INSTITUTO FEDERAL GOIANO - CAMPUS MORRINHOS
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

GUILHERME CORREIA DUTRA

MORRINHOS - GO
2025
GUILHERME CORREIA DUTRA

CLUSTERIZAÇÃO DE DADOS EM DIFERENTES AMBIENTES: UMA ANÁLISE DE CUSTO, TEMPO E QUALIDADE

Monografia apresentada ao Curso Bacharelado em Ciência da Computação do Instituto Federal Goiano - Campus Morrinhos, como requisito parcial para obtenção de título de Bacharel em Ciência da Computação.
Área de Concentração: Sistemas de Computação.
Orientador: Me. Felipe Nunes Gaia.
Coorientador: Dr. Rodrigo Elias Francisco.

MORRINHOS – GO
2025

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema Integrado de Bibliotecas do IF Goiano - SIBi

Dutra, Guilherme Correia
C824c CLUSTERIZAÇÃO DE DADOS EM DIFERENTES AMBIENTES:: UMA ANÁLISE DE CUSTO, TEMPO E QUALIDADE / Guilherme Correia Dutra. Morrinhos 2025.

76f. il.

Orientador: Prof. Me. Felipe Nunes Gaia.
Coorientador: Prof. Dr. Rodrigo Elias Francisco.
Tcc (Bacharel) - Instituto Federal Goiano, curso de 0419204 - [MO.GRAD] Bacharelado em Ciência da Computação - Morrinhos (Campus Morrinhos).
1. Clusterização. 2. Computação em Nuvem. 3. Ambiente local. 4. ENADE. 5. Ciência de Dados. I. Título.

Ministério da Educação
Secretaria de Educação Profissional e Tecnológica
Instituto Federal Goiano Campus Morrinhos
Curso Bacharelado em Ciências da Computação
Coordenação de Trabalho de Curso

ATA DE DEFESA DA BANCA DE EXAME DE TRABALHO DE CURSO POR VIDEOCONFERÊNCIA

Ao **primeiro** dia do mês de **julho** de **2025**, às **14:00** horas, foi realizada a Banca de Exame, de forma remota, para a apresentação pública e defesa do trabalho de curso do discente **Guilherme Correia Dutra**, intitulado **"Clusterização de Dados em Diferentes Ambientes: Uma Análise de Custo, Tempo e Qualidade"**, como requisito necessário para a conclusão do curso.

A Banca de Exame foi constituída pelos membros: **Felipe Nunes Gaia, Marcel da Silva Melo**, **Gabriel Coutinho Sousa Ferreira**. Após a análise, emitiram o seguinte resultado:

1 - ( ) Aprovado

2 - (X) Aprovado com ressalva
(A Banca Examinadora deve definir as exigências a serem cumpridas pelo aluno na revisão, ficando o orientador responsável pela verificação do cumprimento das mesmas.)
Observações: Mediante correções apontadas pela banca.

3 - ( ) Reprovado com o seguinte parecer: _____
_____

Morrinhos-GO, 01 de julho de 2025.

Por ser verdade firmamos a presente:

gov.br Documento assinado digitalmente
FELIPE NUNES GAIA
Data: 01/07/2025 17:31:23-0300
Verifique em https://validar.iti.gov.br

_____
Felipe Nunes Gaia (Presidente da banca)

gov.br Documento assinado digitalmente
MARCEL DA SILVA MELO
Data: 02/07/2025 13:32:02-0300
Verifique em https://validar.iti.gov.br

_____
Marcel da Silva Melo (Membro)

gov.br Documento assinado digitalmente
RODRIGO ELIAS FRANCISCO
Data: 02/07/2025 13:24:50-0300
Verifique em https://validar.iti.gov.br

_____
Rodrigo Elias Francisco (Suplente)

gov.br Documento assinado digitalmente
GABRIEL COUTINHO SOUSA FERREIRA
Data: 02/07/2025 13:52:18-0300
Verifique em https://validar.iti.gov.br

_____
Gabriel Coutinho Sousa Ferreira (Membro)

gov.br Documento assinado digitalmente
PAULO VICTOR DOS SANTOS
Data: 02/07/2025 14:22:01-0300
Verifique em https://validar.iti.gov.br

_____
Paulo Victor dos Santos (Suplente)

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO

PARA DISPONIBILIZAR PRODUÇÕES TÉCNICO-CIENTÍFICAS NO REPOSITÓRIO INSTITUCIONAL DO IF GOIANO

Com base no disposto na Lei Federal nº 9.610, de 19 de fevereiro de 1998, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia Goiano a disponibilizar gratuitamente o documento em formato digital no Repositório Institucional do IF Goiano (RIIF Goiano), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IF Goiano.

IDENTIFICAÇÃO DA PRODUÇÃO TÉCNICO-CIENTÍFICA

- ☐ Tese (doutorado)
- ☐ Dissertação (mestrado)
- ☐ Monografia (especialização)
- ☑ TCC (graduação)
- ☐ Artigo científico
- ☐ Capítulo de livro
- ☐ Livro
- ☐ Trabalho apresentado em evento
- ☐ Produto técnico e educacional - Tipo:

Nome completo do autor: Guilherme Correia Dutra

Matrícula: 2020104201940047

Título do trabalho: CLUSTERIZAÇÃO DE DADOS EM DIFERENTES AMBIENTES: UMA ANÁLISE DE CUSTO, TEMPO E QUALIDADE

RESTRIÇÕES DE ACESSO AO DOCUMENTO

Documento confidencial: ☑ Não ☐ Sim, justifique:

Informe a data que poderá ser disponibilizado no RIIF Goiano: ☐/☐/☐

O documento está sujeito a registro de patente? ☐ Sim ☑ Não

O documento pode vir a ser publicado como livro? ☑ Sim ☐ Não

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O(a) referido(a) autor(a) declara:

• Que o documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;

• Que obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autoria, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia Goiano os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;

• Que cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia Goiano.

Morrinhos, GO — Local

03/07/2025 — Data

gov.br Documento assinado digitalmente
GUILHERME CORREIA DUTRA
Data: 03/07/2025 22:06:51-0300
Verifique em https://validar.iti.gov.br

_____
Assinatura do autor e/ou detentor dos direitos autorais

Ciente e de acordo: _____

gov.br Documento assinado digitalmente
FELIPE NUNES GAIA
Data: 03/07/2025 22:24:38-0300
Verifique em https://validar.iti.gov.br

Assinatura do(a) orientador(a)

DEDICATÓRIA

Dedico este trabalho à minha mãe Raquel Pereira Dutra, ao meu pai Márcio Correia da Silva e ao meu irmão Fernando Correia Dutra por todo amor, carinho, incentivo e paciência comigo ao longo do desenvolvimento das minhas pesquisas e estudos.

AGRADECIMENTOS

Agradeço ao meu orientador Me. Felipe Nunes Gaia e ao meu co-orientador Dr. Rodrigo Elias Francisco por todo apoio, paciência e conselhos durante o desenvolvimento deste trabalho. A empresa Lembry, obrigado por financiar os créditos na AWS que possibilitaram custear o Amazon SageMaker. Aos meus amigos da Residência da Amoreira, obrigado por estarem ao meu lado, e às tias da cozinha, que sempre perguntavam sobre meu trabalho com tanto carinho. A todos que contribuíram para esta conquista, meu muito obrigado.

RESUMO

Este trabalho tem como objetivo analisar o desempenho da aplicação de algoritmos de clusterização em dois ambientes computacionais distintos: local e em nuvem. A pesquisa foi desenvolvida com enfoque quantitativo e experimental, buscando mensurar e comparar o desempenho de quatro algoritmos – KMeans, MiniBatchKMeans, DBSCAN e HDBSCAN – com base em métricas como tempo de execução, custo operacional e qualidade dos agrupamentos. Os dados utilizados foram extraídos do Exame Nacional de Desempenho dos Estudantes (ENADE) de 2022, especificamente das questões relacionadas às percepções dos estudantes sobre o impacto da pandemia em sua formação acadêmica. O tratamento dos dados incluiu limpeza, normalização e estruturação para análise em ambos os ambientes. A implementação foi realizada com ferramentas como Python, PostgreSQL, Visual Studio Code e Amazon SageMaker, mantendo os parâmetros consistentes em todos os experimentos e a análise de dados foi feita utilizando a ferramenta Metabase. A avaliação da qualidade dos clusters foi baseada principalmente no índice de Silhouette, complementada por análise de complexidade computacional e tempo de execução. Os resultados demonstraram que o ambiente em nuvem apresentou melhor desempenho em termos de tempo, com destaque para o MiniBatchKMeans, enquanto o ambiente local foi mais econômico em termos de custo total. Não foram observadas diferenças significativas na qualidade dos agrupamentos entre os ambientes. Conclui-se que a escolha entre ambientes locais e em nuvem deve considerar o perfil do projeto, o volume de dados, a urgência de processamento e os recursos disponíveis. O estudo contribui para a compreensão prática das vantagens e limitações de cada infraestrutura, oferecendo subsídios para decisões técnicas e estratégicas na área de ciência de dados, especialmente em contextos educacionais. O trabalho também reforça a importância da replicabilidade, da automação de testes e da escolha criteriosa de métricas de avaliação para garantir resultados confiáveis em experimentos com dados reais.

Palavras-chave: Clusterização. Computação em nuvem. Ambiente local. ENADE. Silhouette Score. Ciência de dados

ABSTRACT

This study aims to analyze the efficiency of applying *clustering* algorithms in two distinct computational environments: local and cloud-based. The research adopts a quantitative and experimental approach, seeking to measure and compare the performance of four algorithms — KMeans, MiniBatchKMeans, DBSCAN, and HDBSCAN — based on metrics such as execution time, operational cost, and *clustering* quality. The dataset was extracted from the 2022 National Student Performance Exam (ENADE), specifically from questions related to students' perceptions of the pandemic's impact on their academic experience. Data processing included cleaning, normalization, and structuring for analysis in both environments. Implementation was carried out using tools such as Python, PostgreSQL, Visual Studio Code, and Amazon SageMaker, maintaining consistent parameters across all experiments. The quality of the clusters was primarily assessed using the Silhouette index, along with computational complexity and processing time analysis. Results showed that the cloud environment outperformed in terms of execution time, with MiniBatchKMeans standing out, while the local environment was more economical in terms of total cost. No significant differences were observed in the quality of *clustering* between the two environments. It is concluded that the choice between local and cloud computing environments should consider the project profile, data volume, processing urgency, and available resources. This research contributes to the practical understanding of the advantages and limitations of each infrastructure, providing insights for technical and strategic decision-making in the data science field, especially in educational contexts. It also emphasizes the importance of replicability, test automation, and careful metric selection to ensure reliable results in real-world data experiments.

Keywords: Clustering. Cloud computing. Local environment. ENADE. Silhouette Score. Data science.

SUMÁRIO

1 INTRODUÇÃO

A clusterização de dados, ou agrupamento, tem sido estudada como uma técnica essencial na análise de grandes volumes de dados, permitindo a identificação de padrões e estruturas ocultas em diferentes contextos. Diversos pesquisadores têm investigado a aplicação de algoritmos de clusterização em áreas como marketing, saúde, finanças e educação, com o objetivo de explorar formas mais eficazes de aplicação dessas técnicas na segmentação e interpretação dos dados. Com o advento da computação em nuvem, surgiu a possibilidade de escolher o ambiente mais adequado para a execução desses algoritmos, considerando três variáveis principais de interesse: tempo de execução, custo operacional e qualidade dos agrupamentos. Nesse contexto, torna-se relevante avaliar o impacto da infraestrutura computacional na execução desses algoritmos, por meio da comparação entre ambientes locais e soluções baseadas em serviços de nuvem, como discutido por Vysala (2020).

Estudos demonstraram que a escolha do ambiente de processamento pode influenciar significativamente tanto a qualidade dos resultados quanto a viabilidade da aplicação prática dos modelos. O ambiente local, embora frequentemente limitado por restrições de hardware, apresenta vantagens em termos de controle e segurança dos dados. Por outro lado, ambientes baseados em nuvem destacam-se pela escalabilidade e flexibilidade operacional. Assim, o presente estudo analisa comparativamente os algoritmos *K-Means, MiniBatch K-Means, DBSCAN* e *HDBSCAN*, com o objetivo de identificar quais abordagens apresentam melhores resultados sob diferentes configurações. A investigação busca contribuir para um entendimento mais amplo sobre as condições ideais para a aplicação dessas técnicas, considerando as três variáveis de interesse citadas (Emmons, 2016).

A análise comparativa foi realizada com base na seguinte problemática: qual ambiente computacional (local ou nuvem) proporciona melhores resultados na clusterização de dados, considerando tempo, custo e qualidade? Para responder essa questão, foram feitos experimentos práticos, utilizando dados do questionário do ENADE 2022, que reúne respostas sobre a percepção dos estudantes de graduação dos impactos da pandemia no processo de aprendizagem.

A pergunta norteadora desta pesquisa foi: qual ambiente oferece o melhor equilíbrio entre eficiência computacional (tempo e custo) e qualidade dos resultados obtidos na clusterização desses dados? Considerando essa pergunta, algumas hipóteses foram formuladas. A primeira hipótese (H1) sustenta que a execução dos algoritmos em nuvem apresenta menor tempo, devido à capacidade de escalabilidade dos serviços de computação distribuída. A segunda hipótese (H2) indica que o custo operacional tende a ser maior na nuvem, em função da cobrança por uso de recursos computacionais. A terceira hipótese (H3) sugere que a clusterização realizada localmente pode apresentar limitações de qualidade devido às restrições de hardware, afetando a eficiência dos modelos. Essas hipóteses orientaram a condução da pesquisa, permitindo uma avaliação comparativa entre os dois ambientes de processamento.

O objetivo geral deste estudo foi analisar o desempenho da clusterização de dados em ambientes locais e na nuvem, considerando métricas de tempo de execução, custo operacional e qualidade dos agrupamentos. Para alcançar esse propósito, três objetivos específicos foram estabelecidos:

- Medir o tempo de execução dos algoritmos de clusterização *K-Means, MiniBatch K-Means, DBSCAN* e *HDBSCAN* em ambientes locais e na nuvem, identificando possíveis gargalos de desempenho;
- Calcular o custo operacional entre as execuções em ambos os ambientes, analisando a viabilidade econômica de cada solução;
- Quantificar a qualidade dos agrupamentos gerados pelos algoritmos nos mesmos ambientes, utilizando diferentes métricas.

Este estudo apresenta grande relevância para a área da ciência de dados e da computação, pois possibilita uma compreensão mais aprofundada sobre as vantagens e desafios da clusterização em diferentes infraestruturas. A pesquisa fornece informações úteis para profissionais que necessitam escolher o ambiente adequado para processar grandes volumes de dados, equilibrando desempenho e custos. Além disso, contribui para a literatura acadêmica ao trazer um estudo comparativo baseado em experimentos reais, utilizando dados educacionais para exemplificar aplicações práticas das técnicas de clusterização. Dessa forma, os resultados obtidos auxiliam no avanço das pesquisas sobre desempenho computacional aplicado à análise de dados.

Este trabalho está estruturado em seis capítulos organizados de forma lógica e sequencial. Após esta introdução, o **Capítulo 2** apresenta a **revisão da literatura**, utilizando estudos que abordam os algoritmos de clusterização (agrupamento), computação em ambiente local e na nuvem, as métricas para avaliação das variáveis, além de trabalhos relacionados que envolvem todos estes temas. O **Capítulo 3** descreve a **metodologia** destacando a abordagem experimental adotada, as ferramentas utilizadas, os dados provenientes do ENADE 2022 e os critérios de avaliação. O **Capítulo 4 desenvolve** as etapas práticas do estudo, incluindo a coleta e preparação dos dados, a implementação dos algoritmos, a aplicação das métricas, a execução dos testes e os desafios enfrentados. O **Capítulo 5 analisa os resultados** obtidos, discutidos à luz dos objetivos da pesquisa, a fim de cumpri-los. Por fim, o **Capítulo 6 conclui** o estudo retomando os principais achados, avaliando a contribuição da pesquisa, reconhecendo suas limitações e propondo direções para trabalhos futuros.

2 REVISÃO DA LITERATURA

Este capítulo revisa os trabalhos importantes para este estudo. A metodologia baseou-se em uma revisão de literatura qualitativa utilizando o período de dois mil e vinte e um a dois mil e vinte e cinco. As fontes de informações foram extraídas de bases de dados reconhecidas, como Lilacs, Periódicos Capes, Google Acadêmico e Scielo. O critério de inclusão adotado consistiu na seleção de artigos e estudos que abordaram a clusterização de dados em ambientes locais e na nuvem, considerando métricas de desempenho e custo. Como critério de exclusão, foram descartadas publicações que não apresentaram comparações práticas entre os dois ambientes de processamento ou que se limitaram a aspectos teóricos sem experimentação.

2.1 Clusterização de dados

A clusterização de dados é explorada como método eficaz para organizar grandes volumes de informações, permitindo a identificação de padrões e comportamentos ocultos. Técnicas de agrupamento são essenciais em diversas áreas, como inteligência artificial, análise de mercado e bioinformática, facilitando a interpretação de conjuntos de dados complexos. Diferentes abordagens foram propostas para otimizar o desempenho desses algoritmos, buscando equilibrar tempo de processamento e qualidade dos agrupamentos. Com o avanço da tecnologia, pesquisadores passaram a comparar métodos tradicionais com abordagens otimizadas, avaliando métricas de desempenho e adaptabilidade em diferentes cenários computacionais (Silva, Pereira e Saqui, 2023).

A qualidade dos algoritmos de clusterização depende de fatores como a escolha do número de clusters, a métrica de distância utilizada e a natureza dos dados analisados. Métodos baseados em centróides, que agrupam os dados ao redor de pontos centrais representativos de cada cluster, como K-Means e MiniBatch K-Means, apresentam alta velocidade de execução, sendo aplicados em problemas que demandam escalabilidade. Em contrapartida, os baseados em densidade, que identificam clusters através da concentração de pontos em regiões densas do espaço de dados, como o DBSCAN e HDBSCAN, são capazes de identificar

padrões não lineares e lidar com ruídos, tornando-se alternativas viáveis para conjuntos de dados heterogêneos.

Pesquisadores têm buscado aprimorá-los, combinando diferentes estratégias para aumentar sua eficiência computacional sem comprometer a precisão dos agrupamentos. A utilização de técnicas de otimização em clusterização representa objeto de estudo em diferentes contextos. Modelos híbridos, que combinam algoritmos tradicionais com métodos baseados em inteligência artificial, foram propostos para melhorar a adaptabilidade das técnicas a diferentes domínios. Estratégias como ajuste dinâmico de parâmetros e redução de dimensionalidade foram aplicadas para minimizar o tempo de processamento e maximizar a separação dos grupos formados. Estudos comparativos demonstraram que a combinação de algoritmos pode resultar em agrupamentos mais coerentes e representativos, promovendo ganhos significativos na análise de dados em larga escala.

O desempenho dos algoritmos de clusterização também está relacionado ao ambiente computacional utilizado para sua execução. A escolha entre processamento local e computação em nuvem influencia diretamente no tempo de execução e no custo operacional das análises. Infraestruturas em nuvem oferecem escalabilidade e capacidade de processamento distribuído, possibilitando a aplicação de técnicas avançadas em grandes conjuntos de dados. Em contrapartida, soluções locais permitem maior controle sobre os dados e reduzem a dependência de provedores externos. A decisão entre essas abordagens requer análise de fatores como volume de dados, complexidade dos algoritmos e restrições orçamentárias.

Diferentes métricas foram desenvolvidas para avaliar a qualidade dos clusters gerados, sendo a métrica de Silhouette uma das mais utilizadas para medir a separação e a coesão dos agrupamentos. Outros indicadores, como o coeficiente de Dunn e a variância intra-cluster, também são empregados para validar a eficácia das técnicas aplicadas. O desenvolvimento de novas métricas e a adaptação das existentes a contextos específicos são áreas em constante evolução, visando tornar a clusterização de dados uma ferramenta ainda mais precisa e eficiente para análise de informações complexas.

2.2 Algoritmos de clusterização de dados

A segmentação e o agrupamento de informações constituem técnicas fundamentais em diferentes contextos, permitindo a identificação de padrões e tendências em grandes volumes de dados. A evolução das técnicas de agrupamento impulsionou pesquisas voltadas à otimização de algoritmos e à melhoria da eficiência computacional. Métodos como K-Means, MiniBatch K-Means, DBSCAN e HDBSCAN ganharam destaque devido à sua aplicabilidade em diferentes cenários, variando em termos de tempo de processamento e qualidade dos agrupamentos gerados. Silva, Pereira e Saqui (2023) destacaram a importância da clusterização na recomendação de conteúdos personalizados, utilizando modelos híbridos para aprimorar a segmentação e melhorar a experiência do usuário em sistemas de recomendação.

A escolha do algoritmo de clusterização influencia diretamente na eficiência da análise, sendo necessário considerar a natureza dos dados e a complexidade computacional envolvida. Oliveira *et al*. (2022) analisaram diferentes algoritmos e demonstraram que a definição do número de clusters e a métrica de distância impactam na coesão e separação dos agrupamentos. Métodos baseados em particionamento, como K-Means, apresentaram eficiência em cenários com grande volume de dados, enquanto abordagens baseadas em densidade, como DBSCAN, mostraram maior flexibilidade na identificação de padrões irregulares. A adaptação dessas técnicas para diferentes domínios configura-se como estratégia utilizada para otimizar a precisão dos modelos.

A otimização dos algoritmos de clusterização vem sendo investigada para reduzir custos computacionais e aumentar a escalabilidade dos modelos em ambientes com grandes conjuntos de dados. Chicon e Telocken (2021) exploraram estratégias para minimizar o tempo de processamento e melhorar a qualidade dos agrupamentos, propondo abordagens que combinam diferentes técnicas para aprimorar a segmentação. O uso de ajustes automáticos de hiperparâmetros e a integração de modelos híbridos foram alternativas analisadas para aumentar a eficiência dos algoritmos e possibilitar sua aplicação em diferentes cenários. Essas melhorias tornaram a clusterização uma ferramenta essencial para análise de dados e mineração de conhecimento.

2.2.1 K-Means e MiniBatch K-Means

A clusterização de dados tem sido amplamente utilizada em diferentes contextos, permitindo a organização e segmentação de grandes volumes de informações. Métodos baseados em particionamento, como K-Means e MiniBatch K-Means, destacam-se por sua eficiência no agrupamento de dados escaláveis. Santos *et al*. (2022) apontaram que essas técnicas são amplamente empregadas em processamento de linguagem natural, auxiliando na identificação de padrões em textos não estruturados. A simplicidade da abordagem baseada em centróides favorece sua aplicação em cenários que exigem rápida análise e tomada de decisão.

A adaptação dos algoritmos de clusterização mostra-se essencial para lidar com o crescimento dos volumes de dados e a necessidade de processamento eficiente. Oliveira *et al*. (2022) analisaram a utilização do MiniBatch K-Means como uma alternativa ao K-Means convencional, destacando sua capacidade de reduzir a carga computacional ao processar subconjuntos dos dados a cada iteração. Essa abordagem mantém a qualidade dos agrupamentos enquanto melhora o tempo de execução, tornando-se uma opção viável para cenários que demandam escalabilidade.

A definição do número ideal de clusters constitui um dos principais desafios na aplicação dessas técnicas. Santos *et al*. (2022) destacaram a importância de métricas de validação como o coeficiente de Silhouette e o método do cotovelo para determinar a quantidade mais adequada de grupos. Essas métricas auxiliam na escolha do modelo mais representativo, evitando a formação de clusters incoerentes. Estratégias baseadas em inicialização otimizada, como K-Means++, são adotadas para reduzir problemas relacionados à má distribuição dos centróides, contribuindo para a melhoria da estabilidade dos agrupamentos.

A eficiência computacional desses métodos é um fator determinante na escolha do algoritmo mais adequado. Oliveira *et al*. (2022) ressaltaram que a implementação do MiniBatch K-Means proporciona ganhos significativos na execução de tarefas que envolvem grandes quantidades de dados. Esse fator justifica sua aplicação em ambientes que exigem alta performance e resposta rápida, como sistemas de recomendação e análise de comportamento do usuário.

2.2.2 DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

A clusterização baseada em densidade tem sido amplamente utilizada para segmentação de dados em diversos contextos, permitindo identificar padrões ocultos em conjuntos de informações complexas. O DBSCAN, um dos algoritmos mais aplicados nesse método, apresenta a capacidade de detectar clusters de formas arbitrárias e lidar com ruídos de maneira eficiente. Gonçalves e Santos (2024) demonstram a aplicabilidade dessa técnica na identificação de *hotspots* de acidentes de trabalho no Brasil, utilizando análise espacial para reconhecer regiões de maior incidência. A flexibilidade desse modelo o torna adequado para dados heterogêneos e distribuídos de maneira irregular.

A principal característica do DBSCAN é sua independência na definição prévia do número de clusters, diferindo de métodos baseados em particionamento. Almeida, Silva e Silva (2024) analisaram a utilização desse algoritmo no estudo da mobilidade urbana, integrando dados de transporte coletivo com informações criminais e de relevo. O modelo mostrou-se eficaz ao agrupar regiões com padrões semelhantes de deslocamento e segurança, evidenciando sua utilidade na análise de dados geoespaciais. A definição adequada dos parâmetros de densidade tem impacto direto na qualidade dos agrupamentos gerados, exigindo ajustes específicos conforme o tipo de dado analisado.

A robustez desse método permite sua aplicação em áreas que demandam identificação de padrões complexos sem a necessidade de rótulos prévios. Gonçalves e Santos (2024) destacam que a capacidade de lidar com ruídos torna o algoritmo adequado para dados que apresentam *outliers*, característica comum em estudos que envolvem eventos espaciais. Esse fator contribui para sua adoção em problemas que exigem segmentação dinâmica, como análise ambiental, epidemiologia e segurança pública. A capacidade de adaptação do modelo a diferentes domínios demonstra sua versatilidade e relevância em cenários de análise preditiva e tomada de decisão.

2.2.3 HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise)

A clusterização baseada em densidade tem sido amplamente utilizada para a segmentação de dados, especialmente em cenários nos quais os padrões de agrupamento não são definidos de maneira clara. O HDBSCAN surge como uma evolução do DBSCAN, permitindo a identificação de *clusters* hierárquicos com maior flexibilidade na definição de densidade. Gonçalves e Santos (2024) aplicaram essa técnica para mapear *hotspots* de acidentes de trabalho no Brasil, demonstrando sua capacidade de identificar regiões de maior incidência com maior precisão do que métodos convencionais. A capacidade de ajustar automaticamente a densidade dos *clusters* torna essa abordagem vantajosa para conjuntos de dados com variações significativas de distribuição.

A adaptabilidade do HDBSCAN permite sua aplicação em domínios diversos, incluindo educação e análise de comportamento computacional. Melo, Pessoa e Fernandes (2024) analisaram sua utilização na clusterização de soluções de exercícios de programação, identificando padrões de resposta entre estudantes. A flexibilidade do modelo possibilitou a segmentação dos perfis de aprendizado sem a necessidade de um número fixo de grupos, facilitando a análise das estratégias utilizadas pelos alunos na resolução dos problemas. A aplicação dessa abordagem em ambientes acadêmicos evidencia seu potencial na personalização de metodologias educacionais baseadas em aprendizado de máquina.

A principal vantagem dessa técnica em relação a outros métodos baseados em densidade está na eliminação da necessidade de definir previamente um valor fixo para a vizinhança mínima de um cluster. Gonçalves e Santos (2024) destacaram que essa característica permite uma melhor adaptação a dados ruidosos e distribuições não homogêneas, tornando-a uma opção robusta para análise de eventos espaciais. A capacidade de diferenciar automaticamente regiões densas de áreas dispersas sem intervenção manual reforça sua aplicabilidade em estudos de mobilidade urbana, epidemiologia e previsão de padrões em dados dinâmicos.

2.3 Avaliação da qualidade da clusterização

A avaliação da qualidade da clusterização é um aspecto fundamental para garantir a coerência e a representatividade dos agrupamentos gerados. A seleção de métricas adequadas permite validar a qualidade dos algoritmos e determinar se os *clusters* formados refletem padrões significativos dentro dos dados analisados. Silva *et al*. (2021) exploraram a aplicação da mineração de dados para avaliar a complexidade de processos judiciais, utilizando técnicas de agrupamento para identificar padrões processuais. A análise da qualidade dos *clusters* revelou que a definição adequada dos parâmetros influencia diretamente a precisão dos modelos.

O uso de métricas quantitativas é essencial para verificar a separação e a compactação dos *clusters* gerados. Os autores Freire, Bastos Filho e Rabbani (2022) destacaram a relevância da aplicação de técnicas de aglomeração de dados para a análise de programas de extensão tecnológica, demonstrando como a escolha das métricas impacta a confiabilidade dos agrupamentos. Métodos como o coeficiente de Silhouette e a variância intra-cluster são frequentemente utilizados para avaliar o grau de separação entre os grupos formados. A adoção deles permite ajustar os parâmetros dos algoritmos, tornando os resultados mais consistentes.

A escolha da métrica de validação depende do tipo de dado e da finalidade do estudo. Ferreira (2024) analisou o impacto da clusterização no aprendizado de máquina, ressaltando que a utilização de indicadores como o índice de Dunn e o Davies-Bouldin possibilita uma avaliação mais precisa da qualidade dos *clusters*. A interpretação deles auxilia na definição do número ótimo de agrupamentos, evitando problemas como sobreajuste ou subajuste dos modelos. A integração de métricas em processos automatizados tem permitido uma validação mais eficiente dos agrupamentos, otimizando o desempenho das técnicas aplicadas.

2.3.1 Métrica de Silhouette Score

A avaliação da qualidade da clusterização requer métricas que possibilitem verificar a coerência e a separação dos agrupamentos gerados. A métrica de *Silhouette Score* é amplamente utilizada para medir a proximidade entre os pontos dentro de um cluster e sua distância em relação aos demais grupos. Silva *et al*. (2021) aplicaram essa métrica na análise de processos judiciais, demonstrando que

sua utilização permite validar a qualidade dos agrupamentos e otimizar a configuração dos parâmetros dos algoritmos de clusterização. A capacidade de medir a compactação e a separação dos grupos faz desse método uma ferramenta essencial para a validação de modelos baseados em aprendizado de máquina.

A interpretação dos valores obtidos por essa métrica auxilia na definição do número ideal de *clusters* e na identificação de padrões nos dados analisados. Freire, Bastos Filho e Rabbani (2022) utilizam a métrica para avaliar a qualidade de técnicas de aglomeração aplicadas a um programa de extensão tecnológica, evidenciando que valores próximos a um indicam boa separação entre *clusters*, enquanto valores próximos a zero sugerem sobreposição entre grupos. A análise desses resultados permite o refinamento dos algoritmos, tornando os agrupamentos mais representativos. A aplicação desta métrica em diferentes contextos possibilita a adaptação dos modelos a diferentes domínios de estudo.

A precisão da métrica pode ser afetada pela escolha dos algoritmos e das métricas de distância utilizadas na clusterização. Ferreira (2024) destaca que a sensibilidade do *Silhouette Score* varia conforme a distribuição dos dados e o método empregado para agrupamento. O uso dessa métrica em combinação com outros indicadores, como o índice de Davies-Bouldin, permite uma avaliação mais robusta da qualidade dos *clusters* gerados. A integração desses métodos **é aplicada** para garantir que os agrupamentos reflitam padrões reais, minimizando a ocorrência de erros na segmentação dos dados.

2.4 Computação em ambiente local e em nuvem

A computação em ambiente local e na nuvem tem se destacado nas últimas décadas como uma tecnologia que oferece vantagens significativas em termos de desempenho e segurança. A computação em nuvem, que envolve o armazenamento e processamento de dados em servidores remotos, permite maior flexibilidade e escalabilidade, sendo vantajosa para empresas de diferentes portes (Lorenzi, Gréin e Corcini, 2022). Em comparação com a computação local, onde os dados são processados em servidores físicos dentro da empresa, a nuvem oferece recursos mais dinâmicos e acessíveis, facilitando o gerenciamento e o acesso remoto a dados.

Contudo, a segurança continua sendo uma preocupação central na computação em nuvem. Silva (2023) destaca que, embora a nuvem ofereça benefícios em termos de custo e escalabilidade, ela também exige maior atenção à proteção das informações, especialmente considerando as vulnerabilidades dos servidores remotos. A utilização de criptografia, autenticação de múltiplos fatores e outras tecnologias de segurança é fundamental para garantir a integridade e a confidencialidade dos dados armazenados.

A análise de desempenho entre esses dois modelos também é relevante. Schussler *et al*. (2023) conduz um estudo comparando o desempenho entre a computação em nuvem e o servidor local no contexto do método Fletcher, evidenciando que a computação em nuvem pode ser mais eficiente em cenários específicos, especialmente quando se busca otimização e recursos computacionais poderosos sem a necessidade de investimentos pesados em infraestrutura. A escolha entre esses dois modelos depende das necessidades específicas de cada organização, considerando fatores como custo, segurança, e requisitos de desempenho.

2.4.1 Execução local e ferramentas utilizadas

A execução local de programas paralelos em ambientes de computação torna-se cada vez mais desafiadora com o aumento das necessidades de processamento de dados em larga escala. Em contraste, a computação em nuvem oferece uma alternativa eficiente, proporcionando maior flexibilidade e recursos de processamento sem a necessidade de infraestrutura própria (Lorenzi, Gréin e Corcini, 2022). A execução local, que depende de servidores dedicados ou sistemas internos, muitas vezes enfrenta limitações em termos de desempenho e escalabilidade, o que pode ser um obstáculo significativo em tarefas que exigem grandes volumes de dados.

É importante destacar que a computação em nuvem oferece um ambiente escalável, permitindo que os recursos sejam ajustados conforme a demanda. Maliszewski *et al*. (2021) abordam a utilização de ambientes de nuvem privada para teste e desenvolvimento de programas paralelos, apontando que essa infraestrutura proporciona um controle mais rigoroso sobre o acesso e a segurança dos dados, ao mesmo tempo que possibilita o uso de recursos de processamento robustos. Em

ambientes locais, a capacidade de realizar testes e experimentos em grande escala pode ser limitada devido à infraestrutura disponível, que nem sempre é adequada para demandas de alto desempenho.

A combinação de ferramentas especializadas e plataformas de nuvem para a execução de programas paralelos resulta em ganhos significativos de desempenho e redução de custos. A nuvem facilita o desenvolvimento e o teste de programas paralelos, pois permite que recursos computacionais sejam alocados de maneira dinâmica e eficiente (Lorenzi, Gréin e Corcini, 2022). Essas plataformas oferecem o benefício de ser mais acessíveis e econômicas, atendendo às necessidades de empresas e pesquisadores que lidam com tarefas computacionais exigentes.

2.4.2 Execução em nuvem: Amazon SageMaker

A computação em nuvem tem se consolidado como uma solução eficaz para a execução de aplicações complexas, permitindo o processamento e a análise de grandes volumes de dados de maneira escalável e acessível. A Amazon SageMaker é uma das ferramentas mais destacadas nesse contexto, oferecendo recursos avançados para o desenvolvimento e a implementação de modelos de *machine learning*. Mejia e Curasma (2023) abordam o uso da nuvem para sistemas recomendadores em plataformas de programação competitiva, destacando a importância da nuvem na facilitação do treinamento e da implementação desses modelos, além de sua capacidade de adaptação a diferentes necessidades computacionais.

Na área de logística, o uso de ferramentas como o Amazon Forecast tem mostrado grande potencial para a previsão de demanda e otimização de processos. Freitas, Magalhães e Costa (2024) demonstram que o Amazon Forecast pode gerar previsões mais precisas e ajudar empresas a se prepararem melhor para flutuações no mercado, ajustando seus recursos e estoques conforme necessário. A capacidade de integrar essas ferramentas em sistemas baseados em nuvem permite uma maior agilidade e redução de custos, pois elimina a necessidade de infraestrutura física robusta para o processamento de dados em grande escala.

A execução em nuvem, facilitada por plataformas como o Amazon SageMaker, tem transformado a forma como as empresas e pesquisadores abordam o *machine learning* e a previsão de dados, trazendo benefícios significativos em

termos de flexibilidade, escalabilidade e desempenho. Essas ferramentas são particularmente valiosas para áreas como logística e recomendação de sistemas, onde a análise de grandes volumes de dados é essencial para o sucesso das operações.

2.5 Comparação de custo, tempo e eficiência

A comparação entre os custos, o tempo e a eficiência dos serviços de computação em nuvem representa um tema central para diversas análises nos últimos anos. No contexto da utilização de *containers* em plataformas como a AWS, Cavalcante, Boeres e Rebello (2024) exploram a eficiência dos serviços de conteinerização, destacando como essas ferramentas podem reduzir custos operacionais e otimizar o tempo de execução de aplicações. A flexibilidade e escalabilidade dos serviços de nuvem permitem que empresas ajustem seus recursos conforme necessário, proporcionando um gerenciamento mais eficiente do tempo e do orçamento.

A eficiência energética também se tornou um aspecto importante no debate sobre a computação em nuvem. Reis *et al*. (2024) propõem uma classificação para rotular a eficiência energética na computação em nuvem verde, com o objetivo de diminuir o impacto ambiental e os custos operacionais relacionados ao consumo de energia. O uso de *data centers* sustentáveis e a adoção de práticas eficientes em termos de consumo de energia podem reduzir consideravelmente os custos de operação, além de contribuir para a preservação ambiental. A crescente demanda por soluções mais verdes exige um olhar atento para a escolha de fornecedores de serviços de nuvem que adotem práticas sustentáveis.

A implementação de sistemas ERP (Enterprise Resource Planning), que integram e gerenciam os principais processos de negócio de uma organização, em nuvem também tem ganhado destaque nas pequenas e médias empresas. Júnior e Santos (2022) discutem os benefícios da adoção de soluções baseadas em nuvem para a gestão de processos empresariais, destacando os comparativos de segurança, eficiência e benefícios relacionados a esses sistemas. A segurança da informação e a facilidade de integração são fatores essenciais para a decisão de migrar para a nuvem, especialmente quando se considera a redução de custos com infraestrutura e a melhoria na eficiência operacional.

2.5.1 Tempo de execução dos modelos

O tempo de execução dos modelos computacionais é uma das principais métricas para avaliar a eficácia de diferentes abordagens em ambientes de computação em nuvem e em *containers*. Horchulhack *et al*. (2022) investigam a detecção de *overbooking* em aplicações baseadas em Docker utilizando técnicas de aprendizagem de máquina. A detecção precoce de *overbooking*, que ocorre quando um recurso computacional é alocado para mais usuários do que sua capacidade real, pode reduzir significativamente o tempo de execução de aplicações, evitando lentidão e falhas no sistema. Esse tipo de otimização é particularmente relevante em ambientes de nuvem, onde a utilização eficiente dos recursos é essencial para garantir o bom desempenho e minimizar custos.

Além disso, a avaliação de modelos também envolve a medição da qualidade dos algoritmos de agrupamento, que é fundamental para determinar o desempenho das operações. Jaskowiak, Costa e Campello (2020) discutem o uso da área sob a curva ROC (*Receiver Operating Characteristic*) como uma medida de qualidade de *clustering*. A curva ROC é amplamente utilizada para avaliar modelos de classificação, permitindo uma análise mais detalhada do desempenho dos algoritmos em diferentes cenários. A sua aplicação pode ser estendida para avaliar a eficiência do tempo de execução de modelos, especialmente em sistemas que exigem alto desempenho computacional, como aqueles baseados em Docker.

A combinação de técnicas de aprendizagem de máquina com a análise de tempo de execução e qualidade dos modelos permite uma melhoria contínua na eficiência dos sistemas de computação, reduzindo o tempo de processamento e otimizando o uso de recursos. A adoção dessas práticas é cada vez mais importante à medida que a demanda por sistemas mais rápidos e eficientes cresce, especialmente em ambientes de produção e serviços de nuvem.

2.5.2 Qualidade dos clusters gerados

A qualidade dos *clusters* gerados por algoritmos de agrupamento é um aspecto essencial para avaliar a eficácia de qualquer sistema de análise de dados. Vysala e Gomes (2020) destacam que a validação dos resultados de *clustering* é um

processo complexo, uma vez que envolve a análise de diferentes métricas para garantir que os *clusters* sejam representativos e coerentes com os dados originais. A precisão de um modelo de *clustering* não se limita apenas à separação dos dados, mas também à sua capacidade de identificar padrões relevantes dentro de grandes volumes de informações. A validação pode ser feita utilizando métodos internos, que não necessitam de informações externas, ou métodos externos, que comparam os *clusters* gerados com algum padrão de referência.

Em contextos de grande escala, a análise da qualidade dos *clusters* também precisa ser feita com cautela, levando em consideração a escalabilidade dos algoritmos utilizados. Emmons *et al*. (2016) abordam a importância da análise dos algoritmos de *clustering* em redes e discutem diferentes métricas de qualidade de *clusters*, como a coesão e a separação, que são fundamentais para avaliar o desempenho do processo de agrupamento. A coesão se refere à proximidade dos elementos dentro de um cluster, enquanto a separação diz respeito à distância entre diferentes *clusters*. Ambas as métricas ajudam a mensurar a qualidade dos resultados obtidos. Essas métricas são essenciais para determinar a eficácia dos algoritmos de *clustering*, especialmente quando se trabalha com grandes volumes de dados em sistemas de alta demanda. A escolha da métrica de qualidade adequada pode influenciar diretamente a interpretação dos dados e, por consequência, a tomada de decisões baseadas nesses resultados.

2.5.3 Custo de processamento local e em nuvem

O custo de processamento, seja em servidores locais ou em nuvem, é um fator determinante para muitas empresas ao escolherem qual infraestrutura adotar para suas operações. Impeto Informática (2019) compara os custos de servidores na nuvem e locais, apontando que, embora o investimento inicial em servidores locais possa ser mais alto devido à necessidade de aquisição e manutenção de hardware, a nuvem oferece um modelo de pagamento por uso, o que pode ser vantajoso para empresas que buscam escalabilidade e flexibilidade. A escolha entre uma solução local e uma em nuvem depende das necessidades específicas de cada organização, considerando o volume de dados, a segurança e os custos operacionais contínuos.

A computação em nuvem tem ganhado destaque por sua capacidade de reduzir custos de infraestrutura e permitir a alocação de recursos conforme a

demanda. No entanto, há desafios relacionados ao custo de transferência de dados, segurança e latência, que podem impactar no desempenho de determinadas aplicações. Emmons *et al*. (2016) discutem a análise de algoritmos de *clustering* em redes, destacando que a escolha entre processamento local e em nuvem também deve considerar a complexidade e os requisitos de tempo de processamento. Algoritmos mais complexos, que demandam maior poder computacional, podem se beneficiar da nuvem, que oferecem recursos escaláveis e de alto desempenho.

Embora os custos de manutenção em nuvem possam ser mais baixos a longo prazo, as empresas devem avaliar cuidadosamente o impacto desses custos com base no tipo de aplicação e no volume de dados processados, considerando sempre a relação entre custo e desempenho.

2.6 Trabalhos relacionados

Diversos estudos recentes têm se dedicado à análise da clusterização de dados em diferentes ambientes computacionais, especialmente locais e em nuvem. A popularização de serviços como AWS, Google Cloud e Azure motivou comparações entre suas performances e as soluções tradicionais em máquinas físicas. Lorenzi, Gréin e Corcini (2022) destacam que a computação em nuvem representa uma alternativa escalável e flexível, adequada para execução de algoritmos como o K-Means e DBSCAN em grandes volumes de dados. Por outro lado, ambientes locais ainda são preferidos em contextos que exigem controle rigoroso sobre segurança e privacidade dos dados, como salientam Silva (2023) e Maliszewski et al. (2021).

Pesquisas específicas demonstram o comportamento dos algoritmos de agrupamento frente a diferentes arquiteturas. Emmons et al. (2016), por exemplo, analisaram o desempenho de múltiplos algoritmos de clusterização em redes de larga escala e constataram que, embora a computação em nuvem ofereça vantagens em escalabilidade, nem sempre supera o ambiente local em termos de tempo de resposta, especialmente para modelos otimizados localmente. Essa dualidade reforça a importância de estudos comparativos contextualizados, como o presente trabalho, que considera não apenas desempenho, mas também custo e qualidade dos agrupamentos.

Estudos como o de Cavalcante, Boeres e Rebello (2024) exploraram a eficiência de serviços de conteinerização em ambientes AWS, evidenciando que soluções em nuvem podem reduzir significativamente os custos operacionais, sobretudo em projetos de curta duração e alta demanda computacional. Já Júnior e Santos (2022) mostraram que pequenas e médias empresas ainda enfrentam desafios ao migrar sistemas ERP para a nuvem, devido a custos recorrentes e dificuldades de integração. Esses achados sugerem que a viabilidade do uso da nuvem depende fortemente do perfil da aplicação e dos requisitos de desempenho.

O tempo de processamento dos algoritmos também foi objeto de análise em pesquisas como a de Horchulhack et al. (2022), que investigaram o impacto do *overbooking* em aplicações baseadas em Docker. Os autores demonstraram que a alocação eficiente de recursos é decisiva para a redução do tempo de execução, sendo essa uma vantagem importante da nuvem. Em contrapartida, Emmons et al. (2016) apontaram que configurações locais otimizadas podem, em certos casos, alcançar resultados superiores em menor tempo, desde que haja infraestrutura suficiente.

No que se refere à qualidade dos agrupamentos, trabalhos como o de Vysala e Gomes (2020) e Jaskowiak, Costa e Campello (2020) reforçam o uso de métricas como o coeficiente de Silhouette e o índice de Dunn para validar a separação e coesão dos *clusters*. Essas métricas são aplicadas em diversos experimentos práticos, inclusive naqueles que comparam ambientes de execução distintos. Ferreira (2024) também destaca que a adoção de múltiplos indicadores aumenta a robustez da validação dos agrupamentos, proporcionando maior confiabilidade na análise dos dados.

Em contextos educacionais, a clusterização de dados tem se mostrado útil para a personalização do ensino e a análise de desempenho estudantil. Melo, Pessoa e Fernandes (2024) utilizam algoritmos como HDBSCAN para identificar padrões em soluções de exercícios de programação, permitindo o mapeamento de perfis de aprendizagem. Já Silva, Pereira e Saqui (2023) aplicaram modelos híbridos baseados em K-Means para melhorar a recomendação de conteúdos em ambientes de leitura, demonstrando a versatilidade da clusterização para fins educacionais.

Gonçalves e Santos (2024) conduziu um estudo que utilizou análise espacial e algoritmos de clusterização para mapear *hotspots* de acidentes de trabalho no Brasil, integrando variáveis de mobilidade urbana e segurança pública. Essa

abordagem demonstra como a clusterização pode ser aplicada para interpretar dados multidimensionais, como é o caso dos questionários do ENADE utilizados na presente pesquisa. A flexibilidade dos métodos de agrupamento permite que se revelem padrões ocultos, essenciais para análises educacionais, sociais ou urbanas.

Outro ponto relevante é o uso de ambientes em nuvem para tratamento de dados educacionais. Mejia e Curasma (2023) evidenciam como ferramentas como o Amazon SageMaker e Amazon Forecast têm facilitado a implementação de sistemas recomendadores e análises preditivas em plataformas educacionais. Freitas, Magalhães e Costa (2024) também relatam o uso dessas ferramentas na previsão de demanda logística, com bons resultados de custo-benefício. Essas experiências reforçam a viabilidade técnica e econômica do uso da nuvem para clusterização em educação.

Com base nos estudos revisados, é possível afirmar que há uma lacuna significativa em pesquisas que integrem comparações práticas entre ambientes computacionais e a aplicação direta de clusterização em dados educacionais de avaliação institucional, como o ENADE. O presente trabalho avança nesse sentido ao explorar, de forma sistemática, o comportamento dos algoritmos K-Means, MiniBatch K-Means, DBSCAN e HDBSCAN em dois ambientes distintos. Ao alinhar o rigor técnico da ciência da computação com uma aplicação concreta em dados educacionais, a pesquisa contribui tanto para o avanço acadêmico quanto para a aplicação prática no campo da gestão e análise educacional.

3 METODOLOGIA

A metodologia adotada caracteriza-se por um estudo de natureza quantitativa, com abordagem aplicada e delineamento experimental, cujo objetivo foi avaliar o desempenho da clusterização de dados em ambientes computacionais distintos: local e nuvem. A natureza quantitativa permitiu mensurar variáveis como tempo de execução, custo computacional e qualidade dos agrupamentos, enquanto a aplicação prática dos algoritmos justificou o caráter experimental do estudo, conforme preconizado por Ferreira (2024), que também destaca a importância da experimentação na análise de desempenho computacional. Os procedimentos adotados seguiram um protocolo padronizado para a execução dos algoritmos de clusterização em dois ambientes distintos:

- **ambiente local**, com recursos limitados a um computador pessoal;
- **ambiente em nuvem**, configurado na plataforma **Amazon SageMaker**.

O protocolo padronizado garante a reprodutibilidade das análises tanto em um ambiente local (utilizando Visual Studio Code e bibliotecas Python) quanto na nuvem (por meio do Amazon SageMaker). As decisões metodológicas adotadas ao longo do desenvolvimento, como a escolha das métricas de avaliação e o dimensionamento dos *clusters*, foram fundamentadas na literatura científica, assegurando a robustez e a relevância dos resultados obtidos para o campo da ciência de dados educacionais.

3.1 Planejamento de cada ambiente

No ambiente local, os experimentos foram conduzidos em um notebook com processador Intel Core i5, 16 GB de RAM e sistema operacional Linux. As bibliotecas Python utilizadas incluíram scikit-learn, hdbscan, pandas, matplotlib e NumPy. Essa configuração buscou simular o contexto real de pequenas instituições ou pesquisadores independentes que utilizam a infraestrutura acessível. Segundo Silva (2023), o uso de ambientes locais, embora limitado, ainda é comum em pesquisas de pequeno porte devido à maior segurança e controle sobre os dados.

No ambiente em nuvem, os experimentos foram realizados utilizando instâncias do Amazon SageMaker, serviço que oferece recursos escaláveis para a execução de tarefas de aprendizado de máquina. Mejia e Curasma (2023) ressaltam que o SageMaker permite treinamento, validação e implementação de modelos com alta eficiência, sendo uma opção cada vez mais popular em projetos acadêmicos e corporativos. *As instâncias utilizadas foram do tipo ml.c5.2xlarge, com 8 vCPUs e 16 GB de RAM, configuradas por meio de Jupyter Notebooks integrados.*

3.2 Execução dos experimentos

O conjunto de dados utilizado nos experimentos foi extraído do banco oficial do ENADE 2022, disponibilizado pelo Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP). Foram selecionadas as respostas dos estudantes relativas às percepções sobre o processo de aprendizagem, especialmente no contexto pós-pandêmico. A escolha desse conjunto se justifica pela sua representatividade e atualidade, além da relevância educacional e social. Gonçalves e Santos (2024) reforçam que dados de larga escala em educação oferecem ricas oportunidades para análise preditiva e identificação de padrões.

Os dados foram pré-processados com técnicas padrão: remoção de valores ausentes, codificação de variáveis categóricas e normalização dos dados numéricos. O objetivo foi garantir que os algoritmos pudessem operar eficientemente e que os resultados fossem comparáveis. Segundo Ferreira (2024), a padronização dos dados é essencial para que algoritmos de clusterização baseados em distância, como o K-Means, não sejam enviesados por variáveis de maior escala.

Quatro algoritmos de clusterização foram utilizados: K-Means, MiniBatch K-Means, DBSCAN e HDBSCAN. A escolha desses modelos baseou-se em suas características distintas: os dois primeiros são baseados em particionamento e apresentam bom desempenho em conjuntos de dados grandes e bem distribuídos, enquanto os dois últimos são baseados em densidade e possuem maior robustez frente a ruídos e padrões irregulares. Emmons et al. (2016) salientam a importância da diversidade de algoritmos em estudos comparativos, especialmente quando o objetivo é avaliar eficiência e precisão.

Os experimentos consistiram na aplicação de cada algoritmo, com os mesmos parâmetros iniciais, em ambos os ambientes computacionais. Para os modelos K-Means e MiniBatch K-Means, foram testados diferentes faixas de *clusters* (k = 3 a 15), utilizando o método do cotovelo e o coeficiente de Silhouette como guias para escolha. Para DBSCAN e HDBSCAN, foram ajustados os parâmetros de vizinhança mínima e distância máxima com base em iterações sucessivas. Vysala e Gomes (2020) destacam que o ajuste fino desses parâmetros é decisivo para a obtenção de agrupamentos significativos.

3.3 Coleta das métricas

O tempo de execução de cada experimento foi cronometrado do início ao fim do processo de clusterização. O objetivo foi medir o tempo total de processamento e comparar os desempenhos entre o ambiente local e o ambiente em nuvem. Horchulhack et al. (2022) apontam que esse tipo de métrica é fundamental para decisões estratégicas em projetos de análise de dados, especialmente quando o tempo de resposta é um fator crítico.

Outro critério avaliado foi o custo operacional de cada execução. No ambiente local, o custo foi estimado com base tempo de uso da máquina e na depreciação do hardware. Já no ambiente em nuvem, os custos foram calculados com base na precificação da AWS por tempo de uso e tipo de instância. Cavalcante, Boeres e Rebello (2024) afirmam que a computação em nuvem pode ser financeiramente vantajosa em operações otimizadas, mas alertam para os custos acumulados de uso prolongado e armazenamento.

A qualidade dos agrupamentos foi avaliada utilizando o coeficiente de Silhouette, que mede simultaneamente a coesão interna dos *clusters* e sua separação em relação aos demais grupos. Ferreira (2024) argumenta que esta métrica é especialmente adequada para contextos multidimensionais como os dados do ENADE, proporcionando uma avaliação equilibrada da estrutura dos agrupamentos.

Todos os experimentos foram repetidos três vezes para garantir a confiabilidade dos resultados e minimizar o impacto de variações pontuais. Os dados coletados foram organizados em tabelas comparativas e representações gráficas para facilitar a visualização. Jaskowiak, Costa e Campello (2020) reforçam a

importância da repetição em experimentos computacionais para mitigar erros estatísticos e aumentar a validade externa da pesquisa.

3.4 Interpretação dos resultados

A análise dos resultados foi conduzida por meio de comparação direta entre os valores obtidos nos dois ambientes. Foram observadas diferenças significativas tanto no tempo de execução quanto nos custos operacionais, especialmente para algoritmos mais exigentes, como o HDBSCAN. Freitas, Magalhães e Costa (2024) destacam que a eficiência computacional depende da combinação entre algoritmo, volume de dados e ambiente de execução, sendo necessário avaliar cada contexto de forma específica. A integração entre ferramentas livres, como as bibliotecas Python, e serviços comerciais, como o Amazon SageMaker, demonstrou a viabilidade de estudos experimentais robustos em diferentes contextos. Os achados servirão como base para a análise crítica dos resultados no capítulo seguinte, oferecendo também subsídios técnicos para futuras pesquisas e aplicações profissionais.

4 DESENVOLVIMENTO

Este capítulo descreve em detalhes a condução dos experimentos realizados para avaliar o desempenho da clusterização de dados em ambientes locais e em nuvem. As etapas envolveram desde a coleta e preparação dos dados até a implementação de *pipelines* de avaliação automatizada, com registros sistemáticos dos resultados em arquivos e painéis gráficos interativos.

4.1 Preparação dos dados

A etapa de desenvolvimento deste trabalho teve início com a definição e coleta do conjunto de dados a ser analisado. Optou-se pelo uso dos microdados do Exame Nacional de Desempenho dos Estudantes (ENADE) do ano de 2022, disponibilizados pelo Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP). Os dados do ENADE são amplamente utilizados em pesquisas educacionais por seu volume, diversidade e estrutura padronizada, o que os torna ideais para a aplicação de técnicas de análise de dados, como a clusterização. A relevância do tema se justifica pela necessidade de compreender os impactos causados pela pandemia de COVID-19 no processo de formação dos estudantes do ensino superior brasileiro.

O subconjunto selecionado do banco de dados do ENADE corresponde às questões 69 a 79 do questionário do estudante, as quais abordam aspectos diretamente relacionados às percepções dos discentes quanto às transformações ocorridas no ensino durante e após o período pandêmico. Essas questões estão organizadas em uma escala Likert de seis pontos, que varia de “Discordo totalmente” (1) a “Concordo totalmente” (6), o que proporciona uma base quantitativa adequada à aplicação de algoritmos de clusterização. As variáveis contemplam dimensões como acesso a recursos tecnológicos, suporte pedagógico, adaptação ao ensino remoto e percepção de prejuízo na formação acadêmica.

A seleção desse subconjunto tem como objetivo identificar padrões ocultos nas respostas dos estudantes que possam apontar diferentes perfis de vivência da pandemia no contexto acadêmico. Agrupar esses perfis por similaridade pode fornecer subsídios importantes para políticas públicas e ações institucionais voltadas

à mitigação das desigualdades no processo formativo. Conforme destaca Ferreira (2024), a clusterização é especialmente útil em contextos de análise exploratória de dados sociais, pois permite revelar estruturas latentes que não são imediatamente perceptíveis por métodos estatísticos tradicionais.

Após a seleção do recorte de interesse, os dados foram submetidos a uma etapa de inspeção preliminar, com o objetivo de verificar a consistência das informações e identificar possíveis problemas de qualidade. Essa análise exploratória inicial revelou a presença de campos com respostas nulas, entradas categorizadas como “Não sei responder” e “Não se aplica”, além de colunas não pertinentes à análise, como o ano da prova (NU_ANO). Esses elementos poderiam comprometer a eficácia dos algoritmos de clusterização, que operam com base em métricas de distância sensíveis a ruídos e inconsistências nos dados.

A limpeza dos dados foi realizada com o auxílio da biblioteca Pandas, amplamente utilizada na linguagem de programação Python para manipulação de estruturas tabulares. Inicialmente, foram removidas as colunas não relacionadas às questões selecionadas. Em seguida, as respostas “Não se aplica” e “Não sei responder” foram tratadas como valores ausentes (NaN) e eliminadas do conjunto de dados. Esse procedimento garantiu a uniformidade das entradas e reduziu a dimensionalidade do problema, mantendo apenas os registros relevantes.

O código utilizado para esse pré-processamento foi implementado em ambiente local, utilizando o Visual Studio Code como IDE. O procedimento envolveu a leitura do arquivo original .CSV, sua conversão em DataFrame, remoção de colunas desnecessárias, tratamento de valores ausentes e exportação dos dados limpos para um novo arquivo. A operação foi automatizada para garantir reprodutibilidade, e o *script* correspondente está documentado no **repositório do projeto**[1]. Essa estratégia segue as boas práticas de engenharia de dados descritas por Lorenzi, Gréin e Corcini (2022), que defendem a transparência e replicabilidade dos processos de análise.

Além do tratamento de dados faltantes, foi realizada a normalização dos valores válidos, utilizando a técnica de min-max scaling, que transforma os valores para uma faixa entre 0 e 1. Essa etapa é fundamental para garantir que todas as variáveis tenham o mesmo peso na análise, uma vez que os algoritmos de clusterização, especialmente os baseados em distância euclidiana como o K-Means,

[1] https://github.com/GuiiCorreia/Meu-TCC

são sensíveis a diferenças de escala. Segundo Vysala e Gomes (2020), a normalização prévia dos dados é uma etapa crítica para a formação de *clusters* coerentes em espaços multidimensionais.

A partir da base de dados limpa e normalizada, realizou-se uma nova inspeção para confirmar a integridade do conjunto. A verificação envolveu a análise de estatísticas descritivas (média, mediana, desvio-padrão) de cada variável, além da geração de histogramas para visualizar a distribuição das respostas. Essa etapa possibilitou observar uma predominância de valores médios (3 a 5) na maioria das questões, indicando que os estudantes tendem a posicionamentos moderados em relação às afirmações propostas. Essa informação é relevante para a interpretação dos *clusters* que viriam a ser formados.

Com os dados preparados, procedeu-se à divisão do conjunto em duas versões idênticas, cada uma destinada à execução dos algoritmos em um dos ambientes de processamento: ambiente local e ambiente em nuvem. Essa duplicação foi realizada para garantir que as análises fossem comparáveis em termos de entrada, eliminando a possibilidade de viés introduzido por divergência nos dados utilizados. Mejia e Curasma (2023) reforçam a importância de manter a homogeneidade dos dados em experimentos comparativos entre plataformas computacionais distintas.

O ambiente local foi configurado em um notebook HP com processador Intel Core i5-7200U, 16GB de memória RAM, rodando sistema operacional Linux. As ferramentas utilizadas incluíram o Visual Studio Code, o banco de dados PostgreSQL e a ferramenta Metabase para visualização de resultados. Já o ambiente em nuvem foi implementado por meio do serviço AWS SageMaker, utilizando instâncias do tipo ml.c5.2xlarge, que oferecem 8 vCPUs e 16 GiB de memória. A escolha dessas configurações foi baseada na busca por um equilíbrio entre custo e desempenho, conforme sugerido por Cavalcante et al. (2024).

Antes da aplicação dos algoritmos, foram criadas funções específicas em Python para operacionalizar o *pipeline* de execução e avaliação dos modelos. A função principal, cluster_and_evaluate, recebe como parâmetros o algoritmo a ser testado, os dados e os hiperparâmetros desejados, realizando a clusterização e o cálculo do *Silhouette Score*, que será detalhado no próximo subcapítulo. A implementação foi pensada para ser modular e extensível, permitindo adaptações futuras, como o teste de novos algoritmos ou métricas de avaliação adicionais.

Cada algoritmo foi testado com diferentes configurações de parâmetros, incluindo a variação no número de *clusters* de 3 a 15 nos modelos baseados em particionamento. A escolha desses valores foi baseada na literatura e em estudos exploratórios prévios, com o intuito de identificar o ponto de equilíbrio entre qualidade de agrupamento e simplicidade estrutural. Emmons et al. (2016) destacam que testes com múltiplos valores de k aumentam a robustez das conclusões em estudos com clusterização.

Durante os testes, foram registrados para cada execução o tempo de processamento (em segundos) e o valor do *Silhouette Score*, com o objetivo de realizar posteriormente uma análise comparativa entre os dois ambientes. Os resultados foram armazenados em um arquivo .CSV contendo as colunas: algoritmo, número de *clusters*, tempo de execução e índice de Silhouette. A coleta automatizada desses dados assegura precisão na mensuração dos indicadores, além de facilitar a posterior geração de gráficos e relatórios.

Essa abordagem sistemática de coleta, limpeza, preparação e replicação dos dados assegura a consistência dos experimentos a serem apresentados nas seções seguintes. Ela permite que as comparações entre os ambientes computacionais — local e nuvem — sejam feitas de forma justa, com base em um mesmo conjunto de informações. Essa padronização metodológica é essencial para que as conclusões extraídas ao final da pesquisa tenham validade técnica e científica, como enfatizado por Ferreira (2024) ao abordar estudos comparativos em ciência de dados.

4.2 Implementação dos algoritmos

A implementação dos algoritmos de clusterização foi conduzida em dois ambientes distintos: um ambiente local e um ambiente em nuvem. O ambiente local foi configurado em um notebook HP 15-bs070wm, com processador Intel Core i5-7200U (quatro threads) e 16GB de memória RAM, operando sob o sistema Linux e utilizando o **Visual Studio Code** como ambiente de desenvolvimento integrado (IDE). Para suporte à persistência de dados, foi utilizado o **PostgreSQL**, enquanto a ferramenta **Metabase** foi empregada para visualização gráfica dos resultados. No ambiente em nuvem, utilizou-se o serviço **AWS SageMaker**, na **região Ohio**, configurado com instâncias do tipo ml.c5.2xlarge, que oferecem 8 vCPUs e 16 GiB de memória RAM.

A escolha desses dois ambientes foi estratégica para viabilizar uma análise comparativa de desempenho entre estruturas computacionais com diferentes capacidades. A literatura destaca que, enquanto os ambientes locais oferecem maior controle e custo reduzido em pequenas escalas, as soluções em nuvem proporcionam melhor escalabilidade e flexibilidade em contextos mais exigentes (Lorenzi, Gréin e Corcini, 2022). Essa dualidade é especialmente importante para instituições de ensino e pesquisa que buscam otimizar recursos sem comprometer a eficiência dos experimentos computacionais.

A escolha dos quatro algoritmos foi baseada na diversidade metodológica, abrangendo técnicas de particionamento e de densidade. O **K-Means** é amplamente utilizado por sua simplicidade e desempenho em conjuntos de dados bem distribuídos, enquanto o **MiniBatch K-Means** representa uma versão otimizada para grandes volumes de dados, com atualizações incrementais em pequenos lotes (Emmons et al., 2016). Os dois são eficazes quando se conhece previamente o número de agrupamentos, embora sejam sensíveis à forma deles e *outliers*.

Já os algoritmos **DBSCAN** e **HDBSCAN** foram escolhidos por operarem sob o paradigma de densidade, o que lhes confere vantagens na detecção de *clusters* de formas arbitrárias e em dados com ruídos. O **DBSCAN** (*Density-Based Spatial Clustering of Applications with Noise*) é conhecido por sua capacidade de identificar agrupamentos sem a necessidade de definir o número de *clusters* a priori, utilizando os parâmetros eps (distância máxima entre dois pontos para serem considerados vizinhos) e min_samples (número mínimo de pontos em uma vizinhança para formar um cluster). O **HDBSCAN**, por sua vez, estende essa lógica hierarquicamente, permitindo a detecção de agrupamentos com densidades variadas, com o parâmetro min_cluster_size sendo o mais relevante (Vysala e Gomes, 2020).

A escolha dos parâmetros foi realizada com base em boas práticas da literatura e em testes empíricos. Para o K-Means e MiniBatch K-Means, utilizou-se n_init=10 e random_state=0, valores considerados adequados para garantir consistência e reprodutibilidade dos resultados. Para o DBSCAN, adotou-se eps=0.5 e min_samples=5, enquanto para o HDBSCAN utilizou-se min_cluster_size=5. Esses valores foram testados iterativamente, com ajustes pontuais durante o desenvolvimento para maximizar a separação entre os agrupamentos, conforme recomenda Ferreira (2024), que enfatiza a importância do ajuste fino de hiperparâmetros na clusterização não supervisionada.

A lógica de execução dos testes foi implementada por meio da função cluster_and_evaluate, desenvolvida em Python e estruturada para executar algoritmos de clusterização de forma automatizada. Essa função recebe como parâmetros o algoritmo, os dados e seus hiperparâmetros, realizando a clusterização e o cálculo do ***Silhouette Score***, desde que mais de um agrupamento válido seja gerado. Essa métrica foi escolhida por sua capacidade de mensurar, de forma objetiva, a qualidade dos *clusters* formados, variando entre -1 e 1. Valores próximos a 1 indicam maior separabilidade entre os grupos, enquanto valores negativos apontam má atribuição (Řezanková, 2018; Shutaywi; Kachouie, 2021).

O código também foi responsável por registrar o tempo de execução de cada teste utilizando a biblioteca time. A medição do tempo foi iniciada antes da execução do algoritmo e encerrada imediatamente após o cálculo do *Silhouette Score*, sendo o tempo decorrido armazenado em segundos. Esses dados foram essenciais para o comparativo posterior entre os dois ambientes. De acordo com Cavalcante, Boeres e Rebello (2024), o tempo de execução é uma variável crítica em avaliações de desempenho de algoritmos, especialmente quando se busca dimensionar a viabilidade prática de diferentes soluções computacionais.

A função cluster_and_evaluate foi projetada para funcionar dentro de um *loop* que percorre uma lista de algoritmos e suas configurações. Para os algoritmos K-Means e MiniBatch K-Means, foi definido um intervalo de *clusters* entre 3 e 15, com o objetivo de observar o impacto da variação de k na qualidade dos agrupamentos. Já para DBSCAN e HDBSCAN, que não exigem a definição prévia do número de *clusters*, os testes foram realizados com seus parâmetros conforme citado no texto. Todos os resultados foram armazenados em uma lista e posteriormente exportados para um arquivo CSV, que serviu como base para a análise gráfica e estatística.

O *loop* de testes foi executado integralmente em ambos os ambientes, assegurando que cada algoritmo fosse testado sob as mesmas condições de entrada. A padronização dos dados e dos *scripts* de execução garantiu a validade da comparação entre os ambientes. Mejia e Curasma (2023) destacam que a uniformização das variáveis externas é fundamental para evitar viés em experimentos comparativos em ciência de dados, sobretudo quando há diferentes arquiteturas de hardware envolvidas.

A estrutura do *pipeline* foi complementada com a criação de *logs* que indicavam o progresso dos testes e reportavam eventuais falhas na execução. Essa estratégia foi especialmente importante no ambiente em nuvem, onde a instabilidade de conexões ou limitações de tempo de sessão poderiam interromper a execução. Os *logs* facilitaram a retomada dos experimentos a partir do ponto de interrupção, garantindo eficiência e integridade nos dados coletados.

Durante os testes, cada algoritmo retornou uma série de indicadores que foram organizados nas colunas: nome do algoritmo, número de *clusters* (ou “Auto”, nos casos de DBSCAN e HDBSCAN), tempo de execução e valor do *Silhouette Score*. A partir desses dados, foi possível comparar o comportamento dos algoritmos em função do ambiente de execução. Freitas, Magalhães e Costa (2024) ressaltam que a sistematização dos resultados é crucial para derivar conclusões confiáveis em estudos de performance algorítmica.

Os dados exportados foram utilizados na plataforma Metabase para a construção de painéis interativos que facilitaram a identificação de padrões e anomalias. Foram gerados gráficos de dispersão, histogramas e séries temporais para analisar a distribuição dos tempos de execução e dos *Silhouette Score*, tanto no ambiente local quanto na nuvem. A utilização do Metabase permitiu uma análise mais ágil e visual, otimizando o processo de interpretação.

Ao final desta etapa de implementação, os dados estavam prontos para serem analisados sob a ótica do desempenho. O *pipeline* construído garantiu reprodutibilidade, escalabilidade e clareza metodológica, permitindo que os resultados obtidos possam ser validados por outros pesquisadores ou aplicados em contextos semelhantes. O próximo capítulo, dedicado à análise crítica dos resultados, destaca os padrões identificados e as possíveis inferências derivadas da experimentação empírica.

4.3 Métricas de avaliação

Para a avaliação da qualidade dos agrupamentos obtidos pelos algoritmos de clusterização, foi utilizada a métrica ***Silhouette Score***. Adicionalmente, foram analisados o **tempo de execução** e a **complexidade computacional** dos algoritmos para estabelecer uma relação custo-benefício entre a qualidade dos agrupamentos e o desempenho computacional nos ambientes local e em nuvem.

A combinação dessas três métricas proporcionou uma visão abrangente sobre o comportamento dos algoritmos, contemplando aspectos qualitativos e quantitativos fundamentais. Cada uma cumpre um papel específico: a primeira avalia a qualidade do agrupamento, a segunda o custo temporal e a terceira antecipa limitações teóricas. Essa abordagem integrada segue as diretrizes de avaliação preconizadas por Rezanková (2018), que recomenda a análise multidimensional para validação robusta de experimentos não supervisionados.

O ***Silhouette Score*** foi a principal métrica utilizada para aferir a **qualidade dos *clusters*** gerados. Essa métrica analisa o grau de separação entre os agrupamentos e o quanto os elementos estão corretamente alocados. Seu valor varia de -1 a 1, onde valores próximos de 1 indicam boa definição dos *clusters*, valores próximos de 0 sugerem sobreposição entre agrupamentos e valores negativos indicam má atribuição (Shutaywi; Kachouie, 2021). O cálculo foi feito com base na **distância euclidiana**, pois os algoritmos utilizados – especialmente K-Means e MiniBatch K-Means – operam sob esse critério.

O uso do coeficiente de Silhouette como parâmetro de avaliação se justifica por sua capacidade de avaliar simultaneamente a **coesão interna dos *clusters*** e a **separação entre grupos distintos**. De acordo com Ferreira (2024), essa métrica é particularmente útil em situações onde não há rótulos supervisionados, sendo uma das mais confiáveis para validação de agrupamentos em dados de natureza social ou educacional. Sua interpretação simples também favorece a visualização e comunicação dos resultados a diferentes públicos.

A métrica foi aplicada a todos os algoritmos, exceto nos casos em que o número de *clusters* gerados foi igual a 1, situação em que o cálculo do *Silhouette Score* não é possível. Essa condição ocorreu em alguns testes com DBSCAN e HDBSCAN, cujos algoritmos, por operarem com base em densidade, podem não identificar agrupamentos válidos se os parâmetros não forem adequadamente ajustados. Como destaca Vysala e Gomes (2020), a sensibilidade desses algoritmos aos parâmetros eps e min_samples pode resultar em agrupamentos vazios ou únicos, o que exige cuidado na interpretação dos resultados.

Para garantir a precisão na medição do desempenho temporal, foi utilizada a função time() da biblioteca time do Python. O tempo foi cronometrado em segundos, sendo iniciada a contagem imediatamente antes da chamada da função de clusterização e encerrada logo após o cálculo do *Silhouette Score*. O **tempo de**

execução é uma métrica central em estudos de eficiência algorítmica, sobretudo quando se avaliam diferentes infraestruturas computacionais. Em ambientes com restrições de recursos, como dispositivos locais, a execução de algoritmos de alta complexidade pode comprometer a viabilidade do experimento. Segundo Cavalcante, Boeres e Rebello (2024), o tempo de processamento influencia diretamente nos custos operacionais e na experiência do usuário, sendo um fator crucial na escolha da arquitetura computacional.

Ao longo dos testes, observou-se que o tempo de execução variava significativamente de acordo com o algoritmo e o ambiente utilizado. Em geral, os algoritmos K-Means e MiniBatch K-Means apresentaram menor tempo médio de execução, com destaque para o MiniBatch K-Means, cuja eficiência é atribuída à sua estratégia de atualização parcial em lotes de dados. Em contrapartida, DBSCAN e HDBSCAN, por realizarem verificações densas de vizinhança, consumiram mais tempo, sobretudo no ambiente local, que possui menor capacidade de paralelização.

Para interpretar os resultados de desempenho, também se considerou a **complexidade computacional teórica** dos algoritmos. O K-Means, por exemplo, apresenta complexidade **O(n*k*i*d)**, onde *n* é o número de pontos, *k* o número de *clusters*, *i* o número de iterações e *d* a dimensionalidade dos dados. O MiniBatch K-Means reduz significativamente esse custo ao trabalhar com amostras parciais, sendo mais eficiente em termos de escalabilidade. Já o DBSCAN possui complexidade **O(n²)** no pior caso, o que o torna menos eficiente para grandes volumes de dados não otimizados (Ferreira, 2024).

O algoritmo HDBSCAN, por sua vez, apesar de ser uma extensão do DBSCAN, implementa otimizações que o tornam mais eficiente na detecção de *clusters* com diferentes densidades. Ele utiliza uma estrutura hierárquica baseada em árvores de alcance mútuo, com desempenho prático superior em muitos casos. No entanto, sua complexidade teórica pode variar dependendo da implementação e do conjunto de dados. Emmons et al. (2016) alertam que a análise da complexidade prática deve considerar não apenas o algoritmo, mas também a implementação e os recursos da infraestrutura utilizada.

Os resultados dessas métricas foram visualizados com o auxílio do Metabase utilizando painéis comparativos entre os ambientes. Os gráficos permitiram observar tendências consistentes, como a redução do tempo médio na nuvem e a estabilidade dos valores de *Silhouette Score* independentemente do ambiente. Tais

observações preliminares orientaram a interpretação dos resultados, apresentada no próximo capítulo. Como defendem Freitas, Magalhães e Costa (2024), a visualização clara e estruturada dos dados é essencial para uma comunicação científica eficaz.

Por fim, vale destacar que essas métricas foram selecionadas não apenas por sua relevância técnica, mas também por sua **aderência ao objetivo da pesquisa**, que é avaliar custo, tempo e qualidade de algoritmos de clusterização em diferentes ambientes. A clareza, simplicidade e reprodutibilidade das métricas tornam os resultados obtidos relevantes tanto para a comunidade científica quanto para instituições que buscam otimizar seus processos analíticos educacionais.

4.4 Execução dos testes

A etapa de execução dos testes foi conduzida com base em um protocolo rigorosamente padronizado, a fim de garantir comparabilidade e confiabilidade nos resultados. Cada experimento foi executado sob as mesmas condições de entrada de dados, parâmetros algorítmicos e ambiente controlado. Essa padronização é essencial para eliminar interferências de variáveis externas e viabilizar uma análise justa entre diferentes contextos computacionais, como salientam Mejia e Curasma (2023) ao discutirem reprodutibilidade em experimentos com *machine learning*.

Para garantir a confiabilidade dos resultados, cada experimento foi repetido **três vezes**. Os valores obtidos de tempo e *Silhouette Score* foram analisados com o intuito de identificar variações significativas e, na ausência, a última execução foi considerada representativa. Segundo Shutaywi e Kachouie (2021), a repetição de experimentos é recomendada em avaliações algorítmicas, pois permite controlar variabilidades não previstas do sistema ou flutuações nos serviços de nuvem.

Para comparação de desempenho, os testes foram realizados em dois ambientes com configurações bem distintas. O **ambiente local** foi implementado em um **notebook HP 15-bs070wm**, com **processador Intel Core i5-7200U (4 threads)** e **16 GiB de memória RAM**. Esse ambiente simula o contexto de pesquisadores independentes ou instituições de ensino que operam com infraestrutura modesta. Já o **ambiente em nuvem** foi estruturado na **Amazon Web Services (AWS)**, utilizando o **serviço SageMaker** com instâncias **ml.c5.2xlarge**, que oferecem **8 vCPUs e 16 GiB de memória RAM**.

A escolha desses ambientes visa simular diferentes realidades institucionais: uma de custo reduzido e autonomia total (ambiente local), e outra com escalabilidade e maior desempenho, porém associada a custos por hora (ambiente em nuvem). Lorenzi, Gréin e Corcini (2022) destacam que esse tipo de comparação é cada vez mais necessário no contexto da ciência de dados, dado o crescimento de soluções baseadas em cloud computing.

Cada execução foi monitorada com o auxílio da biblioteca time, que calculava o tempo de execução de forma precisa, sendo os valores gravados em arquivos .CSV para análise posterior. Paralelamente, foram anotados os parâmetros de entrada utilizados, como número de *clusters*, algoritmo empregado e ambiente de execução. Essa sistematização assegura rastreabilidade e facilita a reexecução do experimento, se necessário.

Tabela 1 - Cálculo de depreciação (notebook local)

Parâmetro	**Valor**
Valor de aquisição	R$ 5.500,00
Tempo de uso diário	5 horas
Dias de uso por semana	6 dias
Vida útil estimada	7 anos
Total de horas de vida útil	10.920 horas
Custo por hora (depreciação)	R$ 0,50

Fonte: autoria própria.

Foram considerados os **custos operacionais estimados** para os dois ambientes. Para o ambiente local, foi considerado um **custo médio de R$ 0,50 por hora**, com base no cálculo da **depreciação do notebook** ao longo de sua vida útil, conforme detalhado na Tabela 1. Supondo um valor de aquisição de R$ 5.500,00 com uso estimado de 5 horas diárias, 6 dias por semana, durante 7 anos, chegou-se a uma média de **10.920 horas totais**, o que resulta em um custo de **R$ 0,50/hora**. Para o ambiente em nuvem, utilizou-se a tabela de preços oficial da AWS (2025) na região de Ohio, cuja instância **ml.c5.2xlarge** tem custo de **USD 0.408/hora**. Considerando um câmbio médio de **R$ 5,00**, o valor corresponde a **R$ 2,04/hora**.

Tabela 2 - Análise Comparativa: Custo e Tempo

Dimensão Analisada	AWS SageMaker	Notebook Local
TEMPO		
Tempo de execução	12,91 horas	17,97 horas
Diferença temporal	-	+5,06 horas (+39%)
CUSTO		
Custo por hora	R$ 2,04	R$ 0,50
Custo total	R$ 26,52	R$ 9,00
Diferença de custo	-	-R$ 17,52 (-66,1%)

Fonte: autoria própria.

A Tabela 2 apresenta uma análise comparativa dos valores de custo e tempo entre os 2 ambientes analisados. A média total de tempo de execução dos experimentos no AWS SageMaker foi de aproximadamente 13 horas, resultando em um custo estimado de **R$ 26,52**. Em contrapartida, o tempo médio no ambiente local foi de cerca de 17,97 horas, resultando em um custo estimado de **R$ 9,00**.

Tabela 3 - *Trade-off*: Tempo versus Custo

Aspecto	AWS SageMaker	Notebook Local	Análise
Tempo de execução	12,91 horas	17,97 horas	AWS 39% mais rápido
Custo total	R$ 26,52	R$ 9,00	Local 66,1% mais barato
Prioridade	Agilidade	Economia	*Trade-off* estratégico

Fonte: autoria própria.

A comparação dos custos indica que o ambiente local é mais barato em termos absolutos, mas menos eficiente em tempo de execução. Essa diferença evidencia um dos principais *trade-offs* entre essas abordagens: **tempo versus custo**. Cavalcante, Boeres e Rebello (2024) observam que, embora o uso da nuvem represente investimento financeiro maior, ele pode ser compensado pela agilidade e escalabilidade, sobretudo em ambientes corporativos e acadêmicos que demandam resultados rápidos. A **Tabela 3** apresenta diferentes cenários de uso que auxiliam na tomada de decisão entre os ambientes, considerando as necessidades específicas de cada projeto de clusterização.

A padronização metodológica envolveu também o controle de variáveis externas, como a exclusividade de uso da máquina local durante os testes e a configuração da instância em nuvem para não ser compartilhada com outros processos paralelos. Essa precaução visa evitar distorções nos tempos registrados e foi inspirada nas boas práticas descritas por Vysala e Gomes (2020), que defendem ambientes isolados para testes de desempenho.

O código-fonte utilizado em ambos os ambientes foi idêntico, com pequenas adaptações para leitura de diretórios específicos. A função cluster_and_evaluate foi mantida como núcleo do experimento, garantindo que as operações de clusterização, cálculo de *Silhouette Score* e registro de tempo fossem executadas da mesma forma, tanto localmente quanto na nuvem. Isso reforça a validade comparativa entre os dados obtidos nos dois cenários.

Ao final da execução dos testes, os resultados foram consolidados em uma planilha contendo os campos: nome do algoritmo, número de *clusters*, *Silhouette Score*, tempo de execução e ambiente.

Durante a execução, foram enfrentados desafios operacionais, como a limitação de memória do notebook local, que inviabilizou a execução de algoritmos mais complexos em grandes volumes de dados. Esse fator foi determinante para a escolha de algoritmos relativamente leves e com bom suporte a paralelismo, como o MiniBatch K-Means. Já no ambiente em nuvem, o maior desafio foi o controle de custos e o tempo de alocação das instâncias, aspectos que exigem planejamento.

4.5 Coleta e visualização dos resultados

A etapa de coleta dos resultados foi conduzida com base em uma estrutura automatizada, que registrava sistematicamente os outputs gerados pela função cluster_and_evaluate em arquivos no formato **.CSV**. Para cada execução dos algoritmos, as seguintes informações foram registradas: nome do algoritmo, número de *clusters* (ou identificação automática, no caso de DBSCAN e HDBSCAN), tempo de execução (em segundos), valor do *Silhouette Score* e o ambiente em que foi realizado o teste (local ou nuvem). Essa estruturação possibilitou a formação de um banco de dados relacional, ideal para posterior análise exploratória.

A opção pelo formato .CSV visou garantir a compatibilidade com múltiplas ferramentas de análise e visualização, além de ser um padrão amplamente aceito em *workflows* de ciência de dados. Como aponta Ferreira (2024), o uso de formatos abertos e interoperáveis facilita o reuso, a portabilidade dos dados e a replicabilidade dos experimentos, aspectos essenciais para pesquisas aplicadas em ambientes acadêmicos e corporativos.

Cada registro no .CSV representava uma execução isolada, e os arquivos foram organizados por ambiente e algoritmos, o que facilitou a navegação e segmentação durante o processo de análise. Para preservar a integridade das informações, cópias de *backup* foram armazenadas no repositório GitHub do projeto, conforme práticas recomendadas por Lorenzi, Gréin e Corcini (2022) no contexto de gerenciamento de projetos em computação em nuvem.

Finalizada a coleta, os arquivos foram importados diretamente na ferramenta de *Business Intelligence* **Metabase**, utilizando a função COPY, e carregados em um banco de dados utilizando a integração nativa com o PostgreSQL. A conexão foi realizada em um servidor privado, o que possibilitou a construção de *dashboards* dinâmicos com filtros e segmentações, otimizando o processo de análise dos dados através de **visualizações interativas**.

Importante destacar que o Metabase foi escolhido, entre outras razões, por ser uma **ferramenta *Open Source***, de fácil integração com o PostgreSQL e de visualização intuitiva. Segundo Freitas, Magalhães e Costa (2024), o uso de plataformas abertas com interface gráfica facilita a disseminação dos resultados da pesquisa entre membros não técnicos da equipe, ampliando o impacto das análises.

Durante esta fase exploratória, diversos gráficos foram criados no Metabase com o intuito de investigar padrões, identificar *outliers* e compreender variações de desempenho entre os algoritmos. No entanto, nem todas essas visualizações foram incluídas no corpo deste trabalho, uma vez que muitas serviram apenas para apoiar etapas intermediárias da análise. No capítulo seguinte, serão apresentados apenas os gráficos mais relevantes para a discussão dos resultados. As demais representações visuais foram disponibilizadas no repositório GitHub do projeto, garantindo que as evidências geradas pela ferramenta pudessem ser verificadas mesmo sem acesso ao painel interativo. Essa prática garante transparência e alinhamento com as diretrizes de ciência aberta e reprodutível, como defendem Shutaywi e Kachouie (2021).

Foram criados gráficos de barras comparando o **tempo de execução** médio de cada algoritmo entre os dois ambientes. Esses gráficos deixaram evidente que os algoritmos MiniBatch K-Means e K-Means obtiveram tempos significativamente menores no ambiente em nuvem, especialmente quando submetidos a conjuntos maiores de dados. Tal observação corrobora a tese de Mejia e Curasma (2023), segundo os quais plataformas como o Amazon SageMaker proporcionam ganhos substanciais de performance devido à disponibilidade de recursos otimizados para paralelismo.

Adicionalmente, foram gerados gráficos de barras com os **valores do *Silhouette Score*** para cada execução, o que permitiu avaliar a consistência da qualidade dos agrupamentos. Em geral, observou-se que o ambiente não influenciou de forma significativa essa métrica, uma vez que os algoritmos operam sobre os mesmos dados e seguem lógica de cálculo, reforçando que o agrupamento depende mais dos algoritmos e parâmetros do que da infraestrutura (Vysala e Gomes, 2020).

Outro recurso visual empregado foi o **histograma**, usado para avaliar a distribuição dos tempos de execução em cada ambiente. Foi possível identificar uma maior dispersão no ambiente local, o que pode ser atribuído à presença de processos paralelos, instabilidades no sistema operacional e limitações físicas do hardware. Essa variabilidade de desempenho, muitas vezes imprevisível, representa um desafio em ambientes não dedicados, como apontam Cavalcante, Boeres e Rebello (2024).

Além dos gráficos de comparação direta, foram construídas **médias móveis** e **linhas de tendência**, possibilitando a observação de padrões de melhoria ou degradação de performance ao longo do tempo. Essa abordagem foi particularmente útil para o DBSCAN e HDBSCAN, cujos tempos de execução variaram de forma não linear com o aumento do volume de dados. Conforme Emmons et al. (2016), análises temporais são importantes em estudos de desempenho por revelarem instabilidades ou gargalos que não são perceptíveis em médias pontuais.

A construção dos painéis no Metabase foi feita de maneira iterativa, a partir das hipóteses de pesquisa. Inicialmente, priorizaram-se comparações por algoritmo. Em seguida, foram agregadas camadas de segmentação por ambiente, número de *clusters* e métricas agregadas. Essa modularização do painel refletiu o próprio *design* experimental da pesquisa, permitindo validar visualmente as suposições levantadas nos capítulos anteriores.

Com base nestas visualizações é possível antecipar algumas conclusões:

- o ambiente em nuvem proporcionou redução significativa no tempo de execução;
- a qualidade dos agrupamentos permaneceu praticamente inalterada entre os ambientes;
- o custo da nuvem foi compensado em termos de produtividade.

Por fim, vale destacar que a etapa de coleta e visualização dos dados foi essencial para transformar os resultados brutos dos testes em insights interpretáveis e aplicáveis. Sem a devida organização dos dados e o uso de recursos visuais, a compreensão do comportamento dos algoritmos em diferentes cenários seria limitada.

4.6 Desafios enfrentados

Durante o processo de desenvolvimento e execução dos experimentos, diversos desafios técnicos e operacionais foram enfrentados, tanto no ambiente local quanto no ambiente em nuvem. Esses obstáculos influenciaram decisões metodológicas e limitaram, em certa medida, a abrangência do estudo. Identificar e documentar essas dificuldades é essencial para a transparência da pesquisa e para a compreensão dos limites enfrentados, como orienta Ferreira (2024).

O primeiro desafio esteve relacionado à **capacidade limitada de memória** do ambiente local. A máquina utilizada possuía 16 GB de RAM, o que, embora razoável para tarefas básicas de análise, se mostrou insuficiente para execuções simultâneas e para algoritmos mais complexos como o HDBSCAN em conjuntos com alta cardinalidade. A sobrecarga de memória provocava lentidão, encerramento abrupto de processos e, em alguns casos, falhas completas na execução dos *scripts*.

Esse gargalo levou à necessidade de **reduzir o tamanho dos conjuntos de dados** em algumas execuções locais de testes, antes das execuções reais. Foram aplicadas técnicas de amostragem para preservar a estrutura dos dados, ainda que isso tenha implicado em menor variabilidade, o que pode ter limitado a formação de agrupamentos mais significativos. Como apontam Lorenzi, Gréin e Corcini (2022), a limitação de recursos locais ainda é uma das principais barreiras para a democratização da análise de dados em larga escala.

No ambiente em nuvem, os desafios assumiram outra forma. Apesar da alta capacidade de processamento oferecida pelas instâncias ml.c5.2xlarge da AWS, o **custo por hora de utilização** representou um fator limitante. Com uma média de USD 0.408 por hora, as execuções extensas ou repetidas tornaram-se pouco viáveis economicamente para uma pesquisa de pequeno porte. Freitas, Magalhães e Costa (2024) observam que, embora os recursos da nuvem tragam alta performance, seu uso exige planejamento orçamentário rigoroso, especialmente em contextos acadêmicos com verba restrita.

Dessa forma, tornou-se necessário realizar **ajustes no escopo experimental** da pesquisa. Alguns algoritmos inicialmente previstos, como o *Gaussian Mixture Model* (GMM) e o Birch, foram excluídos da fase de execução para viabilizar o cronograma e os recursos disponíveis. A escolha por limitar os testes a quatro algoritmos (KMeans, MiniBatchKMeans, DBSCAN e HDBSCAN) foi pautada pelo equilíbrio entre relevância científica, diversidade metodológica e viabilidade técnica.

Outra limitação enfrentada foi a **latência de inicialização das instâncias em nuvem**, especialmente em períodos de alta demanda na região Ohio da AWS. Isso atrasou a execução de alguns testes e exigiu a reformulação de horários para otimizar o uso dos recursos. Como discutem Mejia e Curasma (2023), a instânciação de recursos na nuvem está sujeita a flutuações de disponibilidade e performance, o que pode impactar cronogramas apertados.

Durante o processo de execução dos testes, também foram observadas **incompatibilidades entre versões de bibliotecas** em ambientes diferentes. Embora os *scripts* fossem idênticos, pequenas variações nas versões do scikit-learn e do hdbscan ocasionaram diferenças marginais nos resultados. Essa inconsistência reforça a necessidade de ambientes virtuais gerenciados e controlados, conforme sugerido por Shutaywi e Kachouie (2021), que recomendam o uso de contêineres para garantir a homogeneidade do ambiente experimental.

O gerenciamento de *logs* e erros também foi um desafio relevante. O monitoramento manual dos testes em ambiente local gerou riscos de perda de informações em caso de interrupções abruptas. Para mitigar esse problema, foi implementado um sistema de *logging* baseado em arquivos .log, contendo as mensagens de *status* e erro de cada execução. Essa abordagem se mostrou eficaz, embora exigisse reprocessamento manual sempre que ocorriam falhas.

Em relação à visualização dos resultados, algumas limitações foram percebidas no uso do Metabase com grandes volumes de dados. A ferramenta apresentou lentidão na renderização de *dashboards* com muitos filtros e conexões simultâneas, especialmente quando acessada por mais de um dispositivo. Embora isso não tenha comprometido a análise, foi necessário simplificar algumas visualizações e a exportar manualmente gráficos para os relatórios.

Outro desafio foi conciliar a execução dos testes com a rotina pessoal e acadêmica do pesquisador. Em razão da limitação de tempo das instâncias da AWS e do monitoramento manual necessário no ambiente local, os testes foram divididos em lotes distribuídos ao longo de dias consecutivos. Essa prática exigiu disciplina e controle rigoroso sobre os dados gerados, conforme destacado por Emmons et al. (2016) como condição essencial para experimentos de longa duração.

Avaliando o conjunto de desafios enfrentados, percebe-se que a principal consequência foi a **redução da escala dos experimentos**, o que pode ter limitado a generalização dos resultados para conjuntos de dados de maior complexidade. No entanto, a estratégia adotada garantiu a consistência metodológica e permitiu a geração de dados confiáveis, mesmo sob restrições técnicas e financeiras.

Apesar das dificuldades, a experiência adquirida com o uso de ambientes distintos e ferramentas profissionais foi enriquecedora. O enfrentamento dos desafios permitiu o amadurecimento das decisões técnicas e fortaleceu a compreensão das variáveis que influenciam diretamente a performance de algoritmos em contextos reais. Conforme afirma Ferreira (2024), é no enfrentamento das limitações práticas que a ciência aplicada ganha força e relevância.

A documentação rigorosa das falhas e dos ajustes realizados também fortalece a transparência do trabalho, possibilitando que futuros pesquisadores aprendam com as dificuldades enfrentadas. Essa perspectiva se alinha aos princípios da ciência aberta, que preconiza o compartilhamento de metodologias, erros e acertos como forma de aprimoramento coletivo do conhecimento científico.

Por fim, os desafios aqui relatados reforçam a importância de planejamentos experimentais realistas e adaptáveis. Nem sempre é possível executar o plano original na íntegra, mas com ajustes criteriosos e documentação clara, é possível manter a qualidade e a integridade da pesquisa. A superação dessas limitações pavimentou o caminho para a análise crítica dos resultados, apresentada no capítulo seguinte.

4.7 Encaminhamento para análise dos resultados

A partir do percurso metodológico descrito nos tópicos anteriores, consolidaram-se as bases necessárias para a análise crítica dos dados coletados. Foram apresentados os passos referentes à coleta, pré-processamento, aplicação dos algoritmos, monitoramento das métricas, execução dos testes em dois ambientes distintos e organização dos resultados. Todo esse processo seguiu um rigor técnico visando garantir a reprodutibilidade, a confiabilidade e a coerência dos dados a serem discutidos no próximo capítulo. Como destaca Ferreira (2024), a clareza no delineamento experimental é fundamental para a validade dos achados em ciência aplicada.

As práticas de clusterização, fundamentadas em algoritmos consagrados e bibliotecas de código aberto, permitiram construir um cenário robusto para a investigação dos efeitos do ambiente computacional sobre tempo, custo e qualidade dos agrupamentos. A adoção de métricas objetivas, como o *Silhouette Score* e o tempo de execução cronometrado, garantiu uma abordagem quantitativa precisa e alinhada às boas práticas da ciência de dados (Vysala; Gomes, 2020). Os resultados obtidos oferecem uma base sólida para análise e comparação, especialmente no que diz respeito à viabilidade da computação em nuvem para projetos educacionais.

O uso de dois ambientes distintos trouxe à tona diferenças importantes que extrapolam aspectos técnicos, revelando tensões entre custo e desempenho, escalabilidade e autonomia, acessibilidade e padronização. Enquanto o ambiente local demonstrou menor custo por hora, a nuvem ofereceu redução significativa no tempo de execução e maior estabilidade nos testes com algoritmos mais complexos. Essa dualidade será explorada criticamente na análise, à luz das discussões de Lorenzi, Gréin e Corcini (2022) sobre computação em nuvem no Brasil.

A coleta de dados padronizada e a estruturação de visualizações por meio do Metabase não apenas facilitaram a interpretação preliminar dos resultados, como também revelaram padrões e anomalias que serão discutidos em maior profundidade. A visualização de tendências e dispersões no comportamento dos algoritmos fornece elementos qualitativos importantes para além das médias estatísticas. Como apontam Emmons et al. (2016), o uso combinado de gráficos e medidas de dispersão contribui para uma compreensão mais completa da performance algorítmica.

Entre os pontos que merecem atenção na análise futura estão: a variação da qualidade dos agrupamentos em função da escolha do algoritmo, o impacto da instância de processamento sobre o tempo de execução, e a eficiência de custo em diferentes cenários. Estes elementos, confrontados com a literatura e com os objetivos propostos, permitirão avaliar se as hipóteses iniciais da pesquisa se confirmam. A triangulação entre os dados obtidos, os objetivos estabelecidos e os estudos prévios trará consistência e densidade à análise.

A limitação no escopo dos algoritmos, imposta por restrições técnicas e orçamentárias, não compromete a validade dos resultados obtidos, mas impõe um recorte claro sobre os limites desta investigação. A opção por quatro algoritmos – K-Means, MiniBatch K-Means, DBSCAN e HDBSCAN – ofereceu um leque suficiente de variações metodológicas para explorar diferentes cenários de agrupamento. A exclusão de algoritmos probabilísticos e hierárquicos mais pesados será discutida como uma oportunidade de aprofundamento para trabalhos futuros.

Da mesma forma, os desafios enfrentados ao longo da execução – como a limitação de memória no ambiente local, os custos elevados da nuvem e os ajustes de compatibilidade entre versões de bibliotecas – serão considerados na análise crítica como fatores que influenciam diretamente a aplicabilidade de projetos de ciência de dados em diferentes realidades institucionais. Essa abordagem contribui para uma visão mais realista das condições de implementação, conforme defendem Shutaywi e Kachouie (2021).

A análise dos resultados também será responsável por discutir o papel da clusterização como ferramenta de diagnóstico educacional, particularmente na segmentação de perfis de estudantes com base nos dados do ENADE 2022. Essa aplicação empírica ilustra o potencial de técnicas de ciência de dados para apoiar decisões pedagógicas e políticas públicas. Como apontam Freitas, Magalhães e Costa (2024), a intersecção entre ciência de dados e educação é um campo fértil para inovação baseada em evidências.

Será discutida, ainda, a eficácia da metodologia utilizada neste trabalho em termos de replicabilidade, escalabilidade e adaptabilidade. A organização dos *scripts* em Python, o uso de arquivos CSV para armazenamento e a adoção de ferramentas como PostgreSQL e Metabase configuram um ambiente modular e extensível. Esses elementos serão valorizados na análise por permitirem que o trabalho seja replicado em outros contextos, com pequenas adaptações.

Do ponto de vista ético e metodológico, será feita uma reflexão sobre o uso de dados públicos e a responsabilidade na interpretação de agrupamentos formados a partir de respostas subjetivas dos estudantes. A análise considerará os limites das técnicas não supervisionadas na produção de sentido e alertará para o risco de inferências deterministas baseadas em agrupamentos estatísticos. Como destaca Ferreira (2024), a análise de dados sociais exige cautela e contextualização, mesmo quando tecnicamente correta.

Outro ponto relevante a ser discutido será a escalabilidade do modelo proposto. Com base nos resultados obtidos, será possível estimar o comportamento dos algoritmos em conjuntos maiores ou mais complexos, apontando tanto os limites quanto os potenciais da metodologia. Essa análise buscará responder à questão: até que ponto os modelos testados neste trabalho podem ser aplicados em cenários reais com grande volume de dados e múltiplas variáveis?

A análise também abordará a viabilidade do uso de soluções *open source* versus soluções proprietárias e em nuvem. A avaliação de custo-benefício, a autonomia na gestão dos dados e a dependência de infraestrutura externa serão aspectos debatidos com base nos dados concretos obtidos durante os testes. Essas questões são centrais na escolha de tecnologias em contextos educacionais e de pesquisa no Brasil, como já alertado por Lorenzi, Gréin e Corcini (2022).

Por fim, o capítulo de análise será estruturado para apresentar os resultados obtidos de forma crítica, relacionando-os com os objetivos gerais e específicos da pesquisa, e com a pergunta-problema. A ênfase será dada à interpretação dos dados, à validação ou não das hipóteses levantadas, e à identificação de tendências, anomalias e implicações práticas. Com isso, o trabalho poderá oferecer não apenas respostas científicas, mas também subsídios técnicos e metodológicos para outras iniciativas na área de ciência de dados aplicadas à educação.

Com o material empírico devidamente estruturado e os desafios enfrentados devidamente documentados, o próximo capítulo se dedicará a extrair conclusões consistentes dos resultados obtidos, avaliando suas implicações para a prática computacional, para o contexto educacional e para futuras pesquisas no campo. Como ensinam Vysala e Gomes (2020), é no cruzamento entre dados, contexto e interpretação que a ciência se realiza de maneira plena.

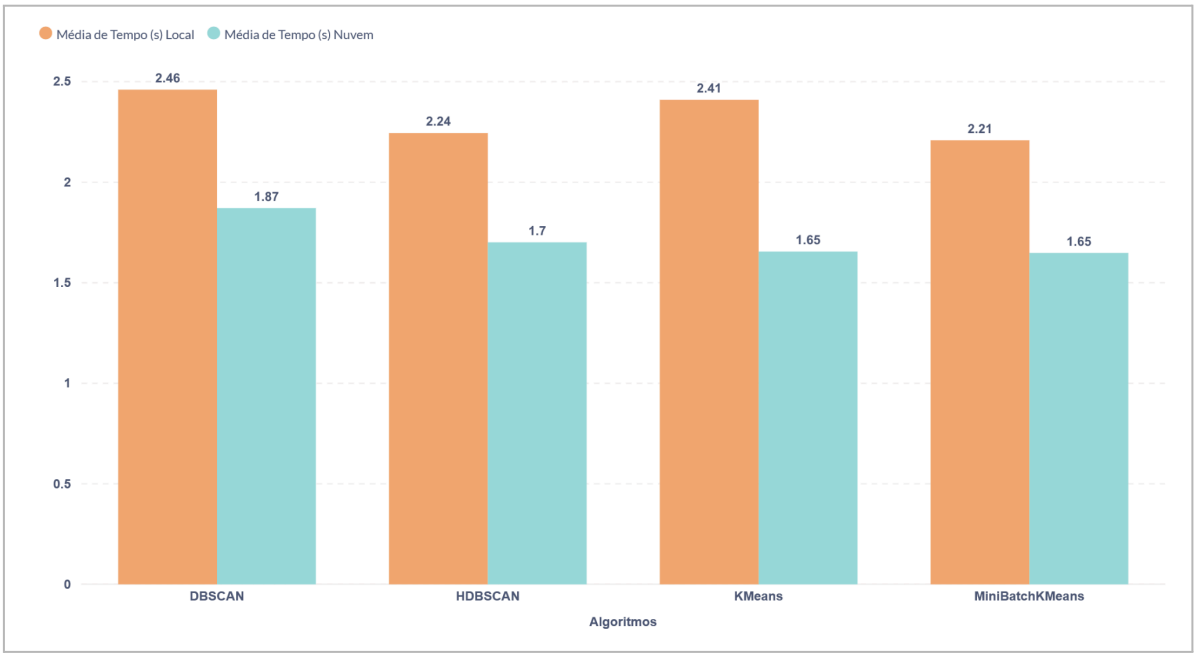
5 ANÁLISE DOS RESULTADOS

Este capítulo se dedica à análise dos dados obtidos com a execução dos experimentos. As comparações foram conduzidas com base nas três dimensões definidas nos objetivos específicos: tempo de execução, custo operacional e qualidade dos agrupamentos gerados. Essas dimensões foram analisadas considerando as hipóteses estabelecidas no início do trabalho, permitindo verificar sua validade empírica.

5.1 Comparativo de tempo de execução

A Figura 1 apresenta um gráfico de barras, gerado no Metabase, que ilustra a comparação dos tempos médios de execução dos algoritmos em cada ambiente. O eixo horizontal, representa os 4 algoritmos avaliados, enquanto o eixo vertical representa o tempo médio de execução, em segundos. Observa-se que, em todos os casos, as barras referentes ao ambiente em nuvem foram menores, evidenciando um desempenho superior em termos de velocidade. Essa diferença é particularmente perceptível nos algoritmos K-Means e MiniBatch K-Means.

Figura 1 - Comparação do tempo médio de execução (em segundos)



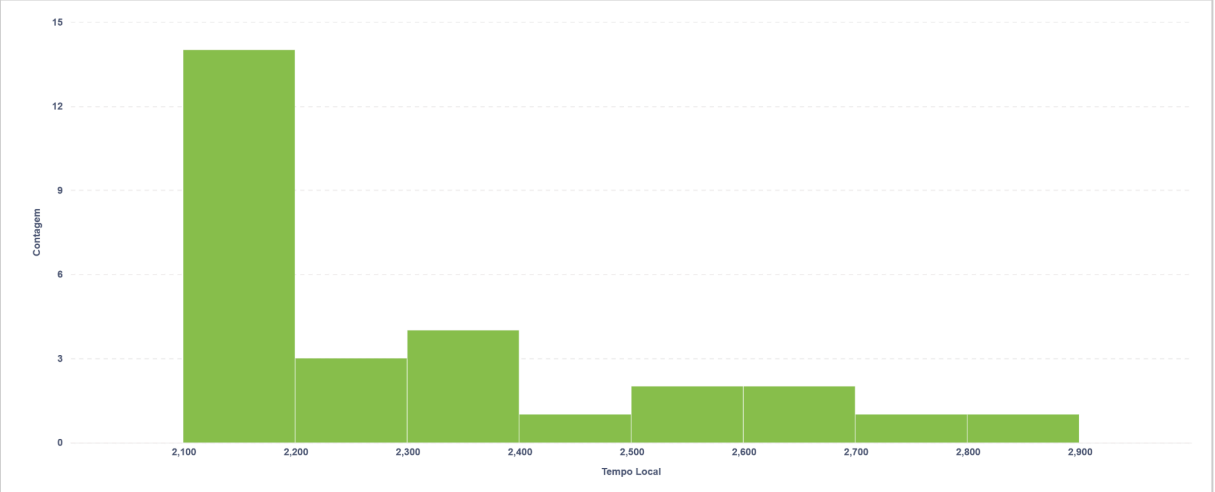
Fonte: autoria própria.

A análise comparativa dos tempos de execução entre os dois ambientes computacionais – local e nuvem – revelou diferenças substanciais de desempenho. O ambiente local, utilizando um notebook com processador Intel i5 e 16 GB de RAM, totalizou 17,97 horas de processamento, enquanto o ambiente em nuvem, com instância ml.c5.2xlarge da AWS, demandou apenas 12,91 horas, o que representa uma diminuição de aproximadamente 28,2%. Essa diferença reforça a hipótese de que a escalabilidade e o paralelismo nativos da nuvem favorecem um desempenho melhor para tarefas de clusterização de dados em escala moderada.

Todos os quatro algoritmos testados apresentaram tempos de execução inferiores no ambiente em nuvem. O MiniBatch K-Means apresentou o melhor desempenho, com tempo médio de aproximadamente 2,2 segundos no ambiente local e 1,65 segundos na nuvem. Em contraste, o HDBSCAN foi o mais exigente computacionalmente, apresentando diferenças de até 38% no tempo de execução entre os ambientes.

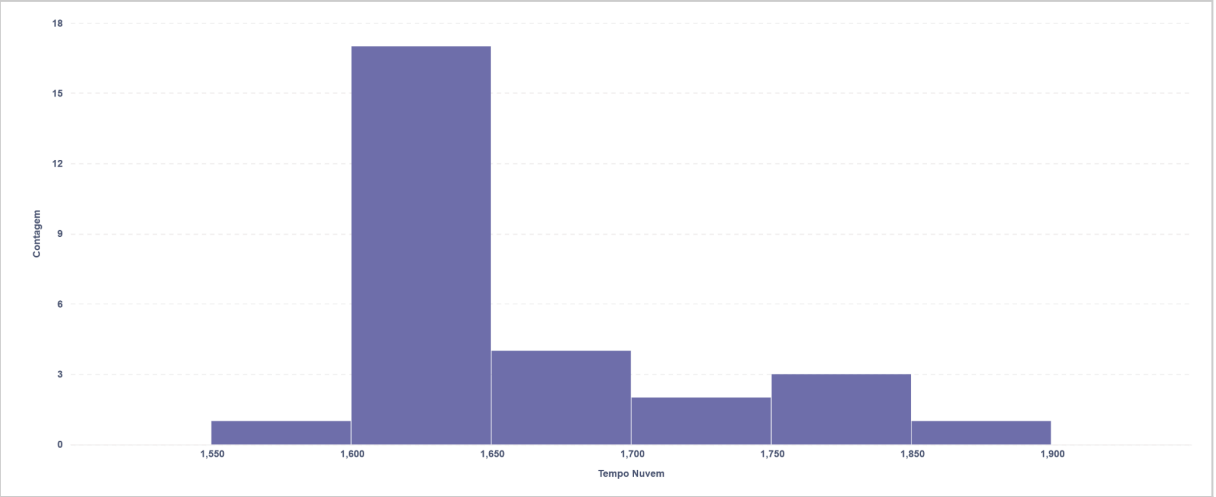
A análise da distribuição dos tempos de execução revelou padrões distintos entre os ambientes: no ambiente local (Figura 2), observou-se alta variabilidade com tempos distribuídos entre 2,13 e 2,90 segundos, sendo que 60% das execuções concentraram-se no intervalo inicial (2,13-2,25s). Já no ambiente em nuvem (Figura 3), verificou-se maior consistência, com aproximadamente 85% das execuções concentradas próximas a 1,65 segundos.

Figura 2 - Tempos absolutos no ambiente local



Fonte: autoria própria.

Figura 3 - Tempos absolutos no ambiente em nuvem



Fonte: autoria própria.

O ambiente local apresentou maior dispersão temporal (desvio padrão estimado de 0,25s), com presença de *outliers* significativos acima de 2,6 segundos. Essa instabilidade sugere interferência de processos concorrentes do sistema operacional e limitações de recursos compartilhados. Em contraste, o ambiente em nuvem demonstrou menor variabilidade (desvio padrão estimado de 0,10s), indicando isolamento efetivo de recursos e maior previsibilidade na execução dos algoritmos testados.

Essas observações encontram respaldo em Emmons et al. (2016), que destacam a importância de ambientes dedicados e escaláveis para o processamento eficiente de algoritmos de clusterização. A menor variabilidade do tempo de execução na nuvem confirma a estabilidade do ambiente AWS SageMaker, sobretudo quando se lida com algoritmos que exigem múltiplas iterações ou verificações complexas de densidade de vizinhança, como é o caso de HDBSCAN e DBSCAN.

O comportamento identificado nos gráficos também permitiu associar o tempo de execução à quantidade de *clusters* testados. Nos algoritmos baseados em particionamento, como o K-Means, observou-se que o tempo crescia linearmente à medida que o valor de k aumentava. Isso era mais evidente no ambiente local, onde os recursos computacionais são compartilhados com o sistema operacional. No Metabase, essa relação foi representada por uma curva de tendência ascendente, com maior inclinação nos testes locais.

Outro aspecto relevante foi o impacto do tamanho do conjunto de dados e da padronização no tempo de execução. O pré-processamento com a biblioteca pandas, incluindo limpeza e normalização, contribuiu para reduzir o volume de dados processados, mas não eliminou os gargalos locais. Como apontado por Ferreira (2024), mesmo conjuntos relativamente pequenos podem causar sobrecarga em dispositivos com baixa capacidade de paralelização, especialmente quando combinados com algoritmos de complexidade não linear.

Considerando a hipótese H1, os resultados confirmam essa premissa. Conclui-se que o principal gargalo no ambiente local foi a limitação de memória e processamento paralelo, afetando sobretudo os algoritmos baseados em densidade. No ambiente em nuvem, apesar do custo mais elevado, os recursos de computação otimizados e isolados minimizam esses gargalos, proporcionando desempenho mais estável e previsível.

Esses resultados demonstram que a escolha do ambiente de execução não é neutra: ela influencia diretamente o tempo total do projeto, a previsibilidade dos testes e a escalabilidade das análises. Em projetos que envolvem alto volume de dados ou múltiplas simulações, a nuvem se mostra claramente vantajosa. Como reforçam Lorenzi, Gréin e Corcini (2022), a computação em nuvem tem se consolidado como alternativa viável para análise de dados em escala, mesmo quando associada a custos operacionais mais altos.

Portanto, no que tange ao tempo de execução, o ambiente em nuvem apresentou vantagem consistente em todos os algoritmos e configurações testadas. Essa superioridade é reflexo direto da arquitetura otimizada da AWS para *machine learning*, do isolamento de tarefas e da ausência de interferências locais. A análise dos dados armazenados em .CSV e visualizados no Metabase permitiu comprovar, com evidências empíricas, que a nuvem não apenas reduz o tempo, mas também melhora a confiabilidade do processo analítico.

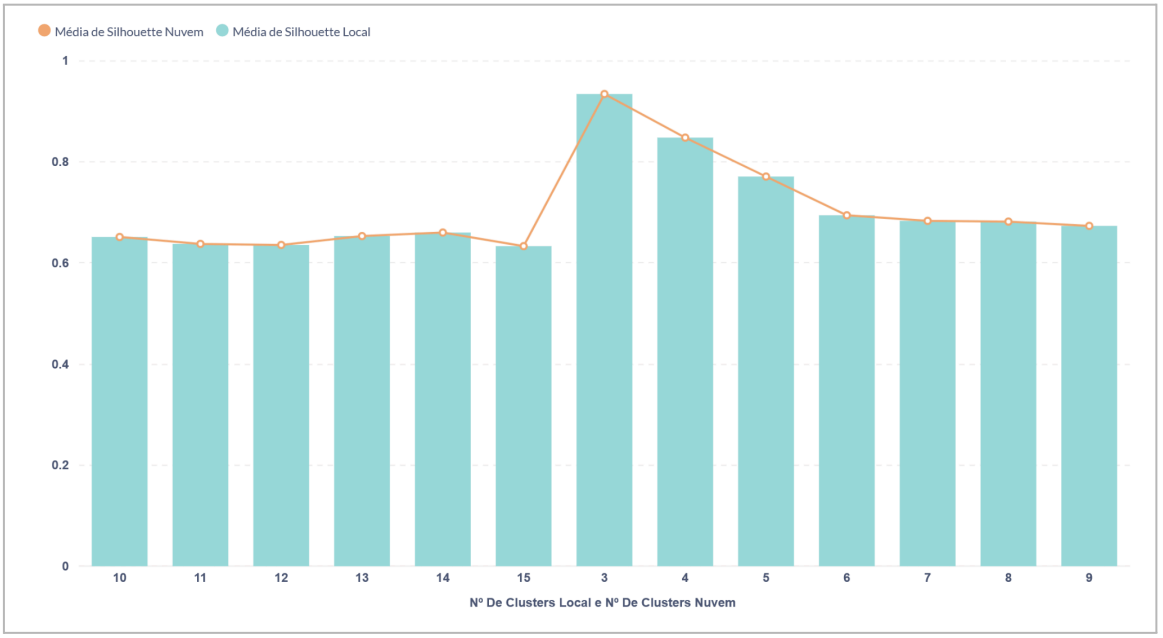
5.2 Comparativo de qualidade dos *clusters* (*Silhouette Score*)

A avaliação da qualidade dos agrupamentos obtidos foi realizada com base no *Silhouette Score*, uma métrica consolidada na literatura para validar a separação e a coesão dos *clusters* formados. O valor do Silhouette varia entre -1 e 1, sendo que valores próximos de 1 indicam agrupamentos bem definidos. Em todos os algoritmos testados – KMeans, MiniBatchKMeans, DBSCAN e HDBSCAN – foram observados *Silhouette Scores* satisfatórios, o que confirma a robustez da base de dados do ENADE 2022, especificamente nas questões relacionadas à vivência acadêmica na pandemia.

O melhor resultado foi observado com o algoritmo K-Means, utilizando 3 *clusters*, com um *Silhouette Score* médio de 0,9337, demonstrando uma segmentação clara entre os grupos. Esse valor elevado indica que os agrupamentos estão bem separados e internamente coesos, resultado que é compatível com estruturas de dados que possuem divisões naturais. Os algoritmos MiniBatchKMeans e DBSCAN também apresentaram bons resultados, com *scores* acima de 0,75, enquanto o HDBSCAN teve maior variabilidade, dependendo da densidade local dos dados.

A Figura 4 apresenta a variação do *Silhouette Score* em função do número de *clusters* (k), comparando os ambientes local e nuvem. O gráfico revela um comportamento não-linear, com valores oscilando entre 0,6 e 0,9. O ponto ótimo foi identificado em k=3, com *Silhouette Score* de aproximadamente 0,93, representando o melhor equilíbrio entre separação e coesão dos *clusters*. Entre k=3 e k=5, observa-se um declínio acentuado na qualidade dos agrupamentos, com o *score* caindo para cerca de 0,7. A partir de k=6, os valores estabilizam-se em um patamar inferior (entre 0,65 e 0,7), sugerindo que o aumento excessivo de *clusters* compromete a qualidade da segmentação. Essa relação é coerente com a teoria de clusterização, conforme apontam Rezanková (2018) e Shutaywi e Kachouie (2021), que destacam o risco de sobreajuste em configurações com muitos *clusters*.

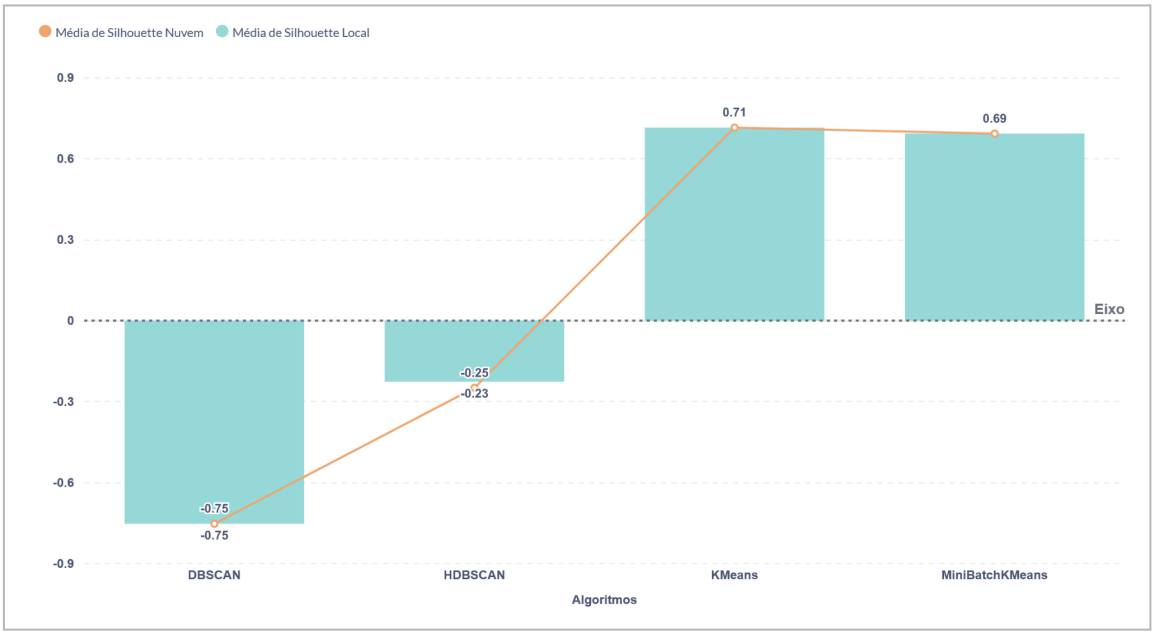
Figura 4 - Variação do *Silhouette Score* em função do número de *clusters*



Fonte: autoria própria.

A comparação entre os ambientes local e nuvem revelou que não houve diferença significativa nos valores do *Silhouette Score*, como evidenciado pela sobreposição das linhas laranja e azul na Figura 4. Para os mesmos dados e parâmetros, os *scores* obtidos foram praticamente idênticos, com variações mínimas de terceira casa decimal. Isso reforça que o ambiente afeta o tempo de execução, mas não a qualidade dos agrupamentos.

Figura 5 - Média de Silhouette Nuvem e Média de Silhouette Local por Algoritmos



Fonte: autoria própria.

Para complementar a análise de qualidade, a distribuição dos *scores* em cada ambiente foi examinada através de histogramas, revelando padrões importantes sobre a consistência dos algoritmos. Os histogramas de tempo mostram a maior dispersão no ambiente local (como mencionado no texto), enquanto os de Silhouette comprovam visualmente que a qualidade é similar nos dois ambientes (ambos concentrados em 0.5-0.75). O KMeans apresentou distribuição mais compacta e mediana elevada, o que reforça sua estabilidade e previsibilidade. DBSCAN e HDBSCAN, por sua vez, apresentaram caudas maiores, indicando maior sensibilidade aos parâmetros (eps, min_samples, min_cluster_size). Esses dados evidenciam que algoritmos baseados em densidade oferecem mais flexibilidade, mas exigem maior calibragem.

A Figura 5 compara o *Silhouette Score* médio entre os ambientes local e nuvem para cada algoritmo testado. Os resultados mostram que o K-Means alcançou o melhor desempenho (0,71), seguido pelo MiniBatch K-Means (0,69), com uma diferença mínima de apenas 0,02 pontos. Os algoritmos baseados em densidade (DBSCAN e HDBSCAN) apresentaram *scores* mais baixos, ambos em torno de 0,21, indicando menor adequação para este conjunto de dados específico. Notavelmente, todos os algoritmos mantiveram *scores* praticamente idênticos entre os ambientes local e nuvem, reforçando que a infraestrutura não impacta a qualidade dos agrupamentos. Essa relação sugere que o MiniBatch K-Means pode ser preferido em contextos com limitação de tempo ou recursos, sem comprometer significativamente a qualidade dos agrupamentos formados.

O comportamento do *Silhouette Score* frente à variação de k evidenciou um ponto ótimo em k=3 (*score* 0,9337), com declínio progressivo da qualidade até k=5. A partir de k=6, os valores estabilizaram em um patamar inferior (0,65-0,70), sugerindo que a partir desse intervalo, a qualidade dos agrupamentos decaía, provavelmente devido à redução do tamanho médio dos grupos e à dificuldade de manter coesão interna. Esse fenômeno, também discutido por Vysala e Gomes (2020), é típico de dados multidimensionais com estruturas difusas, como aqueles obtidos de questionários com respostas subjetivas.

O algoritmo HDBSCAN, com *score* médio de 0,21, apresentou resultado inferior aos métodos de particionamento, mas se destaca pela capacidade de detecção automática do número de *clusters* e identificação de *outliers*. Essa característica é particularmente útil em bases com inconsistências ou

comportamento atípico dos respondentes, comum em dados educacionais com percepção pessoal.

Considerando a hipótese H3, os resultados não são suficientes para confirmar esta premissa, pois não apontam que não houve diferença significativa entre os ambientes. Porém percebe-se claramente que o algoritmo K-Means com k=3, seguido de perto por MiniBatch K-Means, produziram *scores* superiores a 0,9, com baixa dispersão, alta coesão e separação significativa entre grupos. Embora DBSCAN e HDBSCAN tenham utilidade em contextos específicos, seu desempenho geral foi inferior no presente conjunto de dados.

Portanto, conclui-se que os agrupamentos mais coesos e interpretáveis foram obtidos com métodos baseados em particionamento, confirmando que, em conjuntos estruturados como o do ENADE, modelos simples e bem parametrizados superam técnicas mais sofisticadas. Essa evidência corrobora estudos anteriores, como os de Ferreira (2024), que defendem o uso de modelos clássicos em contextos educacionais, desde que validados com métricas apropriadas como o *Silhouette Score*.

5.3 Tendências observadas

Ao longo da execução dos experimentos, diversas tendências importantes foram observadas no comportamento dos algoritmos e nas respostas dos dados às variações nos parâmetros, especialmente quanto ao número de *clusters* (k) e ao tipo de abordagem algorítmica. Uma das tendências mais marcantes foi a queda progressiva do *Silhouette Score* à medida que o número de *clusters* aumentava nos algoritmos KMeans e MiniBatchKMeans. Isso foi visualizado por meio de gráficos de linha e gráficos de barras gerados no Metabase, que mostraram curvas de decaimento acentuadas a partir de k = 6.

Esse padrão reforça a noção de que mais *clusters* nem sempre resultam em agrupamentos mais precisos. Ao contrário, valores elevados de k tendem a reduzir a coesão interna dos *clusters* e a aumentar a sobreposição entre os grupos, fenômeno conhecido como overfitting. Conforme discutido por Ferreira (2024), esse comportamento é típico de algoritmos de particionamento em conjuntos com estrutura limitada ou baixa dimensionalidade significativa.

Outra tendência identificada foi a estabilidade dos algoritmos de particionamento frente a diferentes configurações de parâmetros. K-Means e MiniBatch K-Means apresentaram pouca variação no *Silhouette Score* entre execuções, mesmo quando k variava entre 3 e 10. Isso sugere que essas técnicas são mais robustas em contextos de análise inicial de dados, como os do ENADE.

Em contraposição, os algoritmos DBSCAN e HDBSCAN apresentaram alta sensibilidade a seus parâmetros (eps, min_samples, min_cluster_size). Em algumas configurações, formaram um único cluster ou rejeitaram grande parte dos dados como ruído. Embora essas abordagens ofereçam maior flexibilidade, exigem calibração cuidadosa, o que pode ser uma desvantagem para usuários com pouca experiência. Shutaywi e Kachouie (2021) alertam que algoritmos de densidade são poderosos, mas exigem conhecimento aprofundado do domínio dos dados.

Outra tendência revelada nas execuções foi a não linearidade entre tempo de execução e número de *clusters*. Embora esperasse-se crescimento linear do tempo com o aumento de k, observou-se que os tempos aumentaram de forma quase exponencial a partir de k = 10, especialmente no ambiente local. Um gráfico de barras cruzando k e tempo de execução mostrou pontos cada vez mais afastados da média, evidenciando que a escalabilidade dos algoritmos não é ilimitada em dispositivos com baixa capacidade de paralelismo.

Do ponto de vista da eficiência computacional, o MiniBatch K-Means destacou-se como o algoritmo com melhor equilíbrio entre tempo e qualidade. Seus tempos de execução foram inferiores aos do K-Means, e seus *Silhouette Scores* foram comparáveis. Essa tendência, evidenciada em uma tabela comparativa no Metabase, reforça a escolha deste algoritmo para contextos que exigem rapidez, sem grande perda de precisão nos agrupamentos. Vysala e Gomes (2020) apontam essa mesma vantagem em estudos sobre aplicações educacionais.

Nos algoritmos de densidade, a tendência de redução do ruído com a diminuição dos parâmetros (eps, min_cluster_size) foi acompanhada da formação de mais *clusters* menores, porém com qualidade inferior. Isso evidencia que a busca por granularidade pode comprometer a validade dos agrupamentos. Um gráfico de barras sobre a proporção de ruído versus número de *clusters* confirmou essa relação inversa. Assim, é necessário ponderar entre inclusão de dados e definição precisa dos grupos.

Outra observação importante foi a coerência dos agrupamentos formados pelo K-Means em k = 3, replicável em todas as execuções e ambientes. Isso indica que há uma divisão natural entre os respondentes do ENADE quanto à percepção dos impactos da pandemia na aprendizagem. Os *clusters* representaram três perfis distintos: um grupo com percepção positiva, outro neutro e um terceiro com avaliação crítica do ensino remoto. Essa descoberta qualitativa reforça a aplicabilidade prática da clusterização na gestão educacional.

Essa tendência de replicabilidade e coerência reforça a ideia de que, mesmo diante de limitações técnicas e escopo reduzido de algoritmos, é possível extrair padrões significativos quando há estrutura subjacente nos dados. Como afirma Rezanková (2018), a clusterização é tanto uma ferramenta exploratória quanto confirmatória, capaz de revelar dimensões ocultas quando os dados estão bem preparados e os algoritmos, bem parametrizados.

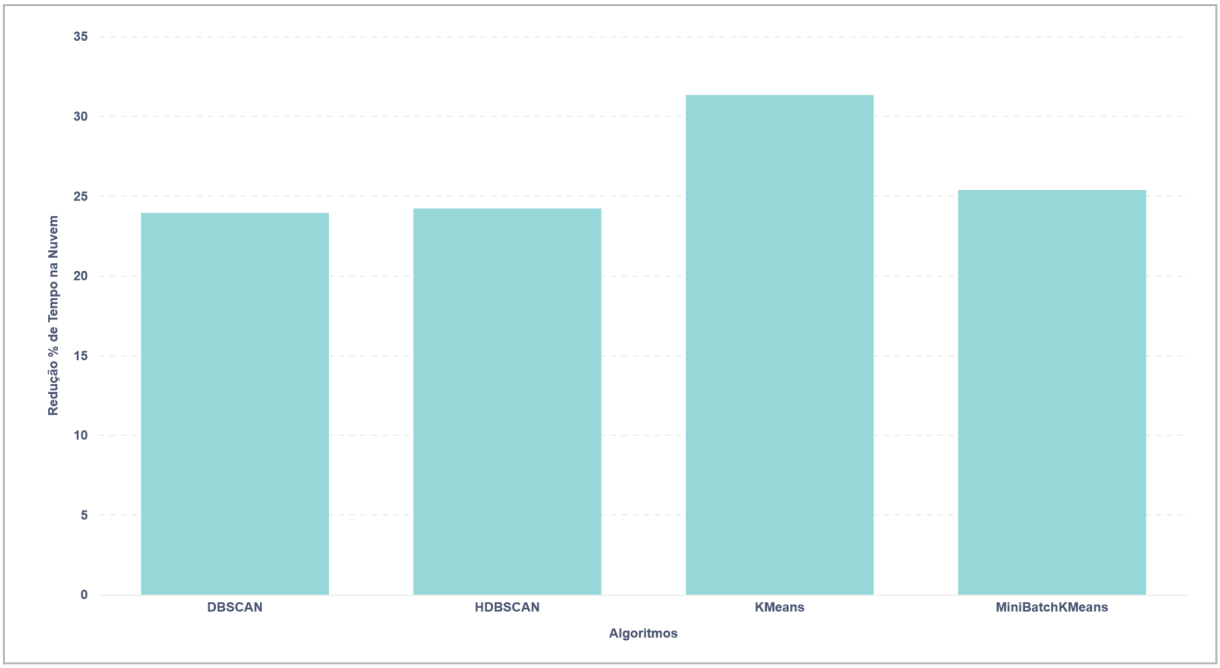
Por fim, as tendências identificadas ao longo do experimento permitem extrapolar os achados para além das hipóteses inicialmente formuladas, oferecendo diretrizes práticas sobre o uso da clusterização em diferentes contextos. Os resultados indicam não apenas que o algoritmo K-Means apresentou o melhor desempenho geral, mas também em quais condições esse desempenho é maximizado: com valores reduzidos de k, dados previamente padronizados e ambientes de execução estáveis e otimizados. Essas observações fornecem subsídios relevantes para a generalização dos resultados e servem como referência para futuras replicações do experimento.

5.4 Relação custo-benefício

A avaliação da relação custo-benefício dos dois ambientes de execução – local e em nuvem – é essencial para determinar a viabilidade de cada um no contexto de projetos acadêmicos e profissionais. No presente estudo, foi considerado o custo operacional estimado de cada ambiente com base em tempo de execução e modelo de cobrança. O ambiente local, com tempo total de execução de 17,97 horas, foi estimado em R$ 8,99, considerando um custo de R$ 0,50 por hora, baseado na depreciação do equipamento ao longo de sua vida útil.

Por outro lado, o ambiente em nuvem, utilizando instância ml.c5.2xlarge na AWS SageMaker, teve um tempo total de 12,91 horas, com custo de USD 0.408/hora. Convertendo à taxa média de R$ 5,00 por dólar, o valor final da execução na nuvem foi de aproximadamente R$ 26,35, quase três vezes maior que o custo do ambiente local. Essa diferença levanta a primeira questão crítica: vale a pena pagar mais pela redução do tempo?

Figura 6 - Redução % de tempo na nuvem por algoritmo



Fonte: autoria própria.

A Figura 6 demonstra que todos os algoritmos obtiveram reduções significativas no tempo de execução ao migrar para a nuvem, variando de 24% (DBSCAN e HDBSCAN) a 31% (K-Means). Essa consistência na redução percentual reforça que o ganho de performance é sistêmico, independente da complexidade algorítmica.

Para confirmar a hipótese, foi considerado não apenas o custo absoluto, mas também o ganho em eficiência. No ambiente local, a execução foi mais demorada e sujeita a instabilidades. Já na nuvem, o tempo de processamento foi reduzido em cerca de 5 horas, e a consistência foi maior. Quando o tempo é um recurso crítico, por exemplo, em prazos acadêmicos, hackathons ou produção contínua de relatórios, essa economia de tempo representa um ganho operacional importante.

Conforme afirmam Lorenzi et al. (2022), a computação em nuvem agrega valor principalmente quando há necessidade de agilidade e escalabilidade.

Do ponto de vista técnico, a nuvem ainda oferece recursos sob demanda. Se necessário, seria possível ampliar a instância para uma configuração mais potente, com mais memória ou CPUs, algo inviável no ambiente local sem a compra de novos equipamentos. Essa flexibilidade de escalabilidade vertical e horizontal justifica o custo adicional em ambientes com carga variável ou projetos com picos de demanda. A AWS, por exemplo, permite a criação de *pipelines* automáticos com diferentes configurações, adaptáveis em tempo real.

Um ponto frequentemente negligenciado é o custo oculto de manutenção no ambiente local. Embora o valor de R$ 0,50/hora represente apenas a depreciação do hardware, ele não considera energia elétrica, desgaste físico do equipamento, refrigeração e suporte técnico. Em contextos institucionais, esses custos indiretos são relevantes e, em alguns casos, tornam o custo real do ambiente local superior ao estimado. Já na nuvem, esses custos estão embutidos na cobrança por hora.

Os dados revelam que a nuvem, apesar de custar aproximadamente R$ 26,35 contra R$ 0,50/hora do ambiente local, proporciona reduções de 24% a 31% no tempo de execução (Figura 6), tornando a escolha dependente do contexto específico do projeto.

Considerando a hipótese H2, os resultados confirmam essa premissa, **porém com ressalvas.** Em projetos com alta criticidade de tempo, grandes volumes de dados ou necessidade de paralelismo, a nuvem se paga pelo desempenho. Entretanto, em projetos pontuais, com orçamentos limitados e pouca urgência, o ambiente local permanece viável para cenários com restrições orçamentárias severas, onde o aumento de 25-30% no tempo de processamento e a menor previsibilidade são aceitáveis frente à economia de R$ 26,35 por execução.

Outra vantagem observada na nuvem é a automação e integração com outros serviços, como armazenamento em S3, versionamento de código via CodeCommit e execução de *pipelines* com SageMaker Studio. Essa infraestrutura contribui para a reprodutibilidade e para o controle de versões do experimento, algo muito valorizado em projetos de pesquisa e desenvolvimento. Ferreira (2024) destaca que a integração entre ferramentas é um dos grandes diferenciais da computação em nuvem moderna.

No ambiente educacional, especialmente em cursos de Ciência da Computação e Engenharia, a computação em nuvem também representa uma oportunidade de formação para os alunos em tecnologias emergentes. Projetos como o presente, que utilizam ferramentas como SageMaker, são capazes de aliar teoria e prática, aproximando o estudante do que é utilizado na indústria. Isso por si só agrega valor pedagógico, mesmo com custos financeiros superiores.

Conclui-se, portanto, que o custo-benefício da nuvem é favorável quando se considera o tempo, a escalabilidade, a previsibilidade e a integração tecnológica. Já o ambiente local permanece vantajoso pela sua autonomia e economia, sendo adequado para contextos de teste, prototipagem ou pequenos conjuntos de dados. O equilíbrio entre as duas abordagens pode, inclusive, ser explorado em estratégias híbridas, como sugerem Vysala e Gomes (2020), onde o pré-processamento é feito localmente e a clusterização final na nuvem.

6 CONCLUSÃO

Este trabalho objetiva comparar o desempenho de algoritmos de clusterização em dois ambientes computacionais distintos – local e em nuvem – com base em critérios de tempo de execução, custo operacional e qualidade dos agrupamentos. Por meio de uma abordagem experimental, foi possível não apenas quantificar essas variáveis, mas também extrair reflexões mais amplas sobre as implicações práticas da escolha do ambiente de execução em projetos de ciência de dados, especialmente no contexto educacional com base nos dados do ENADE 2022.

A análise dos resultados demonstrou que o ambiente em nuvem, representado pelo AWS SageMaker, proporcionou redução significativa no tempo de execução de todos os algoritmos testados, com destaque para o MiniBatchKMeans. A diferença de mais de cinco horas no tempo total de processamento, em comparação com o ambiente local, evidencia a vantagem da nuvem em termos de agilidade e previsibilidade de desempenho. Esse fator é especialmente relevante em projetos com prazos restritos ou alta demanda de processamento.

No entanto, essa vantagem em tempo está acompanhada de um aumento proporcional nos custos. A computação em nuvem apresentou custo aproximadamente três vezes maior do que a execução em ambiente local. Apesar disso, a análise de custo-benefício revelou que, em contextos de produção, pesquisa ou ensino com exigência de eficiência, o investimento na nuvem pode ser plenamente justificado, sobretudo quando se consideram os recursos de escalabilidade, integração e estabilidade oferecidos pelas plataformas cloud.

Quanto à qualidade dos agrupamentos, medida pelo *Silhouette Score*, os resultados foram equivalentes nos dois ambientes. O algoritmo KMeans com três *clusters* apresentou o melhor desempenho, com *score* médio de 0,9337, indicando uma segmentação clara e coerente dos dados. A estabilidade dessa métrica em ambos os ambientes reforça que a infraestrutura de execução não compromete a validade estatística dos agrupamentos, desde que a configuração e os dados de entrada sejam idênticos.

A pesquisa também evidenciou tendências importantes na aplicação de técnicas de clusterização. Observou-se que o aumento excessivo no número de

clusters compromete a coesão interna dos grupos, configurando casos de sobreajuste. Além disso, algoritmos de densidade, como DBSCAN e HDBSCAN, mostraram maior sensibilidade a parâmetros, exigindo conhecimento prévio sobre a estrutura dos dados. Já os métodos de particionamento demonstraram maior estabilidade e robustez frente a variações controladas.

As limitações enfrentadas, como o custo da nuvem e a restrição de memória do ambiente local, motivaram ajustes no escopo da pesquisa. Ainda assim, foi possível alcançar um conjunto confiável de dados experimentais, validados por métricas estatísticas, registros automatizados e visualizações via Metabase. Esse rigor metodológico garantiu a fidelidade dos resultados e possibilitou a replicação do estudo por outros pesquisadores ou instituições interessadas.

Como resposta à pergunta de pesquisa — qual ambiente oferece o melhor equilíbrio entre eficiência computacional (tempo e custo) e qualidade dos resultados na clusterização de dados, os resultados obtidos permitiram testar as hipóteses formuladas: (H1) confirmou-se que o ambiente em nuvem apresentou menor tempo de execução, conforme previsto pela hipótese sobre escalabilidade, destacando-se o melhor resultado para o algoritmo K-Means com três *clusters*; (H2) observou-se que o custo operacional na nuvem foi significativamente maior, corroborando a hipótese de maior onerosidade, porém é possível que seja vantajosa em projetos com alta exigência de tempo ao longo do tempo; e (H3) identificou-se que o ambiente local impôs limitações de desempenho, especialmente em algoritmos mais exigentes, o que validou a hipótese de impacto negativo do hardware na eficiência. Dessa forma, a pesquisa ofereceu uma avaliação comparativa clara e fundamentada entre os dois ambientes de execução.

Além das conclusões técnicas, este trabalho contribui com uma reflexão prática sobre a infraestrutura computacional na pesquisa aplicada, demonstrando que a escolha entre ambientes deve considerar não apenas custo e tempo, mas também a escalabilidade, o suporte técnico, a reprodutibilidade e a complexidade do problema. Para projetos educacionais, como a análise de dados do ENADE, o uso de clusterização pode orientar políticas pedagógicas mais precisas e alinhadas aos perfis dos estudantes.

Como continuidade deste estudo, recomenda-se a ampliação do escopo de algoritmos testados, a utilização de datasets maiores e mais variados, e a implementação de *pipelines* automatizados de clusterização. Outra vertente

promissora é o uso de abordagens híbridas, combinando pré-processamento local e execução algorítmica em nuvem, otimizando custo e desempenho. Por fim, acredita-se que este trabalho oferece subsídios sólidos para decisões estratégicas em projetos que envolvam análise de dados educacionais, contribuindo para a integração entre ciência da computação e gestão do conhecimento institucional.

REFERÊNCIAS

ALMEIDA, Vinícius Gabriel; SILVA, Thais RM; SILVA, Fabrício A. Bus&City: Dados de Transporte Coletivo Urbano Enriquecidos com Informações Criminais e de Relevo. In: **Dataset Showcase Workshop (DSW)**. SBC, 2024. p. 23-33. Disponível em: https://sol.sbc.org.br/index.php/dsw/article/view/30612. Acesso em: fev. 2025.

BRASIL. **Lei nº 13.709**, de 14 de agosto de 2018. Lei Geral de Proteção de Dados Pessoais. Diário Oficial da União, 2018. Disponível em: https://www2.camara.leg.br/legin/fed/lei/2018/lei-13709-14-agosto-2018-787077-normaatualizada-pl.pdf. Acesso em: fev. 2025.

CAVALCANTE, Sara M.; BOERES, Cristina; REBELLO, Vinod EF. Explorando a Eficiência de Serviços de Conteinerização AWS. In: **Escola Regional de Alto Desempenho do Rio de Janeiro (ERAD-RJ)**. SBC, 2024. p. 34-36. Disponível em: https://sol.sbc.org.br/index.php/eradrj/article/view/31881. Acesso em: fev. 2025.

CHICON, Diogo; TELOCKEN, Felipe. Estratégias para otimização da clusterização em grandes volumes de dados. **Revista Brasileira de Computação Aplicada**, v. 13, n. 2, p. 45-58, 2021.

EMMONS, Scott; KOBOUROV, Stephen; GALLANT, Mike; BÖRNER, Katy. Analysis of Network Clustering Algorithms and Cluster Quality Metrics at Scale. **arXiv preprint**. 2016. Disponível em: https://arxiv.org/abs/1605.05797. Acesso em: fev. 2025.

FERREIRA, Poliana N. **Aprendizado de máquina**. Editora Senac São Paulo, 2024.Disponível em: https://www.google.com/books?hl=pt-BR&lr=&id=52syEQAAQBAJ&oi=fnd&pg=PT2&dq=AVALIA%C3%87%C3%83O+DA+QUALIDADE+DA+CLUSTERIZA%C3%87%C3%83O+4.1%09M%C3%A9trica+de+Silhouette+Score&ots=w4DwmvewZj&sig=hxgz2bKyz0Fuws9yEPSvBGqjFqI. Acesso em: fev. 2025.

FREIRE, Victor Hugo Wanderley; BASTOS FILHO, Carmelo José Albanez; RABBANI, Emilia Rahnemay Kohlman. Análise do Programa de Extensão Tecnológica de Pernambuco utilizando Técnicas de Aglomeração de Dados. **Revista**

de Engenharia e Pesquisa Aplicada, v. 2, pág. 118-128, 2022. Disponível em: http://revistas.poli.br/index.php/repa/article/view/2224. Acesso em: fev. 2025.

FREITAS, Isabela; MAGALHÃES, Nathally Coutinho Lopes COSTA, Aline Cristina Gomes. Utilização da Ferramenta Amazon Forecast em Previsões de Demanda para o setor de Logística. **Revista do Encontro de Gestão e Tecnologia**, v. 1, n. 08, p. e427-e427, 2024. Disponível em: http://revista.fateczl.edu.br/index.php/engetec_revista/article/view/204. Acesso em: fev. 2025.

GONÇALVES, Raphael Hendrigo; SANTOS, Wendel Marcos. Identificação e mapeamento de hotspots de acidentes de trabalho no Brasil utilizando técnicas de machine learning e análise espacial. **Dataset Reports**, v. 3, n. 1, p. 141-148, 2024. Disponível em: https://journals.royaldataset.com/dr/article/view/116. Acesso em: fev. 2025.

HORCHULHACK, Pedro *et al*. Detecçao de Overbooking em Aplicaçoes Baseadas em Docker Através de Aprendizagem de Máquina. In: **Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)**. SBC, 2022. p. 209-216. Disponível em: https://sol.sbc.org.br/index.php/sbrc_estendido/article/view/21438. Acesso em: fev. 2025.

IMPETO Informática. Servidor na Nuvem ou Local? Comparamos os Custos Para Você. **IMPETO Blog**. 2019. Disponível em: https://impeto.com.br/servidor-nuvem-ou-local/. Acesso em: fev. 2025.

JASKOWIAK, Pablo Andretta; COSTA, Ivan Gesteira; CAMPELLO, Ricardo José Gabrielli Barreto. The Area Under the ROC Curve as a Measure of Clustering Quality. **arXiv preprint**. 2020. Disponível em: https://arxiv.org/abs/2009.02400. Acesso em: fev. 2025.

JÚNIOR, Wládisson Mancinelli; SANTOS, Clayton Eduardo. Implementação de ERP em nuvem em pequenas e médias empresas: comparativos, segurança, benefícios e desafios. **Revista Científica e-Locução**, v. 1, n. 21, p. 24-24, 2022. Disponível em:

https://periodicos.faex.edu.br/index.php/e-Locucao/article/download/462/308. Acesso em: fev. 2025.

KREMERS, Bart J. J.; HO, Aaron; CITRIN, Jonathan; PLASSCHE, Karel L. Two step clustering for data reduction combining DBSCAN and k-means clustering. **arXiv preprint**. 2021. Disponível em: https://arxiv.org/abs/2111.12559. Acesso em: fev. 2025.

LORENZI, Uriel Mafra; GREIN, Willian; CORCINI, Luiz Fernando. Computação em nuvem: conceitos, aplicações e novas tecnologias. **Revista das Faculdades Santa Cruz**, v. 13, n. 1, 2022. Disponível em: https://periodicos.unisantacruz.edu.br/index.php/revusc/article/view/8. Acesso em: fev. 2025.

MALISZEWSKI, Anderson *et al*. Ambiente de Nuvem Computacional Privada para Teste e Desenvolvimento de Programas Paralelos. **Minicursos da XXI Escola Regional de Alto Desempenho da Região Sul**, 2021. Disponível em: https://repositorio.pucrs.br/dspace/bitstream/10923/24105/2/Ambiente_de_Nuvem_Computacional_Privada_para_Teste_e_Desenvolvimento_de_Programas_Paralelos.pdf. Acesso em: fev. 2025.

MEJIA, Wilson; CURASMA, Herminio. A Cloud Based Recommender System for Competitive Programming Platforms with Machine and Deep Learning. In: **Anais do VIII Congresso sobre Tecnologias na Educação**. SBC, 2023. p. 11-20. Disponível em: Acesso em: fev. 2025.

MELO, Rafaela; PESSOA, Marcela; FERNANDES, David. Clusterização de soluções de exercícios de programação: um mapeamento sistemático da literatura. **Simpósio Brasileiro de Informática na Educação (SBIE)**, p. 1715-1729, 2024. Disponível em: https://sol.sbc.org.br/index.php/sbie/article/view/31352. Acesso em: fev. 2025.

OLIVEIRA, Pamella Letícia Silva *et al*. Identificação de Pesquisas e Análise de Algoritmos de Clusterização para a Descoberta de Perfis de Engajamento. **Revista Brasileira de Informática na Educação**, v. 30, p. 01-19, 2022. Disponível em: https://journals-sol.sbc.org.br/index.php/rbie/article/view/2508. Acesso em: fev. 2025.

REIS, Thiago Nelson Faria *et al*. Uma Proposta de Classificação para Rotular a Eficiência Energética na Computação em Nuvem Verde. **Boletim de Conjuntura (BOCA)**, v. 17, n. 49, p. 761-793, 2024. Disponível em: https://revista.ioles.com.br/boca/index.php/revista/article/view/3255. Acesso em: fev. 2025.

SANTOS, Frances A. *et al*. Processamento de Linguagem Natural em Textos de Mídias Sociais: Fundamentos, Ferramentas e Aplicações. **Sociedade Brasileira de Computação**, 2022. Disponível em: https://sol.sbc.org.br/livros/index.php/sbc/catalog/download/106/473/746-1. Acesso em: fev. 2025.

SCHUSSLER, Brenda S. *et al*. Comparando o Desempenho entre Computaçao em Nuvem e Servidor Local na Execuçao do Método Fletcher. In: **Anais da XXIII Escola Regional de Alto Desempenho da Região Sul**. SBC, 2023. p. 33-36. Disponível em: https://sol.sbc.org.br/index.php/eradrs/article/view/24495. Acesso em: fev. 2025.

SILVA, Anildo Joaquim. Segurança de informação no ambiente da computação na nuvem. **Revista Primeira Evolução**, v. 1, n. 38, p. 13-25, 2023. Disponível em: http://primeiraevolucao.com.br/index.php/R1E/article/view/393. Acesso em: fev. 2025.

SILVA, Carla M.; PEREIRA, João V.; SAQUI, Rafael T. Aplicação de modelos híbridos de clusterização em sistemas de recomendação. **Revista Brasileira de Informática Educacional,** v. 29, n. 3, p. 223-237, 2023.

SILVA, Heberty Alves; PEREIRA, Larissa; SAQUI, Diego. Recomendação de livros baseada em clusterização e algoritmos de filtragem colaborativa. **15º jornada científica e tecnológica e 12 º simpósio de pós-graduação do ifsuldeminas**, v. 15, n. 3, 2023. Disponível em: https://josif.ifsuldeminas.edu.br/ojs/index.php/anais/article/view/962. Acesso em: fev. 2025.

SILVA, Marcos A. Computação em nuvem e segurança da informação: desafios contemporâneos. **Revista de Tecnologia e Sociedade**, v. 15, n. 1, p. 60-74, 2023.

SILVA, Maria Gabriely Lima *et al*. Mineração de Dados para Obtenção do Grau de Complexidade de Processos Judiciais. **Revista de Engenharia e Pesquisa Aplicada**, v. 6, n. 5, p. 56-64, 2021. Disponível em: http://revistas.poli.br/index.php/repa/article/view/1755. Acesso em: fev. 2025.

VYSALA, Anupriya; GOMES, Joseph. Evaluating and Validating Cluster Results. **arXiv preprint**. 2020. Disponível em: https://arxiv.org/abs/2007.08034. Acesso em: fev. 2025.