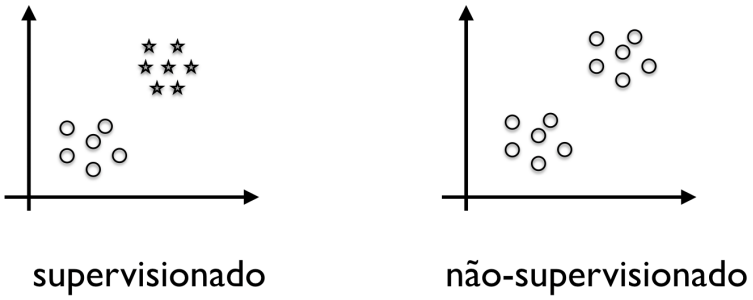
Universidade Federal de Viçosa
Centro de Ciências Exatas e Tecnológicas
Departamento de Informática

Avaliação de Models de Aprendizagem
Levi H. S. Lelis
levi.lelis@ufv.br

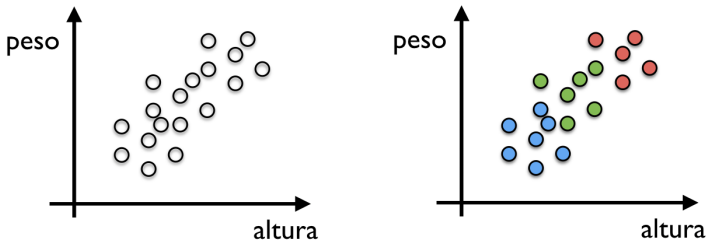
Algoritmos de Agrupamento

Em aprendizagem supervisionada os algoritmos recebem como entrada um conjunto de dados rotulados. E a tarefa é induzir um modelo para realizar uma tarefa de **classificação** ou **regressão**. Nessa aula veremos uma tarefa diferente: os dados fornecidos como entrada não possuem rótulo e o objetivo é tentar extrair a "estrutura" dos dados. Essa tarefa é conhecida como **agrupamento**. Problemas de agrupamento são conhecidos como problemas **não supervisionados**, por não existirem rótulos para os dados.

A figura abaixo contrasta um problema de classificação e um problema de agrupamento.



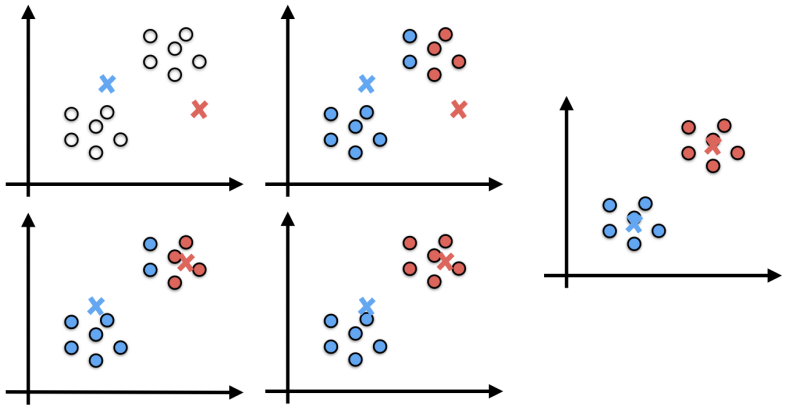
Exemplo Considere o problema enfrentado por uma fábrica de camisetas. Dado a distribuição de altura e peso das pessoas, a fábrica gostaria de descobrir 3 tamanhos de camisetas a serem fabricadas.



Outros exemplos: agrupar clientes de acordo com preferências, análise de redes sociais, descoberta de expressões de genes, dentre outros.

k-Médias

O algoritmo k-Médias recebe como entrada um um número k de grupos a serem encontrados e seleciona aleatoriamente k centróides (pontos no espaço de características). O algoritmo iterativamente assinala cada instância ao centróide mais próximo. Uma vez que todos as instâncias possuem um grupo assinalado, novos centróides são calculados através através da média das características das instâncias em cada grupo. O algoritmo converge quando nenhuma modificação é feita com relação aos grupos das instâncias.



```
1 def k-Médias(k, {x_1, x_2, ··· , x_m}):
2   inicialize k centróides C_1, C_2, ··· , C_k de forma arbitrária
3   grupo = {0, 0, ··· , 0} #vetor de m posições
4   convergiu = False
5   while not convergiu:
6     convergiu = True
7     for i in range(1, m+1):
8       antigo = grupo[i]
9       grupo[i] = arg min_j dist(C_i, x_i)
10      if antigo ≠ grupo[i]:
11        convergiu = False
12    for i in range(1, k+1):
13      C_i = média de todos os x em i
```

Função Objetivo

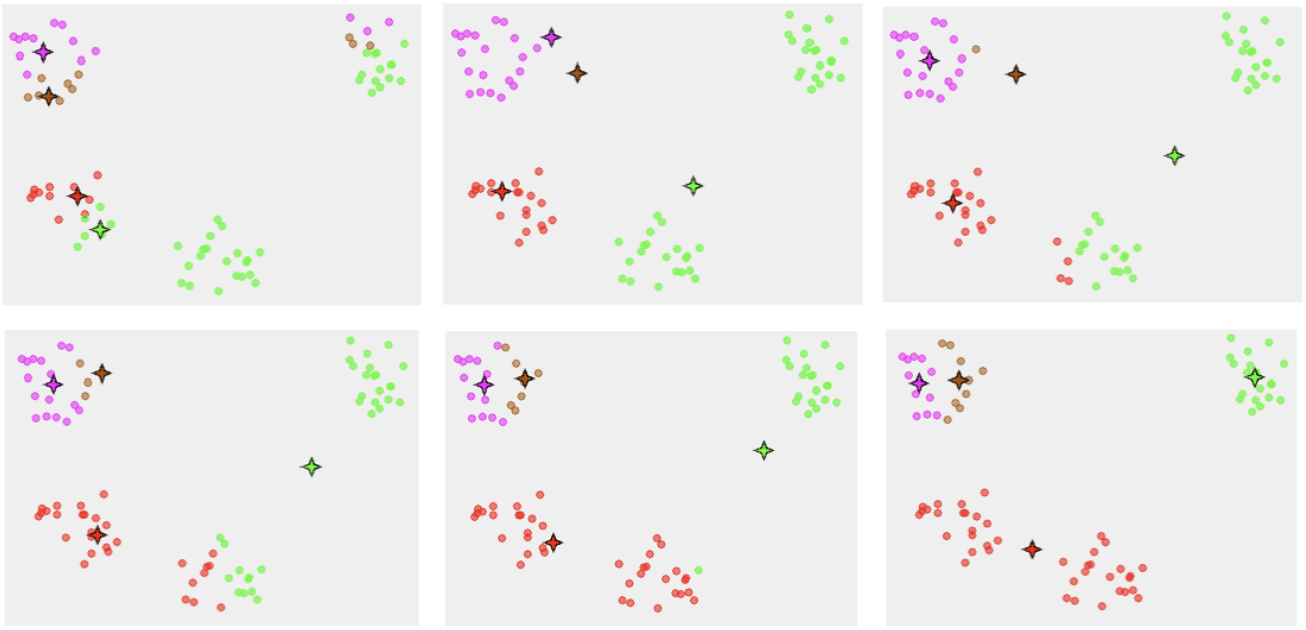
O k-Médias pode ser visto como um processo de otimização que tenta minimizar a soma das distâncias das instâncias em um grupo i e C_i. Para um problema com n características, nós temos a seguinte função objetivo.

$$J(C_1, C_2, \cdots, C_k) = \frac{1}{m} \sum_{i=1}^{m} ||x_i - M(x_i)||^2 ,$$

onde $M(x_i)$ é o centróide do grupo a qual x_i pertence e $||x_i - M(x_i)||^2$ é a distância Euclidiana entre x_i e $M(x_i)$.

Mínimos Locais

O processo de otimização não é convexo e o algoritmo k-médias pode ficar preso em mínimos locais, como demonstrado nas figuras abaixo.[1]



Para evitar os mínimos locais podemos utilizar o seguinte procedimento.

```
1 def k-Médias(T):
2   melhor_J = ∞
3   melhor_grupos = {}
4   for i in range(0, T):
5     inicialize os centróides aleatoriamente
6     grupos, J = execute k-médias com os centróides
7     if J < melhor_J:
8       melhor_J = J
9       melhor_grupos = grupos
10  return grupos
```

Inicialização dos Centróides

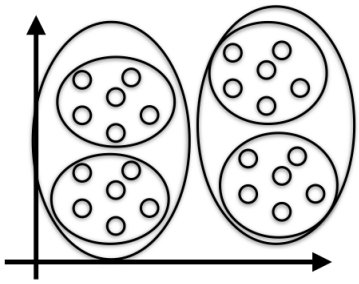
Uma heurística de inicialização aleatória de centróides que pode dar bons resultados é a de inicializar os centróides com valores de instâncias do conjunto fornecido como entrada. Assim certificamos que os centróides estão em regiões do espaço de características que possuem instâncias a serem agrupadas.

[1] http://en.wikipedia.org/wiki/K-means_clustering

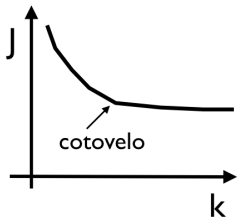
Número de Grupos

Como escolher o número de grupos (valor de k)? Muitas vezes já sabemos a priori o número de grupos a serem encontrados, como o problema das camisetas. E se não soubermos a priori, como escolher o valor de k?

De uma forma geral não existem formas corretas ou incorretas de escolher o valor de k. Por exemplo, no conjunto de dados abaixo, quantos grupos escolher?



Uma heurística bastante utilizada é a do "cotovelo". Na heurística do cotovelo variamos os valores de k e observamos os valores resultantes de J, como na figura abaixo.



Em um determinado ponto a redução no valor de J satura com o aumento do valor de k, esse é o ponto do "cotovelo". Esse ponto sugere um número de grupos que consegue extrair melhor a estrutura do problema.

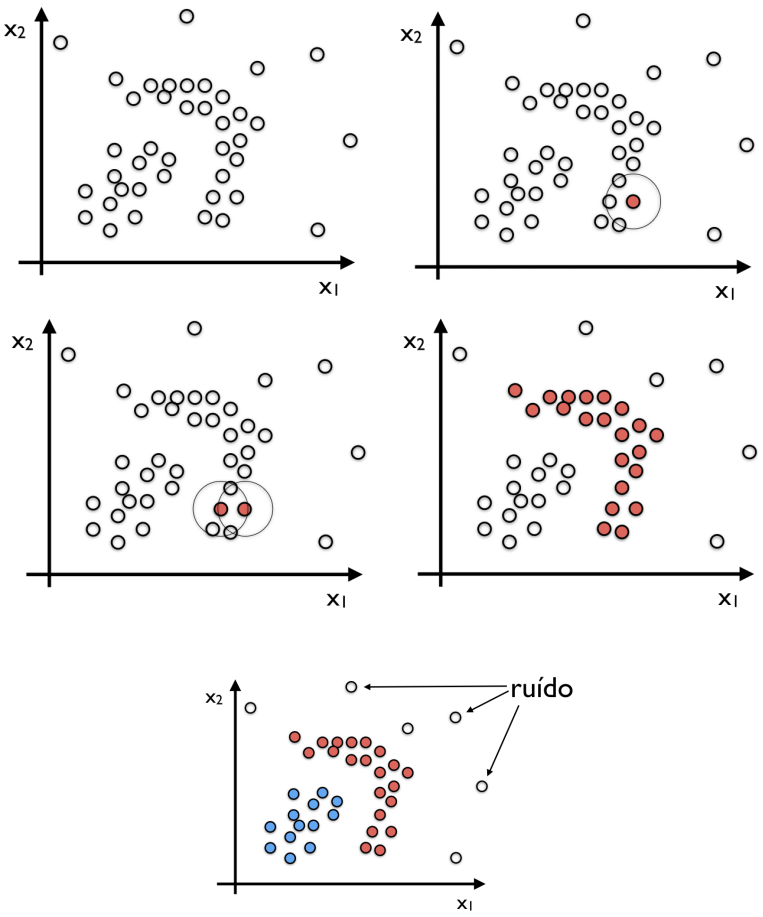
Agrupamento Baseado em Densidade

Desvantagens do k-Médias:

- Grupos encontrados são esféricos.
- Não encontra ruídos nos dados.

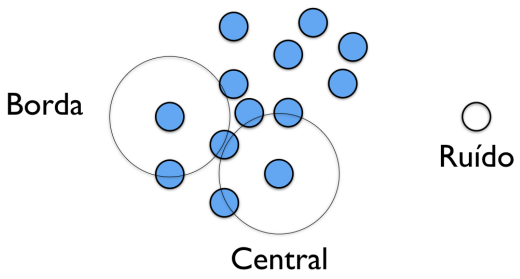
Algoritmos baseados em densidade conseguem lidar com esses problemas. Um exemplo desse tipo de algoritmo é o DBSCAN.

No exemplo abaixo o DBSCAN começa o seu procedimento de agrupamento selecionando um objeto e verificando se em sua vizinhança (raio ϵ, um parâmetro do algoritmo) existe uma "boa densidade" de pontos. Se sim, DBSCAN seleciona um outro objeto dentro da vizinhança do objeto anterior o marca como sendo do mesmo grupo. Esse processo é repetido até encontrarmos todos os objetos dentro dessas "vizinhanças densas". Objetos que não estão em vizinhanças densas são marcados como ruído.

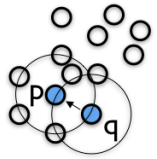


Definições:

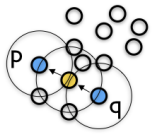
- ϵ raio de vizinhança de um objeto.
- $MinPts$ número mínimo de pontos em uma vizinhança.
- $N_\epsilon(p) = \{q | dist(p, q) \leq \epsilon\}$, é a vizinhança de p.
- q é um **objeto central** se $|N_\epsilon(p)| \geq MinPts$.
- q é um **objeto borda** se q está na vizinhança de um objeto central mas não é um objeto central.
- q é ruído se ele não é nem borda e nem central. Considere o problema abaixo com $MinPts = 4$.



- p é **diretamente alcançável** a partir de q para valores ϵ e $MinPts$ se:
  1. $p \in N_\epsilon(q)$
  2. q é central



- p é **alcançável** a partir de q se existe um caminho diretamente alcançável entre p e q.




```
1 def DBSCAN(ϵ, MinPts, {x_1, x_2, ··· , x_m}):
2   for i in range(0, m):
3     if x_i não tem um grupo:
4       if x_i é central:
5         Colete todos os objetos alcançáveis de x_i de acordo com ϵ e MinPts
6         Remova objetos coletados de consideração.
7       else:
8         Marque x_i como ruído.
```

- Vantagens:
  1. Grupos podem ter formas e tamanhos distintos.
  2. Número de grupos é determinado automaticamente.
  3. Separa ruídos.
- Desvantagem:
  1. Parâmetros ϵ e $MinPts$ podem ser difíceis de serem ajustados.