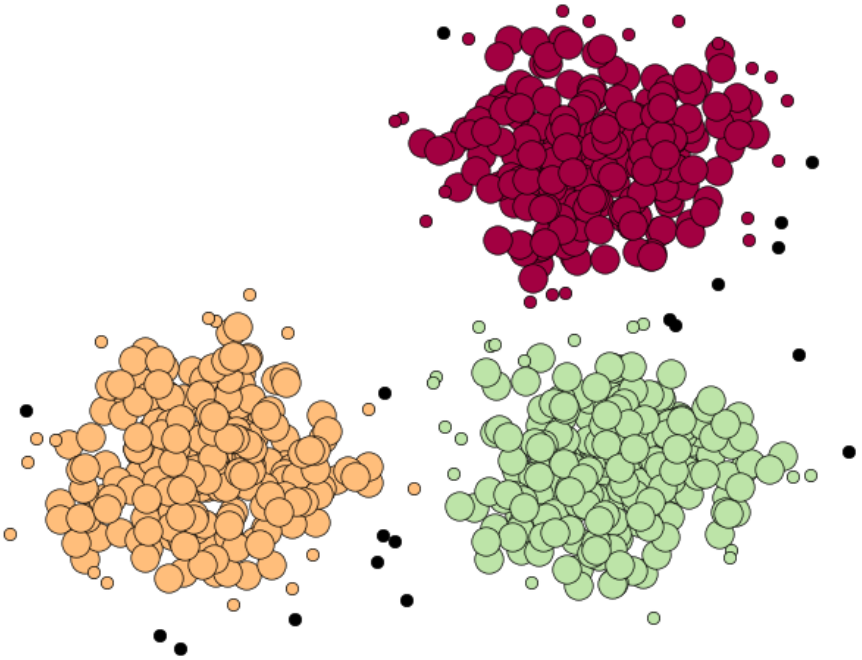
Universidade do Minho

3 de Abril de 2020

O Método *Elbow*

Clusters particionais com dados numéricos



Bruno Jácome, A89515
Carolina Barros, A84950
Dinis Gomes, A87993
Joana Gouveia, A85650
João Silva, A84617
Jorge Gonçalves, A84133
Pedro Peixoto, A89602

Índice

Lista de Ilustrações

Tabelas

1 | Introdução

Este trabalho foi realizado no âmbito da Unidade Curricular de Matemática das Coisas e tem como objetivo primordial o estudo do *Clusters* particionais com dados numéricos (centróide) atráves do *The Elbow Method*.

O presente relatório divide-se essencialmente em 4 partes. Primeiramente, no Capítulo 2, será feita uma contextualização do assunto, apresentan-se a definição de clusters no geral e, mais em concreto, de clusters particionais.

Seguidamente, no Capítulo 3, será descrito o conceito de centróides, bem como outros aspetos relevantes relativos.

Depois, no Capítulo 4, será abordado o *The Elbow Method*, com a apresentação da definição teórica e a sua aplicação mais prática.

No capítulo seguinte, a parte teórica será aplicada em exemplos mais práticos, de forma a melhor entendermos a aplicação dos tópicos referidos nos capítulos anteriores.

Para finalizar, expor-se-á uma breve conclusão do trabalho apontando-se os aspetos mais enriquecedores para o nosso conhecimento.

2 | Clusters

2.1 O que são?

Um **cluster** é um conjunto de objetos similares entre si e dissimilares em relação a objetos noutros clusters. A análise de clusters ou o seu conceito, é um procedimento humano normal, muitas vezes usado de forma inconsciente. [6][7]

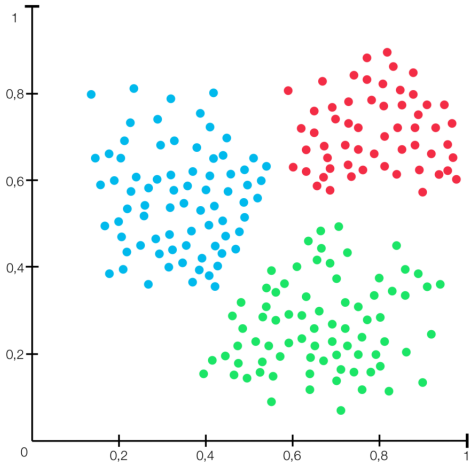


Figura 2.1: Figura ilustrativa do agrupamento de clusters.

Muito cedo nas escolas, os alunos aprendem a classificar e agrupar, por exemplo distinguir entre gatos e cães, entre animais e planta, progredindo num refinamento de classificação que tem subjacente teorias de *clustering*. A análise de clusters é usada em inúmeras aplicações, tais como no reconhecimento de padrões (*machine learning*), processamento de imagem e pesquisa de mercado.

2.2 Clustering

O *clustering* é o conjunto de técnicas de prospeção de dados, isto é, exames minuciosos e metódicos, que fazem agrupamentos automáticos de dados segundo o seu grau de semelhança. Normalmente o usuário do sistema deve escolher a priori o número de grupos a serem detetados. Alguns algoritmos mais sofisticados pedem apenas o nú-

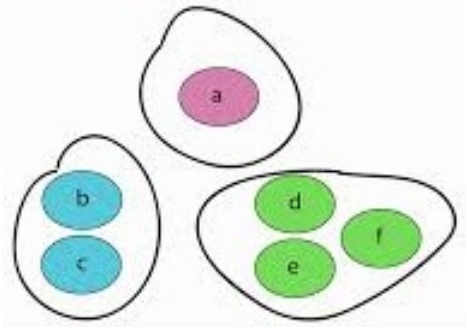


Figura 2.2: Clustering.

mero mínimo e outros tem a capacidade de subdividir um grupo em dois. Existem vários tipos de agrupamentos, mas o que será analisado com mais detalhe serão os **particionais**.

2.2.1 Clustering na História

O primeiro registo publicado sobre um método de clustering foi feito em 1948, com o trabalho de *SORENSEN* (1948) sobre o Método Hierárquico de Ligação Completa. Desde então mais de uma centena de algoritmos distintos de clustering já foram definidos.

2.3 Clusters Particionais

2.3.1 O que são?

Cluster particional define-se, especificamente, pelo facto de ao utilizar o agrupamento particional estar a dividir objetos de dados em subconjuntos sem sobrepor grupos, o que leva a que cada dado esteja exatamente num subconjunto.

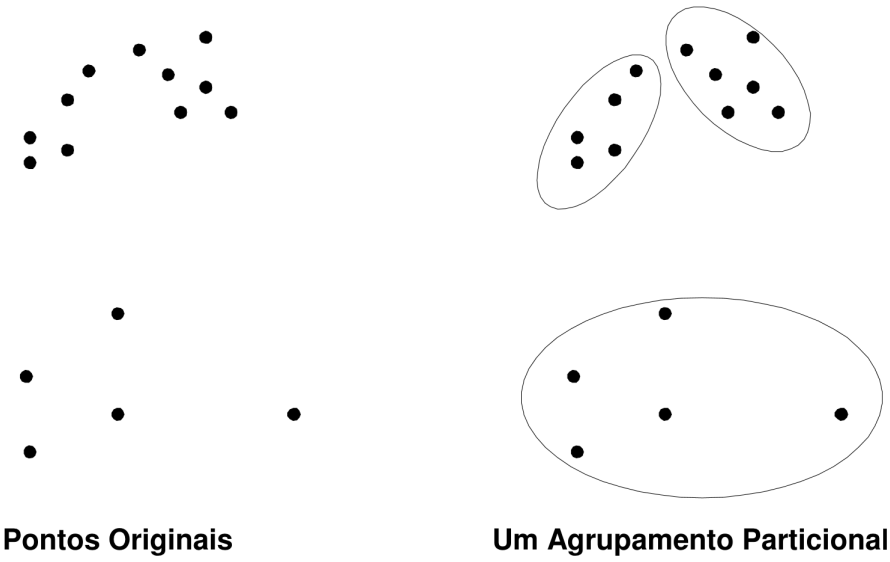


Figura 2.3: Cluster Particional

3 | Centróides

3.1 O que são?

Um **centróide** é o ponto que representa o *centro* de todos os pontos pertencentes a um cluster. No que diz respeito aos modelos centróides, a noção de similaridade deriva da proximidade dos pontos com o centróide do *cluster*.

Além disso, os centróides são obtidos através de operações algébricas (somas e multiplicações por escalares) e, em regra, estes não pertencem à base de dados. Logo, são uma mera interpretação de resultados e que dependem maioritariamente da definição de proximidade entre dois objetos de estudo.

3.2 Relação entre centróide e média

3.2.1 Semelhanças

... a *média* de um cluster é o mesmo que o centróide, contudo o termo **centróide** é mais preciso quando se estuda *multivariate dada*, isto é, dados multivariados.

Um centróide é ás vezes denominado de **centro de massa** ou **barycenter**(centro de gravidade), baseado na sua interpretação física. Assim como a média, a localização do centróide **minimiza a *sum-squared distance* entre os outros pontos**.

3.2.2 Medóide

Uma ideia semelhante é a de **medóide**, que é o ponto de dado que é *menos parecido* de todos os outros pontos de dados.

Ao contrário do centróide, a medóide tem de ser um dos pontos originais.

3.2.3 Diferenças

Há, no entanto, uma diferença entre **distância de centróide** e **distância média** quando se comparam clusters. A distância de centróide entre dois quaisquer clusters A e B é simplesmente a distância entre o centróide de A e o centróide de B. Já a distância média é calculada encontrando-se a distância média entre todos os pares de pontos de cada cluster.

$$\text{dist}(A, B) = \frac{\sum_{ij} \text{dist}(a_i, b_j)}{\#A \times \#B}, \quad \forall a_i \in A, b_j \in B$$

Métrica de Clusters: Distância média

$$\text{dist}(A, B) = \text{dist}\left(\frac{\sum_i a_i}{\#A}, \frac{\sum_i b_i}{\#B}\right), \quad \forall a_i \in A, b_i \in B$$

Métrica de Clusters: Distância entre centróides

Estes dois cálculos são duas métricas possíveis para calcular a distância entre dois clusters, mas existem mais métodos.[8]

3.2.4 Exemplo

3.3 Como se determinam?

4 | Os Algoritmos de Clusters Particionais

O Cluster Particional tem dois algoritmos: o K-means e o K- medoids. Estes algoritmos tem as suas diferenças. Uma delas é o facto de K-means temos a soma máxima das distâncias, em K-medoids temos a soma mínima nas distancias.

4.1 Representação dos Dados

- K representa o número e clusters
- $(m^1, ..., m^k, ..., m^K)$ K pontos distintos de D
- $(x^1, ..., x^n, ..., x^N)$ representa a base de dados
- d representa uma função distância

4.2 K-Means

4.2.1 O que é?

O *K-Means* é um algoritmo de *clustering* bastante comum e popular usado por numerosos investigadores em todo o mundo. Este tem por objetivo por em partes n observações dentro de k clusters, onde cada observação está dentro do cluster com que tem a média mais próxima, usando o Diagrama de Voronoi. Nestes modelos, os números de clusters necessários no final (n) têm de ser mencionados com antecedência, o que torna importante o conhecimento prévio do conjunto de dados.

Na maioria das vezes, o método *Elbow* é usado ou com a soma de erros quadrados (sse) ou com a soma dos erros do cluster (wcss) (EXPLICAR O QUE CADA UM É!).

4.2.2 Diagrama de Voronoi

O diagrama de Voronoi relaciona-se com o algoritmo K-means pelo facto de que o diagrama é uma parte do conjunto de dados com alguns pontos centrais, que se denominam de centroides. Estes centroides não pertencem a base de dados, e um centroide é a localização (pode ser real ou imaginária) do centro de um cluster.

4.2.3 Restrições

Uma restrição que este algoritmo tem é o facto de apenas funcionar com atributos quantitativos, necessita de fazer operações algébricas, como somas e multiplicações por escalar, que dará origem uma matriz que é a "matriz da partição". A nível de pontos que se encontram fora da curva, tem que se ter cuidado devido ao facto de os mesmos poderem facilmente influenciar o valor da média e levar a mesma a alterar-se.

4.2.4 Determinação do K-Means

1º Passo: Temos de determinar os clusters que estão associados a M, e para isso temos de ter cuidado com os algoritmos fora do grupo, para não se influenciar a mesma e para isso podemos utilizar um número mediano.

$$M = (m^1, ..., m^k, ..., m^K) \rightarrow P = (P^1, ..., P^k, ..., P^K)$$
$$P^k = x^i \in D : d(x^i, m^k) < d(x^i, m^j), j \in (1, ..., k-1, k+1, ..., K)$$

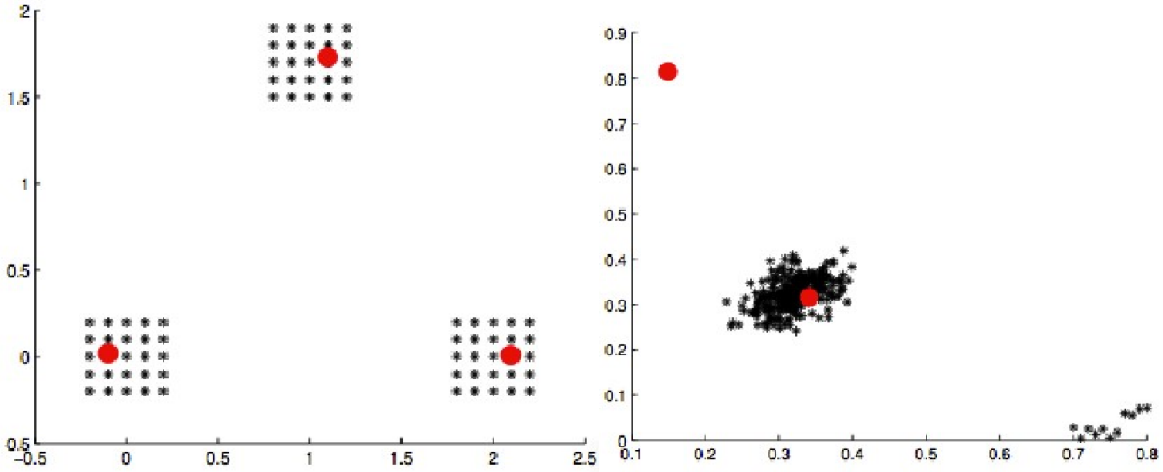
2º Passo: Temos de determinar os novos centroides associados a P, pegando no primeiro conjunto de dados e fazer uma seleção aleatória de k pontos de dados para se verificar onde se encontra o centro.

$$P = (p^1, ..., p^k, ..., p^K) \rightarrow M = (m^1, ..., m^k, ..., m^K)$$

3º Passo: Repetir os dois passos anteriores até se verificar que nenhum cluster muda de grupo.

$$m^k = \frac{1}{|P^k|} \sum_{x \in P^k} x$$

Fórmula do K-Means



(a) Gráfico bem resolvido por K-Means

(b) Gráfico mal resolvido por K-Means

4.3 K-Medoids

4.3.1 O que é?

idk

4.4 Diferença entre K-Means e K-Medoids

4.4.1 A nível de técnica

O *K-medoid* é uma técnica de cluster particional que trata de juntar dados dentro de um conjunto de n objetos em k agrupamentos, tendo em conta que k é conhecido a priori, o que exige que o programador deve especificar k antes de executar o algoritmo. Para a escolha de K temos dois métodos que podemos utilizar: **The Elbow Method** e o **The silhouette Method**. Já no *k-means* o k é escolhido previamente.

4.4.2 A nível de sensibilidade

O *K-medoid* lida melhor com os *outliers* (pontos fora da curva) do que *K-means*, é menos sensível a eles, porque minimiza a soma das diferenças contrariamente a *k-means*, que maximiza.

4.4.3 A nível de Centróide

O centro de *k-medoids* não é o ponto médio mas sim um ponto real, porque é o objeto mais centralmente localizado do cluster, que como já referi, tem somas mínimas de distancia.

4.4.4 A nível atributos

Os atributos de *K-medoids* podem ser atributos quantitativos, tal como *k-mean*, mas também podem ser atributos qualitativos, o que leva a que não exista uma necessidade e obrigação do uso de operações algébricas neste algoritmo. Estes atributos encontram-se representados na base de dados.

5 | O Método Elbow

5.1 O que é?

Uma etapa fundamental para qualquer aprendizagem não-supervisionada é determinar o número ideal de clusters segundo os quais os dados podem ser agrupados: **K**.

O **The Elbow Method** é uma heurística, uma vez que, é um método criado para encontrar soluções sobre um problema, neste caso, para determinar o número ideal de clusters no *k-means clustering*. Este método parcela o valor da função custo produzida pelos diferentes valores de **K**. Ora, isto só é possível, ignorando parte da informação com o objetivo de tornar a escolha mais fácil e rápida.

Sendo assim, não há uma resposta universal para este problema já que o número ideal de *clusters* é de alguma forma subjetivo e depende do método usado para medir as similaridades e os parâmetros usados para particionar. Portanto, em algumas situações, pode ser considerado ambíguo e pouco confiável. Nesse caso, é preferível utilizar-se outras abordagens para determinar o número de clusters.

5.2 Como se aplica?

5.2.1 Pré-aplicação

Numa fase inicial, criar um dendrograma, ou seja, um diagrama que organize as variáveis, agrupando-as de forma hierárquica ascendente - o que em termos gráficos se assemelha aos ramos de uma árvore.

Seguidamente, inspecionar o dendrograma produzido usando o cluster hierárquico para verificar se ele sugere um número específico de clusters. (Todavia, esta abordagem também é subjetiva.)

Estes métodos, apresentados a seguir, incluem métodos diretos e teste estatístico:

- **Métodos diretos**: consistem em otimizar um critério, como a somas de erros

quadrados dentro do *cluster* ou a média silhouette. Os métodos correspondentes são denominados métodos de *Elbow* e silhouette, respetivamente.

- **Métodos de teste estatístico**: consiste em comparar evidências contra hipóteses nulas. Um exemplo é a estatística de gap.

É importante referir ainda que, a ideia básica por detrás dos métodos de particionamento, como o *k-means clustering*, é definir clusters de forma que a variação total intra-cluster, ou a soma total quadrada dentro do cluster (WSS), seja minimizada.

5.2.2 WCSS

O código abaixo é uma maneira fácil de obter o valor wcss para diferentes números de clusters.

```
from sklearn.cluster import KMeans
        for i in range(1, 11):
                kmeans = KMeans(n_clusters = i, init = 'k-means++',
                                                    random_state = 42)
                kmeans.fit(X)
                wcss.append(kmeans.inertia_)
```

Figura 5.1: Código para obtenção de valor wcss

Assim como o nome sugere, **wcss** é o somatório da distância de cada cluster entre esses clusters específicos e cada um dos pontos contra o centróide do cluster.

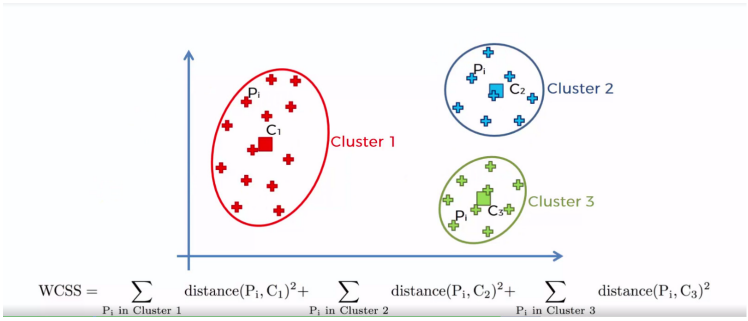


Figura 5.2: Cálculo do valor wcss para três conjuntos de dados de *cluster*

5.2.3 Processos

O método de Elbow considera o WSS total como uma função do número de clusters: deve-se escolher um número de clusters para que a adição de outro cluster não melhore muito mais o WSS total.

O número ótimo de clusters pode ser obtido da seguinte forma:

1. Calcular o algoritmo de *clustering*, por exemplo, *k-means clustering*, para diferentes valores de *k*. Por exemplo, variando *k* de 1 a 10 clusters;
2. Para cada *k*, calcular a soma total quadrada (WSS) dentro do *clusters*;
3. Fazer o gráfico (curva) de wws de acordo com o número de *clusters k*;
4. A localização de uma curva, curva joelho, provavelmente, é uma curva com uma dobra acentuada, é geralmente considerada um indicador do número apropriado de *clusters*.

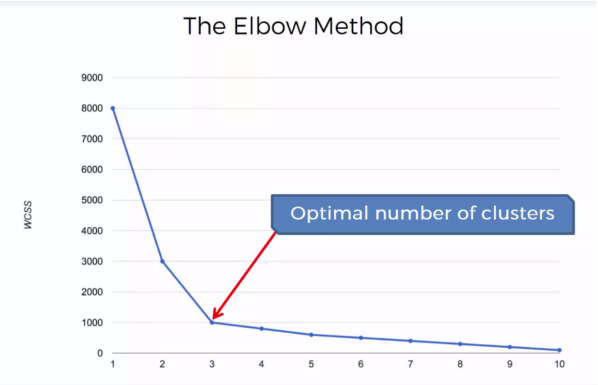


Figura 5.3: Valor de wcss *versus* número de *clusters*

Uma alternativa é o método de *silhouette* média (*Kaufman e Rousseeuw* [1990]), que também pode ser usado com qualquer abordagem de *clustering*.

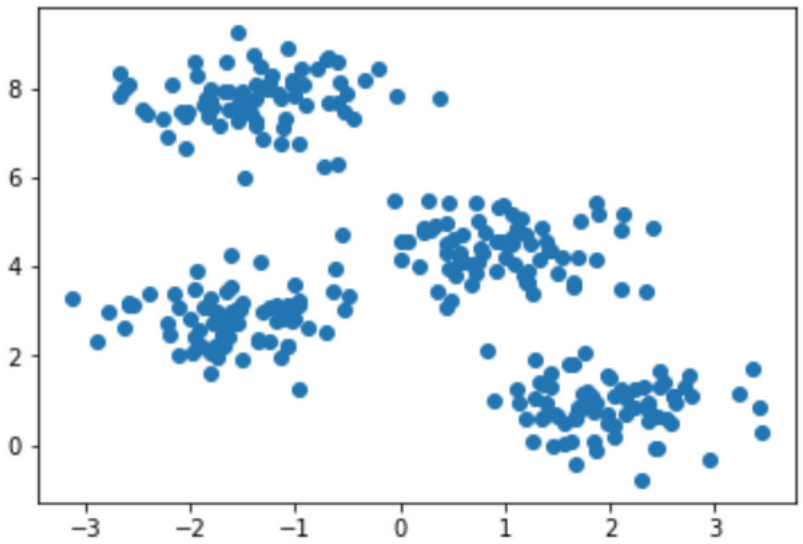
6 | Aplicações práticas

6.1 Exemplo 1

6.1.1 Como aplicar Clustering:

Para aplicar o algoritmo, precisamos de primeiro criar alguns conjuntos aleatórios de pontos e distribui-los com algum espaçamento.

```
points = make_blobs(n_samples=300, centers=4, cluster_std=0.60, random_s
points.scatter(distance=1.5);
```



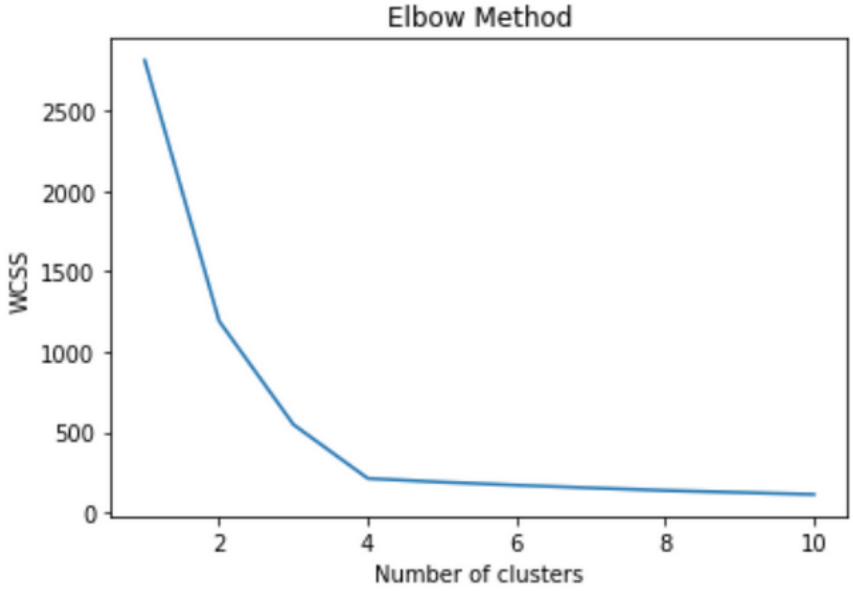
De seguida vamos aplicar kmeans aos nossos pontos. Vamos aplicar a função várias vezes, para numeros de clusters desde 1 até 9 e vamos guardar o valor de WCSS de cada resultado.

```
int wcss[10];

for(int i=1; i<10; i+=1) {
        kmeans = points.KMeans(n_clusters=i, init="k-means++", max_iter=3
```

```
            wcss[i] = kmeans.getWCSS();
}
```

Com os falores de WCSS obtidos podemos gerar um gráfico que os relaciona com o respetivo numero de clusters.



7 | Conlusões

Bibliografia

[1] *What is "Within cluster sum of squares by cluster" in K-means*

https://discuss.analyticsvidhya.com/t/what-is-within-cluster-sum-of-squares-by-cluster-in-k-means/2706

[2] *Elbow Method*,

https://www.scikit-yb.org/en/latest/api/cluster/elbow.html

[3] *Determining the optimal number of clusters*,

https://www.datanovia.com/en/lessons/determining-the-optimal-number-of-clusters-3-must-know-methods/#elbow-method

[4] *Finding the optimal number of clusters for K-Means through Elbow method using a mathematical approach compared to graphical approach*,

https://www.linkedin.com/pulse/finding-optimal-number-clusters-k-means-through-elbow-asanka-perera

[5] Lachi, Ricardo Luís & Rocha, Heloísa Vieira da. Fevereiro 2005. *Aspectos básicos de clustering: conceitos e técnicas* . (Brasil).

[6] https://www.maxwell.vrac.puc-rio.br/24787/24787_5.PDF

[7] http://www.dei.isep.ipp.pt/~paf/proj/Julho2003/Clustering.pdf

[8] *Hierarchical Clustering 3: single-link vs. complete-link* https://www.youtube.com/watch?v=VMyXc3SiEqs