


## Lista 6 - Inteligência Artificial

Nome: **Victor Ferraz de Moraes**

Matrícula: **802371**

**Código utilizado para a Lista:**

 **Kmeans.ipynb**

**Questão 1.1)**

### **Qualidade dos agrupamentos**

#### **Silhueta**

O algoritmo calcula o Score para diferentes valores de k, variando de 2 até a raiz quadrada da metade do número de instâncias. Um score mais alto indica melhor qualidade do agrupamento, com valores mais próximos de 1 sendo ideais.

Ao analisar os scores impressos, podemos identificar o valor de k que resulta no maior Silhouette Score, indicando o número ideal de clusters segundo esse método.

#### **Elbow**

O algoritmo primeiramente calcula o Within-Cluster Sum of Squares (WCSS) para diferentes valores de k e após isso, o WCSS é definido como a soma dos quadrados das distâncias entre cada ponto e o centroide do seu cluster.

O método busca o ponto de inflexão na curva do WCSS, onde o ganho em termos de redução do WCSS diminui significativamente. Esse ponto de inflexão geralmente indica o número ideal de clusters. A biblioteca kneed é utilizada para identificar automaticamente o ponto de inflexão (elbow/cotovelo).

### **Caracterizando os agrupamentos**

Após determinar o número ideal de clusters (neste caso, 3), o código aplica o K-means com esse valor de k.

1. Os pontos são então atribuídos aos seus respectivos clusters.
2. O código gera um gráfico de dispersão mostrando os clusters e seus centroides.

Através da análise visual do gráfico e dos dados originais, podemos caracterizar cada cluster com base nas características dos pontos que ele contém. Por exemplo, podemos observar se um cluster contém pontos com valores específicos para determinadas features, como comprimento e largura das pétalas.

## Questão 1.2)

### Silhueta

A métrica avalia a qualidade de um agrupamento considerando a distância entre um ponto e os pontos do seu próprio cluster (coesão) e a distância entre esse ponto e os pontos dos outros clusters (separação).

Para um ponto de dado  $i$ , a Silhueta é calculada da seguinte forma:

1. Calcule a distância média entre o ponto  $i$  e todos os outros pontos do mesmo cluster ( $a_i$ ). Essa distância representa a coesão do cluster.
2. Para cada cluster diferente do cluster do ponto  $i$ , calcule a distância média entre o ponto  $i$  e todos os pontos desse cluster.
3. Encontre o mínimo entre essas distâncias médias calculadas no passo 2. Essa distância mínima é chamada de  $b_i$ . Ela representa a separação do ponto  $i$  em relação aos outros clusters.
4. Calcule a silhueta do ponto  $i$  usando a seguinte fórmula:  $(b - a) / \max(A, B)$
5. Para obter a silhueta do agrupamento inteiro, calcule a média das Silhouettes de todos os pontos de dados.

Os valores de silhueta variam de -1 a 1. Um valor próximo de 1 indica que o ponto está bem agrupado, próximo de 0 indica que o ponto está na fronteira entre dois clusters, e próximo de -1 indica que o ponto pode ter sido atribuído ao cluster errado.

### Elbow

O método Elbow utiliza a soma dos quadrados das distâncias dentro dos clusters (WCSS - Within-Cluster Sum of Squares) para avaliar a qualidade do agrupamento. O WCSS é calculado da seguinte forma:

1. Para cada cluster, calcule a soma dos quadrados das distâncias entre cada ponto do cluster e o centroide do cluster.
2. Some os valores obtidos no passo 1 para todos os clusters.

O objetivo do método Elbow é encontrar o número ideal de clusters ( $k$ ) analisando a curva do WCSS em função de  $k$ . A ideia é encontrar o ponto na curva onde o ganho em termos de redução do WCSS diminui significativamente, formando um "cotovelo". Esse ponto geralmente indica o número ideal de clusters, pois adicionar mais clusters além desse ponto não resulta em uma redução significativa do WCSS.

### Questão 1.3)

#### Índice Davies-Bouldin

O Índice Davies-Bouldin mede a similaridade média entre cada cluster e seu cluster mais similar. Ele busca minimizar essa similaridade, indicando melhor separação entre os clusters.

#### Cálculo

1. Para cada cluster  $i$ , calcule a dispersão ( $S_i$ ), que pode ser a distância média entre cada ponto do cluster e o centroide do cluster.
2. Para cada par de clusters  $i$  e  $j$ , calcule a distância entre os centroides dos clusters ( $M_{ij}$ ).
3. Para cada cluster  $i$ , encontre o cluster  $j$  mais similar (com maior valor de  $R_{ij}$ ), onde  $R_{ij}$  é definido como:

$$R_{i,j} = \frac{S_i + S_j}{M_{i,j}}$$

4. Calcule o Índice Davies-Bouldin (DB) como a média dos valores máximos de  $R_{ij}$  para cada cluster  $i$ :

$$DB \equiv \frac{1}{N} \sum_{i=1}^N D_i$$

Onde  $n$  é o número de clusters.

#### Interpretação

Valores menores de DB indicam melhor qualidade do agrupamento, com clusters mais separados e compactos. O valor ideal de DB é 0, o que significa que os clusters são perfeitamente separados e não há sobreposição entre eles.

### Questão 1.4)

Número de Clusters:

K-means: 3 clusters (definido pelo método Elbow e Silhouette)

DBSCAN: 1 clusters + ruído (definido pelos parâmetros  $\epsilon$  e  $\min\_samples$ )

SOM: 3 clusters (considerando cada neurônio como um cluster)

## **Análise dos Grupos**

### **K-means**

Encontrou 3 clusters, que correspondem bem às 3 classes do dataset Iris (setosa, versicolor e virginica). As instâncias incorretamente classificadas estão principalmente na região de sobreposição entre as classes versicolor e virginica. É um algoritmo simples e eficiente, mas pode ter dificuldades com clusters de formas irregulares e ruído.

### **DBSCAN**

Encontrou 1 cluster principal. É mais flexível que o K-means, pois pode encontrar clusters com formas irregulares e lidar com ruído.

### **SOM**

Encontrou 3 clusters. A visualização dos clusters no mapa SOM não é muito intuitiva. Pode ser útil para visualizar a estrutura dos dados e identificar padrões, mas a interpretação dos clusters é mais complexa.

### **Comparação**

Número de Clusters: O K-means e o SOM encontraram o número de clusters mais próximo do número de classes reais do dataset Iris. O DBSCAN encontrou menos clusters..

Qualidade dos Clusters: O K-means e o SOM tiveram o melhor desempenho na separação das classes do Iris, com poucas instâncias incorretamente classificadas. O DBSCAN teve dificuldades em separar as classes de forma clara.

Flexibilidade: O DBSCAN é mais flexível que o K-means, pois pode encontrar clusters com formas irregulares e lidar com ruído, porém isso não aconteceu. O SOM é ainda mais flexível, porém sua interpretação dos clusters é mais complexa..

### **Questão 1.5)**

#### **Região de Sobreposição**

As instâncias incorretas geralmente se encontram na região de sobreposição entre os clusters, onde as características das flores de diferentes classes são semelhantes. O K-means assume que os clusters têm forma esférica e densidade similar. Isso pode levar a erros de classificação em datasets com clusters de formas complexas ou densidades variadas, como o Iris. Com  $k=3$ , o K-means tende a errar na classificação de algumas flores da classe versicolor e virginica, que possuem características semelhantes.

## Questão 1.6)

### Pré-processamento

**Normalização:** Os dados foram normalizados utilizando o MinMaxScaler, transformando os valores para o intervalo [0, 1]. Isso garante que features com diferentes escalas não influenciem desproporcionalmente o processo de agrupamento.

### K-means

Determinação do número de clusters: O método Elbow e a análise da Silhouette foram utilizados para determinar o número ideal de clusters. O método Elbow indicou 3 clusters como o ponto de inflexão na curva do WCSS. A análise da Silhouette também corroborou com 3 clusters, apresentando o maior valor para essa métrica.

Agrupamento: O algoritmo K-means foi aplicado com 3 clusters, utilizando a inicialização 'k-means++' para otimizar a convergência.

**Resultados:** O K-means agrupou os dados em 3 clusters, que correspondem, em grande parte, às 3 classes do dataset (setosa, versicolour e virginica). No entanto, algumas instâncias foram agrupadas incorretamente, principalmente na região de sobreposição entre as classes versicolour e virginica. A visualização dos clusters e das instâncias incorretas permitiu identificar essa sobreposição e as limitações do K-means nesse caso.

### DBSCAN

Ajuste de parâmetros: Os parâmetros eps (raio de vizinhança) e min\_samples (número mínimo de pontos em uma vizinhança) foram ajustados para obter resultados satisfatórios.

Agrupamento: O algoritmo DBSCAN foi aplicado com os parâmetros ajustados.

**Resultados:** O DBSCAN encontrou 1 cluster principal e alguns pontos foram classificados como ruído. Isso indica que o DBSCAN identificou uma estrutura de densidade diferente do K-means, possivelmente agrupando as classes setosa, versicolour e virginica em um único cluster devido à sobreposição entre elas.

### SOM

Ajuste de parâmetros: As dimensões do mapa SOM, sigma (raio de vizinhança) e learning\_rate foram ajustados para obter resultados satisfatórios.

**Resultados:** O SOM gerou um mapa de clusters, onde cada neurônio representa um cluster. A visualização do mapa de distâncias permitiu identificar as regiões de maior e menor densidade.

de dados. A análise dos clusters formados pelo SOM indicou uma estrutura similar à encontrada pelo K-means, com 3 clusters principais.