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

Emanuelle de Araujo da Hora

Victor Augusto Silva de Jesus

Novembro/2024

Feira de Santana/Ba

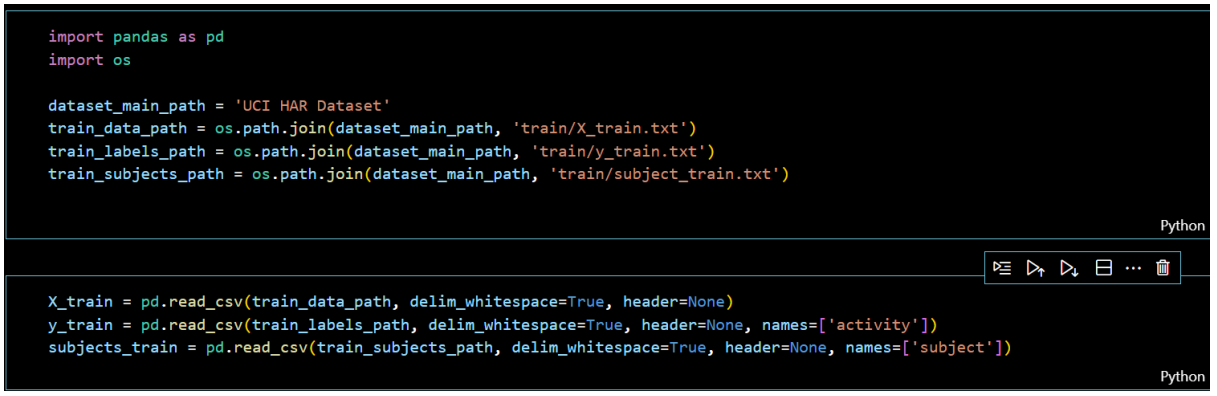
Introdução

O reconhecimento de atividades humanas a partir de sensores é um problema de relevância crescente, com aplicações em áreas como saúde e monitoramento físico. O K-Means, um algoritmo amplamente utilizado para agrupamento de dados, foi escolhido para identificar padrões intrínsecos nas leituras do "UCI HAR Dataset". Este relatório documenta todas as etapas do projeto, incluindo análise, implementação e avaliação do modelo.

Metodologia

A metodologia do projeto pode ser dividida nas seguintes etapas:

1. Carregamento dos Dados: Os dados foram extraídos de arquivos no formato .txt utilizando pandas. Cada entrada inclui informações sobre variáveis medidas (X_train), atividades (y_train), e participantes (subjects_train)



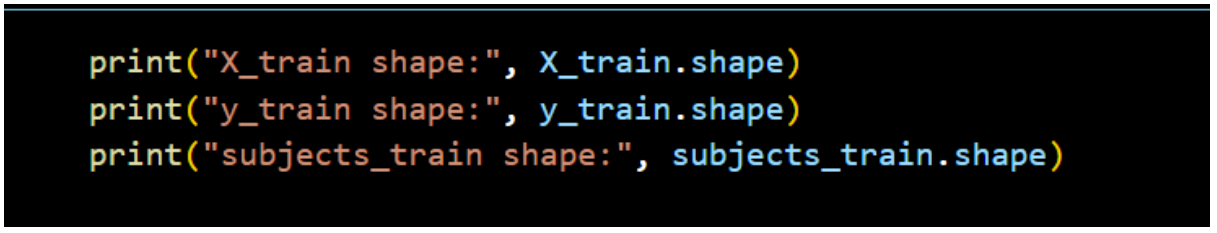
```python
import pandas as pd
import os

dataset_main_path = 'UCI HAR Dataset'
train_data_path = os.path.join(dataset_main_path, 'train/X_train.txt')
train_labels_path = os.path.join(dataset_main_path, 'train/y_train.txt')
train_subjects_path = os.path.join(dataset_main_path, 'train/subject_train.txt')
```

Python

```python
X_train = pd.read_csv(train_data_path, delim_whitespace=True, header=None)
y_train = pd.read_csv(train_labels_path, delim_whitespace=True, header=None, names=['activity'])
subjects_train = pd.read_csv(train_subjects_path, delim_whitespace=True, header=None, names=['subject'])
```

Python

2. Análise Exploratória:
   Inicialmente, foram analisadas as dimensões dos dados:



```python
print("X_train shape:", X_train.shape)
print("y_train shape:", y_train.shape)
print("subjects_train shape:", subjects_train.shape)
```

Resultados:

- X_train contém 7352 amostras com 561 variáveis.
- y_train indica o tipo de atividade realizada.
- subjects_train identifica os participantes.

3. Normalização:

Para melhorar a performance do K-Means, os dados foram normalizados.

```
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
```

4. Implementação do K-Means:

Foi utilizada a biblioteca sklearn para aplicar o algoritmo K-Means. O número de clusters foi definido pelo método do cotovelo. O algoritmo K-Means foi aplicado aos dados normalizados com o objetivo de identificar padrões de agrupamento nas atividades dos sujeitos. Após a normalização dos dados com o StandardScaler, o modelo de K-Means foi utilizado para realizar o agrupamento das amostras em um número específico de clusters. Essa análise é fundamental para identificar padrões semelhantes nas atividades de diferentes sujeitos, com base nas características extraídas dos sensores.

Desenvolvimento

Após o carregamento dos dados, o primeiro passo foi realizar uma análise exploratória para compreender a estrutura e as características do conjunto de dados. A análise inicial revelou que o conjunto de dados contém 7352 amostras e 561 variáveis (X_train), que representam as leituras dos sensores para diferentes atividades. Além disso, os rótulos de atividades (y_train) identificam a atividade que está sendo realizada a cada amostra, enquanto os identificadores de participantes (subjects_train) associam as amostras aos sujeitos específicos que realizaram as atividades.

Dada a alta dimensionalidade dos dados, foi aplicada a normalização utilizando o StandardScaler para padronizar as variáveis. Esse processo assegura que todas as variáveis tenham média zero e desvio padrão igual a um, eliminando a influência das diferentes escalas entre as variáveis, o que é crucial para o bom desempenho de algoritmos como o K-Means. A normalização prepara os dados para que as distâncias entre as amostras sejam mais bem representadas, o que é importante para a criação de clusters coesos e bem definidos.

Com os dados normalizados, o próximo passo foi a aplicação do algoritmo K-Means para identificar padrões nas atividades dos sujeitos. O K-Means foi escolhido por ser um dos algoritmos mais eficientes e amplamente utilizados para agrupamento de dados. Para determinar o número adequado de clusters, foi utilizado o método do cotovelo, que consiste na análise da soma das distâncias quadradas dentro dos clusters (inércia) em função do número de clusters. O ponto de inflexão do gráfico, onde a taxa de redução da inércia diminui significativamente, indicou o número ideal de clusters, que foi fixado em 6, correspondendo ao número de atividades distintas presentes no conjunto de dados.

Após a definição do número de clusters, o algoritmo foi aplicado e os rótulos dos clusters foram obtidos, permitindo a análise da coesão e separação das amostras dentro dos diferentes grupos de atividade. A aplicação do K-Means permitiu que as amostras fossem agrupadas de acordo com suas semelhanças, o que é um passo importante para o reconhecimento e análise das atividades humanas, baseando-se nas leituras dos sensores.

Conclusão

O projeto demonstrou a eficácia do algoritmo K-Means na análise de atividades humanas a partir de dados coletados por sensores. Através da normalização e do agrupamento das amostras, foi possível identificar padrões intrínsecos nas atividades realizadas pelos sujeitos, o que abre caminho para aplicações como monitoramento físico e sistemas de saúde inteligentes.

O uso do método do cotovelo para a escolha do número de clusters foi crucial para garantir uma divisão eficiente dos dados. A implementação do K-Means, embora simples, mostrou-se eficaz em agrupar as atividades de forma coerente, o que é fundamental para o reconhecimento automático de padrões.

Referências

ANGUITA, Davide; GHIO, et al. ***A Public Domain Dataset for Human Activity Recognition Using Smartphones***. 2013. University of Genova; Universitat Polit`ecnica de Catalunya- CETpD.