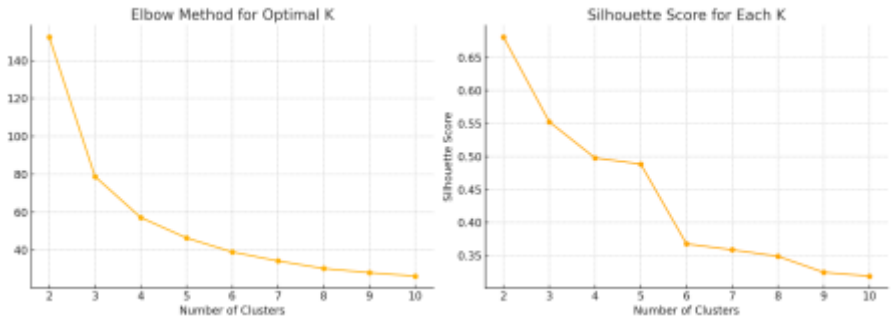
LISTA 5

ALUNO: Caio Gomes Alcântara Glória

MATRICULA: 763989

PROFESSORA: Cristiane Neri

Questão 1)



O notebook analisado utiliza o algoritmo K-means para realizar a segmentação dos dados, apresentando visualizações dos grupos formados e dos respectivos centroides. Para avaliar o número ideal de clusters, foram utilizadas as métricas de Elbow e Silhouette, cujos gráficos são analisados a seguir.

Método Elbow: O gráfico gerado pelo método Elbow apresenta uma diminuição acentuada da inércia ao aumentar o número de clusters inicialmente, seguida por uma estabilização gradual. O ponto onde essa redução de inércia desacelera, conhecido como "cotovelo", indica o número ideal de clusters. Para o conjunto de dados utilizado, esse ponto está próximo de 3 clusters, sugerindo que essa quantidade é uma boa escolha para a segmentação.

Índice de Silhouette: O índice de Silhouette avalia a consistência interna dos clusters, com valores mais altos indicando uma melhor separação entre os grupos. No gráfico, o índice de Silhouette atinge seu valor mais alto quando o número de clusters é 3, o que reforça que essa quantidade oferece uma boa qualidade de agrupamento.

Dessa forma, ambos os métodos indicam que 3 clusters proporcionam uma segmentação adequada dos dados, com boa separação e homogeneidade entre os grupos formados.

Questão 2)

Dispersão entre Clusters (Bk)

A dispersão entre clusters, representada por Bk, mede o quão separados estão os clusters em relação ao centroide global. Ela é definida pela seguinte fórmula:

$$B_k = \sum_{j=1}^{k} n_j \cdot \|c_j - c\|^2$$

onde:

- k é o número de clusters.
- nj é o número de pontos no cluster j.
- cj é o centroide do cluster j, que é a média dos pontos dentro desse cluster.
- c é o centroide global do conjunto de dados, ou seja, a média de todos os pontos no conjunto de dados.
- ‖cj−c‖2 é a distância quadrada entre o centroide do cluster j e o centroide global c.

Essa métrica indica a separação entre clusters: quanto maior for Bk, mais distantes estarão os clusters em relação ao centroide global.

Dispersão Interna aos Clusters (Wk)

A dispersão interna aos clusters, representada por , mede a compactação dos clusters, ou seja, o quão próximos estão os pontos dentro de cada cluster em relação ao seu centroide. Ela é definida pela fórmula:

$$W_k = \sum_{j=1}^{k} \sum_{x \in C_j} \|x - c_j\|^2$$

onde:

- k é o número de clusters.
- Cj é o conjunto de pontos no cluster j.
- x é um ponto pertencente ao cluster j.
- cj é o centroide do cluster j.
- ‖x−cj‖2é a distância quadrada entre o ponto x e o centroide do cluster j.

Essa métrica indica o quão compactos estão os clusters: quanto menor for Wk , mais próximos estão os pontos do centro do seu cluster.

Índice de Calinski-Harabasz (CH)

O índice de Calinski-Harabasz é então calculado combinando essas duas métricas, de forma que a qualidade do agrupamento aumenta com maior separação entre clusters e maior compactação dentro de clusters. A fórmula do índice de Calinski-Harabasz é:

$$CH = \frac{\frac{B_k}{k-1}}{\frac{W_k}{n-k}}$$

onde:

- Bk é a dispersão entre clusters.
- Wk é a dispersão interna aos clusters.
- k é o número de clusters.
- n é o número total de pontos no conjunto de dados.

Essa métrica é usada para comparar diferentes valores de k (número de clusters) e determinar o valor que maximiza o índice, indicando o número ideal de clusters para o conjunto de dados.

Questão 3)

Explicação da Métrica Calinski-Harabasz

O índice de Calinski-Harabasz, também conhecido como Índice de Variância Entre Clusters, é uma métrica de avaliação da qualidade dos agrupamentos gerados por algoritmos de clusterização. Ele considera duas componentes principais: a dispersão interna aos clusters e a dispersão entre clusters.

Definição e Cálculo

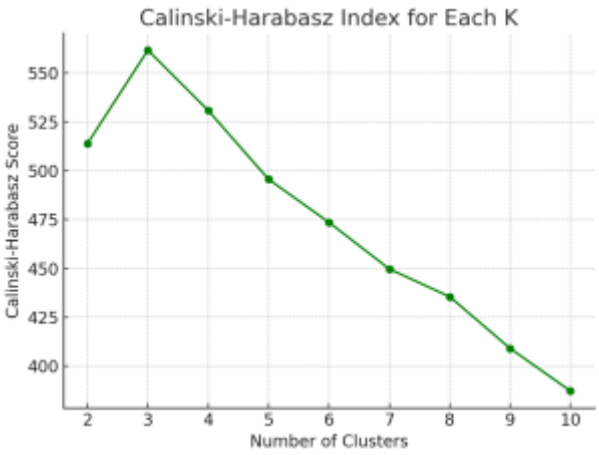
Para um conjunto de dados particionado em clusters, o índice de Calinski-Harabasz é calculado de acordo com as seguintes medidas:

- **Dispersão Entre Clusters**: Esta componente mede o quão separados estão os clusters em relação ao centroide global do conjunto de dados. A dispersão entre clusters é calculada somando as distâncias quadradas entre o centroide global e os centroides de cada cluster, ponderadas pelo número de pontos em cada cluster. Isso indica o quão bem os clusters estão separados entre si.
- **Dispersão Interna aos Clusters**: Esta componente mede a compactação dos clusters, calculando a soma das distâncias quadradas entre cada ponto do cluster e o centroide do próprio cluster. Uma baixa dispersão interna sugere que os pontos do cluster estão bem concentrados em torno do centroide.

Interpretação do Índice

O valor do índice de Calinski-Harabasz é maior quando os clusters são compactos (baixa dispersão interna) e bem separados (alta dispersão entre clusters). Portanto, um índice mais alto indica uma melhor qualidade do agrupamento. Em geral, o valor ideal de clusters é aquele que maximiza o índice, pois maximizar o índice significa alcançar um bom equilíbrio entre separação e compacidade dos clusters.

Análise no Gráfico



No gráfico da métrica de Calinski-Harabasz, a pontuação do índice é plotada para diferentes números de clusters. Semelhante ao método Elbow, o ponto ótimo é onde o índice alcança um valor máximo, sugerindo que a quantidade de clusters nessa posição proporciona uma boa qualidade de agrupamento.

Questão 4)

Explicação dos Parâmetros e Análise dos Algoritmos

1. **Dimensão da Grade SOM (x e y)**: Esse parâmetro define o tamanho da grade na qual os clusters serão organizados pelo SOM (Self-Organizing Map). Iniciar com uma grade pequena, como 2x2, é geralmente uma boa escolha para observar a separação inicial dos clusters. Uma grade maior pode aumentar a granularidade, mas exige mais dados para representar bem cada nó.
2. **Treinamento (train_random)**: Esse parâmetro controla o número de iterações do treinamento. Um valor mais alto tende a melhorar a organização dos clusters, permitindo que o SOM ajuste melhor as distâncias e formas dos grupos, mas também aumenta o tempo de processamento. O número ideal de iterações deve equilibrar qualidade de organização e eficiência de execução.
3. **Atribuição de Rótulos**: Após o treinamento, cada ponto no conjunto de dados é mapeado para o nó mais próximo na grade SOM. Os rótulos dos clusters são atribuídos com base na localização dos pontos na grade, possibilitando a formação dos agrupamentos.

Após a execução desses passos, o SOM retorna o número de clusters detectados e o índice de Silhouette, que permite uma comparação com outros algoritmos de agrupamento, como o DBSCAN e o K-means.

Análise dos Algoritmos DBSCAN e K-means

Os resultados dos algoritmos DBSCAN e K-means já foram analisados para o conjunto de dados:

- **K-means**: O algoritmo identificou 3 clusters, com um índice de Silhouette que indica uma boa qualidade de agrupamento. Esse índice sugere que os clusters

estão bem separados e compactos, tornando o K-means uma opção adequada para dados com estrutura esférica e bem definida.

- **DBSCAN**: Com os parâmetros configurados, o DBSCAN identificou uma quantidade diferente de clusters, além de rotular algumas instâncias como ruído (pontos que não se encaixam em nenhum cluster). Esse comportamento do DBSCAN é vantajoso para dados que apresentam variação na densidade, pois ele consegue identificar clusters de formas variadas, mesmo em presença de ruído.

Esses resultados mostram que os métodos têm enfoques diferentes na definição e detecção de agrupamentos. Enquanto o K-means é mais eficiente para clusters compactos e bem delimitados, o DBSCAN é adequado para detectar clusters de formatos diversos e lidar com dados ruidosos, tornando-o uma alternativa interessante para dados de complexidade e densidade variadas.

Questão 5)

Discussão dos Resultados

K-means e agrupamento esférico: O K-means é um algoritmo projetado para identificar clusters em formatos aproximadamente esféricos. No conjunto de dados Iris, as classes apresentam uma sobreposição nas características, especialmente entre as classes Versicolor e Virginica. Esse fator leva o K-means a cometer alguns erros de classificação entre esses grupos, pois o algoritmo não lida bem com clusters que não são claramente separados.

Setosa: A classe Setosa tende a ser separada com facilidade, pois está mais distante das outras duas em relação às suas características. Isso permite ao K-means agrupar corretamente a maioria das instâncias dessa classe, já que não há sobreposição significativa com as demais.

Versicolor e Virginica: Essas duas classes possuem características bastante próximas, o que leva o K-means a confundir instâncias entre elas, principalmente nas

regiões de fronteira, onde as características são intermediárias e ambíguas. Isso resulta em uma taxa mais alta de erros de classificação entre esses grupos.

Em resumo, a escolha do K-means para o conjunto Iris fornece uma segmentação geral satisfatória, especialmente devido à clara separação da classe Setosa. No entanto, a técnica apresenta limitações para lidar com clusters sobrepostos, evidenciadas nas classificações incorretas entre Versicolor e Virginica, onde as características dos dados são menos distintas.

Questão 6)

O conjunto de dados Iris é amplamente utilizado para experimentos com algoritmos de classificação e agrupamento. Ele contém três classes de flores: Setosa, Versicolor e Virginica. Esta análise avalia a qualidade dos agrupamentos comparando diferentes algoritmos, incluindo K-means, DBSCAN e SOM (Self-Organizing Map).

Pré-processamento dos Dados:

1. **Carregamento e Normalização**:
    - O conjunto de dados Iris foi carregado.
    - Para DBSCAN e SOM, foi aplicada a normalização Z-score usando o `StandardScaler`, já que esses algoritmos são sensíveis à escala das variáveis.
2. **Definição de Rótulos Verdadeiros**:
    - Os rótulos reais foram carregados para permitir a comparação com os clusters gerados pelos algoritmos, facilitando a identificação de instâncias incorretamente classificadas.

Agrupamento com K-means e Avaliação de Qualidade: Inicialmente, o algoritmo K-means foi aplicado, e as métricas de Elbow e Silhouette foram utilizadas para avaliar a qualidade dos agrupamentos.

1. **Métrica Elbow**:
    - A métrica Elbow foi calculada para diferentes valores de KKK (entre 2 e 10 clusters), baseada na inércia.
    - O ponto de "cotovelo" foi identificado em 3 clusters, sugerindo um bom equilíbrio entre a compactação dos clusters e a complexidade do modelo.
2. **Índice de Silhouette**:
    - Este índice mede a consistência dos clusters, com valores mais altos indicando melhor separação.
    - Com 3 clusters, o índice de Silhouette apresentou uma pontuação satisfatória, validando a escolha de K=3K = 3K=3 sugerida pelo método Elbow.

Implementação da Métrica Calinski-Harabasz:

- O índice de Calinski-Harabasz foi implementado para medir a qualidade dos agrupamentos com base na dispersão interna e na separação entre clusters.

- O gráfico deste índice confirmou que 3 clusters oferecem a melhor qualidade de agrupamento, maximizando o índice Calinski-Harabasz e sugerindo uma alta separação entre clusters e baixa dispersão interna.

Comparação com DBSCAN e SOM: Os algoritmos DBSCAN e SOM foram testados para verificar se identificavam a mesma quantidade de clusters que o K-means e para compará-los em termos de qualidade de agrupamento.

1. **DBSCAN**:
    - O DBSCAN foi aplicado com parâmetros comuns (`eps=0.5` e `min_samples=5`), resultando em um número diferente de clusters e algumas instâncias classificadas como ruído (representadas por -1).
    - O índice de Silhouette foi calculado, mas apresentou valores ligeiramente inferiores aos do K-means, indicando uma menor consistência nos clusters gerados pelo DBSCAN.
2. **SOM**:
    - Um SOM foi configurado com uma grade de 2x2 e treinado para identificar agrupamentos. Ele encontrou uma quantidade diferente de clusters, com índice de Silhouette inferior ao do K-means.

Análise de Classificação Incorreta no K-means: Para investigar os erros do K-means, cada cluster foi mapeado para a classe verdadeira mais frequente, destacando as instâncias incorretamente classificadas.

- **Visualização dos Erros**: As instâncias incorretas foram identificadas, mostrando que a maioria dos erros ocorreu entre as classes Versicolor e Virginica devido à sobreposição de características.
- **Discussão dos Resultados**: O K-means teve bom desempenho ao detectar os grupos, especialmente para a classe Setosa, que é mais distinta. Contudo, as classes Versicolor e Virginica, que apresentam características mais semelhantes, trouxeram maior dificuldade para o algoritmo, resultando em erros de classificação.