Uma imagem com texto

Descrição gerada automaticamente

Unsupervised Learning – a3rl

André Filipe Frade Guerra

Novembro/2022

Conteúdo

[Introdução 3](#_Toc119160512)

[Classes e métodos 3](#_Toc119160513)

[Exercício 1 4](#_Toc119160514)

[Exercício 2 6](#_Toc119160515)

[Exercício 3 9](#_Toc119160516)

[Exercício 4 10](#_Toc119160517)

## Introdução

Neste relatório irá ser explicado resumidamente as classes criadas, funcionalidade dos métodos criados e por fim a execução e discussão dos resultados dos exercícios.

Este trabalho tem como objetivo implementar vários algoritmos que tratam informação num conjunto de pontos por clusters.

## Classes e métodos

Para desenvolver os exercícios da terceira prática da disciplina de Introdução à Aprendizagem Automática foi desenvolvido um código em python usando o vsCode para realizar a mesma.



Presente em todas as classes é criada a população de pontos que vai ser trabalhada:

Uma imagem com texto

Descrição gerada automaticamente

## Exercício 1

Presente no ficheiro exer1.py, este ficheiro começa por criar os pontos anteriormente descrito, e escolhe 2 pontos aleatórios:

Uma imagem com texto

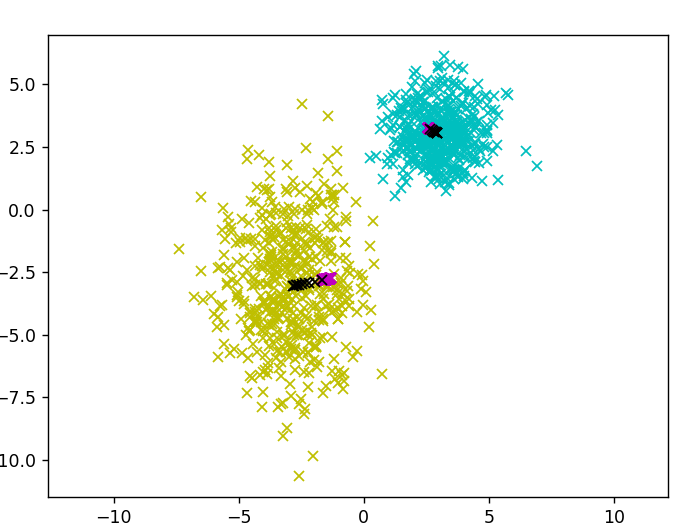
Descrição gerada automaticamente

De seguida, encontra-se o algoritmo implementado, que vai fazer andar os dois pontos escolhidos para dentro do kluster mais próximo:

Uma imagem com texto

Descrição gerada automaticamente

No fim mostra o resultado da localização dos pontos e o percurso que fizeram:



Neste gráfico a roxo, podemos ver a deslocação na primeira interação do ciclo, e a preto as restantes deslocações. Por fim podemos ver que os pontos se encontram mais perto do “meio” dos clusters.

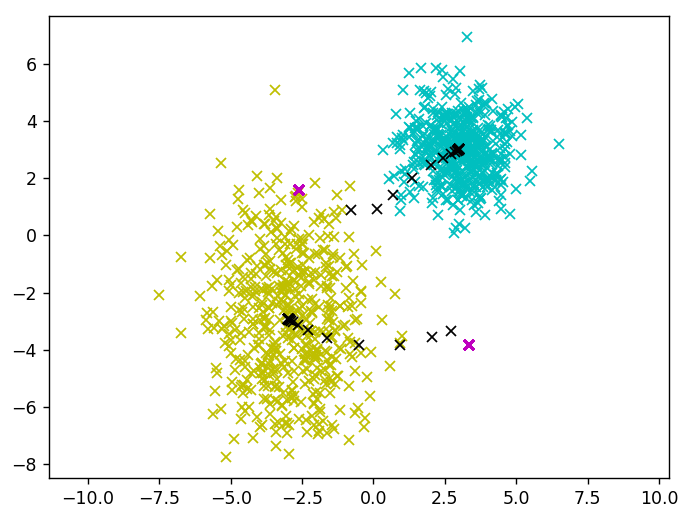
## Exercício 2

Presente no ficheiro exer2.py, este ficheiro começa por declarar vários valores iniciais que definem o algoritmo. Com a diferença do primeiro algoritmo, este apenas atualiza a posição do ponto após percorrer todos os pontos:

Uma imagem com texto

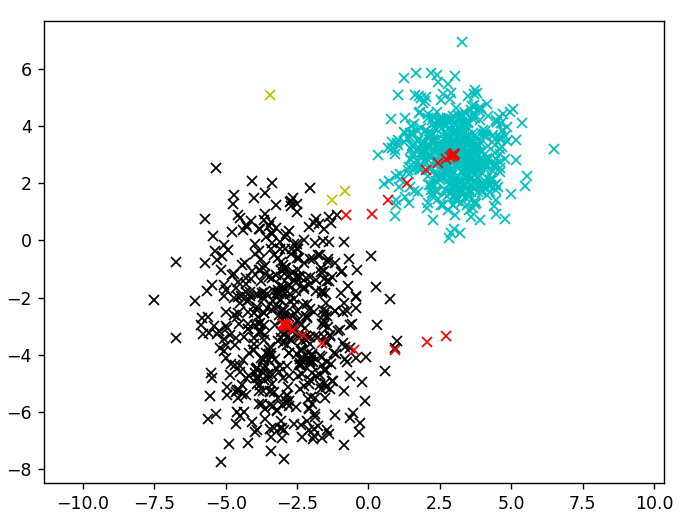
Descrição gerada automaticamente

No fim mostra o resultado da localização dos pontos e o percurso que fez:



Após a conclusão desta parte do código, o restante, verifica se existem pontos do kluster que se encontram mais próximos do outro kluster (representados a amarelo): Uma imagem com texto

Descrição gerada automaticamente



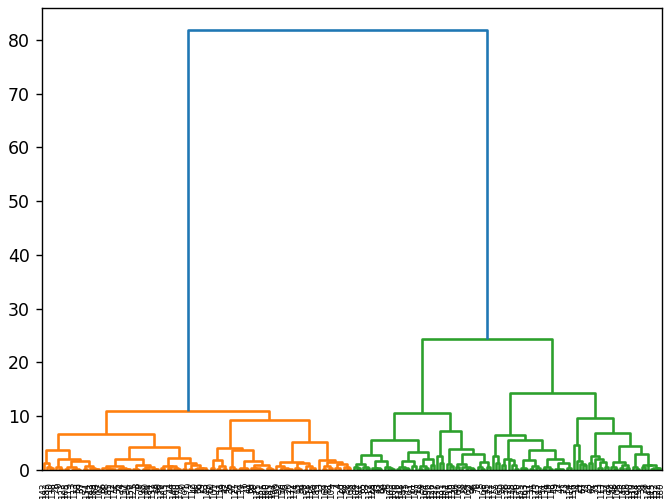
## Exercício 3

Presente no ficheiro exer3.py, este ficheiro começa por declarar vários valores iniciais que definem o algoritmo.

Inicialmente foi implementado um algoritmo que mostra o gráfico em árvore:

Uma imagem com texto

Descrição gerada automaticamente



E de seguida a implementação do algoritmo, sem usar bibliotecas externas, sem mostrar visualmente a árvore:

Uma imagem com texto

Descrição gerada automaticamente

## Exercício 4

Presente no ficheiro exer4.py, este ficheiro começa por declarar vários valores iniciais que definem o algoritmo.

Inicialmente temos uma implementação do algoritmo que guarda nas variáveis “klusterFinalX” e “klusterFinalY”, os pontos de cada kluster encontrado:

Uma imagem com texto

Descrição gerada automaticamente

E por fim uma implementação que usa bibliotecas externas para realizar o algoritmo, que mostra visualmente o resultado de quantos clusters foram encontrados:

Uma imagem com texto

Descrição gerada automaticamente

