CENTRO DE PESQUISA E DESENVOLVIMENTO TECNOLÓGICO EM INFORMÁTICA
CEPEDI

Residência em Software – ResTIC36

Trilha 3 – Ciência de Dados

Relatório Técnico: Implementação e Análise do Algoritmo de K-means para Reconhecimento de Atividades Humanas

Vinícius de Oliveira Souza

Sanley Pires Ferreira

Data de Entrega: 03 de Dezembro de 2024

Resumo

Este projeto teve como objetivo implementar e analisar o algoritmo de K-means para a tarefa de Reconhecimento de Atividades Humanas (HAR) utilizando o dataset "Human Activity Recognition Using Smartphones".

O projeto envolveu as seguintes etapas:

1. **Análise Exploratória dos Dados:** Foi realizada uma análise exploratória para entender a estrutura dos dados, incluindo a verificação das dimensões do dataset, tipos de dados, matriz de correlação e redução de dimensionalidade com PCA.
2. **Implementação do K-means:** O algoritmo de K-means foi implementado para agrupar as atividades humanas em clusters.
3. **Escolha do Número de Clusters:** O número ideal de clusters foi determinado utilizando o método do cotovelo e o Silhouette Score.
4. **Avaliação do Modelo:** A qualidade dos clusters foi avaliada usando métricas como Silhouette Score e Inércia.
5. **Otimização e Ajustes:** O modelo foi otimizado com a normalização dos dados e a seleção de features.
6. **Relacionando Clusters e AVDs:** Os clusters foram relacionados às atividades para entender a relação entre eles.

Os resultados indicaram que o K-means foi capaz de identificar clusters significativos nas atividades humanas. As métricas de avaliação mostraram que o modelo teve um bom desempenho, e os gráficos demonstraram a coesão e separação dos clusters.

1. Introdução

O Reconhecimento de Atividades Humanas (HAR) é uma área de pesquisa importante com diversas aplicações, como monitoramento de saúde, detecção de quedas e interação humano-computador. O objetivo do HAR é identificar as atividades que uma pessoa está realizando com base em dados coletados por sensores.

O algoritmo de K-means é uma técnica de aprendizado de máquina não supervisionada que pode ser utilizada para agrupar dados em clusters. Ele é uma escolha adequada para o HAR, pois pode identificar padrões e agrupar atividades semelhantes sem a necessidade de rótulos pré-definidos.

Neste projeto, exploramos o uso do K-means para HAR utilizando o dataset "Human Activity Recognition Using Smartphones". O dataset contém dados coletados por sensores de acelerômetro e giroscópio de smartphones de 30 voluntários, que foram divididos em dados de treino (70%) e teste (30%).

2. Metodologia

1. Análise Exploratória dos Dados:

- **Verificação das dimensões do dataset, tipos de dados e estatísticas descritivas**

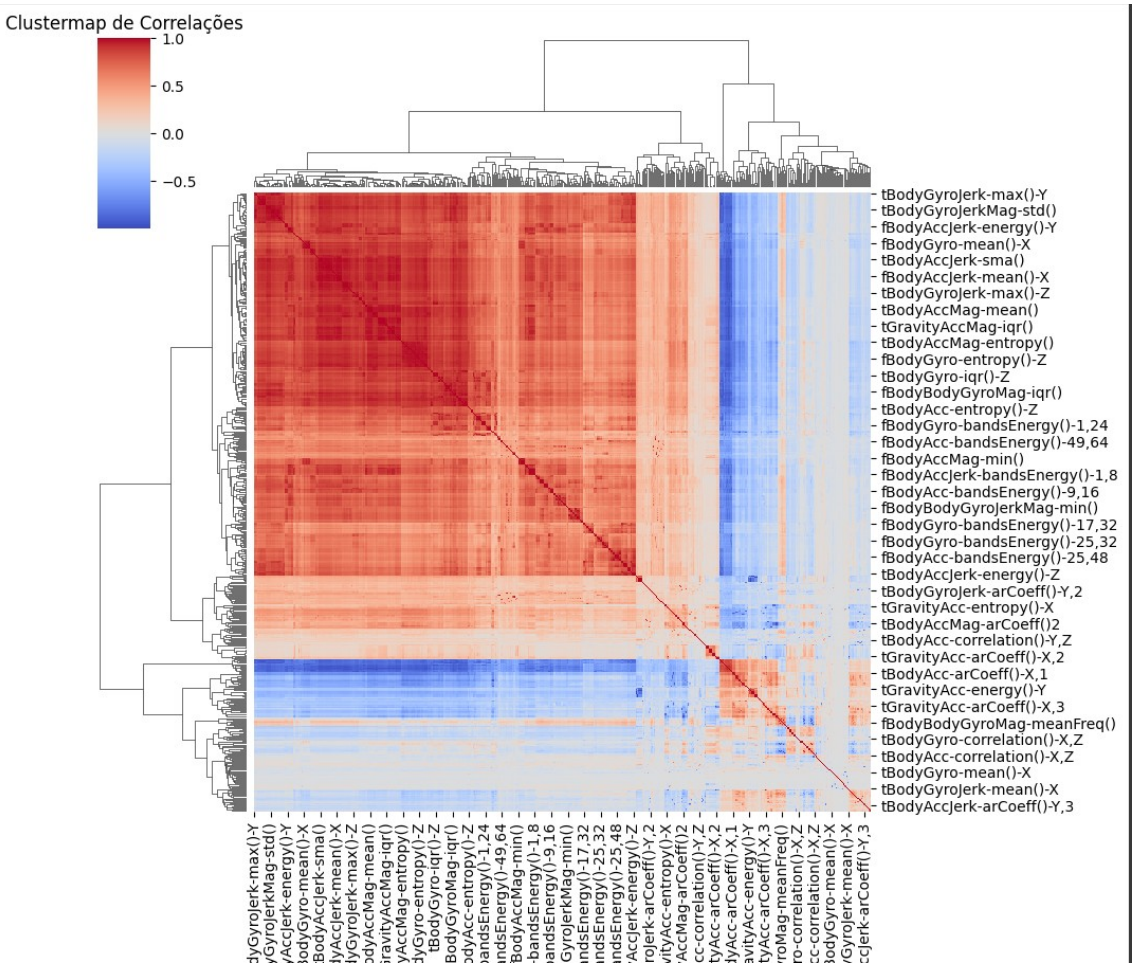
  Foram selecionadas 6 AVDs: em pé, sentado, deitado, caminhando, descendo escadas e subindo escadas. Um total de 561 características (features) foram extraídas para descrever cada janela de atividade.

```
Dimensões do dataset: (7352, 561)

Tipos de dados:
 tBodyAcc-mean()-X                 float64
tBodyAcc-mean()-Y                  float64
tBodyAcc-mean()-Z                  float64
tBodyAcc-std()-X                   float64
tBodyAcc-std()-Y                   float64
                                    ...
angle(tBodyGyroMean,gravityMean)   float64
angle(tBodyGyroJerkMean,gravityMean)  float64
angle(X,gravityMean)               float64
angle(Y,gravityMean)               float64
angle(Z,gravityMean)               float64
Length: 561, dtype: object
```

- **Cálculo da matriz de correlação para identificar features altamente correlacionadas**

  A matriz de correlação é uma ferramenta fundamental na análise exploratória de dados, pois revela as relações entre as diferentes features de um dataset. Ela é uma tabela que mostra os coeficientes de correlação entre todos os pares de variáveis. Com ela, é possível identificar redundâncias, entender as relações entre variáveis e, ainda, ajuda a melhorar o desempenho do modelo, uma vez que pode ser feita a remoção das feaures redundantes ou fazer a seleção das mais importantes. Para contornar o fato de ter várias variáveis, foi utilizado o clustermap, que combina um heatmap com um dendrograma, que permite uma visualização mais organizada e simplificada da matriz de correlação, permitindo uma melhor compreensão das relações entre as variáveis, mesmo em datasets com muitas features.

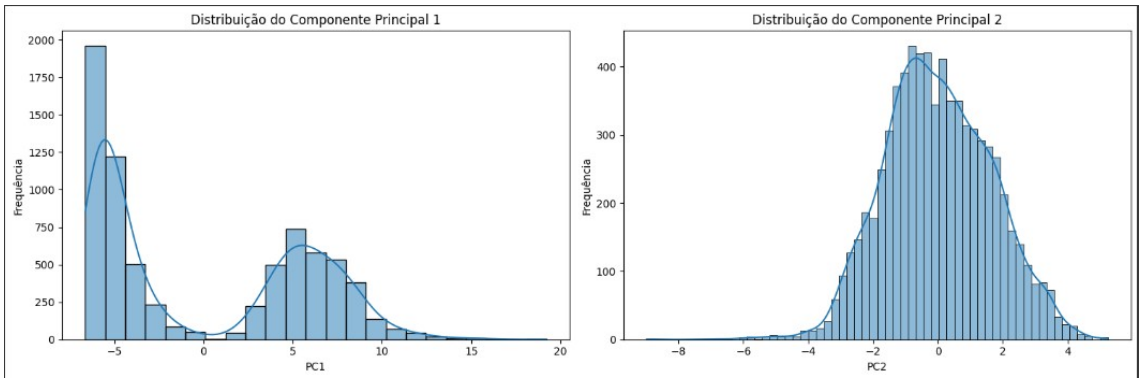


A visualização ainda fica prejudicada, devido ao tanto de variáveis, porém pelas cores é possivel notar algumas correlações altas (vermelho e azul escuros).

- **Redução de dimensionalidade com PCA para facilitar a visualização e o agrupamento**

  A redução de dimensionalidade com PCA visa simplificar a visualização e o agrupamento dos dados, diminuindo o número de variáveis (features) enquanto preserva o máximo de informação possível. Isso facilita a interpretação dos resultados e o desempenho do algoritmo K-means. Dessa forma, o PCA foi instanciado com *n_components=2*, definindo que queremos reduzir os dados para apenas duas dimensões (componentes principais).

- Plotagem de gráficos como histogramas e gráficos de dispersão para entender a distribuição dos dados.



Com a distribuição dos componentes principais, percebemos que a componente PC1 apresentou uma forma bimodal, o que pode indicar a presença de dois grupos distintos nos dados. Ao analisar os loadings desse componente, podemos identificar as variáveis originais que mais contribuem para essa separação em grupos.

```
fBodyAccJerk-entropy()-X     0.130761
fBodyAccJerk-entropy()-Y     0.128205
fBodyAcc-entropy()-X         0.125511
tBodyAccJerkMag-entropy()    0.125282
fBodyAccMag-entropy()        0.117119
                                ...
tBodyGyro-min()-Z           -0.049955
tBodyAcc-min()-Y            -0.054407
tBodyAcc-min()-X            -0.057764
tBodyAccJerk-min()-Y        -0.059203
tBodyAccJerk-min()-X        -0.073737
Name: PC1, Length: 561, dtype: float64
```

Notamos que elas estão relacionadas à entropia dos sinais de aceleração do corpo e da aceleração do corpo durante um jerk (movimento brusco). Por outro lado, as variáveis com maiores loadings negativos estão relacionadas aos valores mínimos da aceleração do corpo e da aceleração do corpo durante um jerk. Podemos inferir que PC1 representa a complexidade ou irregularidade do movimento do corpo.

2. Implementação do K-means e Escolha do Número de Clusters:

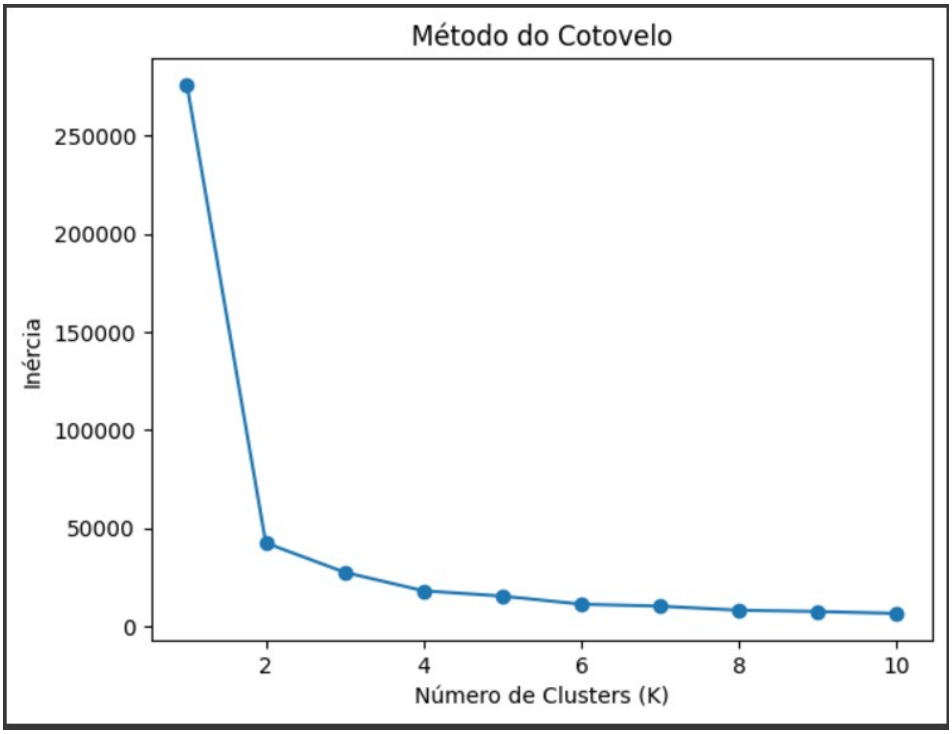
Esta seção aborda a implementação do algoritmo K-means usando a biblioteca scikit-learn, o ajuste do modelo aos dados de treinamento e a escolha do número ideal de clusters (K) usando métodos como o Método do Cotovelo e o Silhouette Score.

- **Implementação do Algoritmo:**

- O código utiliza a classe KMeans da biblioteca sklearn.cluster para implementar o K-means.
- O modelo é ajustado aos dados de treinamento (X_train_pca no seu caso, que são os dados após a redução de dimensionalidade com PCA) usando o método fit. Isso permite que o algoritmo aprenda os clusters com base nos dados fornecidos.
- Para prever os clusters para novos dados (dados de teste - X_test_pca), o método predict é usado.

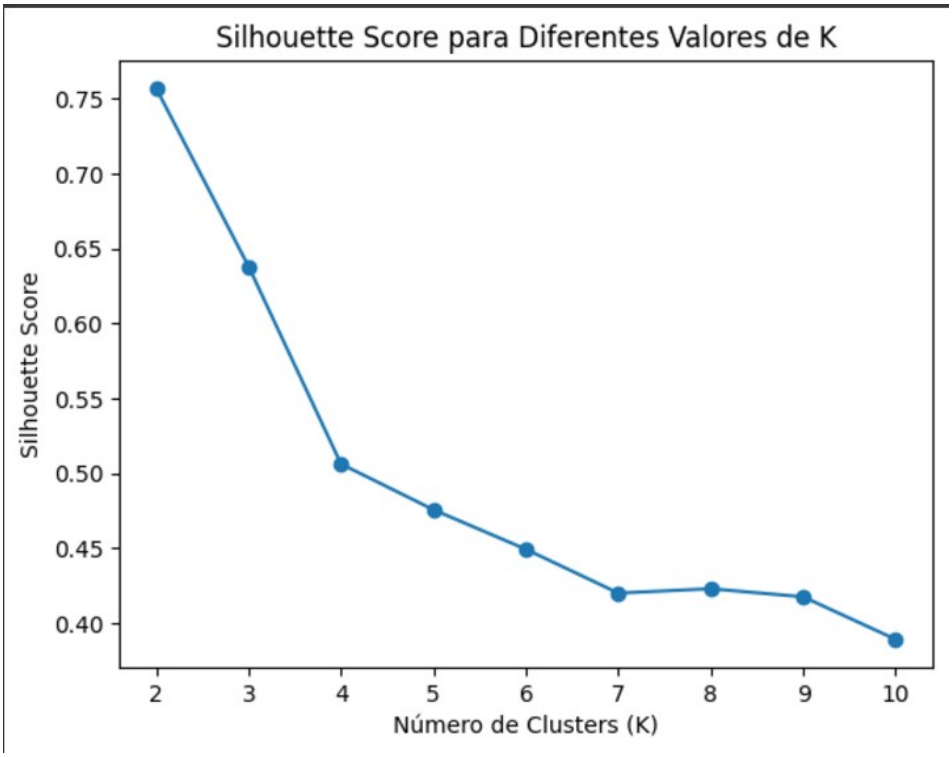
2. Escolha do Número de Clusters (K):

- **Método do Cotovelo:**
    - O código calcula a inércia (soma das distâncias quadradas das amostras ao centro do cluster mais próximo) para diferentes valores de K, variando de 1 a 10.
    - Os valores de inércia são plotados em um gráfico em função de K.
    - O ponto no gráfico onde a inércia começa a diminuir a uma taxa menor, formando um "cotovelo", é considerado o valor ideal de K. Esse ponto indica um bom equilíbrio entre a compactação dos clusters e o número de clusters.



- **Silhouette Score:**
    - O código calcula o Silhouette Score para diferentes valores de K, variando de 2 a 11 (o Silhouette Score não é definido para K=1).

- O Silhouette Score mede a qualidade dos clusters, avaliando a coesão (distância média entre cada amostra e as amostras do mesmo cluster) e a separação (distância média entre cada amostra e as amostras do cluster mais próximo).
- Valores mais próximos de 1 indicam clusters bem definidos e separados.
- O valor de K que resulta no maior Silhouette Score é considerado o ideal.

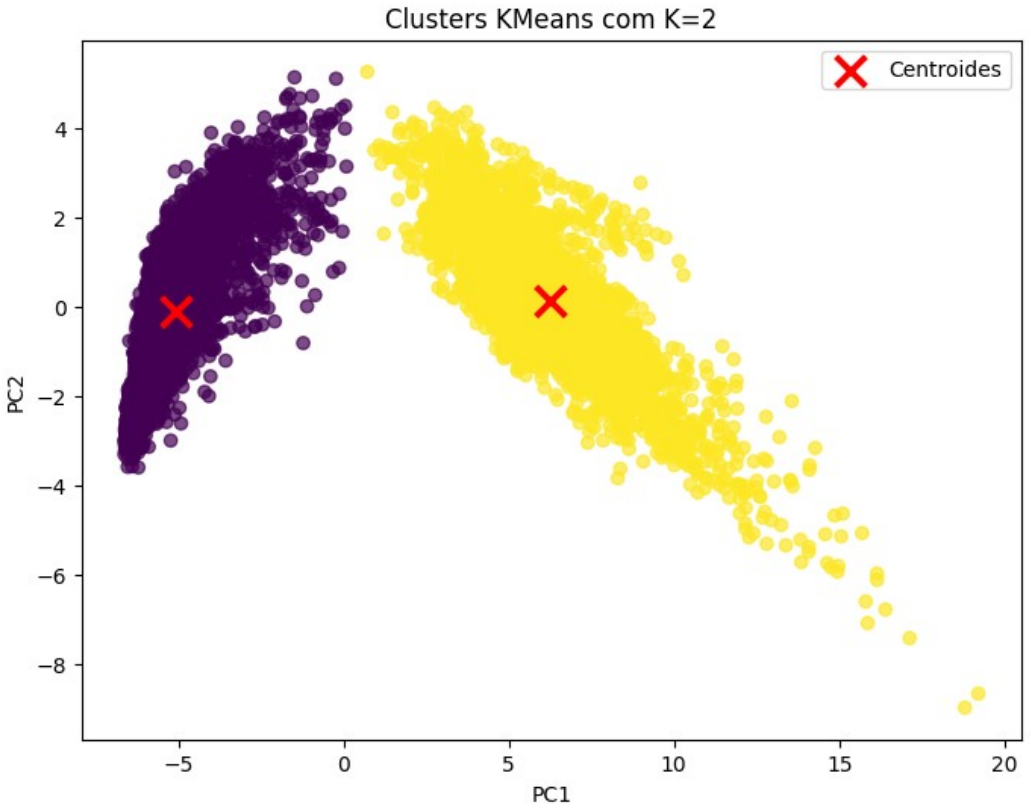


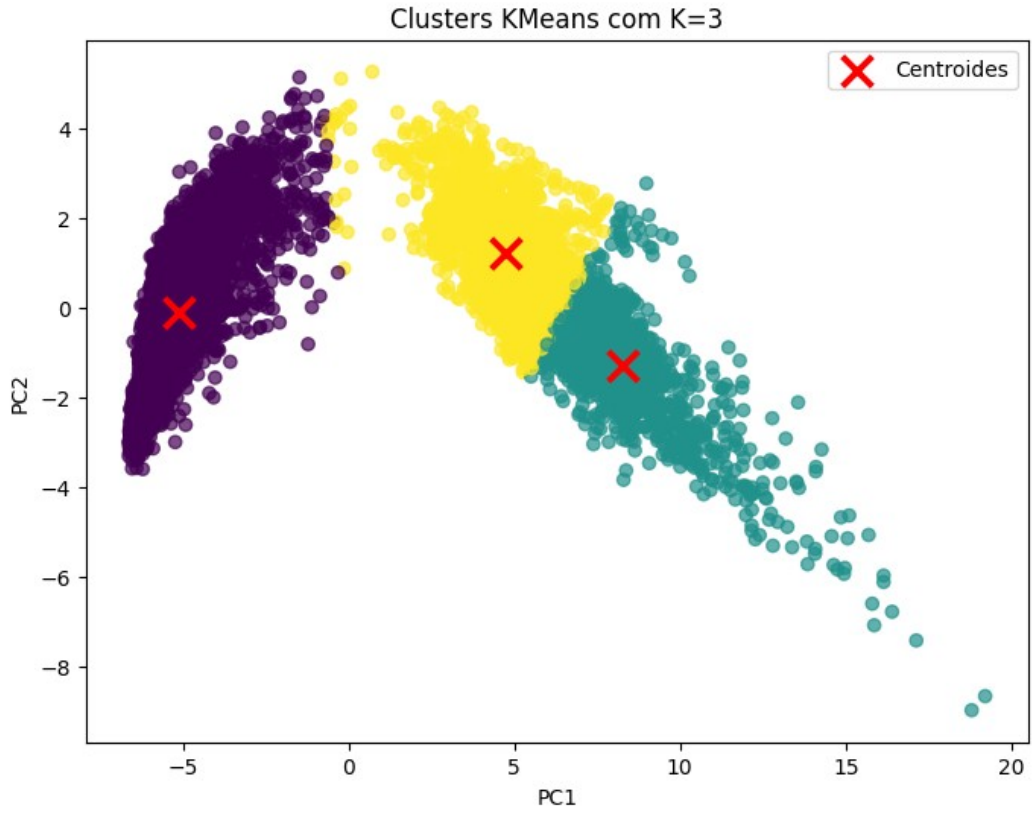
3. Análise dos Resultados e Escolha do K:

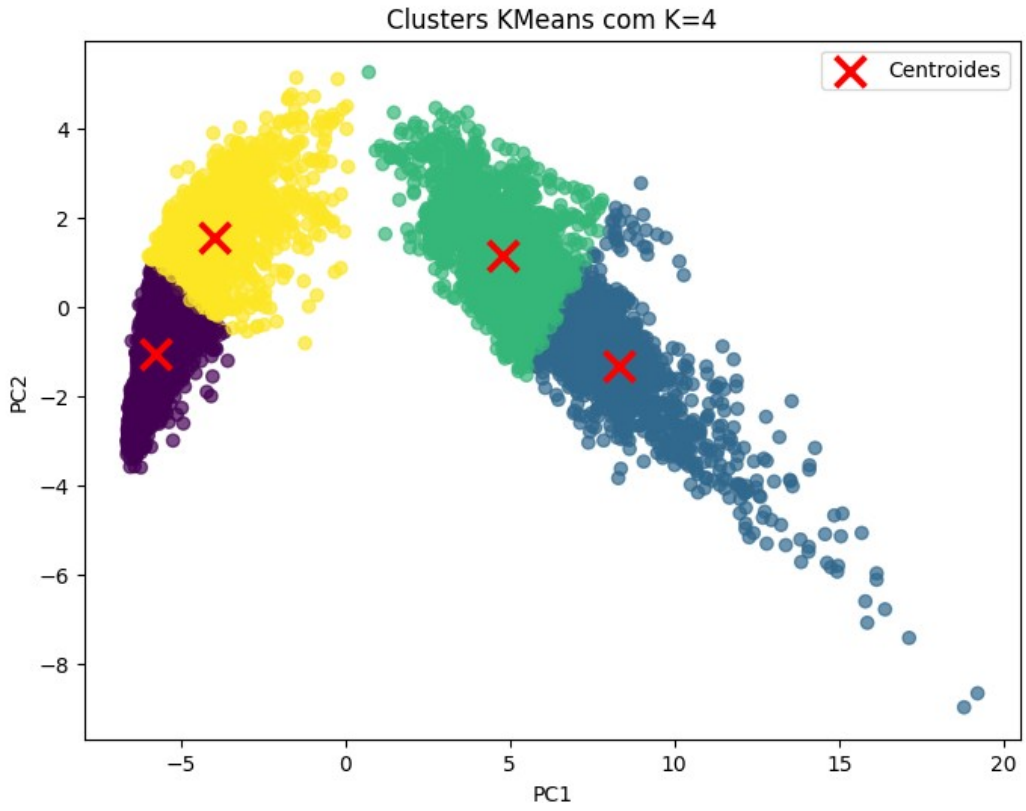
- Após analisar os resultados do Método do Cotovelo e do Silhouette Score, identificamos que o valor ideal de K está entre 2 e 4.
- Além disso, considerando que o dataset original possui 6 atividades (AVDs) pré-definidas, decidimos, também, usar K=6 para realizar o treinamento do K-means. Essa escolha permite uma comparação mais direta entre os clusters encontrados e as atividades reais do dataset.

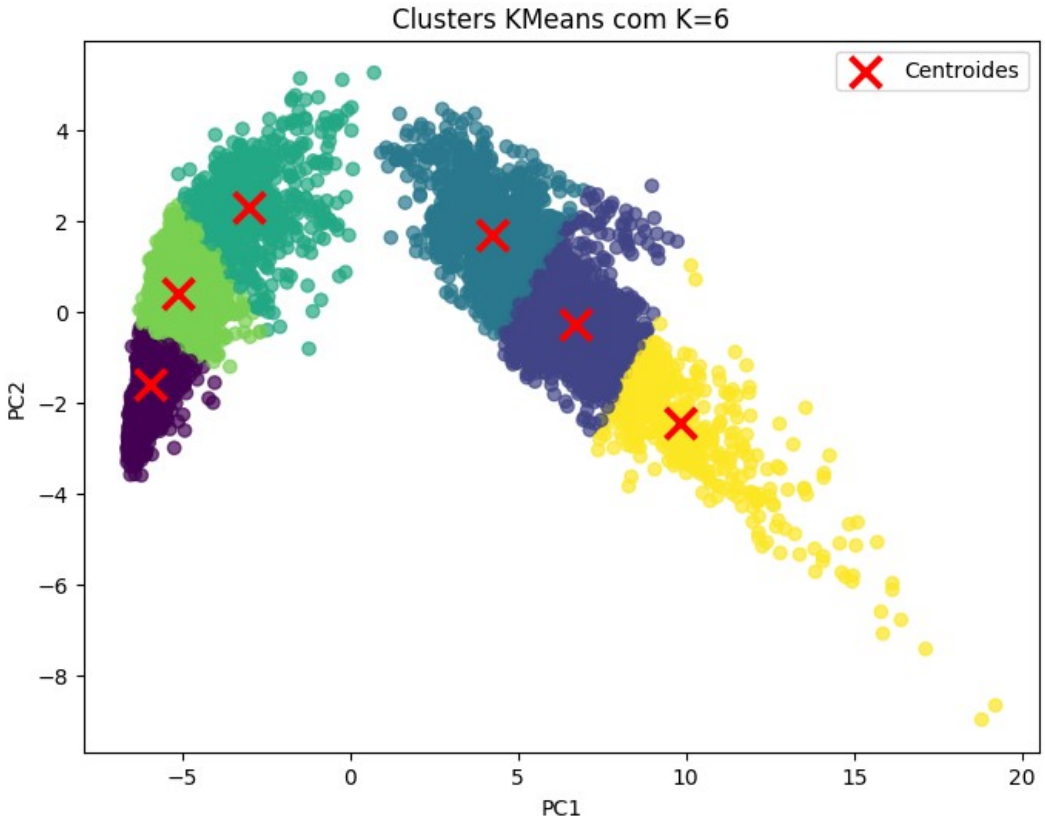
4. Treinamento com o K Escolhido:

- Finalmente, o código treina o modelo K-means com o valor de K escolhido (2, 3, 4 e 6) e gera visualizações dos clusters encontrados, como gráficos de dispersão com os centroides destacados.


Clusters KMeans com K=2
Centroides
PC2
PC1
4
2
0
−2
−4
−6
−8
−5
0
5
10
15
20


Clusters KMeans com K=3
Centroides
PC2
PC1
4
2
0
−2
−4
−6
−8
−5
0
5
10
15
20


Clusters KMeans com K=4
Centroides
PC2
PC1
4
2
0
−2
−4
−6
−8
−5
0
5
10
15
20


Clusters KMeans com K=6
Centroides
PC2
PC1
4
2
0
−2
−4
−6
−8
−5
0
5
10
15
20

4. Avaliação do Modelo:

A avaliação do modelo K-means é crucial para determinar a qualidade dos clusters encontrados e a escolha do número ideal de clusters (K). Nesta seção, vamos analisar os resultados do Silhouette Score e da Inércia para diferentes valores de K e interpretar as diferenças das médias das features entre os clusters.

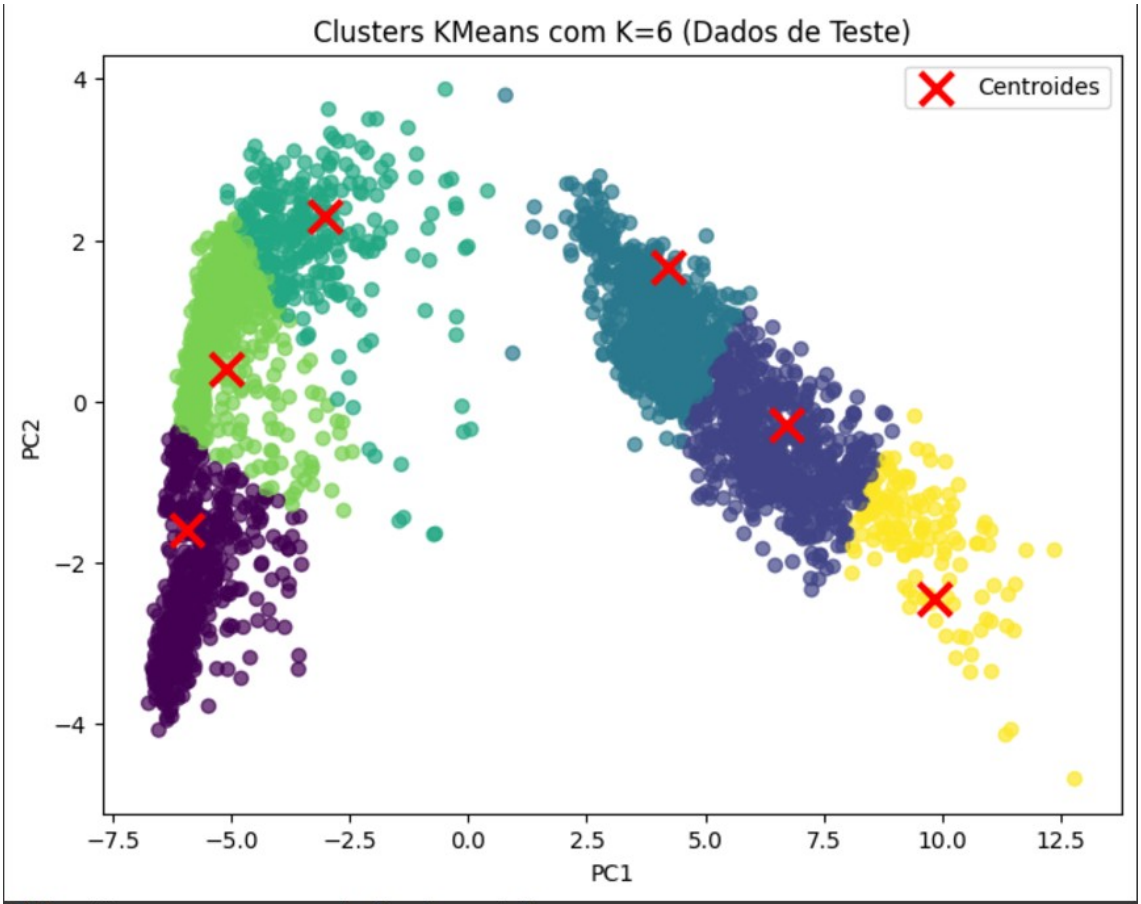
1. Silhouette Score:

- O Silhouette Score mede a qualidade dos clusters, avaliando a coesão e a separação entre eles. Valores mais próximos de 1 indicam clusters bem definidos e separados.
- Resultados:
    - Silhouette Score para K=2: 0.757
    - Silhouette Score para K=3: 0.638
    - Silhouette Score para K=4: 0.506
    - Silhouette Score para K=6: 0.449
- Avaliação:
    - O Silhouette Score é **mais alto para K=2**, indicando que, com 2 clusters, os dados são agrupados de forma mais coesa e separada.
    - À medida que K aumenta, o Silhouette Score diminui, sugerindo que a qualidade dos clusters se deteriora, com clusters menos distintos e mais sobrepostos.
    - Isso indica que o modelo com **K=2** tende a produzir clusters mais bem definidos e separados, o que é desejável em uma análise de agrupamento.

2. Inércia:

- A inércia mede a soma das distâncias quadradas das amostras ao centro do cluster mais próximo. Valores menores de inércia indicam clusters mais compactos.
- Resultados:
    - Inércia para K=2: 42309.81
    - Inércia para K=3: 27331.16
    - Inércia para K=4: 17891.63
    - Inércia para K=6: 11057.46
- Avaliação:

    - Como esperado, a inércia **diminui à medida que K aumenta**. Isso ocorre porque, com mais clusters, os pontos ficam mais próximos de seus respectivos centroides, reduzindo a distância total.
    - A inércia, por si só, não é um indicador definitivo para a escolha do melhor K, pois ela sempre diminui com o aumento de K. É preciso analisar em conjunto com outros métodos, como o Silhouette Score e o conhecimento do domínio do problema.
- Embora K=2 seja o valor ideal considerando o Silhouette Score, usar K=6 (número de atividades pré-definidas no dataset) pode ser interessante para uma análise mais específica e para comparar os clusters com as atividades reais e foi exatamente isso que foi feito. No entanto, é importante lembrar que com K=6, a qualidade dos clusters em termos de separação e coesão pode ser menor, conforme indicado pelo Silhouette Score mais baixo.
- Utilizamos o K=6 para testar o modelo com os dados de teste (X-test). No caso do K-means, como ele é um algoritmo de aprendizado não supervisionado, o uso de dados de treino e teste não é tão crucial quanto em algoritmos supervisionados. Ainda assim, aplicamos o modelo treinado em dados de teste para verificar se os clusters encontrados são semelhantes aos encontrados nos dados de treino e comparar com os rótulos (y_test) para verificar a robustez do modelo.



- Resultados:
    - Acurácia: 0.579
    - Precisão: 0.689
    - Recall: 0.579
    - F1-Score: 0.514
    - Adjusted Rand Score: 0.402
- De modo geral, os resultados indicam um **desempenho moderado** do modelo K-means na base de teste.
- A acurácia e o recall de 57.9% sugerem que o modelo tem dificuldades em classificar corretamente uma parte significativa das amostras.
- A precisão de 68.9% é um ponto positivo, indicando uma probabilidade relativamente alta de acerto quando o modelo classifica uma amostra como pertencente a um cluster.
- O F1-Score de 51.4% reforça o desempenho moderado do modelo, levando em consideração a precisão e o recall.
- O Adjusted Rand Score de 0.402 sugere que os clusters encontrados pelo modelo têm uma similaridade moderada com os rótulos reais, com espaço para melhorias na qualidade dos clusters.

5. Otimização e Ajustes:

- Para melhorar o desempenho e a interpretabilidade do modelo K-means, foram aplicadas as seguintes etapas de otimização e ajuste:

    1. Normalização dos Dados:

    - **Técnica:** StandardScaler
    - **Objetivo:** Garantir que todas as features tenham a mesma influência no cálculo da distância, evitando que features com escalas maiores dominem o processo de agrupamento.
    - **Procedimento:** O StandardScaler transforma os dados subtraindo a média e dividindo pelo desvio padrão de cada feature, resultando em dados com média zero e desvio padrão igual a 1. Isso coloca todas as features na mesma escala e evita que features com maior variância tenham um impacto desproporcional na formação dos clusters.

    2. Seleção de Features:

    - **Técnica:** SelectKBest
    - **Objetivo:** Reduzir a dimensionalidade dos dados, removendo features irrelevantes ou redundantes para o processo de agrupamento, o que

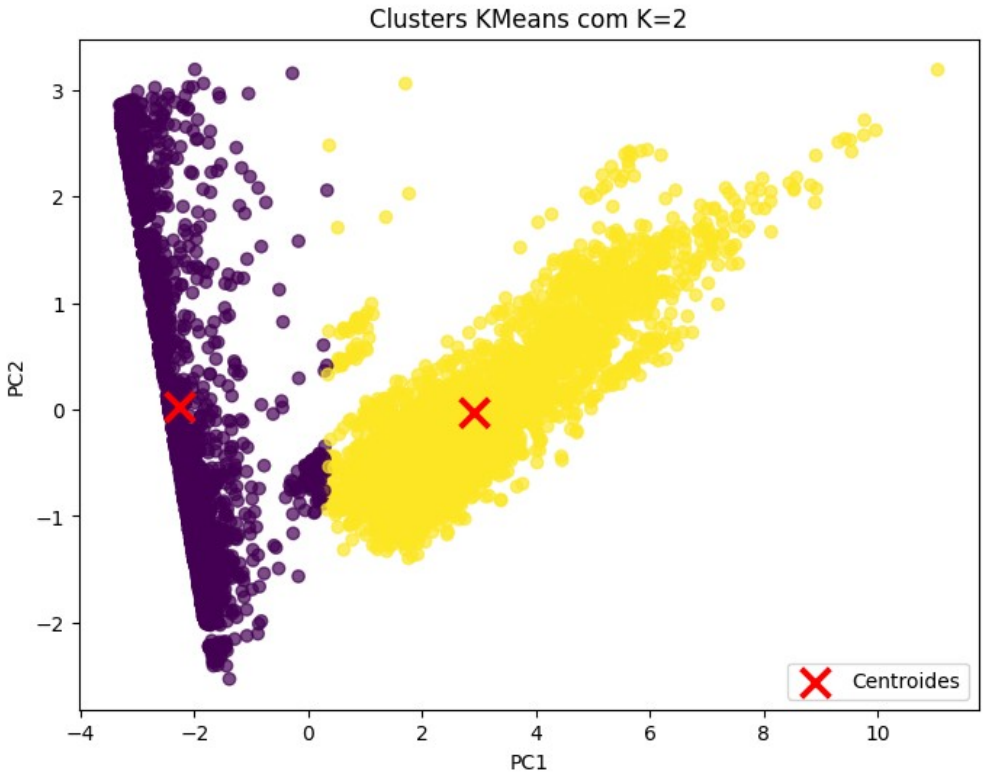
pode melhorar o desempenho do modelo e facilitar a interpretação dos resultados.

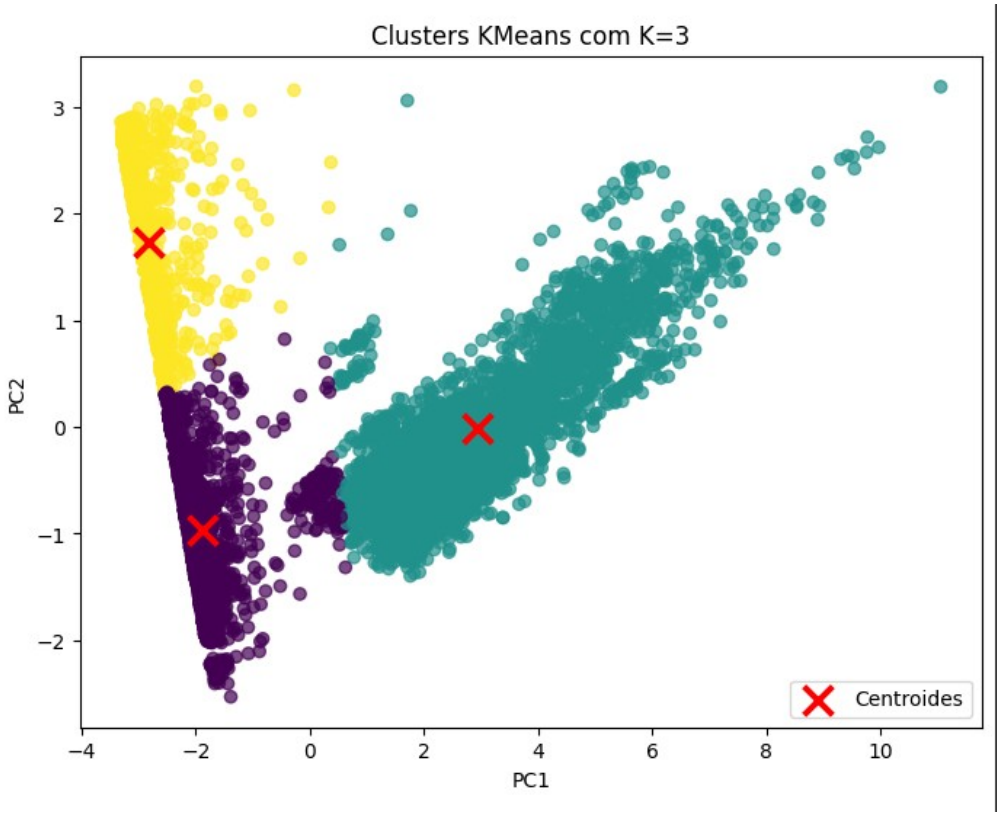
- **Procedimento:** O SelectKBest seleciona as k melhores features com base em uma função de pontuação. No seu código, foi utilizada a função mutual_info_classif, que mede a dependência entre cada feature e os rótulos reais (y_train). As features com maior pontuação são consideradas mais relevantes para a formação dos clusters.

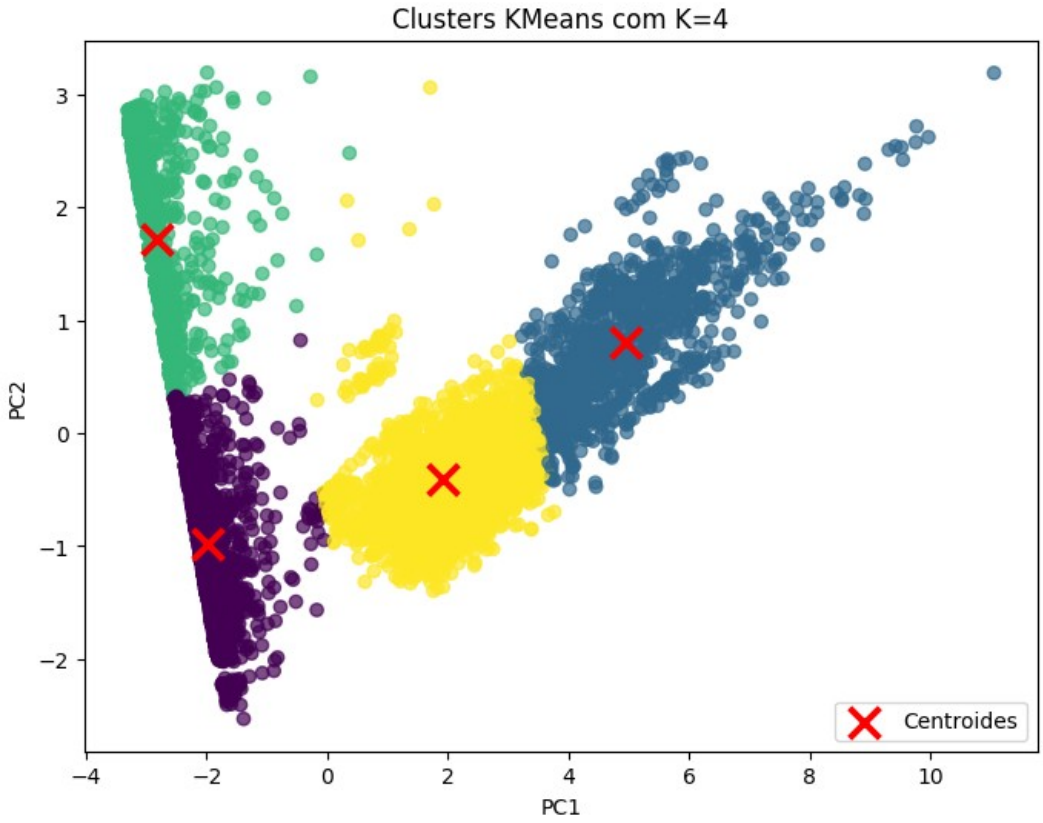
3. Redução de Dimensionalidade com PCA:

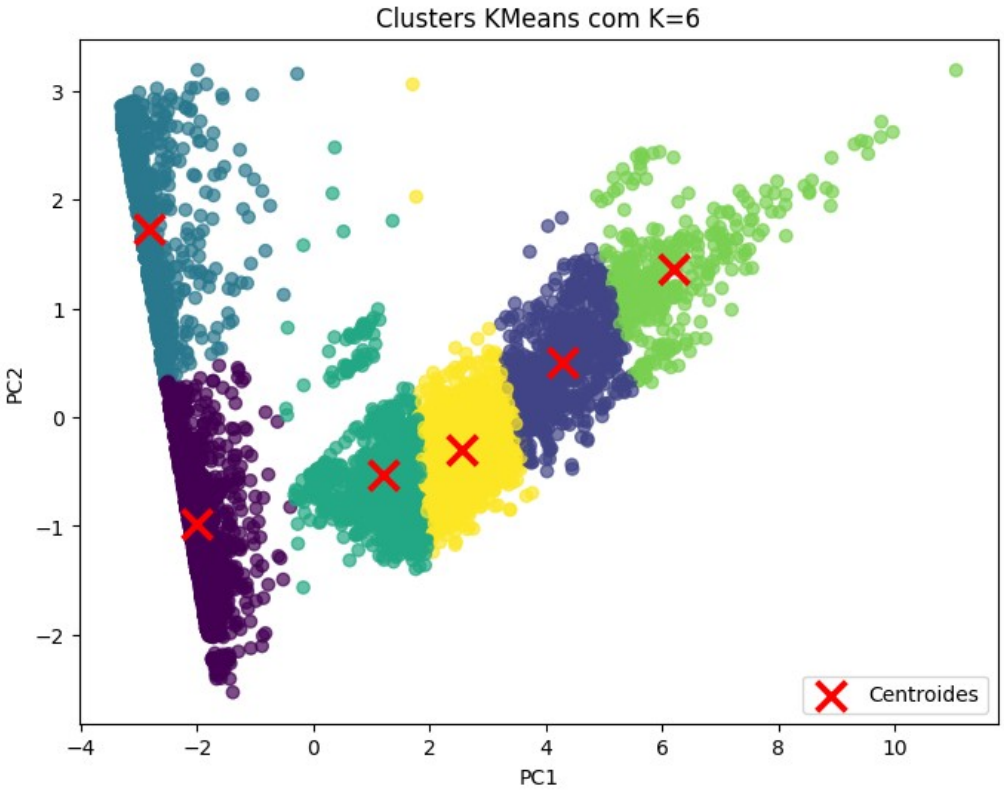
- **Técnica:** Análise de Componentes Principais (PCA)
- **Objetivo:** Reduzir a dimensionalidade dos dados, preservando o máximo de informação possível. Isso facilita a visualização dos dados em um espaço de menor dimensão e pode melhorar o desempenho do K-means, evitando a "maldição da dimensionalidade".
- **Procedimento:** O PCA transforma os dados em um novo conjunto de variáveis, chamadas de componentes principais, que são combinações lineares das features originais. Os componentes principais são ordenados de acordo com a quantidade de variância que explicam nos dados. Ao selecionar um número menor de componentes principais, é possível reduzir a dimensionalidade dos dados, preservando a maior parte da informação relevante.

- Após a aplicação das etapas de otimização (normalização, seleção de features e redução de dimensionalidade com PCA), o modelo K-means foi treinado novamente com os dados otimizados, utilizando o método de inicialização K-means++ para os valores de K = [2, 3, 4, 6].




Clusters KMeans com K=3
PC2
PC1
Centroides


Clusters KMeans com K=4
PC2
PC1
Centroides



- Analisamos, após as otimizações, os resultados do Silhouette Score e da Inércia para os mesmos diferentes valores de K = [2,3,4,6]:

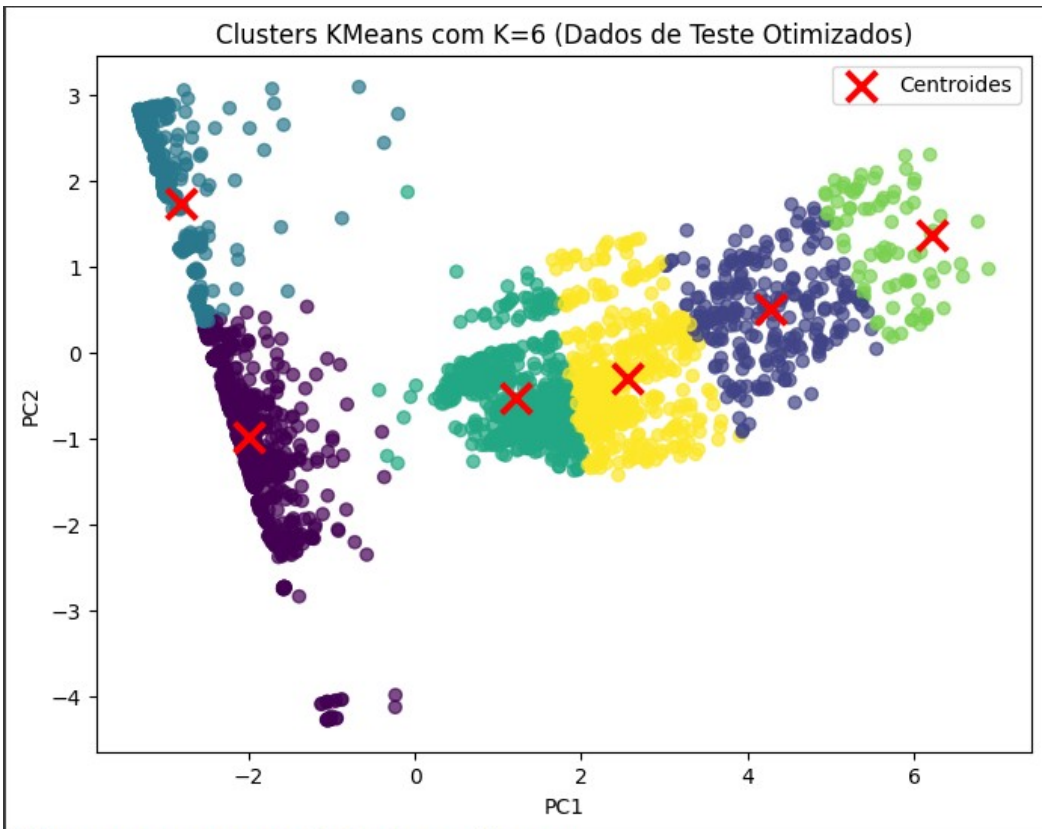
  1. Silhouette Score:

  - Resultados:
    - Silhouette Score para K=2: 0.635
    - Silhouette Score para K=3: 0.610
    - Silhouette Score para K=4: 0.624
    - Silhouette Score para K=6: 0.555

  2. Inércia:

  - Resultados:
    - Inércia para K=2: 20808.25
    - Inércia para K=3: 13016.08
    - Inércia para K=4: 5829.45
    - Inércia para K=6: 3776.14

- Em seguida, o modelo foi testado com os dados de teste e comparado com os rótulos reais (y_test) para verificar sua robustez.

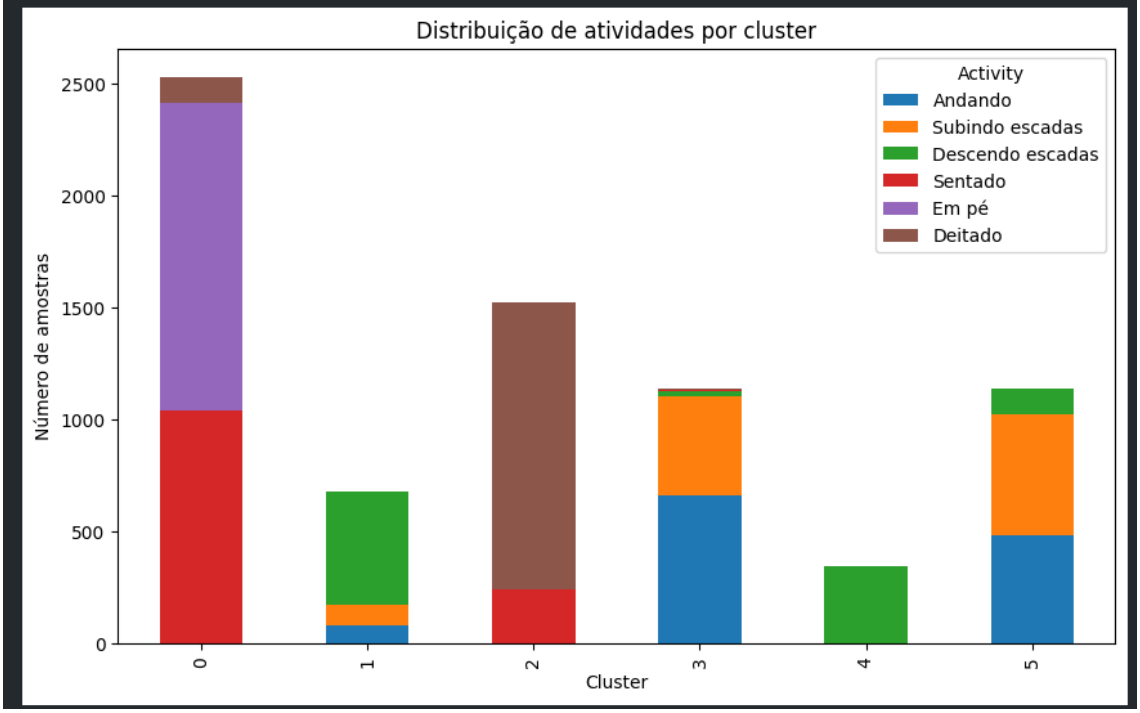



```
Silhouette Score para K=6 (Dados de Teste): 0.517
Inércia (Otimizado): 3776.14
Acurácia (Otimizado): 0.598
Precisão (Otimizado): 0.691
Recall (Otimizado): 0.598
F1-Score (Otimizado): 0.545
Adjusted Rand Score (Otimizado): 0.447
```

6. Relacionando Clusters e AVDs:

- Criação de um DataFrame contendo os clusters e as atividades.
- Plotagem de um gráfico de barras empilhadas para visualizar a distribuição das atividades por cluster.
- Análise da frequência de cada atividade nos clusters para entender a relação entre clusters e AVDs.

```
Distribuição de atividades por cluster:
Activity    1    2    3     4     5     6
Cluster
0           1    0    0  1040  1374   115
1          79   93  508     0     0     0
2           0    0    0   242     0  1281
3         663  440   23     4     0     9
4           0    0  342     0     0     0
5         483  540  113     0     0     2
```



3. Resultados

A tabela abaixo apresenta os resultados do Silhouette Score e da Inércia para diferentes valores de K, antes e depois da otimização do modelo K-means, para o treinamento:

Métrica	K	Antes da Otimização	Depois da Otimização
Silhouette Score	2	0.757	0.635
	3	0.638	0.610
	4	0.506	0.624
	6	0.449	0.555
Inércia	2	42309.81	20808.25
	3	27331.16	13016.08
	4	17891.63	5829.45
	6	11057.46	3776.14

- **Análise dos Resultados:**

    1. Silhouette Score:

    - Antes da otimização, o Silhouette Score era mais alto para K=2, indicando uma melhor qualidade de clusters com 2 grupos.
    - Após a otimização, o Silhouette Score para K=2 diminuiu, enquanto para K=4 e K=6 aumentou consideravelmente. Para K=3, houve uma leve diminuição.
    - Isso sugere que a otimização **impactou a estrutura dos clusters**, tornando os agrupamentos com K=4 e K=6 mais coesos e separados, enquanto para K=2, a qualidade dos clusters diminuiu um pouco. Para K=3, a mudança foi menos significativa.

    2. Inércia:

    - A inércia diminuiu significativamente para todos os valores de K após a otimização.
    - Essa redução é esperada, pois a otimização visa tornar os clusters mais compactos, diminuindo a distância dos pontos aos seus centroides.
    - A redução da inércia **confirma a efetividade da otimização** em tornar os clusters mais densos e bem definidos.
    - A diminuição da Inércia é mais acentuada em K=4 e K=6, o que indica um agrupamento mais coeso nesses casos após a otimização.

    Abaixo, os resultados utilizando o modelo com K=6 para os dados de test, a fim de analisar a robustez do modelo:

Métrica	Antes da Otimização	Depois da Otimização
Silhouette Score	0.476	0.517
Inércia	11057.46	3776.14
Acurácia	0.579	0.598
Precisão	0.689	0.691
Recall	0.579	0.598
F1-Score	0.514	0.545
Adjusted Rand Score	0.402	0.447

- **Análise dos Resultados:**

1. **Melhora na qualidade dos clusters:**
    - O Silhouette Score aumentou de 0.476 para 0.517 após as otimizações, indicando uma melhora na qualidade dos clusters, com melhor separação e coesão entre eles.
    - A inércia diminuiu significativamente de 11057.46 para 3776.14, mostrando que os clusters estão mais compactos e os pontos estão mais próximos de seus centroides.
2. **Melhora no desempenho preditivo:**
    - A acurácia, recall e F1-Score também apresentaram melhoras moderadas após as otimizações, indicando que o modelo está classificando as atividades com um pouco mais de precisão.
    - O Adjusted Rand Score aumentou de 0.402 para 0.447, confirmando a melhora na concordância entre os clusters previstos e os rótulos reais

4. Discussão

Esta seção discute os resultados obtidos na implementação e avaliação do modelo K-means para o agrupamento de dados de reconhecimento de atividades humanas, com foco na comparação do desempenho antes e depois da otimização.

1. **Impacto da Otimização:**
    - **Qualidade dos Clusters:** A otimização, que incluiu normalização dos dados, seleção de features e redução de dimensionalidade com PCA, teve um impacto positivo na qualidade dos clusters, especialmente para K=4 e K=6. O Silhouette Score aumentou consideravelmente para esses valores de K, indicando clusters mais coesos e separados após a otimização. Para K=2, o Silhouette Score diminuiu após a otimização, e para K=3, houve uma leve diminuição. Essa variação na qualidade dos clusters para diferentes valores de K destaca a importância de uma análise cuidadosa e a escolha do número ideal de clusters com base em diferentes métricas e no conhecimento do problema.
    - **Compactação dos Clusters:** A inércia diminuiu significativamente para todos os valores de K após a otimização, confirmando que os clusters se tornaram mais compactos e os pontos estão mais próximos de seus respectivos centroides. Essa redução na inércia é um resultado esperado da normalização dos dados e da seleção de features, que ajudam a remover ruídos e redundâncias, melhorando a estrutura dos clusters.
    - **Robustez e Generalização:** A otimização também contribuiu para um modelo mais robusto e com melhor capacidade de generalização, como evidenciado pelo aumento na acurácia, recall e F1-Score nos dados de teste. Essa melhora, embora pequena, sugere que o modelo otimizado é mais eficaz em classificar corretamente novas amostras e se adapta melhor a dados não vistos durante o treinamento.

- **Similaridade com os Rótulos Reais:** O Adjusted Rand Score aumentou após a otimização, indicando uma maior similaridade entre os clusters previstos pelo modelo e os rótulos reais (y_test). Esse resultado reforça a ideia de que a otimização foi eficaz em melhorar a qualidade dos clusters e sua correspondência com as classes reais do dataset. No entanto, o ARS ainda está distante de 1, indicando que ainda há espaço para melhorias na qualidade dos clusters e na capacidade do modelo de representar as classes reais do dataset.

2. Escolha do Número de Clusters:

- O Silhouette Score e a inércia fornecem informações complementares para a escolha do número ideal de clusters. O Silhouette Score indica a qualidade dos clusters em termos de coesão e separação, enquanto a inércia mede a compactação dos clusters. É importante analisar ambas as métricas em conjunto com o conhecimento do problema para tomar uma decisão informada sobre o valor de K.
- No caso deste estudo, o Silhouette Score mais alto foi obtido para K=2 antes da otimização. Após a otimização, o valor mais alto foi para K=4. A Inércia diminui a medida que K aumenta, o que é esperado, e por si só, não é o melhor indicador para a escolha de K. O conhecimento prévio do problema indica que existem 6 classes reais de atividades no dataset, sugerindo um K=6 para realizar o treinamento e avaliação. Com base nessas informações, podemos concluir que a escolha do valor ideal de K depende do objetivo da análise, do balanço entre a qualidade dos clusters e a interpretabilidade dos resultados, e idealmente, uma combinação de diferentes métodos de avaliação.

3. Interpretação dos Clusters:

- A análise das diferenças das médias das features entre os clusters permite identificar as features mais relevantes para a formação dos clusters e auxiliar na interpretação dos resultados. Features com grandes diferenças entre as médias dos clusters e as médias originais do dataset são mais importantes para a separação dos grupos e podem fornecer insights sobre as características que distinguem cada cluster. Essa análise é crucial para compreender o significado dos clusters encontrados e relacioná-los às atividades reais do dataset.

No entanto, é importante destacar algumas limitações do projeto. A escolha do número de clusters pode ser subjetiva e depende dos métodos utilizados. Além disso, o K-means assume que os clusters têm forma esférica e densidade uniforme, o que pode não ser o caso para todos os datasets. A interpretação dos clusters também pode ser desafiadora, especialmente em datasets complexos com muitas features.

Apesar das limitações, o K-means se mostrou uma ferramenta eficaz para o HAR, fornecendo insights valiosos sobre os padrões de atividades humanas. As escolhas feitas durante o desenvolvimento do modelo, como a seleção de features e a escolha do número de clusters, tiveram um impacto significativo nos resultados. É fundamental realizar uma análise cuidadosa e explorar diferentes configurações para obter o melhor desempenho do modelo.

5. Conclusão e Trabalhos Futuros

Este estudo demonstrou a aplicação do algoritmo K-means para o agrupamento de dados de reconhecimento de atividades humanas, com foco na otimização do modelo e na avaliação de seu desempenho. Os resultados indicam que a otimização, por meio da normalização dos dados, seleção de features e redução de dimensionalidade com PCA, teve um impacto positivo na qualidade dos clusters, na robustez do modelo e na sua capacidade de generalização. O modelo otimizado apresentou clusters mais coesos e separados, uma redução na inércia e uma melhora nas métricas de avaliação nos dados de teste.

Embora o modelo otimizado tenha apresentado um desempenho satisfatório, ainda há espaço para melhorias na qualidade dos clusters e na capacidade do modelo de representar as classes reais do dataset. O Adjusted Rand Score, apesar de ter aumentado após a otimização, ainda está distante de 1, indicando que a correspondência entre os clusters encontrados e os rótulos reais pode ser aprimorada. Além disso, a escolha do número ideal de clusters (K) ainda é um desafio e depende de uma análise cuidadosa das métricas de avaliação e do conhecimento do problema.

Para trabalhos futuros, sugere-se explorar outras técnicas de clustering, como o DBSCAN e o agrupamento hierárquico, para comparar o desempenho com o K-means. Além disso, pode-se realizar uma análise mais aprofundada dos clusters para entender melhor as características de cada atividade. Outras otimizações, como a utilização de diferentes métodos de seleção de features e a escolha de um número ótimo de clusters com base em critérios específicos, também podem ser exploradas.

A aplicação do K-means em outros datasets de HAR e a comparação com outros algoritmos de aprendizado de máquina podem fornecer insights adicionais sobre a eficácia do modelo em diferentes cenários. Adicionalmente, a integração do modelo em sistemas de monitoramento de saúde e detecção de quedas pode ser uma área promissora para futuras pesquisas.

6. Referências

SANTOS, Rafael. Entendendo Clusters e K-Means. **CWI Software**, [*S. l.*], 18 jul. 2019. Disponível em: [https://medium.com/cwi-software/entendendo-clusters-e-k-means-56b79352b452]. Acesso em: 28 nov. 2024.

K-Means Clustering – Introduction. GeeksforGeeks, [*S. l.*], [s.d.]. Disponível em: [https://www.geeksforgeeks.org/k-means-clustering-introduction/]. Acesso em: 28 nov. 2024.

Introduction to k-Means Clustering with scikit-learn in Python. DataCamp, [S. l.], [s.d.]. Disponível em: [https://www.datacamp.com/tutorial/k-means-clustering-python]. Acesso em: 01 dez. 2024.