

Classificação de Atividades Humanas



Sitting

Standing

Walking

Running

Climbing Stairs

Introdução

Objetivo: Exercitar conceitos centrais de uma pipeline de análise de dados, passando pelas fases de preparação de dados, a sua limpeza, a extração de características descritivas, a sua seleção/redução e a aprendizagem computacional.

Contexto: o problema proposto no presente trabalho prático é um problema típico de classificação, nomeadamente, de reconhecimento de atividades físicas humanas. Este é um contexto com uma importância crescente em múltiplas situações, abrangendo, por exemplo, aplicações médicas, recreativas e de bem-estar. Independentemente do problema específico e das suas potenciais aplicações, o este trabalho irá permitir exercitar e interiorizar conceitos centrais em qualquer pipeline de análise dados.



Figura 1: localização dos sensores.

Iremos usar o dataset FORTH-TRACE benchmark¹. Este dataset foi adquirido usando 5 sensores (ver Figura 1), incluindo sensores de aceleração, velocidade angular e variação do campo magnético, quer da parte superior, quer da parte inferior, do corpo. É composto por dados adquiridos de 15 participantes usando um protocolo que envolvia 16 atividades distintas listadas na Tabela 1. Os dados originais podem ser descarregados usando este [link](#), contendo os seguintes ficheiros:

- partX/partXdev1.csv
- partX/partXdev2.csv
- partX/partXdev3.csv
- partX/partXdev4.csv
- partX/partXdev5.csv

em que X corresponde ao ID do participante e 1-5 ao ID do dispositivo (ver Tabela 2).

¹ Katerina Karagiannaki, Athanasia Panousopoulou, Panagiotis Tsakalides, A Benchmark Study on Feature Selection for Human Activity Recognition, UBICOMP/ISWC '16, (<https://dl.acm.org/doi/pdf/10.1145/2968219.2971421>)

Cada ficheiro CSV segue o formato seguinte:

- Coluna 1: Device ID
- Coluna 2: accelerometer x
- Coluna 3: accelerometer y
- Coluna 4: accelerometer z
- Coluna 5: gyroscope x
- Coluna 6: gyroscope y
- Coluna 7: gyroscope z
- Coluna 8: magnetometer x
- Coluna 9: magnetometer y
- Coluna 10: magnetometer z
- Coluna 11: Timestamp
- Coluna 12: Activity Label

Tabela 1: Atividades

Etiqueta	Atividade
1	Stand
2	Sit
3	Sit and Talk
4	Walk
5	Walk and Talk
6	Climb Stair (up/down)
7	Climb Stair (up/down) and talk
8	Stand-> Sit
9	Sit-> Stand
10	Stand-> Sit and talk
11	Sit->Stand and talk
12	Stand-> walk
13	Walk-> stand
14	Stand -> climb stairs (up/down), stand -> climb stairs (up/down) and talk
15	Climb stairs (up/down) -> walk
16	Climb stairs (up/down) and talk -> walk and talk

Tabela 2: Identificadores dos dispositivos

ID	Atividade
1	Pulso esquerdo
2	Pulso direito
3	Peito
4	Perna superior direita
5	Perna inferior esquerda

Elaboração de um conjunto de scripts e funções em Python, NumPy e SciPy para realizar as tarefas de preparação dos dados e engenharia de características

1. Crie um script e grave-o com o nome 'mainActivity.py'. Este script será utilizado na chamada de todas as funções indicadas abaixo.
2. Descarregue os dados através do link indicado em cima e elabore uma rotina que carregue os dados relativos a um indivíduo e os devolva num Array NumPy.

Poderá usar, por exemplo, a biblioteca CSV (<https://docs.python.org/3/library/csv.html>).

3. Análise e tratamento de *outliers*: o objetivo será identificar e tratar *outliers* usando diferentes abordagens univariável e multivariável. Para o efeito iremos utilizar os módulos dos vectores aceleração, giroscópio e magnetómetro. Seja

$$\vec{t} = (t_x, t_y, t_z)$$

o vector aceleração, giroscópio e magnetómetro. O respetivo módulo é determinado recorrendo:

$$\|\vec{t}\| = \sqrt{t_x^2 + t_y^2 + t_z^2}$$

- 3.1. Elabore uma rotina que apresente simultaneamente o *boxplot* de cada atividade (coluna 12 – eixo horizontal) relativo a todos os sujeitos e a uma das seguintes variáveis transformadas: módulos dos vectores de aceleração, giroscópio e magnetómetro. Sugere-se o uso da biblioteca *matplotlib*, por exemplo, a classe *matplotlib.pyplot.boxplot*.
- 3.2. Analise e comente a densidade de *outliers* existentes no *dataset* transformado, isto é, nos módulos dos vetores aceleração, giroscópio e magnetómetro para cada atividade (use somente os sensores do pulso direito). Observe que a densidade é determinada recorrendo

$$d = \frac{n_o}{n_r} \times 100$$

em que n_o é o número de pontos classificados como *outliers* e n_r é o número total de pontos.

- 3.3. Escreva uma rotina que receba um *Array* de amostras de uma variável e identifique os *outliers* usando o teste Z-Score para um k variável (parâmetro de entrada).
- 3.4. Usando o Z-score implementado, assinale todos as amostras consideradas *outliers* nos módulos dos vetores de aceleração, giroscópio e magnetômetro. Apresente *plots* em que estes pontos surgem a vermelho, enquanto os restantes surgem a azul. Use $k=3, 3.5$ e 4 .
- 3.5. Compare e discuta os resultados obtidos em 3.1 e 3.4.
- 3.6. Elabore uma rotina que implemente o algoritmo k-means para n (valor de entrada) clusters.
- 3.7. Determine os *outliers* no *dataset* transformado usando o k-means. Experimente diferentes números de *clusters* e compare com os resultados obtidos em 3.4. Ilustre graficamente os resultados usando gráficos 3D para cada vetor (veja, por exemplo, <https://medium.com/data-science/an-easy-introduction-to-3d-plotting-with-matplotlib-801561999725>).

3.7.1. Bónus: poderá realizar um estudo análogo usando o algoritmo DBSCAN (sugere-se que recorra à biblioteca *sklearn*²)

4. Extração de informação característica: o objetivo será comprimir o espaço do problema, extraíndo informação característica discriminante que permita implementar soluções eficazes do problema de classificação.
 - 4.1. Usando as variáveis aplicadas na alínea 3.1, determine a significância estatística dos seus valores médios nas diferentes atividades. Observe que poderá aferir a normalidade da distribuição usando, por exemplo, o teste Kolmogorov-Smirnov (ver documentação do SciPy). Para rever a escolha de testes estatísticos sugere-se a referência³. Comente.
 - 4.2. Desenvolva as rotinas necessárias à extração do feature set temporal e espectral sugerido no artigo⁴. Para o efeito deverá:

² <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html#sklearn.cluster.DBSCAN>

³ Jean-Baptist du Prel, Bernd Röhrig, Gerhard Hommel, Maria Blettner, Choosing Statistical Tests, Deutsches Arzteblatt, v107(19), 2010. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2881615/>

⁴ Mi Zhang, Alexander A Sawchuk, A. Sawchuk, A feature selection-based framework for human activity recognition using wearable multimodal sensors, : [BodyNets '11: Proceedings of the 6th International Conference on](#)

- Ler o artigo e identificar o conjunto de features temporais e espectrais identificadas por estes autores
- Para cada feature deverá elaborar uma rotina para a respetiva extração
- Usando as rotinas elaboradas no item anterior, deverá escrever o código necessário para extrair o vetor de features em cada instante.

Nota: Poderá usar as bibliotecas NumPy e SciPy. Qualquer outra biblioteca deverá ser identificada.

- 4.3. Desenvolva o código necessário para implementar o PCA de um feature set; poderá usar implementações existentes.
- 4.4. Determine a importância de cada vetor principal na explicação da variabilidade do espaço de features. Note que deverá normalizar as features usando o z-score. Quantas dimensões deverá usar para explicar 75% do feature set?
 - 4.4.1. Indique como poderia obter as features relativas a esta compressão e exemplifique para um instante à sua escolha.
 - 4.4.2. Indique as vantagens e as limitações desta abordagem.
- 4.5. Desenvolva o código necessário para implementar o Fisher feature Score e o ReliefF. Poderá usar implementações existentes.
- 4.6. Identifique as 10 melhores features de acordo com o Fisher Score e o ReliefF e compare os resultados.
 - 4.6.1. Indique como poderia obter as features relativas a esta compressão e exemplifique para um instante à sua escolha.
 - 4.6.2. Indique as vantagens e as limitações desta abordagem.