilizado o método do cotovelo para verificar a quantidade ideal de clusters para os testes. Tanto com 2 quanto com 3 componentes, a quantidade ideal de clusters estaria entre 4 e 6, conforme ilustrado em Gráfico 10 e Gráfico 11. No entanto, como os testes com PCA foram realizados com 2 a 10 clusters, optou-se por utilizar esse mesmo intervalo nos testes com t-SNE.

Gráfico 10 – Método do cotovelo para 2 componentes



Fonte: autoria própria (2025).

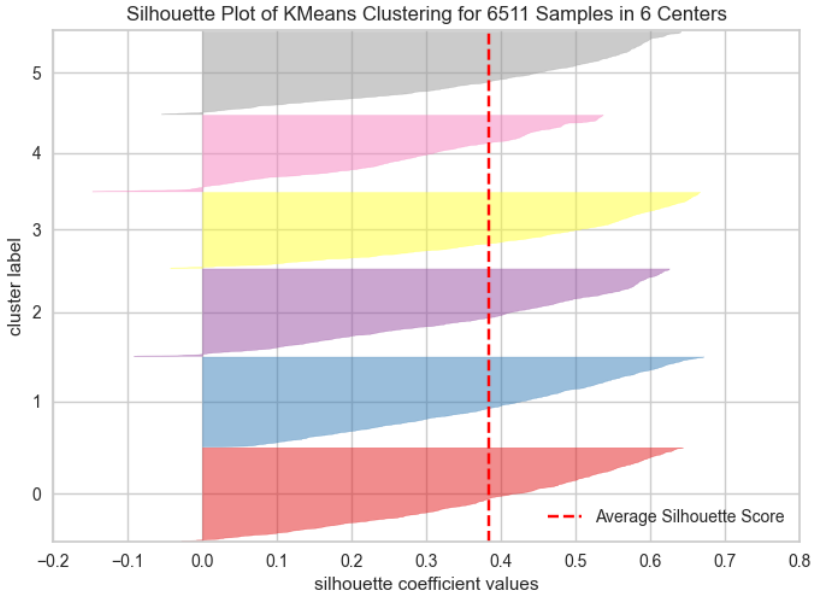
Gráfico 11 – Método do cotovelo para 3 componentes



Fonte: autoria própria (2025).

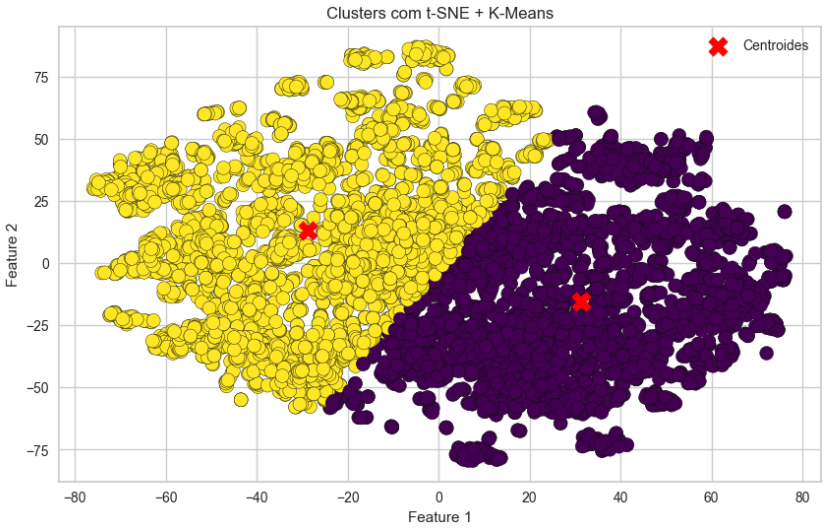
Na análise dos resultados da clusterização K-means após a redução de dimensionalidade com t-SNE, foram testados dois valores para o número de componentes: 2 e 3. Com 2 componentes, o melhor índice de silhueta obtido foi 0,38, com os dados divididos em 6 clusters, como mostra o Gráfico 12. Podemos ver a visualização de clusters nessa configuração no Gráfico 13. Já com 3 componentes, o melhor índice foi 0,30, com um total de 2 clusters, como mostra o Gráfico 14. O resultado da clusterização pode ser vista em Gráfico 15.

Gráfico 12 – Índice de silhueta para KMeans com 6 clusters pós redução de dimensionalidade utilizando t-SNE com 2 componentes



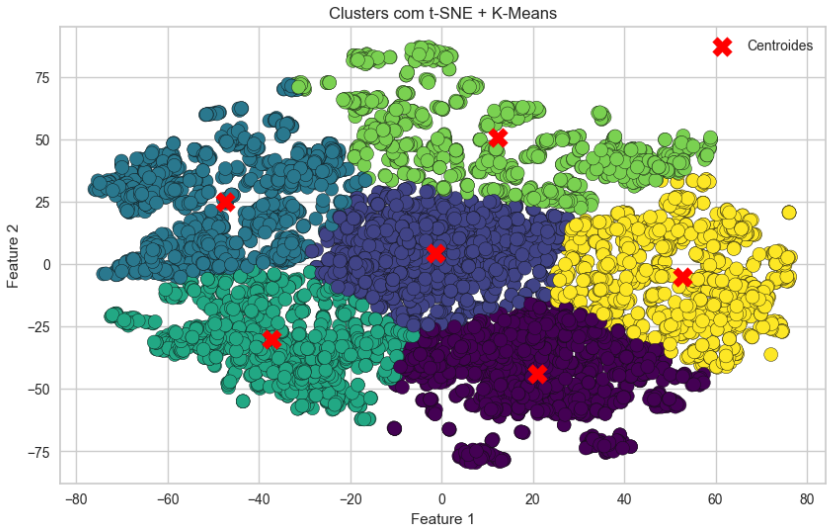
Fonte: autoria própria (2025).

Gráfico 13 – Visualização de clusters com utilização de KMeans com 2 clusters + t-SNE com 2 componentes



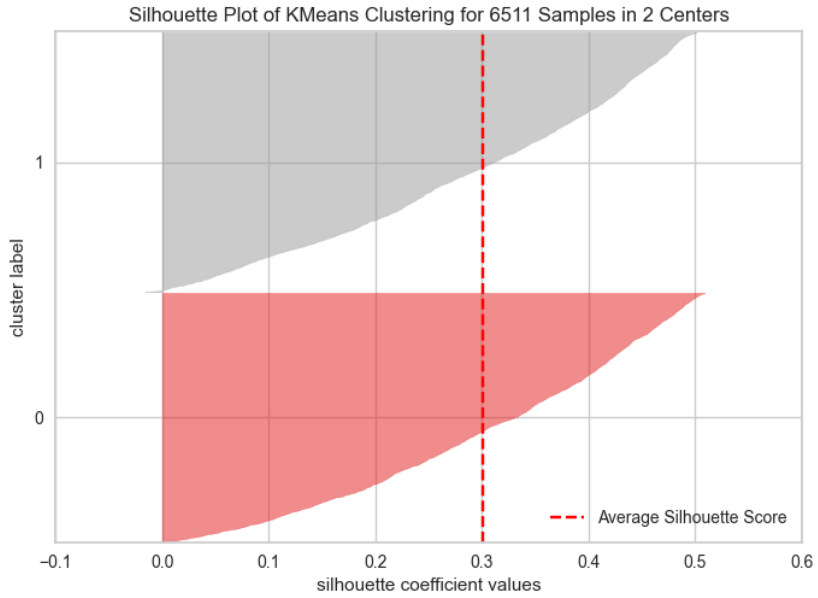
Fonte: autoria própria (2025).

Gráfico 14 – Visualização de clusters com utilização de KMeans com 6 clusters + t-SNE com 2 componentes



Fonte: autoria própria (2025).

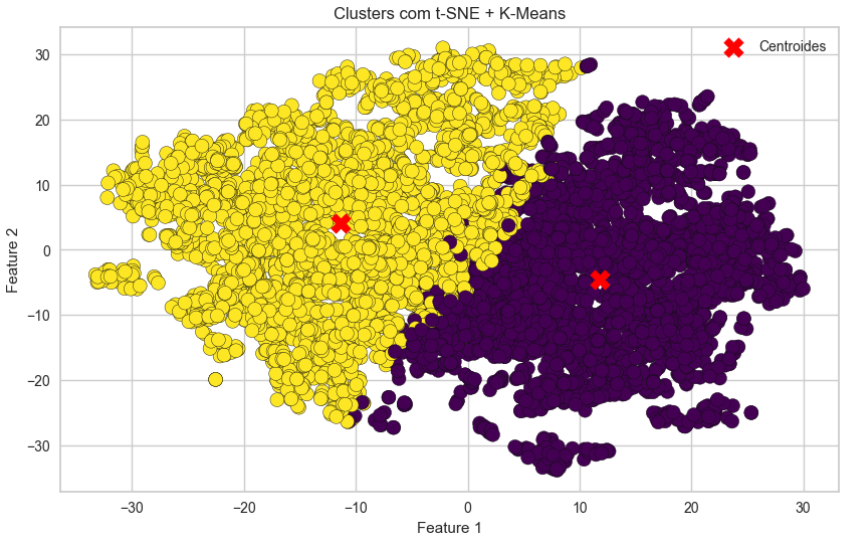
Gráfico 15 – Índice de silhueta para KMeans com 2 clusters pós redução de dimensionalidade utilizando t-SNE com 3 componentes



Fonte: autoria própria (2025).

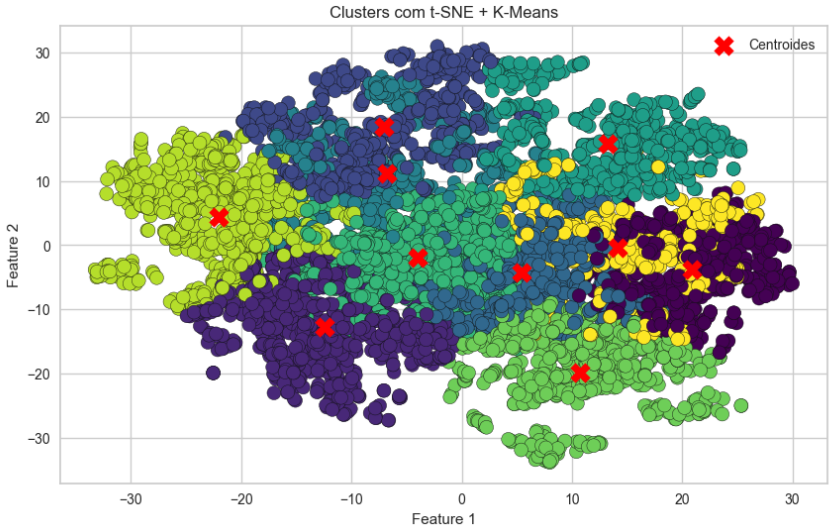
Abaixo, no Gráfico 16 e Gráfico 17, se encontram alguns outros exemplos de resultados da clusterização após redução de dimensionalidade com t-SNE.

Gráfico 16 – Visualização de clusters com utilização de KMeans com 2 clusters + t-SNE com 3 componentes



Fonte: autoria própria (2025).

Gráfico 17 – Visualização de clusters com utilização de KMeans com 10 clusters + t-SNE com 3 componentes

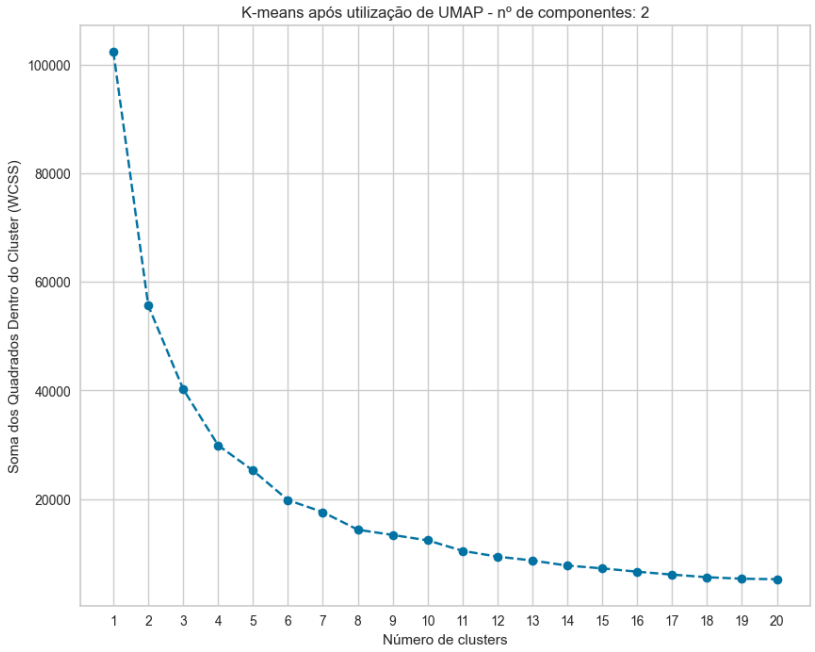


Fonte: autoria própria (2025).

5.3 UMAP

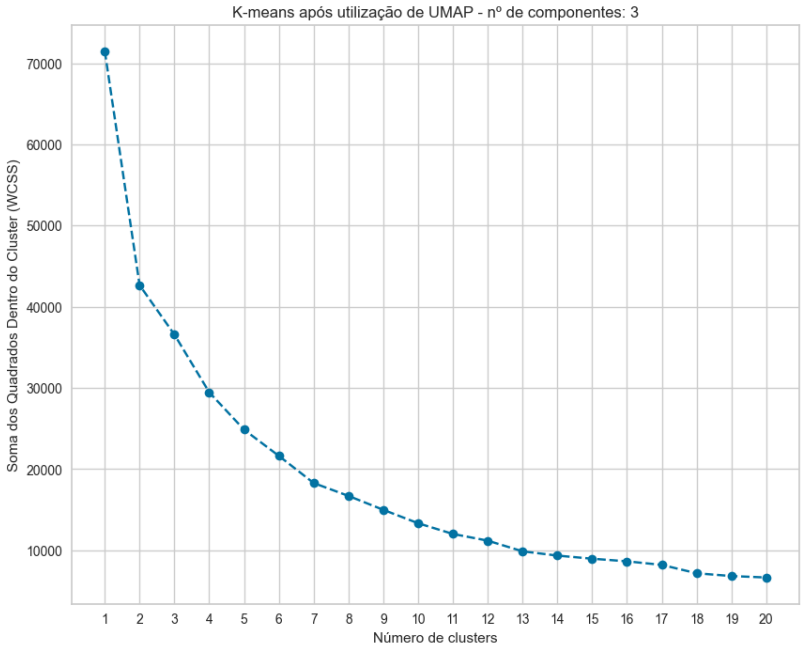
Na análise com o UMAP, foi aplicado o método do cotovelo, assim como no PCA, para diferentes quantidades de componentes (de 2 a 40) e com valores de clusters variando de 2 a 20. Assim como no PCA, em muitos gráficos foi difícil identificar o “cotovelo” que indicaria o melhor número de clusters. Ainda assim, observou-se certa semelhança entre os gráficos gerados com PCA e UMAP, que apontavam um número ideal de clusters entre 3 e 6. Como feito anteriormente, a análise de silhueta foi realizada em um intervalo maior de clusters, de 2 a 10, e com duas quantidades de componentes, 2 e 3, seguindo o mesmo critério adotado tanto no PCA quanto no t-SNE, como mostrado no Gráfico 18 e Gráfico 19.

Gráfico 18 – Método do cotovelo para 2 componentes



Fonte: autoria própria (2025).

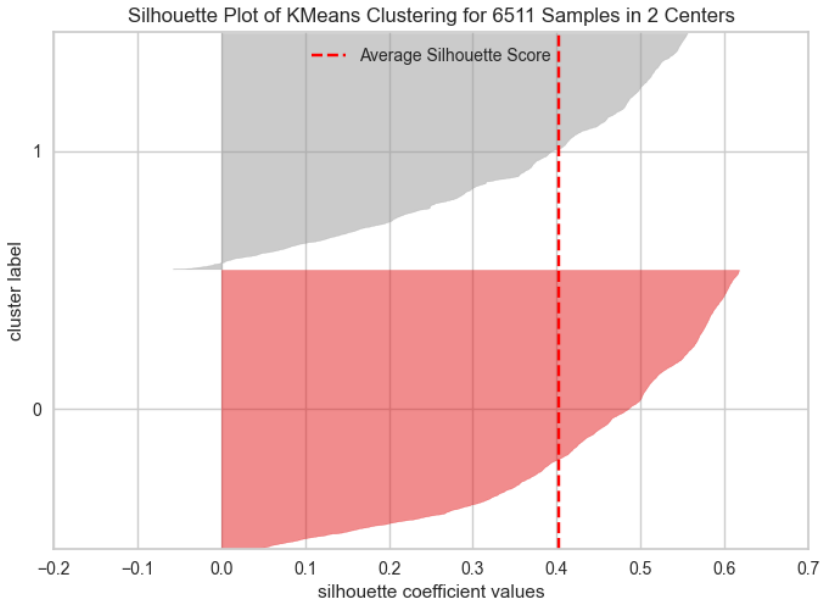
Gráfico 19 – Método do cotovelo para 3 componentes



Fonte: autoria própria (2025).

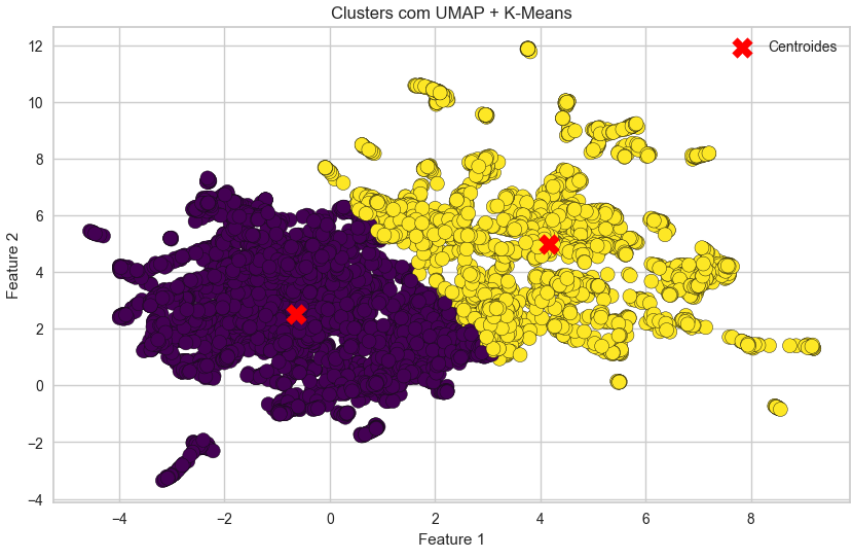
Na análise do método UMAP, com 2 componentes, o melhor índice de silhueta foi 0,40, com os dados divididos em 2 clusters, como mostrado no Gráfico 20. O resultado da clusterização com essa configuração pode ser visto em Gráfico 21. Já com 3 componentes, o melhor resultado foi 0,37, com os dados agrupados em 10 clusters distintos, como evidenciado em Gráfico 22 e o resultado da clusterização pode ser visto em Gráfico 23.

Gráfico 20 – Índice de silhueta para KMeans com 2 clusters pós redução de dimensionalidade utilizando UMAP com 2 componentes



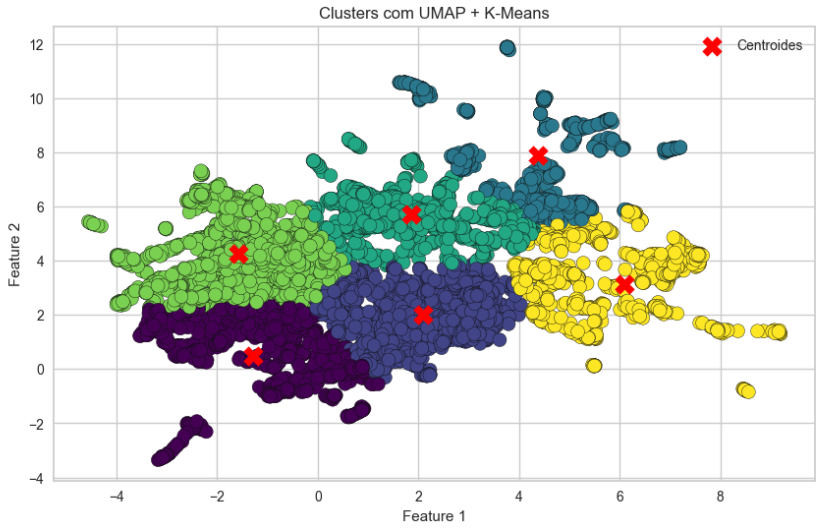
Fonte: autoria própria (2025).

Gráfico 21 – Visualização de clusters com utilização de KMeans com 2 clusters + UMAP com 2 componentes



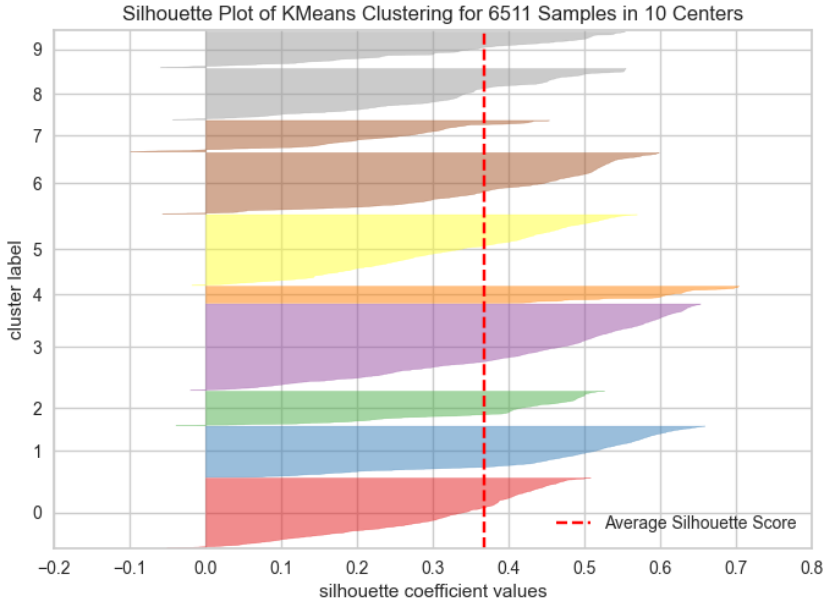
Fonte: autoria própria (2025).

Gráfico 22 – Visualização de clusters com utilização de KMeans com 6 clusters + UMAP com 2 componentes



Fonte: autoria própria (2025).

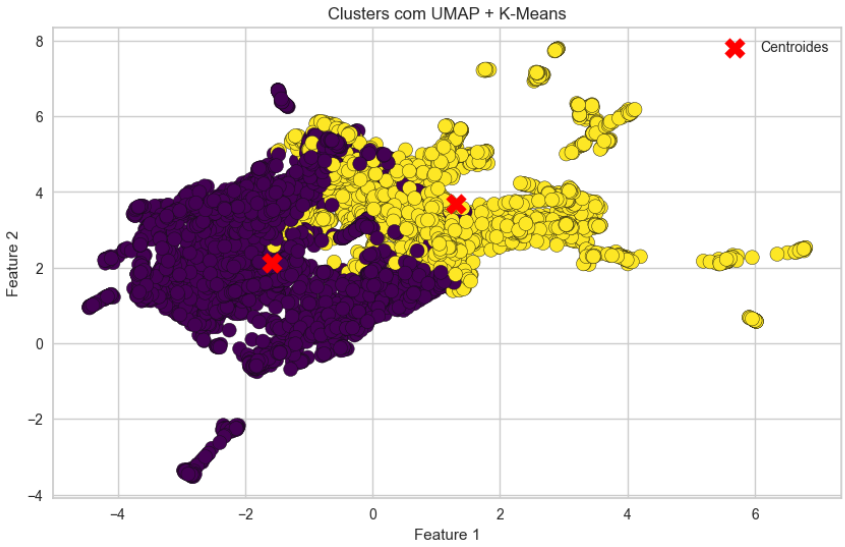
Gráfico 23 – Índice de silhueta para KMeans com 10 clusters pós redução de dimensionalidade utilizando UMAP com 3 componentes



Fonte: autoria própria (2025).

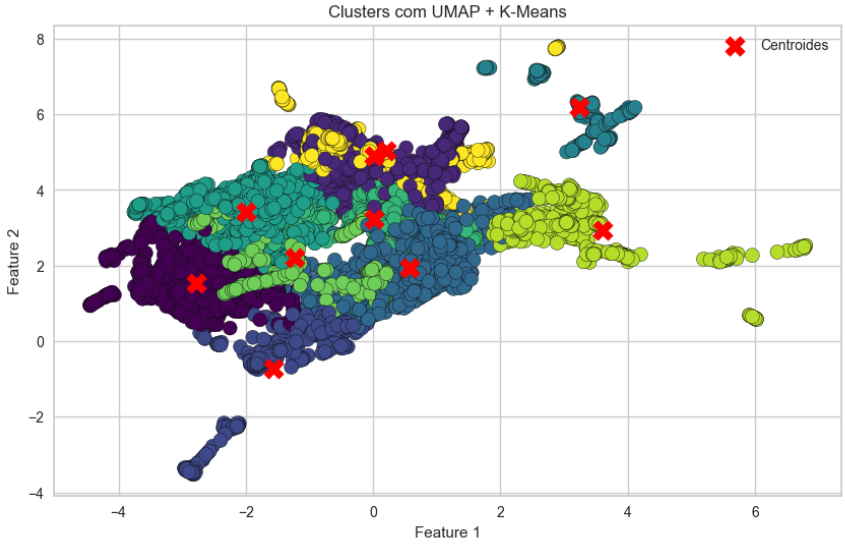
Abaixo, no Gráfico 24 e Gráfico 25, se encontram alguns outros exemplos de resultados da clusterização após redução de dimensionalidade com UMAP.

Gráfico 24 – Visualização de clusters com utilização de KMeans com 2 clusters + UMAP com 3 componentes



Fonte: autoria própria (2025).

Gráfico 25 – Visualização de clusters com utilização de KMeans com 10 clusters + UMAP com 3 componentes



Fonte: autoria própria (2025).

5.4 Comparações entre métodos

Com dois componentes, a combinação de PCA com K-means apresentou índices de silhueta superiores aos obtidos com o t-SNE em todos os testes com diferentes quantidades de clusters, exceto com a quantidade de clusters igual a 7, onde os métodos de redução de dimensionalidade obtiveram o mesmo resultado. Em comparação com o UMAP, o PCA se saiu melhor na maioria dos testes, exceto nos com a quantidade de clusters sendo 3 e 8, onde ambos os métodos obtiveram o mesmo resultado. Já com três componentes, o PCA alcançou índices mais altos em todas as quantidades de clusters testadas em relação ao t-SNE. No entanto, ao comparar com o UMAP, o desempenho foi mais equilibrado: o PCA obteve um melhor resultado em 5 testes (inclusive obteve o valor geral mais alto de índice de silhueta), enquanto o UMAP foi melhor nos outros 4.

Na comparação direta entre t-SNE e UMAP com dois componentes, o UMAP apresentou desempenho superior em quatro dos nove testes, enquanto

o t-SNE foi melhor em três, e houve empate em dois. Ainda assim, o UMAP alcançou o maior índice de silhueta isolado entre os dois métodos nesse cenário. Com três componentes, o UMAP demonstrou desempenho consistentemente superior, superando o t-SNE em todos os testes realizados, com destaque para os cenários com maior número de clusters, onde a diferença entre os métodos foi mais expressiva.

Abaixo, na Tabela 1 e Tabela 2, encontram-se todos os resultados do índice de silhueta para cada método.

Tabela 1 – Índices de silhueta para diferentes métodos de redução de dimensionalidade com 2 componentes e diferentes valores de k (número de clusters)

	Quantidade de Clusters (k)								
Método	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
PCA	**0.52**	**0.39**	**0.42**	**0.41**	**0.39**	**0.37**	**0.38**	**0.38**	**0.37**
t-SNE	0.37	0.36	0.36	0.36	0.38	**0.37**	0.37	0.36	0.36
UMAP	0.40	**0.39**	0.36	0.35	0.37	0.35	**0.38**	0.37	0.36

Fonte: autoria própria (2025).

Tabela 2 – Índices de silhueta para diferentes métodos de redução de dimensionalidade com 3 componentes e diferentes valores de k (número de clusters)

	Quantidade de Clusters (k)								
Método	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
PCA	**0.47**	**0.33**	**0.35**	**0.33**	**0.32**	0.31	0.31	0.31	0.30
t-SNE	0.30	0.27	0.26	0.28	0.27	0.29	0.27	0.29	0.29
UMAP	0.35	0.28	0.27	0.30	0.30	**0.33**	**0.35**	**0.36**	**0.37**

Fonte: autoria própria (2025).

5.5 Análise de tempo

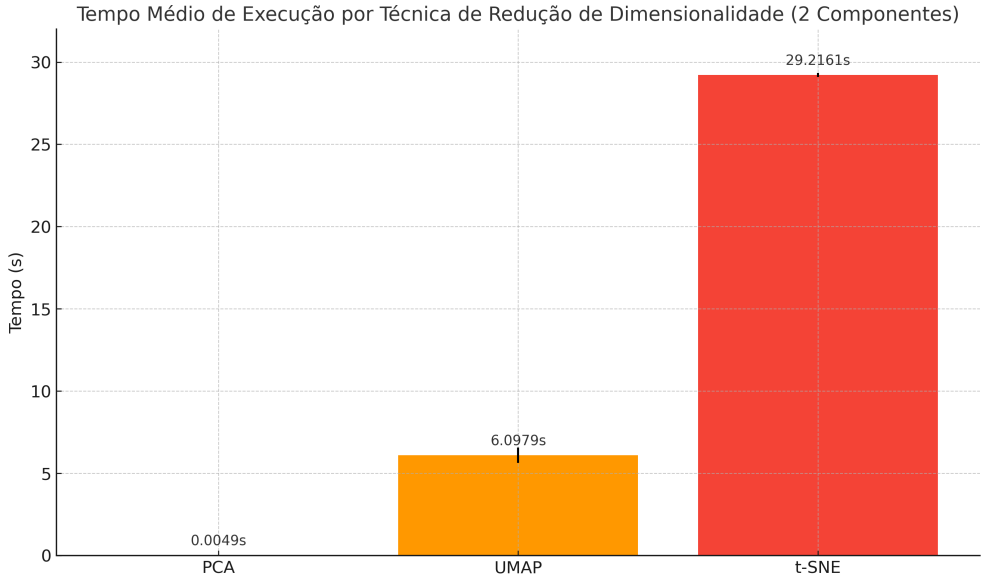
Para garantir a reprodutibilidade dos experimentos e fornecer um contexto para a análise do tempo de execução das técnicas testadas, os experimentos foram realizados em um computador com as seguintes especificações de hardware: processador 12th Gen Intel® Core™ i5-12400F de 2.50 GHz, 16 GB de memória RAM, SSD de 1 TB e GPU NVIDIA GeForce RTX 3060 Ti.

Na análise de tempo foi utilizado o módulo timeit do Python. Todos os métodos de redução de dimensionalidade foram testados com 2 e 3 componentes e com o mesmo número de repetições: 50.

A análise do tempo de execução dos métodos de redução de dimensionalidade revelou diferenças expressivas de desempenho entre o PCA, o t-SNE e o UMAP, tanto com dois quanto com três componentes principais.

Com dois componentes, o PCA apresentou o menor tempo médio de execução, com 0,0049 s (desvio padrão de 0,0004 s), e intervalo de confiança de 95% entre 0,0048 e 0,0050 s. O t-SNE, por sua vez, foi o mais demorado, com tempo médio de 29,2161 s (desvio padrão de 0,4509 s), e intervalo de confiança entre 29,0879 e 29,3442 s. Já o UMAP apresentou desempenho intermediário, com média de 6,0979 s (desvio padrão de 1,6687 s), e intervalo de confiança entre 5,6236 e 6,5721 s. O Gráfico 26 mostra a diferença entre os tempos de cada método.

Gráfico 26 – Tempo Médio de Execução por Técnica de Redução de Dimensionalidade (2 componentes)



Fonte: autoria própria (2025).

A aplicação da ANOVA tipo II confirmou a existência de diferenças estatisticamente significativas entre os tempos médios dos métodos:

$$F(2, 147) = 11922{,}88, \quad p < 0{,}001$$

A análise post hoc com o teste de Tukey HSD indicou diferenças significativas em todas as comparações pareadas, conforme apresentado na Tabela 3.

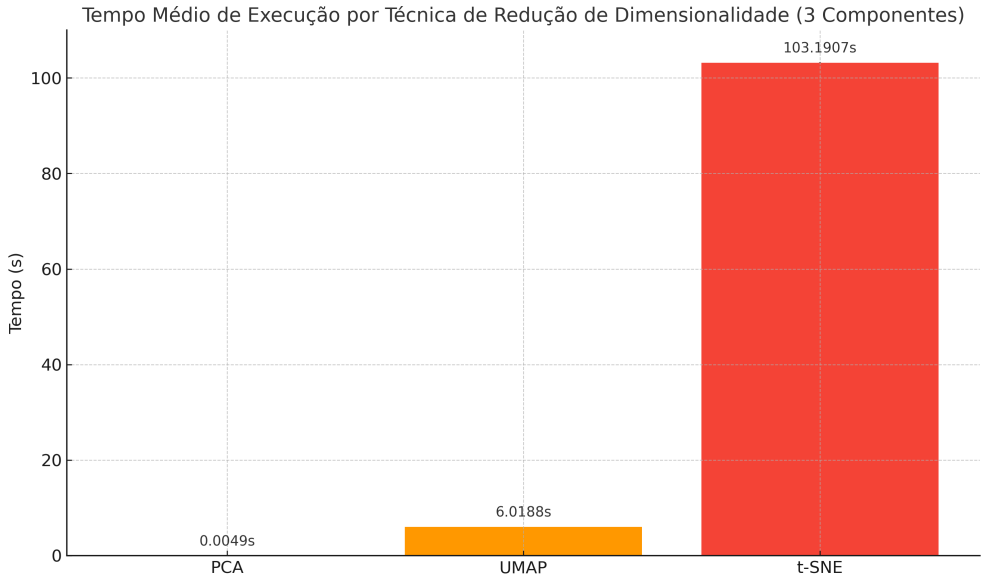
Tabela 3 – Resultados do teste de Tukey para os tempos com dois componentes

Grupo 1	Grupo 2	Diferença	IC 95%	p-valor	Significância
PCA	UMAP	6,0930	(5,6204, 6,5656)	< 0,001	Sim
PCA	t-SNE	29,2112	(28,7386, 29,6838)	< 0,001	Sim
UMAP	t-SNE	23,1182	(22,6456, 23,5908)	< 0,001	Sim

Fonte: autoria própria (2025).

Com três componentes, a ordem de desempenho manteve-se semelhante. O PCA continuou sendo o mais eficiente, com tempo médio de 0,0049 s (desvio padrão de 0,0005 s), e intervalo de confiança entre 0,0048 e 0,0050 s. O t-SNE apresentou novamente o maior tempo, com média de 103,1907 s (desvio padrão de 0,2040 s), e intervalo de confiança entre 103,1328 e 103,2487 s. O UMAP teve média de 6,0188 s (desvio padrão de 0,0468 s), com intervalo de confiança entre 6,0055 e 6,0321 s. O Gráfico 27 evidencia as discrepâncias nos tempos de execução observados entre os métodos avaliados.

Gráfico 27 – Tempo Médio de Execução por Técnica de Redução de Dimensionalidade (3 componentes)



Fonte: autoria própria (2025).

A ANOVA tipo II também indicou diferenças estatisticamente significativas:

$$F(2, 147) = 11481810{,}43, \quad p < 0{,}001$$

O teste de Tukey HSD confirmou que todas as comparações entre os métodos apresentaram diferenças estatisticamente significativas, conforme mostrado na Tabela 4.

Tabela 4 – Resultados do teste de Tukey para os tempos com três componentes

Grupo 1	Grupo 2	Diferença	IC 95%	p-valor	Significância
PCA	UMAP	6,0139	(5,9567, 6,0712)	< 0,001	Sim
PCA	t-SNE	103,1858	(103,1286, 103,2431)	< 0,001	Sim
UMAP	t-SNE	97,1719	(97,1147, 97,2291)	< 0,001	Sim

Fonte: autoria própria (2025).

Esses resultados demonstram que o PCA é significativamente mais rápido que os demais métodos em ambos os cenários avaliados. O t-SNE, por outro lado, apresentou o maior tempo de execução, o que pode representar uma limitação prática em contextos onde o desempenho computacional é um fator crítico. O UMAP, embora mais demorado que o PCA, mostrou-se uma alternativa consideravelmente mais eficiente do que o t-SNE.

5.6 Discussões

Na análise detalhada dos índices de silhueta, observou-se que o PCA apresentou desempenho superior ao t-SNE em praticamente todos os cenários avaliados, tanto com dois quanto com três componentes. Em comparação ao UMAP, o PCA também se destacou na maioria dos testes com 2 componentes. Já com três componentes, os resultados foram mais equilibrados, onde cada técnica foi superior em parte dos casos. Já na comparação entre t-SNE e UMAP, o UMAP apresentou leve vantagem com dois componentes e superou o t-SNE de forma consistente quando três componentes foram utilizados.

Além da qualidade dos agrupamentos, a análise do tempo de execução revelou que o PCA foi a técnica mais eficiente em todos os cenários avaliados. Com tempo médio de 0,0049 s em ambos os casos (2 e 3 componentes), desvio padrão inferior a 0,0005 s e intervalos de confiança com pequena amplitude, o PCA se destacou pela rapidez. O UMAP obteve tempos intermediários, com médias entre 6,01 s e 6,09 s, enquanto o t-SNE apresentou os maiores tempos de execução, com 29,21 s para dois componentes e 103,19 s para três componentes. Os testes estatísticos (ANOVA tipo II e Tukey HSD) confirmaram que

essas diferenças entre os métodos foram estatisticamente significativas, com $p < 0{,}001$ em todas as comparações pareadas.

De modo geral, o PCA demonstrou ser o método mais eficiente e competitivo, apresentando tanto os melhores tempos de execução quanto os índices de silhueta mais elevados em grande parte dos testes. O UMAP, embora mais demorado que o PCA, mostrou-se vantajoso em alguns contextos com três componentes, especialmente com maiores quantidades de clusters. Já o t-SNE, apesar de apresentar relativa estabilidade frente à variação no número de clusters, foi consistentemente o método com pior desempenho, tanto em termos de tempo quanto de qualidade dos agrupamentos. Ressalta-se, porém, que esses resultados se referem exclusivamente à base de dados utilizada neste estudo, podendo variar significativamente com outras bases. Além disso, a quantidade de clusters mostrou-se um fator relevante nos resultados, como observado no desempenho crescente do UMAP com três componentes à medida que se aumentava o número de clusters. Outro ponto relevante é que a escolha dos hiperparâmetros do UMAP e do t-SNE pode impactar de forma significativa os resultados obtidos por esses métodos; entretanto, neste trabalho foram utilizados apenas os valores padrão de cada técnica.

Os resultados apresentados neste trabalho corroboram, em certa medida, as conclusões das pesquisas de Baligodugula e Amsaad (2025) e Hozumi et al. (2021), nas quais o UMAP foi identificado como a técnica de redução de dimensionalidade com melhor desempenho. De forma semelhante, nesta pesquisa, o UMAP obteve os maiores índices de silhueta em alguns cenários, dependendo do número de clusters considerado. No entanto, de forma geral, os melhores resultados foram alcançados com o uso do PCA, que apresentou desempenho superior na maioria dos testes realizados. É importante destacar, contudo, que embora os estudos consultados apresentem resultados distintos dos observados nesta pesquisa, tais diferenças são justificáveis, uma vez que foram utilizadas bases de dados distintas, o que pode influenciar significativamente os resultados obtidos.

6 Conclusão

Neste trabalho, foi analisado o impacto da utilização de técnicas de redução de dimensionalidade na clusterização de dados da Educação Superior do Brasil com o algoritmo K-Means. Para isso, foram comparadas as abordagens PCA, t-SNE e UMAP, considerando dois critérios principais: índice de silhueta, que mede a qualidade dos agrupamentos, e tempo de execução, que avalia a eficiência computacional de cada método.

Com base nos resultados obtidos e enfatizando que esses mesmos resultados podem mudar de acordo com a base de dados utilizada, o PCA se mostra uma opção robusta, considerando os dados utilizados do censo, tanto em termos de qualidade dos agrupamentos quanto de eficiência computacional, sendo uma alternativa viável para cenários em que há necessidade de rapidez na execução. Porém, é necessária atenção na questão da quantidade de clusters e em como cada método se comporta no dataset selecionado. O UMAP pode ser uma alternativa interessante, especialmente quando há interesse em explorar maior variabilidade nos agrupamentos com mais de 2 componentes. O t-SNE, apesar de apresentar bons índices de silhueta em alguns casos, tem como principal limitação o alto tempo de processamento, tornando-o menos eficiente para grandes conjuntos de dados.

É importante enfatizar que, embora o PCA tenha se destacado como a técnica com melhor desempenho na maioria dos cenários analisados — tanto em termos de tempo de execução quanto na qualidade dos agrupamentos — sua superioridade, neste último aspecto, não é absoluta. Observou-se que, especialmente com três componentes e à medida que se aumentava a quantidade de clusters, o UMAP apresentou desempenho competitivo, superando o PCA em alguns casos. As diferenças entre esses dois métodos, nos testes com três componentes, foram geralmente modestas. Técnicas como o t-SNE e o UMAP demonstraram resultados relevantes em situações específicas, o que reforça a recomendação de que, sempre que se aplicar a redução de dimensionalidade,

diferentes métodos sejam testados, considerando as particularidades dos dados e os objetivos da análise.

Como trabalhos futuros, sugere-se alguns pontos:

- Explorar a aplicação dessas técnicas em diferentes conjuntos de dados: o presente estudo foi focado em dados educacionais específicos dos anos 2019, 2021 e 2022, da educação superior pública e presencial do Brasil. Existem diversos outros dados disponíveis sobre educação, então aplicar as mesmas técnicas nesses outros dados permitiria verificar se os padrões observados se mantêm.
- Avaliar o impacto da escolha de hiperparâmetros para t-SNE e UMAP: Tanto o t-SNE quanto o UMAP possuem hiperparâmetros sensíveis, como a perplexidade no t-SNE e o número de vizinhos no UMAP, que podem influenciar diretamente os resultados da redução de dimensionalidade. Embora esta análise não tenha contemplado diferentes configurações desses parâmetros, seria interessante investigar, em estudos futuros, como variações nesses hiperparâmetros afetam a qualidade dos agrupamentos gerados.
- Utilizar outros algoritmos de clusterização: O estudo utilizou apenas o K-Means. No entanto, outros algoritmos como DBSCAN, Clusterização Hierárquica, entre outros, podem lidar melhor com os dados. Comparar os resultados entre diferentes algoritmos pode fornecer insights mais abrangentes sobre a eficácia da redução de dimensionalidade na clusterização.
- Avaliar os resultados do K-Means com e sem a aplicação de técnicas de redução de dimensionalidade, de modo a analisar de forma mais abrangente o impacto desses métodos no desempenho da clusterização.

Referências

ABDI, H.; WILLIAMS, L. J. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, Wiley Online Library, v. 2, n. 4, p. 433–459, 2010. Citado 2 vezes nas páginas 21 e 28.

ALI, Z. M.; HASSOON, N. H.; AHMED, W. S.; ABED, H. N. The application of data mining for predicting academic performance using k-means clustering and naïve bayes classification. *International Journal of Psychosocial Rehabilitation*, v. 24, n. 03, p. 2143–2151, 2020. Citado na página 14.

ALLAOUI, M.; KHERFI, M. L.; CHERIET, A. Considerably improving clustering algorithms using umap dimensionality reduction technique: a comparative study. In: SPRINGER. *International conference on image and signal processing*. [S.l.], 2020. p. 317–325. Citado na página 34.

BAKER, R.; ISOTANI, S.; CARVALHO, A. Mineraçao de dados educacionais: Oportunidades para o brasil. *Revista Brasileira de informática na educação*, v. 19, n. 02, p. 03, 2011. Citado na página 14.

BALIGODUGULA, V. V.; AMSAAD, F. *Unsupervised Learning: Comparative Analysis of Clustering Techniques on High-Dimensional Data*. 2025. Preprint. Disponível em: <https://arxiv.org/abs/2503.23215>. Acesso em: 16 jul. 2025. Citado 3 vezes nas páginas 34, 35 e 66.

DEVASSY, B. M.; GEORGE, S. Dimensionality reduction and visualisation of hyperspectral ink data using t-sne. *Forensic science international*, Elsevier, v. 311, p. 110194, 2020. Citado na página 31.

FERREIRA, L. A.; RODRIGUES, R. L.; SOUZA, R. N. de. Dados abertos educacionais brasileiros: Um mapeamento sistemático da literatura. *Simpósio Brasileiro de Informática na Educação (SBIE)*, SBC, p. 1186–1195, 2021. Citado 2 vezes nas páginas 12 e 15.

GéRON, A. *Mãos à Obra: Aprendizado de Máquina com Scikit-Learn, Keras & TensorFlow: Conceitos, Ferramentas e Técnicas para a Construção de Sistemas Inteligentes*. 2. ed. Rio de Janeiro: Editora Alta Books, 2021. E-book. ISBN 9786555208146. Disponível em: <https://integrada.minhabiblioteca.com.br/reader/books/9786555208146/>. Citado 7 vezes nas páginas 12, 15, 16, 17, 18, 19 e 20.

HINNEBURG, A.; KEIM, D. A. Optimal grid-clustering: Towards breaking the curse of dimensionality in high-dimensional clustering. 1999. Citado na página 19.

HOZUMI, Y.; WANG, R.; YIN, C.; WEI, G.-W. Umap-assisted k-means clustering of large-scale sars-cov-2 mutation datasets. *Computers in biology and medicine*, Elsevier, v. 131, p. 104264, 2021. Citado 6 vezes nas páginas 21, 23, 24, 26, 34 e 66.

Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP). *Conceito Enade*. 2020. Acesso em: 22 mar. 2025. Disponível em: <https://www.gov.br/inep/pt-br/areas-de-atuacao/pesquisas-estatisticas-e-indicadores/indicadores-de-qualidade-da-educacao-superior/conceito-enade>. Citado na página 37.

Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP). *Conceito Preliminar de Curso (CPC)*. 2020. Acesso em: 22 mar. 2025. Disponível em: <https://www.gov.br/inep/pt-br/areas-de-atuacao/pesquisas-estatisticas-e-indicadores/indicadores-de-qualidade-da-educacao-superior/conceito-preliminar-de-curso-cpc>. Citado na página 37.

KHAN, I. K.; DAUD, H. B.; ZAINUDDIN, N. B.; SOKKALINGAM, R.; FAROOQ, M.; BAIG, M. E.; AYUB, G.; ZAFAR, M. Determining the optimal number of clusters by enhanced gap statistic in k-mean algorithm. *Egyptian Informatics Journal*, Elsevier, v. 27, p. 100504, 2024. Citado na página 33.

MAATEN, L. v. d.; HINTON, G. Visualizing data using t-sne. *Journal of machine learning research*, v. 9, n. Nov, p. 2579–2605, 2008. Citado 3 vezes nas páginas 23, 24 e 28.

MAATEN, L. V. D.; POSTMA, E. O.; HERIK, H. J. V. D. et al. Dimensionality reduction: A comparative review. *Journal of machine learning research*, v. 10, n. 66-71, p. 13, 2009. Citado 3 vezes nas páginas 19, 20 e 30.

MCINNES, L. *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*. 2024. Documentação online. Disponível em: <https://umap-learn.readthedocs.io/en/latest/>. Citado 2 vezes nas páginas 26 e 28.

MCINNES, L.; HEALY, J.; MELVILLE, J. *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*. 2018. Preprint. Disponível em: <https://arxiv.org/abs/1802.03426>. Acesso em: 25 jul. 2025. Citado 2 vezes nas páginas 26 e 29.

OGBUABOR, G.; UGWOKE, F. Clustering algorithm for a healthcare dataset using silhouette score value. *Int. J. Comput. Sci. Inf. Technol*, v. 10, n. 2, p. 27–37, 2018. Citado na página 32.

PAL, A. K.; PAL, S. Classification model of prediction for placement of students. *International Journal of Modern Education and Computer Science*, Modern Education and Computer Science Press, v. 5, n. 11, p. 49, 2013. Citado 2 vezes nas páginas 14 e 15.

REDDY, G. T.; REDDY, M. P. K.; LAKSHMANNA, K.; KALURI, R.; RAJPUT, D. S.; SRIVASTAVA, G.; BAKER, T. Analysis of dimensionality reduction techniques on big data. *Ieee Access*, IEEE, v. 8, p. 54776–54788, 2020. Citado na página 30.

RODRIGUES, E. M. *Classificação do risco de evasão no ensino superior a partir do rendimento acadêmico e do agrupamento de cursos*. Dissertação (Dissertação (Mestrado)) — Universidade Federal Rural de Pernambuco, Programa de Pós-Graduação em Informática Aplicada, Recife, 2024. Disponível em: <https://www.ppgia.ufrpe.br/sites/default/files/testes-dissertacoes/CLASSIFICA%C3%87%C3%83O%20DO%20RISCO%20DE%20EVAS%C3%83O%20NO%20ENSINO%20SUPERIOR%20A%20PARTIR%20DO%20RENDIMENTO%20ACAD%C3%8AMICO%20E%20DO%20AGRUPAMENTO%20DE%20CURSOS.pdf>. Citado na página 33.

SAPUTRA, D. M.; SAPUTRA, D.; OSWARI, L. D. Effect of distance metrics in determining k-value in k-means clustering using elbow and silhouette method. In: ATLANTIS PRESS. *Sriwijaya international conference on information technology and its applications (SICONIAN 2019)*. [S.l.], 2020. p. 341–346. Citado na página 31.

VENKAT, N. The curse of dimensionality: Inside out. *Pilani (IN): Birla Institute of Technology and Science, Pilani, Department of Computer Science and Information Systems*, v. 10, n. 10.13140, 2018. Citado na página 19.

WELLA, Y. E.; OKFALISA, O.; INSANI, F.; SAEED, F.; HUSSIN, A. R. C. Service quality dealer identification: the optimization of k-means clustering. *SINERGI*, Fakultas Teknik Universitas Mercu Buana, v. 27, n. 3, p. 433, 2023. Citado na página 33.

YUAN, C.; YANG, H. Research on k-value selection method of k-means clustering algorithm. *J*, MDPI, v. 2, n. 2, p. 226–235, 2019. Citado na página 32.

A Metadados

Coluna	Descrição
no_cine_area_geral_agricultura,_silvicultura,_pesca_e_veterinaria	Curso da área de Agricultura, Silvicultura, Pesca e Veterinária
no_cine_area_geral_artes_e_humanidades	Curso da área de Artes e Humanidades
no_cine_area_geral_ciencias_naturais,_matematica_e_estatistica	Curso da área de Ciências Naturais, Matemática e Estatística
no_cine_area_geral_ciencias_sociais,_comunicacao_e_informacao	Curso da área de Ciências Sociais, Comunicação e Informação
no_cine_area_geral_computacao_e_tecnologias_da_informacao_e_comunicacao_(tic)	Curso da área de Computação e TIC
no_cine_area_geral_educacao	Curso da área de Educação
no_cine_area_geral_engenharia,_producao_e_construcao	Curso da área de Engenharia, Produção e Construção
no_cine_area_geral_negocios,_administracao_e_direito	Curso da área de Negócios, Administração e Direito
no_cine_area_geral_saude_e_bem-estar	Curso da área de Saúde e Bem-estar
no_cine_area_geral_servicos	Curso da área de Serviços

Variável	**Descrição**
tp_grau_academico _bacharelado	Curso com grau de bacharelado
tp_grau_academico _licenciatura	Curso com grau de licenciatura
tp_grau_academico _tecnologico	Curso com grau tecnológico
co_regiao	Código da região geográfica da IES
co_uf	Código da Unidade da Federação
tp_categoria_administrativa	Categoria administrativa da IES (ex: pública federal, privada, etc.)
co_ies	Código da Instituição de Ensino Superior
qt_vg_total	Total de vagas ofertadas
qt_vg_total_diurno	Vagas ofertadas para cursos diurnos
qt_vg_total_noturno	Vagas ofertadas para cursos noturnos
qt_vg_nova	Vagas novas ofertadas
qt_vg_remanesc	Vagas remanescentes ofertadas
qt_inscrito_total_diurno	Total de inscrições em cursos diurnos
qt_inscrito_total_noturno	Total de inscrições em cursos noturnos
qt_insc_vg_nova	Inscrições para vagas novas
qt_insc_vg_remanesc	Inscrições para vagas remanescentes
qt_ing_fem	Quantidade de ingressantes do sexo feminino
qt_ing_masc	Quantidade de ingressantes do sexo masculino
qt_ing_diurno	Ingressantes em cursos diurnos
qt_ing_noturno	Ingressantes em cursos noturnos
qt_ing_vg_nova	Ingressantes por vaga nova
qt_ing_vestibular	Ingressantes via vestibular

`Variável`	**Descrição**
`qt_ing_enem`	Ingressantes via ENEM
`qt_ing_vg_remanesc`	Ingressantes por vaga remanescente
`qt_ing_branca, qt_ing_preta, qt_ing_parda, qt_ing_amarela, qt_ing_cornd`	Ingressantes por raça/cor
`qt_mat_fem, qt_mat_masc`	Matrículas por sexo
`qt_mat_diurno, qt_mat_noturno`	Matrículas por turno
`qt_mat_60_mais`	Matrículas de estudantes com 60 anos ou mais
`qt_mat_branca, qt_mat_preta, qt_mat_parda, qt_mat_amarela, qt_mat_indigena, qt_mat_cornd`	Matrículas por raça/cor
`qt_conc_fem, qt_conc_masc`	Concluintes por sexo
`qt_conc_diurno, qt_conc_noturno`	Concluintes por turno
`qt_conc_branca, qt_conc_preta, qt_conc_parda, qt_conc_cornd`	Concluintes por raça/cor
`qt_ing_nacbras, qt_mat_nacbras, qt_mat_nacestrang, qt_conc_nacbras`	Nacionalidade dos estudantes (brasileira ou estrangeira)
`qt_aluno_deficiente, qt_ing_deficiente`	Alunos/ingressantes com deficiência

Variável	**Descrição**
qt_ing_reserva_vaga, qt_ing_rvredepublica, qt_ing_rvetnico, qt_ing_rvsocial_rf	Ingressantes por sistema de cotas
qt_mat_reserva_vaga, qt_mat_rvredepublica, qt_mat_rvetnico, qt_mat_rvpdef, qt_mat_rvsocial_rf	Matrículas por tipo de cota
qt_conc_reserva_vaga, qt_conc_rvredepublica, qt_conc_rvetnico, qt_conc_rvsocial_rf	Concluintes por tipo de cota
qt_sit_trancada, qt_sit_desvinculado, qt_sit_transferido	Situação de evasão ou interrupção do curso
qt_ing_procescpublica, qt_ing_procescprivada	Forma de ingresso (processo seletivo público ou privado)
qt_mat_procescpublica, qt_mat_procescprivada	Tipo de processo seletivo para matrícula
qt_conc_procescpublica, qt_conc_procescprivada	Tipo de processo seletivo dos concluintes
qt_apoio_social, qt_ing_apoio_social, qt_conc_apoio_social	Apoio social recebido pelos estudantes
qt_ativ_extracurricular, qt_ing_ativ_extracurricular, qt_mat_ativ_extracurricular, qt_conc_ativ_extracurricular	Participação em atividades extracurriculares

Variável	**Descrição**
co_curso	Código do curso
no_curso	Nome do curso
enade_faixa	Faixa de conceito do ENADE (1 a 5)
cpc_faixa	Faixa do Conceito Preliminar de Curso (1 a 5)