APRENDIZADO DE MÁQUINA E INTELIGÊNCIA ARTIFICIAL EM FÍSICA

Primeira atividade

Gustavo Freire Pereira da Silva, nUSP: 10799790

Intituto de Física - Universidade de São Paulo

Abril de 2024

Sumário

1 Introdução

No contexto do desenvolvimento de um algoritmo de aprendizado de máquina, é comum trabalhar com um roteiro bem determinado, representado pelo fluxograma da Figura 1. Dada uma base de dados, a primeira etapa consiste do pré processamento, focado na normalização e escalonamento dos dados, assim como na redução de dimensionalidade em cenários com um grande número de dimensões. Para estudos didáticos, como o caso dessa atividade, não se torna necessário a separação dos dados, nesse caso, em conjunto de treinamento e validação. Ainda assim, foi testado três algoritmos de aprenzidado de máquina distíntos. Para definir a qualidade do modelo, é razoável utilizar métricas de performance para definir se o modelo está pronto, ou precisa de ajuste. A ultima etapa é a predição a partir do modelo, que foge do escopo da atividade.

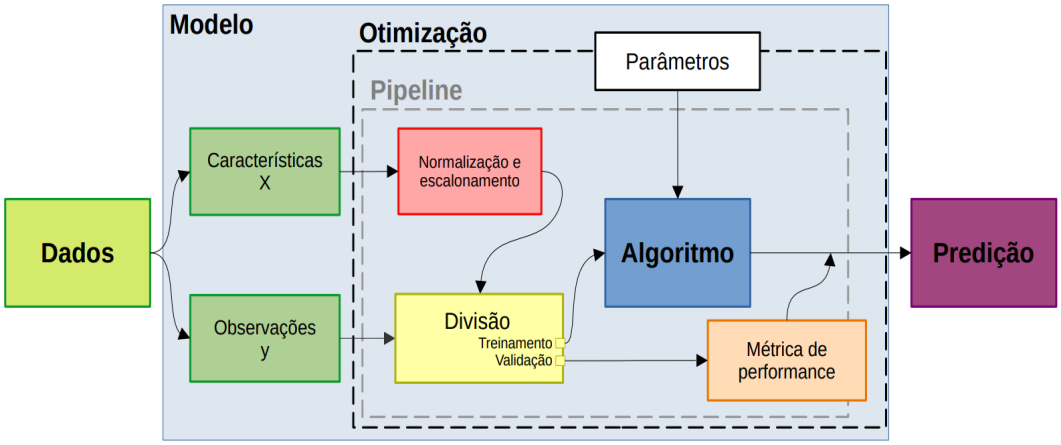

Figura 1: Fluxograma geral de um algoritmo de aprenzado de máquina. Retirado de: [1]

No presente trabalho, foi desenvolvido um estudo dos algoritmos de aprendizado de máquina: Agrupamento Hierárquico, KMeans e DBSCAN. Os métodos se fundamentam no agrupamento de dados (*clusters*) seguindo critérios específicos. O Agrupamento Hierárquico consiste da clusterização dos dados por meio da aproximação entre os dados, o KMeans define os agrupamentos entorno de um valor médio enquanto o DBSCAN se baseia no adensamento dos dados [2]. Foi realizado um pré processamento dos dados e análise de métricas de perfomance, quando possível, para determinar a melhor adequação do algoritmo aos dados.

2 Pré processamento

A base de dados utilizada contém informações sobre características de estrelas, sendo as colunas "*Temperature*", "L", "R", "A_M", "*Color*", "*Spectral Class*"e "*Type*", sem nenhuma legenda que permita interpretar esses dados qualitativamente antes do pré processamento, para analisar a relevância de cada característica na análise dos dados. Nesse caso, a única ferramenta disponível é a análise quantitativa.

A primeira etapa foi verificar a existência de células não preenchidas na tabela de dados, onde foi verificado que não havia informação incompleta no nosso conjunto. A segunda etapa foi verificar a presença de inconsistência quanto ao preenchimento manual de palavras

nas colunas da tabela, onde foi verificado a presença de palavras iguais na coluna "*Color*", mas com diferença de prenchimento com letras maiúsculas e minúsculas, ou traço "-" e espaçamento quando havia palavras compostas. Nesses casos foram transformadas todas as palavras em mínusculas e substituído o traço por espaço entre as palavras.

A segunda etapa consistiu de adequar os dados aos métodos numéricos, para isso notou-se que nas colunas preenchidas com palavras (*strings*) haviam categorias, isso é, os dados podiam ser classificados sob cores ou classe espectral específicas. Portanto, cada categoria foi transformada em uma nova coluna na tabela de dados, sendo atribuído "1.0" caso aquela linha possuía a característica ou "0.0" caso contrário, e removendo as colunas "*Color*" e "*Spectral Class*", evitando redundâncias.

A terceira etapa consistiu na busca de correlações claras entre as grandezas da base de dados. A Figura (2) apresenta a matriz de correlação entre as variáveis, e embora haja correlações entre algumas classes espectrais e cores, por não ser unanime - isso é, cada cor representar uma classe espectral -, não há sentido em analisar conjutamente.

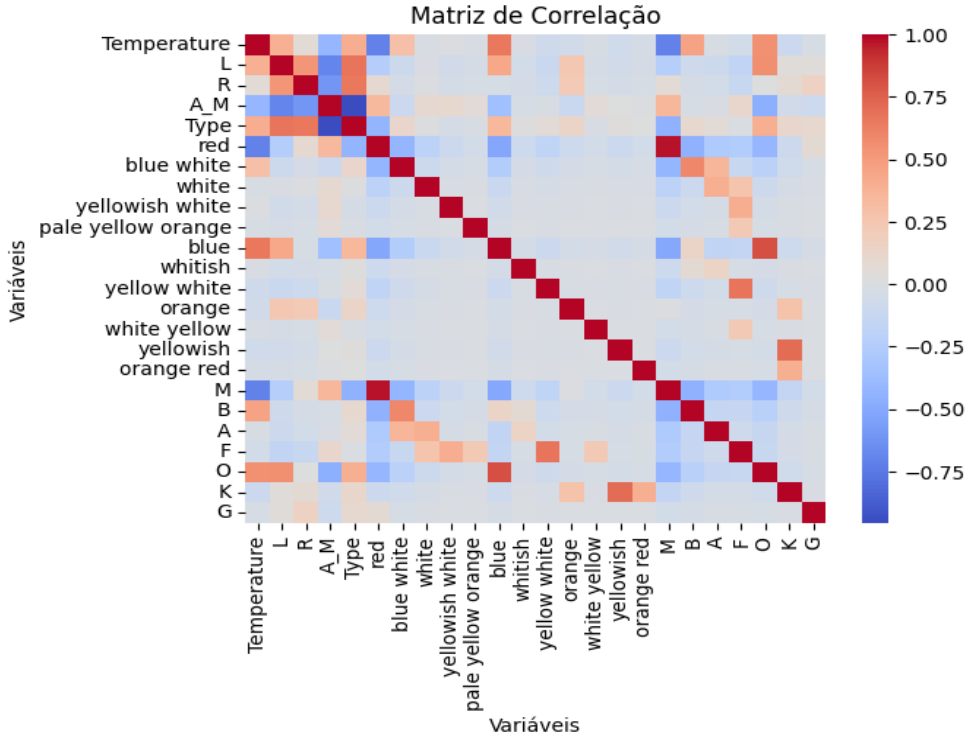

Figura 2: Mapa de correlação entre as variáveis da base de dados. M, B, A, F, O, K e G são as classes espectrais.

A quarta etapa foi verificar a presença de dados inconsistentes a partir da análise do histograma das grandezas, verificando contagens em escalas incompatíveis com restante dos dados, o que não foi observado. Os histogramas foram utilizados também para análise da distribuição dos dados para decidir o algoritmo de escalonamento que seria utilizado, seguindo o raciocínio: caso a distribuição fosse normal ou aproximadamente normal, usaríamos um reescalonamento baseado em normalização dos dados de média zero e desvio padrão 1; caso a distribuição não fosse normal e houvesse a presença de dados máximos e mínimos muito distantes e com presença razoável de outliers (ou uma distribuição muito distinta

da normal), usaríamos um escalonamento baseado no valor máximo e mínimo, baseado nos algoritmos apresentados em: [3], respectivamente StandardScaler e MinMaxScaler.

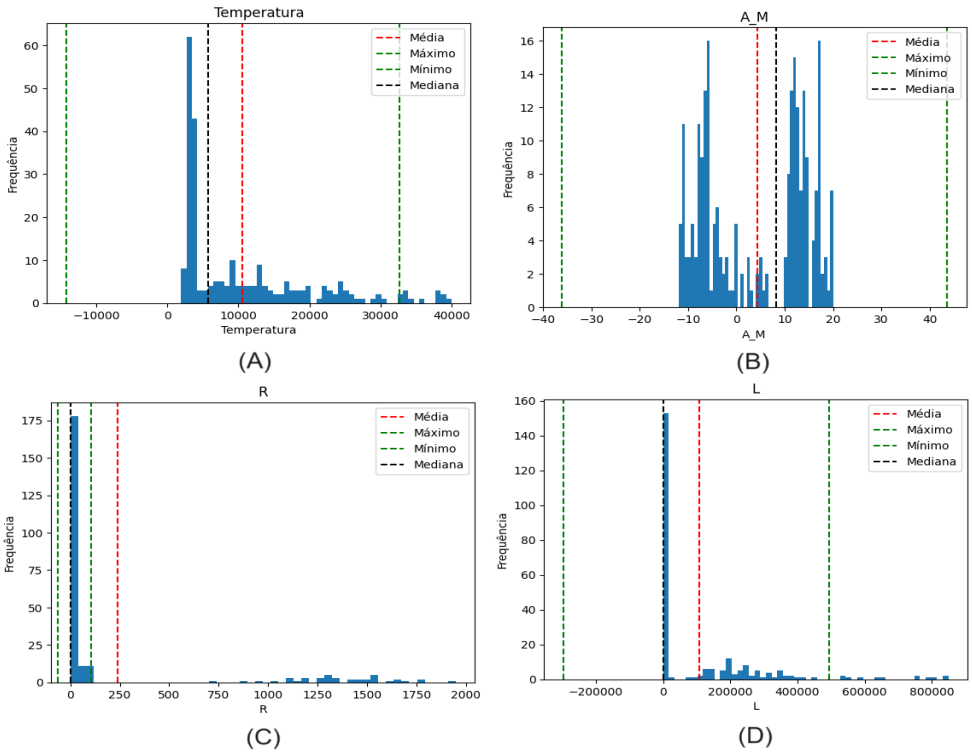

Figura 3: Histogramas da "*Temperature*", (B) "A_M", (C) "R", e (D) "L". A linha tracejada vermelha representa a média, a preta a mediana e as verdes os máximos e mínimos, calculados como o terceiro quartil somando o intervalo interquartil (terceiro quartil menos primeiro quartil) multiplicado por 1.5, e o primeiro quartil subtraindo o intervalo interquartil multiplicado por 1.5, respectivamente. Dados anteriores ou posteriores as linhas verdes podem ser interpretados como *outliers*.

Analisando as Figuras 3, podemos definir qualitativamente que, a distribuição em (A) é pouco afetada pela presença de *outliers*, visto sua proximidade com os dados fora dos extremos, além de que sua média e mediana estão razoavelmente próximas e com um pico de contagem pouco acentuado quando comparado com as figuras (C) e (D), portanto, nesse caso, será aplicado um algoritmo de escalonamento normal. Igualmente será aplicado esse algoritmo na distribuição (B), sendo a mais "normal"avaliando qualitativamente a distribuição; não há presença de outliers, a média e mediana estão próximas e a distribuição parece binormal e simétrica ao redor de 0. A distribuição (C) possui a média fora do limite máximo, possuindo contagens em escala uma ordem de grandeza maior do que no pico (acentuado) de contagem, sendo uma distribuição distinta da normal, tornando necessário o escalonamento a partir da valor máximo e mínimo do conjunto. A distribuição (D) também apresenta um pico bastante acentuado, sendo uma distribuição bastante distinta de uma normal, se aproximando de uma delta de Dirac [4]; nesse caso também foi aplicado uma normalização a partir do valor máximo e mínimo.

A grandeza *Type* apresenta uma distribuição discreta entre 0 e 5, Figura 4, sem *outliers*, mas com uma distribuição diferente da normal. Note que essa variável já é uma classificação do conjunto de dados, logo não parece haver sentido em usá-la numa nova clusterização. Nesse sentido, se há 6 tipos de estrelas no nosso conjunto, é esperado que nos algoritmos de

clusterização, obtenhamos também 6 agrupamentos.

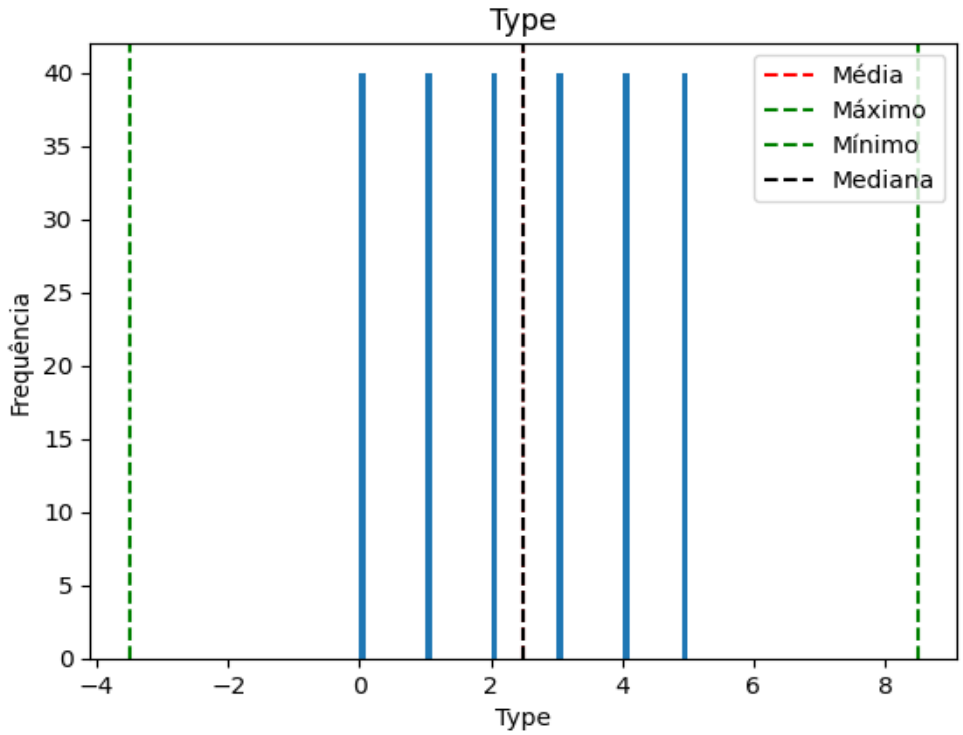

Figura 4: Histogramas do "*Type*". A linha tracejada vermelha representa a média, a preta a mediana e as verdes os máximos e mínimos, calculados como o terceiro quartil somando o intervalo interquartil (terceiro quartil menos primeiro quartil) multiplicado por 1.5, e o primeiro quartil subtraindo o intervalo interquartil multiplicado por 1.5, respectivamente. Dados anteriores ou posteriores as linhas verdes podem ser interpretados como *outliers*.

As grandezas restantes são transformações das colunas *Spectral Class* e *Color*, já escalonadas para 0 ou 1 seguindo critério apresentado previamente.

3 Redução de variáveis

Como apresentado na seção anterior, estamos trabalhando com uma base de dados com 7 colunas. Após o pré processamento, passamos a ter 23 colunas, o que representa 23 dimensões nos nossos dados. Para tratar o conjunto de forma adequada, é essencial aplicarmos uma redução de dimensionalidade, aplicando o algoritmo PCA como ilustrado em [5]. Analisando a Figura 5, podemos notar que para haver 90% da explicabiliadade dos nossos dados, é essencial usar pelo menos 5 componentes principais para representar nossas 23 dimensões.

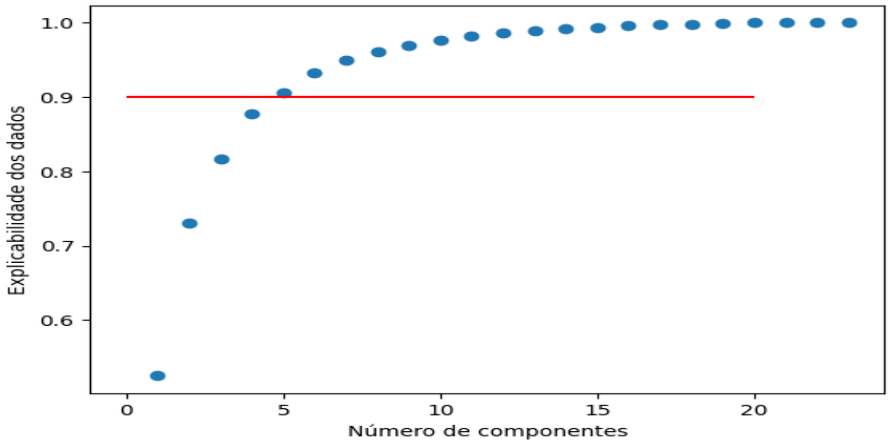

Figura 5: Explicabilidade dos dados em função do número de componentes principais. A linha vermelha representa o limiar para explicabilidade de pelo menos 90% da variância dos dados originais.

Para verificar quais variáveis da tabela são mais importantes para o nosso agrupamento dos dados, foi aplicado o seguinte procedimento: primeiro foi calculado a razão da variância que cada uma das três primeiras componentes principais carrega dos dados originais, e essa razão foi multiplicada pelo peso de cada variável na determinação da componente principal, tornando o *heatmap* abaixo representativo da importância de cada variável no agrupamento do dados. Um valor alto representa uma maior importância enquanto um valor menor, uma menor importância.

Analisando a Figura 6, vemos que as variáveis *Temperature* e R, L, a classe espectral "O"e a cor de estrela "*blue*" apresentaram maior contribuição para a variação total explicada pelas 3 primeiras componentes do algoritmo, sendo as mais importantes para o agrupamento dos dados. No entanto, a cor *blue* e *blue white* destoam muito das outras cores, assim como as classes "B", "M"e "O"com as outras classes. Se elas possuem um peso maior nas variáveis que carregam a variância dos dados, podemos interpretar que esses dados variam mais em comparação aos outros, o que pode representar uma inconsistência na forma de aquisição ou atribuição dessas características. Sem saber a forma de obtenção dos dados fica difícil analisar e categorizar como possível erro humano, mas visto que havia erro de digitação, é uma hipotése que não pode ser descartada.

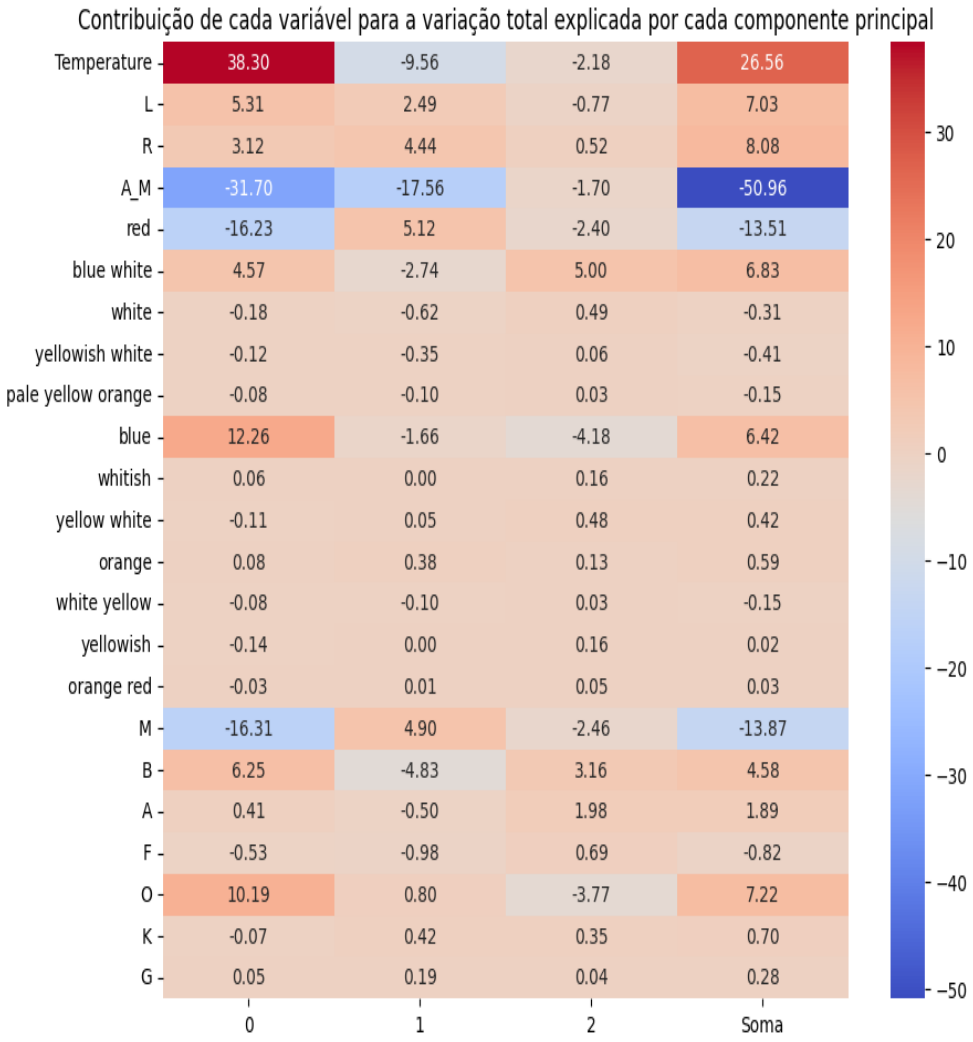

Figura 6: Explicabilidade dos dados em função do número de componentes principais. A linha vermelha representa o limiar para explicabilidade de pelo menos 90% da variância dos dados originais.

Para análise dos algoritmos, usaremos as duas componentes principais para avaliar a clusterização, visto que carregam a porção relativa mais representativa da variância do nosso conjunto de dados. Embora seja uma avaliação subjetiva, é razoável admitir que o ganho da representação dos dados em 3 dimensões não resultaria em ganho suficiente para justificar sua complexidade.

4 Agrupamento Hierárquico

O método de Agrupamento Hierárquico necessita de uma pré-definição do número de *clusters* para o funcionamento do algoritmo. Para definir o melhor número, foi definido uma métrica de performance apresentada em [2], definida pela razão entre o fator de silhueta e a quantidade de pontos mal atribuídos, Figura 7. Dada a natureza dos dados, para o Agrupamento Hierárquico, para 6 *clusters*, não houve pontos mal atribuídos, determinando o número ideal de agrupamento dos dados. O resultado dessa classificação pode ser observado

na Figura 8.

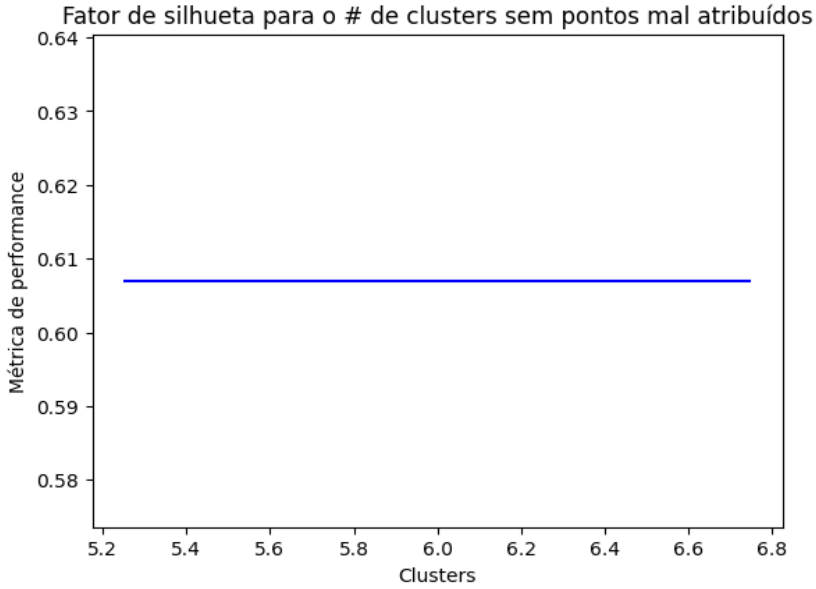

Figura 7: Fator de silhueta em função do número de *clusters* quando não há pontos mal atribuídos.

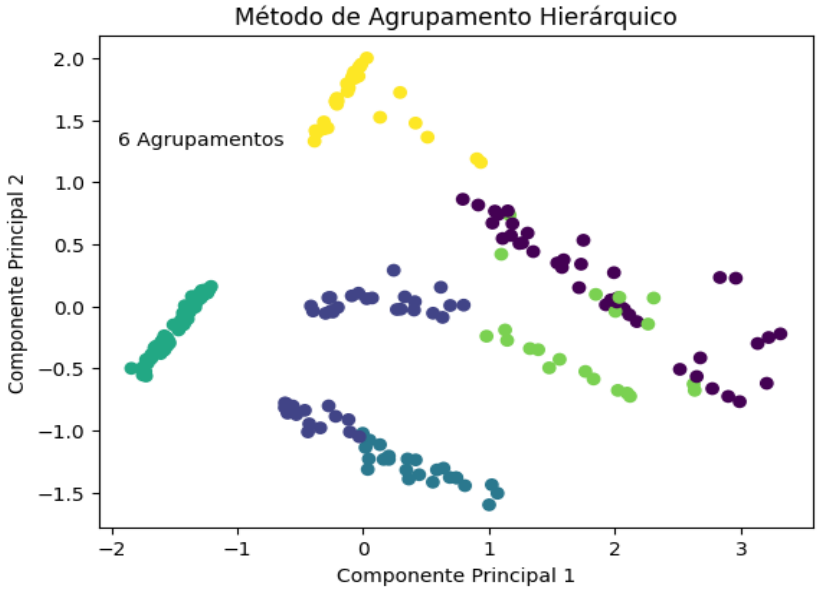

Figura 8: Visualização dos agrupamentos dos dados em função das duas principais componentes.

5 KMeans

Assim como método de Agrupamento Hierárquico necessita de uma pré-definição do número de *clusters*, o KMeans também necessita. Para definir o melhor número, aplicamos a mesma métrica de performance apresentada em [2], definida pela razão entre o fator de silhueta e a quantidade de pontos mal atribuídos, Figura 9. Novamente, dada a natureza dos

dados, para o KMeans, para 6 *clusters*, não houveram pontos mal atribuídos, determinando o número ideal de agrupamento dos dados pelo melhor fator de silhueta. O resultado dessa classificação pode ser observado na Figura 10.

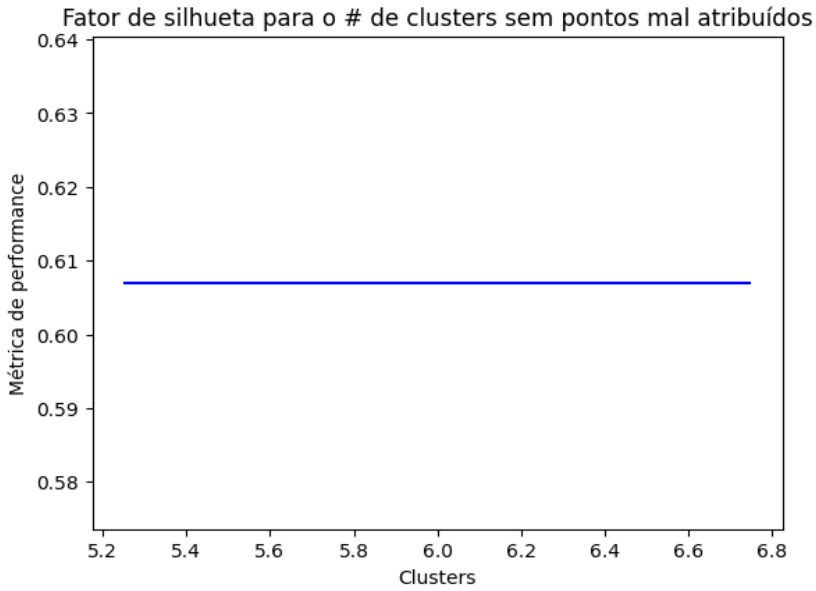

Figura 9: Fator de silhueta em função do número de *clusters* quando não há pontos mal atribuídos.

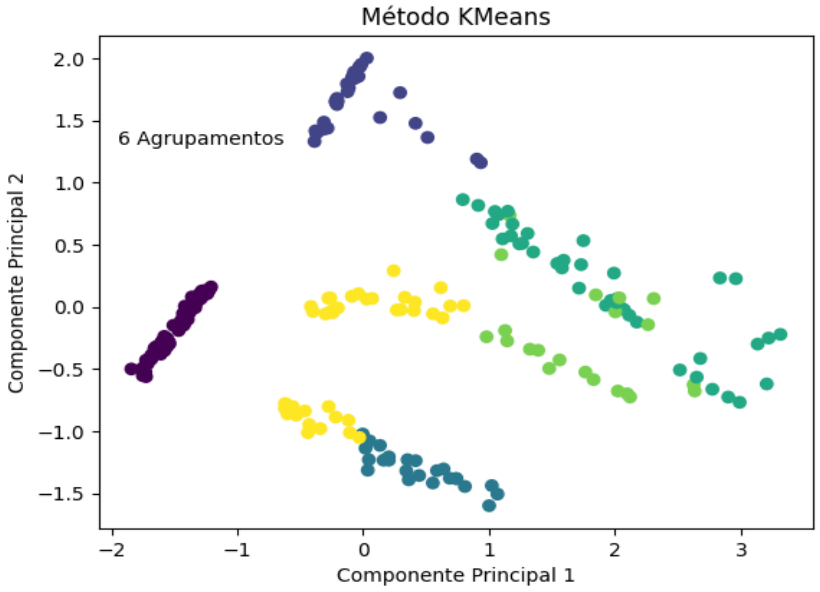

Figura 10: Visualização dos agrupamentos dos dados em função das duas principais componentes.

6 DBSCAN

Nesse método, não há necessidade de definir um número de agrupamentos, mas sim um raio dentro dos quais os dados serão considerados pertencentes ao agrupamento e o

número mínimo necessário de vizinhos para definir um core ou não, como definidos em [2]. Para evitar uma avaliação desses parâmetros meramente qualitativa, foi calculado a distância entre os 2 vizinhos mais próximos para o conjunto de dados, Figura 11. O raio foi determinado no final do "cotovelo"da curva que representa essa grandeza, entendendo que a partir desse ponto, as distância se tornam grandes de mais comparada ao restante do conjunto [6]. O número mínimo foi definido como igual a dimensão dos dados que estamos trabalhando, isso é, como estamos tratando em 5 dimensões, usamos o número mínimo como 5.

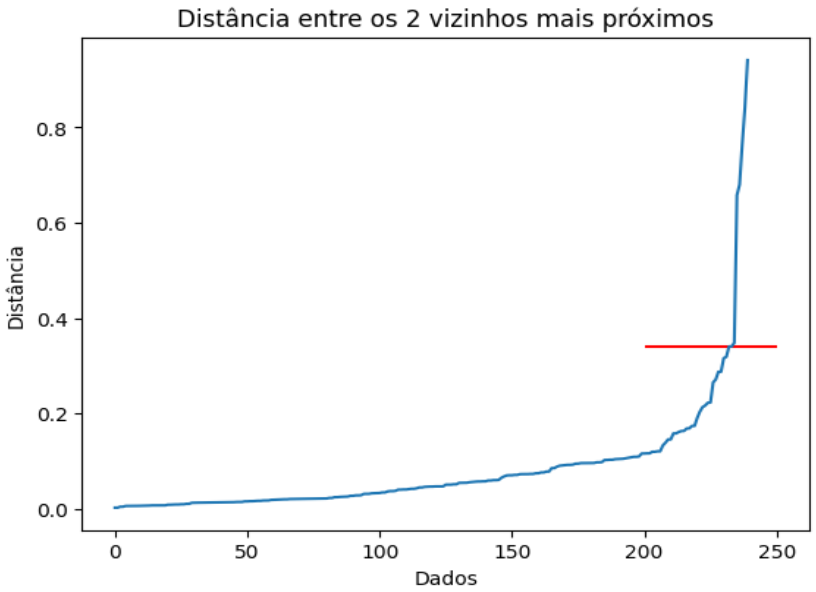

Figura 11: Distância entre os 2 vizinhos mais próximos para cada ponto do conjunto de dados.

O resultado do algoritmo está apresentado na Figura 12. Obtivemos 10 agrupamentos com a presença de 44 *outliers*, representando, aproximadamente, 18% dos dados. Considerando que não há qualquer informação sobre a maneira da obtenção dos dados, e de que os *outliers*, ao analisar a Figura 12, englobam os pontos mais isolados do conjunto de dados, e, somado a isso, ao retornar-mos a análise da Figura 6, uma possibilidade para justificar a presença de pesos em variáveis inexesperadas poderiam ser justificados devido à uma má atribuição dos dados naquelas colunas, concluímos que essa estatística de outliers é razoável ao caracterizar nosso conjunto.

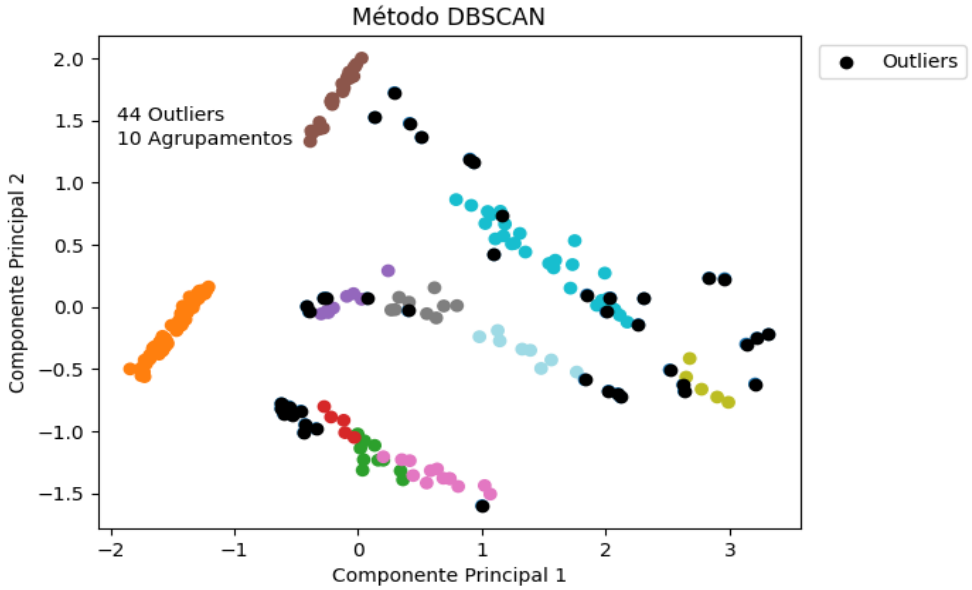

Figura 12: Visualização dos agrupamentos dos dados em função das duas principais componentes.

7 Performance Geral

Ao analisar a performance do algoritmo, um bom parâmetro é o tempo de execução de cada método, Tabela 1. Notamos que o algoritmo mais rápido é o KMeans, enquanto o mais lento é o Agrupamento Hierárquico. No entanto, ao analisarmos as Figuras 8,10 e 12 vemos que no primeiro e segundo método, temos o mesmo número de agrupamentos que foram tipificados no arquivo original, demonstrando que a métrica de performance utilizada garantiu uma compatibilidade entre os métodos e o pré-determinado. A partir de uma análise subjetiva, mesmo tratando o corte bidimensional do nossos dados 5-dimensionais, podemos observar que os métodos KMeans e Agrupamento Hierárquico geraram os mesmos agrupamentos. Como o KMeans é mais rápido, evidentemente se torna a melhor escolha para essa base de dados.

O DBSCAN parece ter tido o resultado menos razoável quanto a categorização dos dados quando comparada a tipificação original, embora represente o método do meio em questão do tempo de execução. No entanto, o método parece ser interessante na identificação de *outliers*, sendo interessante se análisado em conjunto com a Figura 6, pois, como comentando anteriormente, pode ser utíl para identificar pontos mal atribuidos do conjunto de dados originais.

Tabela 1: Tempo de execução de cada algoritmo

Algoritmo	Tempo de Execução (segundos)
Agrupamento Hierárquico	0.432
KMeans	0.306
DBSCAN	0.351

Sem compreender a natureza original dos dados, é inviável concluir categoricamente qual

o melhor método para aplicar ao nosso conjunto de dados. Entretanto, analisando os dados puramente em sua distribuição, o melhor método demonstrou ser o KMeans, dado seu tempo de contribuição e atribuição de *clusters*, e o pior, o Agrupamento Hierárquico, visto que é superado pelo KMeans mas carrega menos informação que o DBSCAN.

Referências

[1] A. SUAIDE, L. RIZZO, and T. FIORINI, "Aprendizado de máquina e inteligência artificial em física: Uma visão geral de um modelo de aprendizado de máquina," Slide de aula, 2024, código da disciplina: PGF5393 4305512. Acesso em: 22/04/2024.

[2] ——, "Aprendizado de máquina e inteligência artificial em física: Algoritmos de clusterização," Slide de aula, 2024, código da disciplina: PGF5393 4305512. Acesso em: 22/04/2024.

[3] ——, "Aprendizado de máquina e inteligência artificial em física: Pré e pós processamento de dados," Slide de aula, 2024, código da disciplina: PGF5393 4305512. Acesso em: 22/04/2024.

[4] Wikipedia, "Dirac delta function," 2024, acesso em: 22/04/2024. [Online]. Available: https://en.wikipedia.org/wiki/Dirac_delta_function

[5] A. SUAIDE, L. RIZZO, and T. FIORINI, "Aprendizado de máquina e inteligência artificial em física: Fatoração de dados e decomposição," Slide de aula, 2024, código da disciplina: PGF5393 4305512. Acesso em: 22/04/2024.

[6] E. DNC, "Clusterização por densidade com dbscan: Guia completo," 2024, acesso em: 22/04/2024. [Online]. Available: https://www.escoladnc.com.br/blog/clusterizacao-por-densidade-com-dbscan-guia-completo/