Relatório Técnico: Implementação e Análise do Algoritmo de K-means com o Dataset Human Activity Recognition

Anna Giulia Gomes Miranda

Victoria Beatriz Silva de Azevedo Reis

03 de novembro de 2024

Resumo

Este projeto implementa e analisa o desempenho do algoritmo K-means no conjunto de dados "Human Activity Recognition Using Smartphones". A metodologia inclui uma análise exploratória, seguida pela redução de dimensionalidade com PCA, além da determinação do número ideal de clusters usando o Método do Cotovelo e o Silhouette Score. Os resultados indicam que o modelo conseguiu identificar padrões de comportamento humano a partir de informações de sensores, com métricas de inércia e Silhouette Score indicando a eficácia do algoritmo. A pesquisa sugere que a combinação de K-means e PCA é eficaz, mas oferece espaço para melhorias futuras.

Palavras-chave: K-means, Reconhecimento de Atividades Humanas, PCA, Sensores de Smartphones.

Introdução

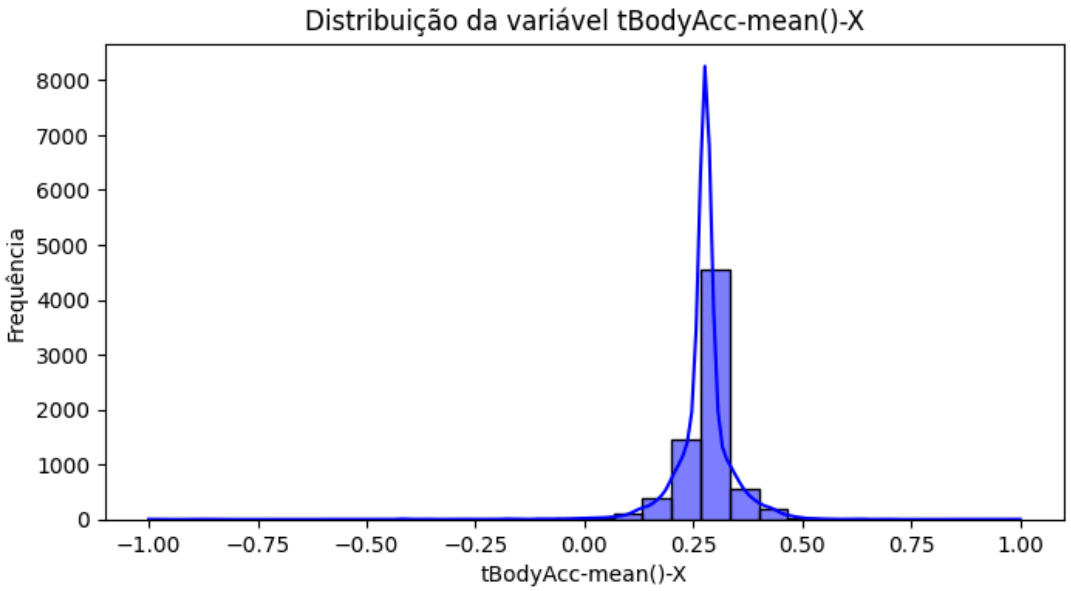
A identificação de atividades humanas por meio de sensores de smartphones tem atraído a atenção devido à sua utilidade em campos como saúde e monitoramento. Informações precisas sobre movimentos podem ser obtidas a partir de dados provenientes de acelerômetros e giroscópios. Este projeto explora a aplicação de aprendizado não supervisionado, focando no algoritmo K-means, para organizar essas atividades. A escolha pelo K- means fundamentada na sua simplicidade e eficiência no processamento de grandes quantidades de dados, especialmente após a implementação de técnicas de redução de dimensionalidade que preservaram 95% da variância dos dados originais.

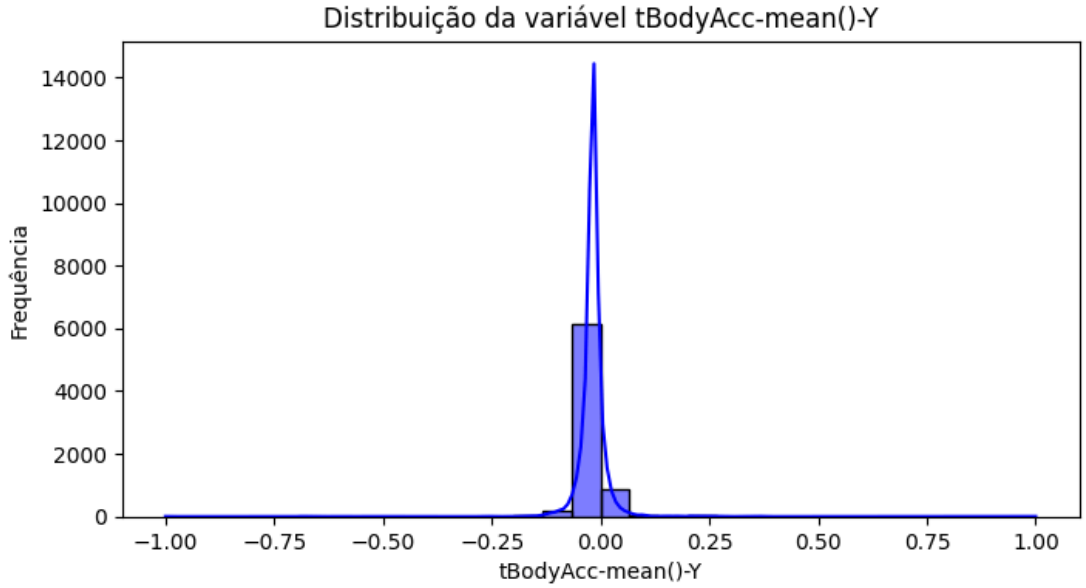
Metodologia

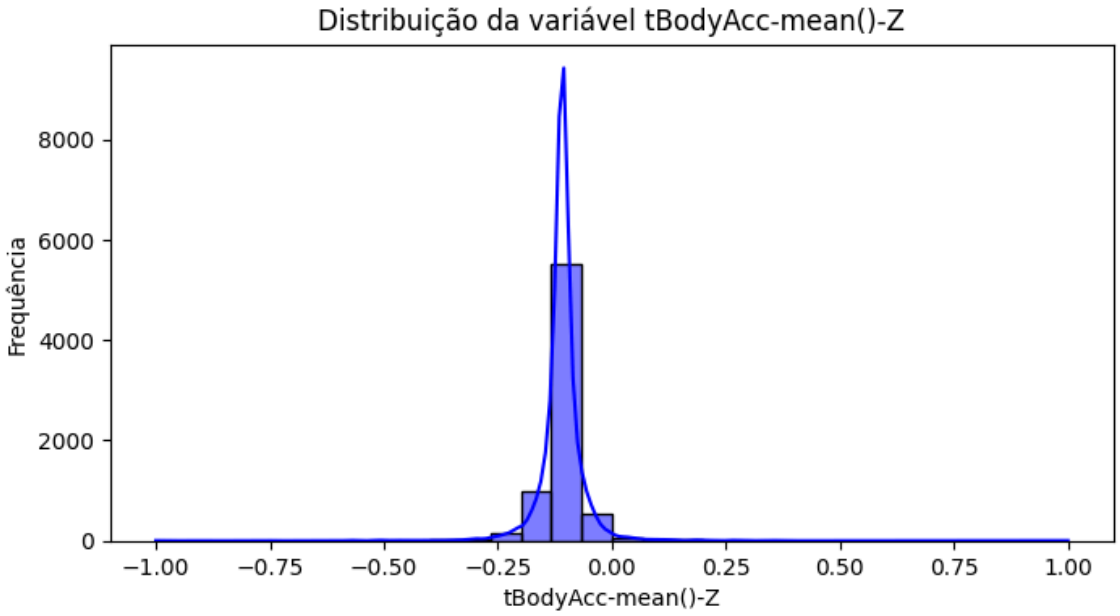
Análise Exploratória

Os datasets de treinamento e teste foram importados e analisados levando em conta a existência de valores ausentes e a correlação entre as variáveis. Foram eliminadas as linhas com valores ausentes, e uma matriz de correlação foi gerada para identificar as relações entre as variáveis dos sensores. Além disso, ao analisar a distribuição dessas variáveis por meio de gráficos de densidade foi revelado padrões relevantes nos dados.

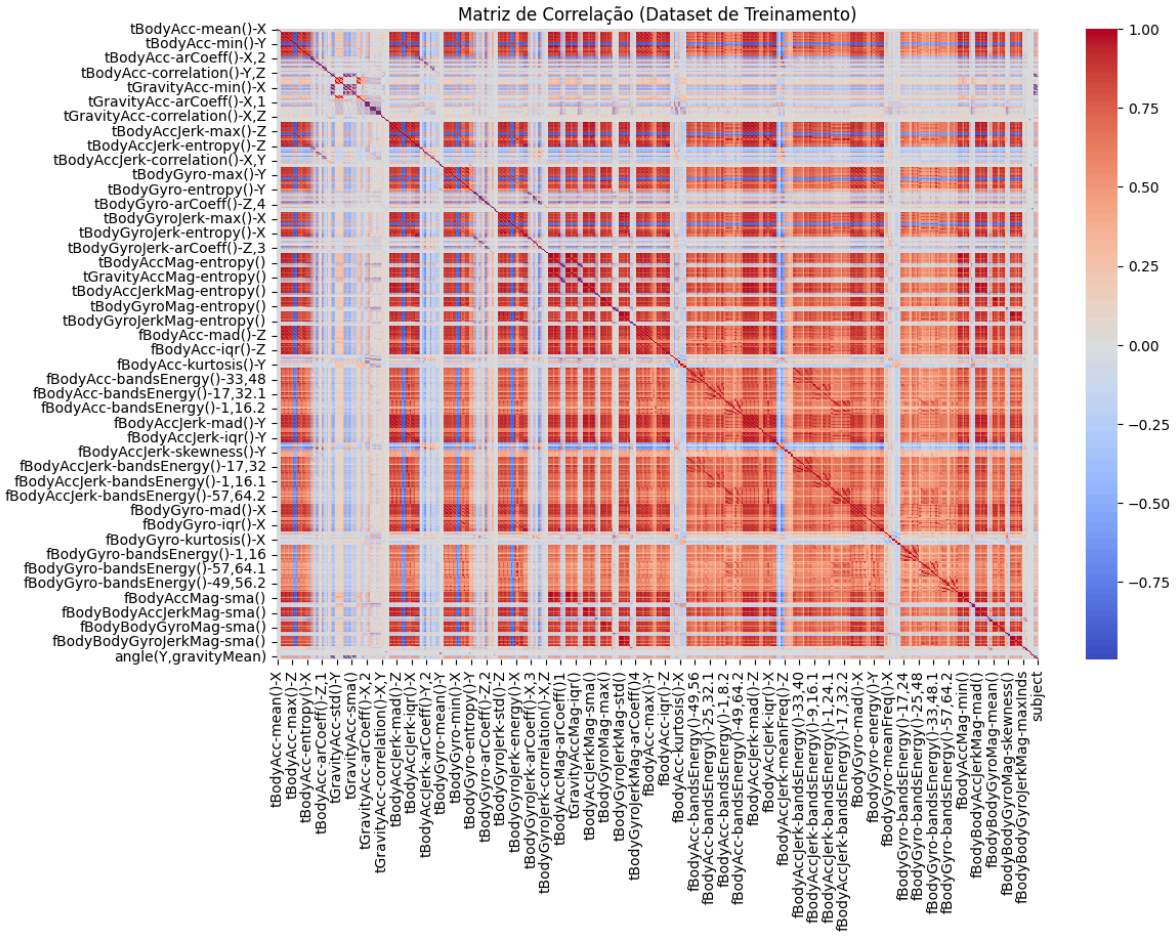
Abaixo podemos visualizar distribuições de algumas variáveis que foram selecionadas exceto a última, pois se assumiu que esta era o alvo:




Distribuição da variável tBodyAcc-mean()-Y
Frequência
14000
12000
10000
8000
6000
4000
2000
0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
tBodyAcc-mean()-Y

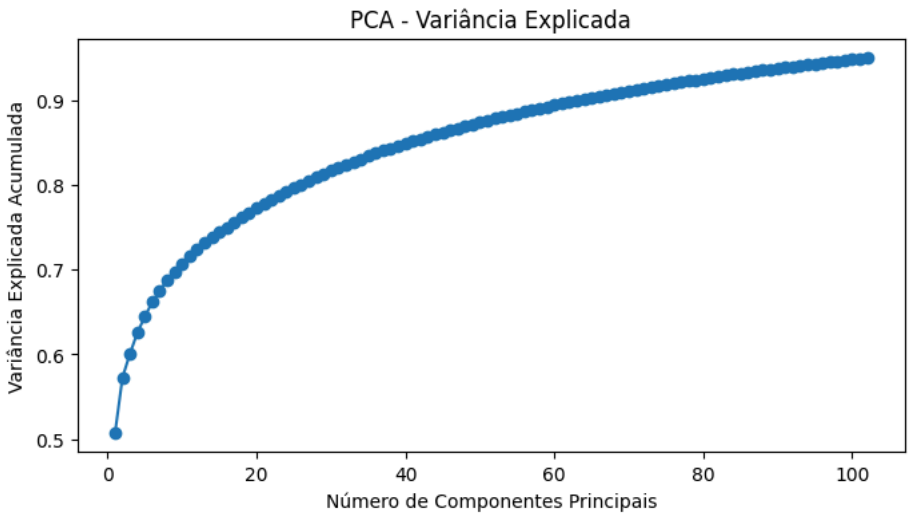

Distribuição da variável tBodyAcc-mean()-Z
Frequência
8000
6000
4000
2000
0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
tBodyAcc-mean()-Z

E a matriz de correlação para as colunas de sensores:

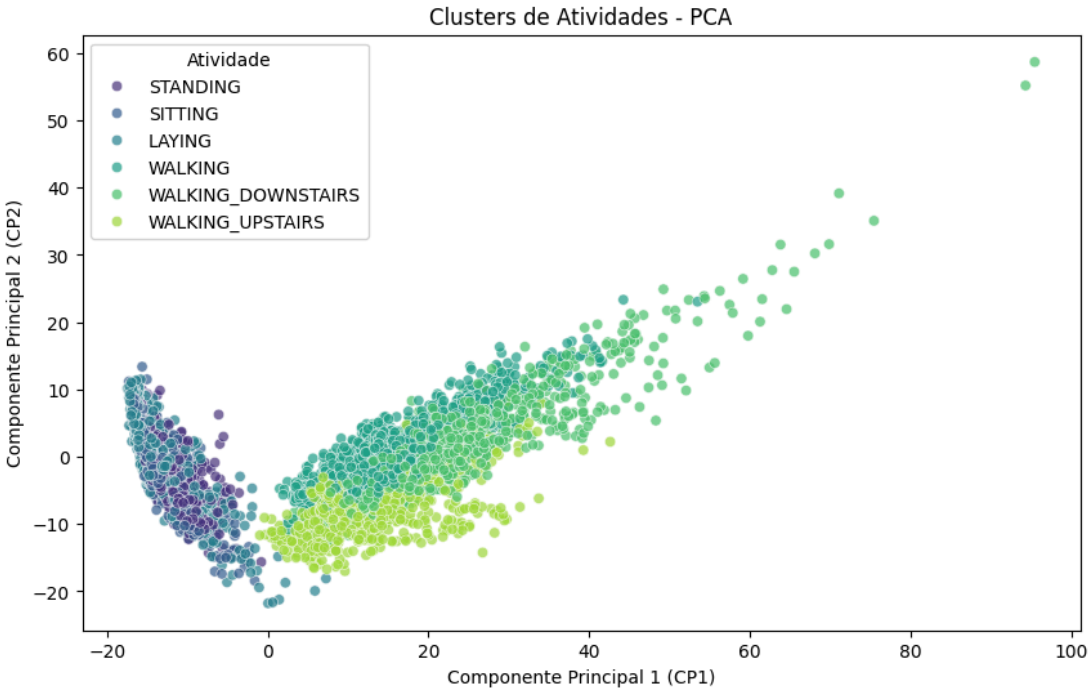


Normalização e Redução de Dimensionalidade

As variáveis foram normalizadas, garantindo que cada uma contribuísse de maneira igual ao agrupamento, uma vez que as medições dos sensores apresentam escalas distintas. Em seguida, foi aplicado a Análise de Componentes Principais (PCA), que preservou 95% da variância dos dados. Esse processo não apenas aumentou a eficiência computacional do algoritmo K-means, mas também simplificou a visualização dos clusters.



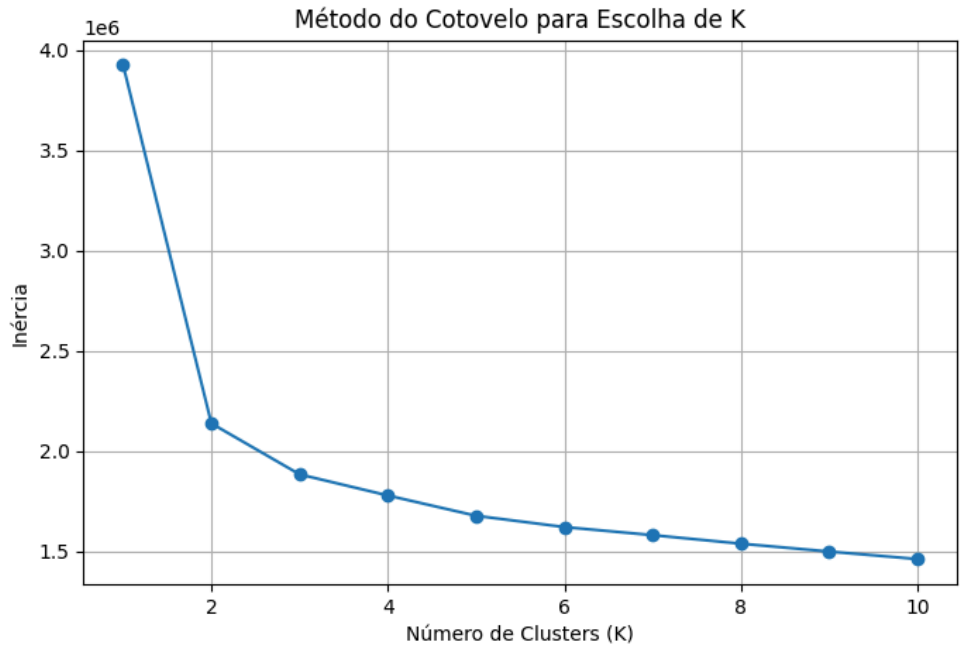
Abaixo temos um DataFrame com os resultados do PCA e as etiquetas de cada atividade, é possível visualizar os dois primeiros componentes principais em um gráfico de dispersão:

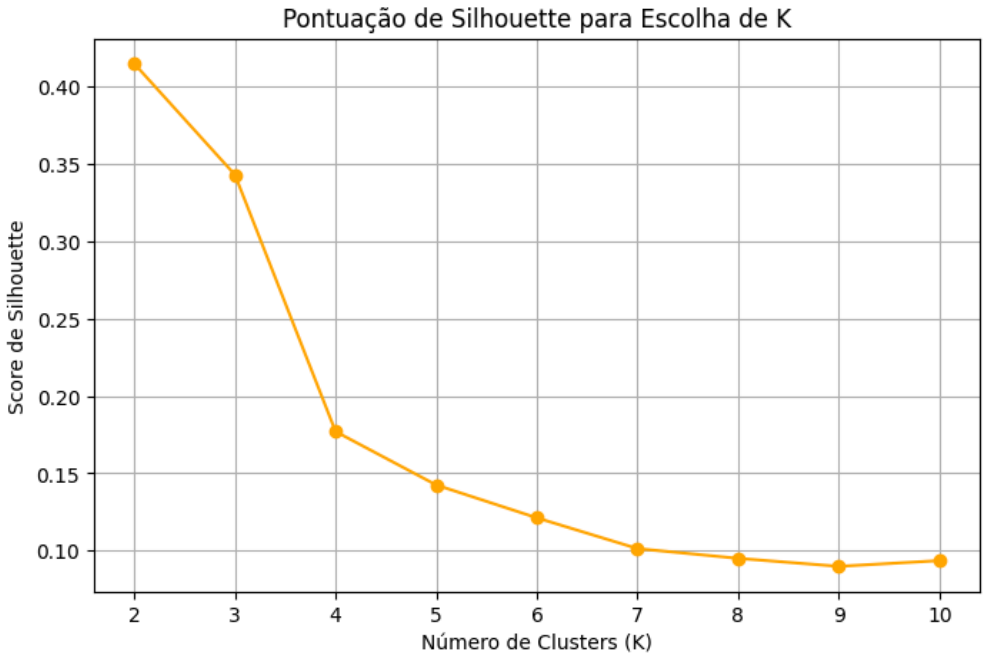


Implementação do Algoritmo K-means

A implementação do algoritmo K-means foi realizada sobre os dados que foram transformados. Para determinar o número ideal de clusters, foram utilizados o Método do Cotovelo e a métrica *Silhouette Score*. O primeiro identificou potenciais pontos de inflexão no gráfico da soma das distâncias quadradas internas, enquanto o segundo

avaliou a eficácia dos clusters gerados com base na coesão interna e separação entre os grupos. O código também mostrou a apresentação dos clusters com o gráfico de dispersão dos dois primeiros componentes principais do PCA, simplificando a avaliação visual dos resultados. Após a análise, optou-se por um número final de clusters com base na maximização do *Silhouette Score*, garantindo melhor separação entre os grupos.





O uso de funções como KMeans para efetuar o agrupamento e PCA para diminuir a dimensionalidade permitiu um fluxo eficaz, com o pré-processamento dos dados sendo essencial para melhorar o desempenho do algoritmo. O código é flexível e modular, possibilitando modificações no número de clusters e reexecuções para verificar a consistência dos resultados.

Resultados

Escolha do Número de Clusters

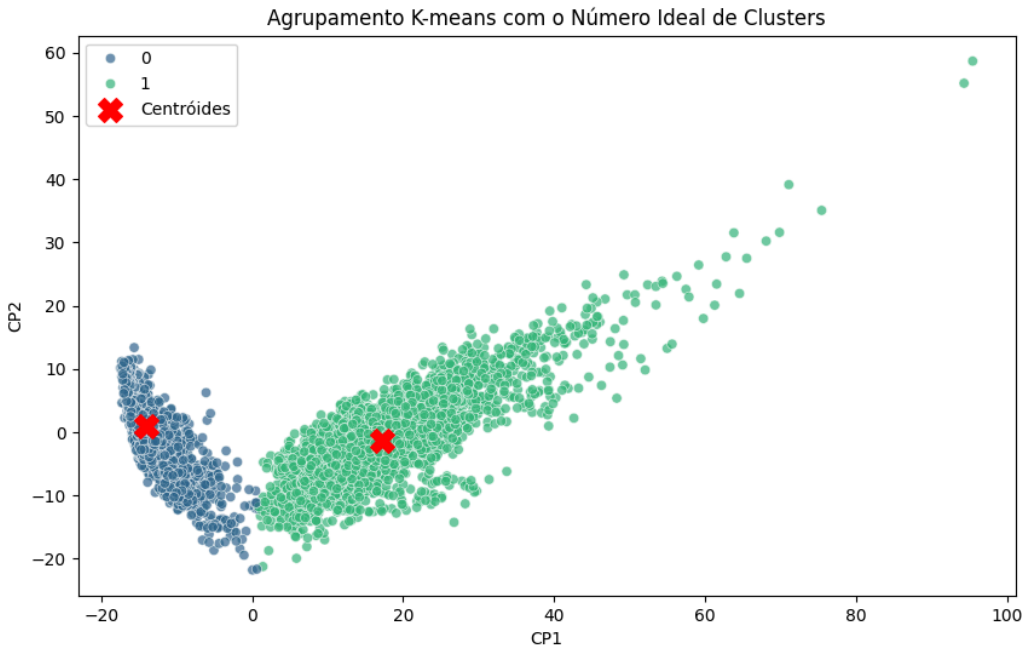
O *Método do Cotovelo* indicou a escolha entre 2 e 3 clusters como pontos de inflexão, mas após realizar o cálculo do *Silhouette Score*, foi identificado que o valor ideal era 2 clusters, com um Silhouette Score de 0.4149, o que indica uma separação moderada entre os grupos. Este resultado mostra uma separação adequada entre os grupos, possibilitando a criação de um modelo com uma melhor coesão intra-cluster e maior separação inter-cluster, alcançando assim um equilíbrio através da maximização do *Silhouette Score*.

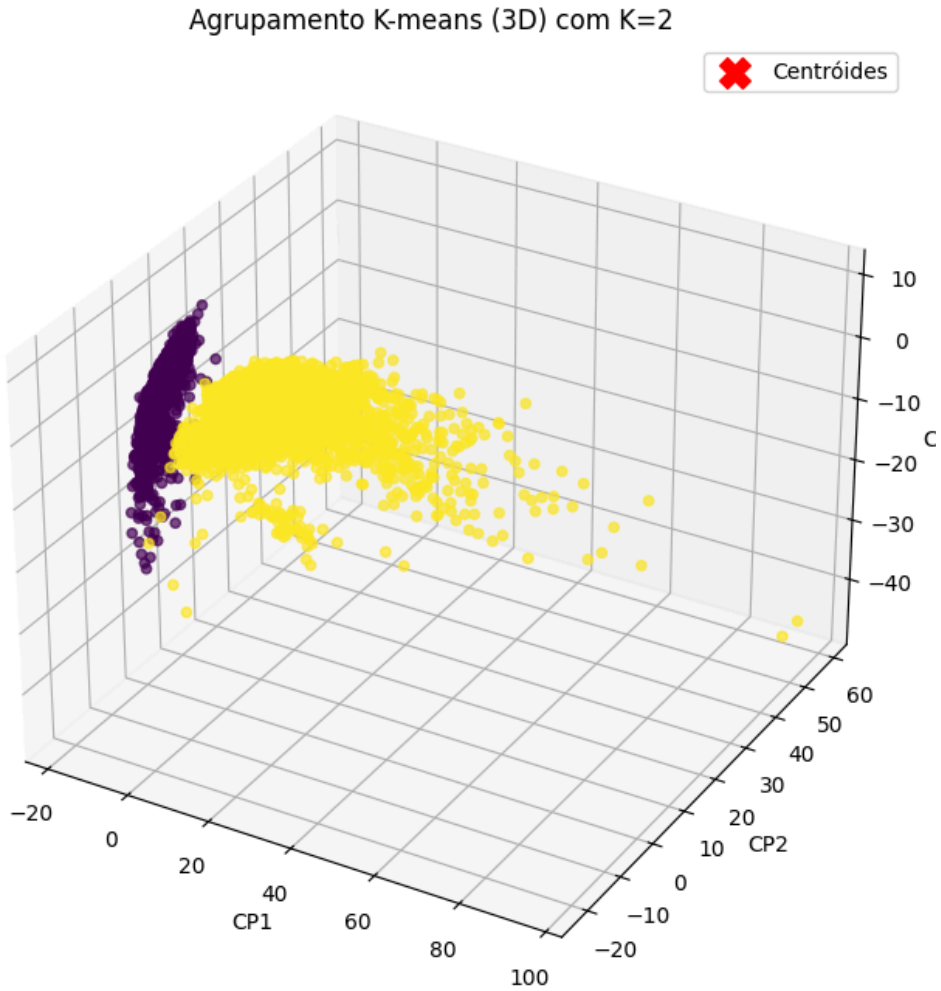
Qualidade do Agrupamento

Os resultados mostram que o modelo final alcançou um coeficiente de inércia de `2137253.221636358`, que representa a soma das distâncias das amostras em relação ao centróide mais próximo dentro dos clusters. O *Silhouette Score* obtido sugere que os clusters têm uma evidente distinção e são claramente definidos. Contudo, foram observados casos de sobreposição parcial entre clusters, sugerindo que embora o agrupamento tenha sido satisfatório, a separação entre os clusters poderia ser melhorada com mais ajustes, talvez considerando um aumento na quantidade de clusters ou explorando mais variáveis que poderiam a ser relevantes.

Visualização dos Clusters

A visualização dos clusters com base nos dois primeiros componentes principais do PCA, mostrou uma evidente separação para a maioria das atividades humanas, que simplificou a interpretação visual do modelo. Os centroides dos clusters foram destacados, oferecendo uma visão clara das posições médias de cada grupo. A consistência dos agrupamentos foi confirmada por meio de execuções repetidas do algoritmo, que demonstraram consistência nos resultados, indicando que o K-means produziu agrupamentos sólidos com os parâmetros utilizados.


Agrupamento K-means com o Número Ideal de Clusters
0
1
Centróides
CP1
CP2


Agrupamento K-means (3D) com K=2
Centróides
CP1
CP2

Discussão

O modelo K-means provou ser eficaz na tarefa de agrupar atividades humanas a partir de dados de sensores. Os clusters gerados mostraram uma boa coesão interna e separação externa, com o Silhouette Score confirmando a adequação de 2 clusters. Contudo, foi identificada uma limitação: a sobreposição de alguns clusters. Essa falha indica que o K-means, por ser um algoritmo que depende da definição de clusters esféricos e da distância euclidiana, pode não capturar completamente as diferenças mais sutis entre atividades que possuem padrões de movimento semelhantes.

Por outro lado, a utilização do PCA para a redução de dimensionalidade, apesar de ter contribuído para melhorar o desempenho computacional e facilitar a visualização dos dados, pode ter acarretado a perda de algumas características relevantes dos dados originais, impactando assim a qualidade do agrupamento. Dados de sensores, como acelerômetros e giroscópios, frequentemente envolvem interações complexas que podem não ser bem representadas nos componentes principais selecionados.

Para superar essas limitações, seria interessante explorar técnicas de agrupamento que tratam melhor a forma e a densidade dos dados, como o algoritmo DBSCAN, que é capaz de identificar clusters de formas arbitrárias e é mais tolerante a ruídos e outliers. Além disso, uma análise mais detalhada sobre a relevância das variáveis, como a utilização de técnicas de ponderação, pode melhorar a separação dos clusters.

Conclusão e Trabalhos Futuros

Este projeto demonstrou que o K-means, aliado à redução de dimensionalidade com PCA, se mostrou uma ferramenta válida para o reconhecimento de atividades humanas usando dados de sensores. A combinação das duas técnicas resultou em agrupamentos visualmente compreensíveis e eficazes sob a perspectiva computacional. No entanto, o processo de agrupamento apresentou desafios, como a sobreposição de grupos e a possível perda de detalhes importantes no processo de redução de dimensionalidade.

Trabalhos futuros poderiam se concentrar no estudo de algoritmos de agrupamento mais avançados, como DBSCAN ou algoritmos hierárquicos, que poderiam lidar melhor com a complexidade dos dados. Além disso, a implementação de técnicas supervisionadas, como redes neurais ou Support Vector Machines (SVM),

poderiam ser utilizadas para melhorar a acurácia da classificação das atividades. Outro ponto a ser explorado é o pré-processamento de dados, utilizando técnicas mais sofisticadas de seleção de características para identificar e preservar variáveis essenciais, o que maximiza a distinção entre os grupos.

Por fim, uma ampliação do conjunto de dados, que inclua sensores adicionais ou diferentes formas de atividades, pode aumentar a robustez do modelo. O uso de diferentes modalidades de sensores (como dados de GPS ou pressão atmosférica) também poderia enriquecer as informações disponíveis para o agrupamento, oferecendo uma visão mais detalhada e precisa das atividades humanas.

Referências

REYES-ORTIZ, JORGE-L. et al. Transition-Aware Human Activity Recognition Using Smartphones. Neurocomputing, v. 171, p. 754–767, jan. 2016.
UCI Machine Learning Repository. Disponível em: <https://archive.ics.uci.edu/dataset/240/human+activity+recognition+using+smartphones>.
JURCZYK, T. Algoritmos de agrupamento (clustering) utilizando scikit-learn em Python. Disponível em: <https://programminghistorian.org/pt/licoes/algoritmos-agrupamento-scikit-learn-python>. Acesso em: 30 nov. 2024.

PCA. Disponível em: <https://scikit-learn.org/dev/modules/generated/sklearn.decomposition.PCA.html>.

BABITZ, K. Introdução ao k-Means Clustering com o scikit-learn em Python. Disponível em: <https://www.datacamp.com/pt/tutorial/k-means-clustering-python> Acesso em: 1 dez. 2024.

KALOYANOVA, E. How to Combine PCA and K-means Clustering in Python? Disponível em: <https://365datascience.com/tutorials/python-tutorials/pca-k-means/>.

GOMES555. Análise Multivariada (PCA e kmeans). Disponível em: <https://www.kaggle.com/code/gomes555/an-lise-multivariada-pca-e-kmeans>. Acesso em: 29 nov. 2024.
JHA, W. Implementation of Principal Component Analysis(PCA) in K Means Clustering. Disponível em: <https://medium.com/analytics-vidhya/implementation-of-principal-component-analysis-pca-in-k-means-clustering-b4bc0aa79cb6>.