Universidade Federal do Paraná (UFPR)
Bacharelado em Informática Biomédica

Aprendizado Não Supervisionado
Regressão

David Menotti
www.inf.ufpr.br/menotti/ci171-182

Hoje

- Aprendizado não supervisionado (*Clustering*)
  - k-Means
  - DBScan

Aprendizado
Não Supervisionado

Agenda

- Aprendizado não supervisionado
  - Clustering
    - Particionamento: k-Means
    - Densidade: DBScan

Aprendizagem Não Supervisionada

- Em problemas de aprendizagem supervisionada, contamos como a tupla [X,y], em que o objetivo é classificar y usando o vetor de características X.
- Na aprendizagem não supervisionada, temos somente o vetor X.
- Nesse caso, o objetivo é descobrir alguma coisa a respeito dos dados
  - Por exemplo, como eles estão agrupados.
- Mais subjetiva que a aprendizagem supervisionada uma vez que não existe um objetivo simples como a classificação.

- **Dados não rotulados**:
  - Obtenção de dados não rotulados não é custosa!

Noção de Grupos pode ser Ambígua

•

Quantos clusters?

Seis Clusters

Dois Clusters

Quatro Clusters

Clustering

- Refere-se a um conjunto de técnicas utilizada para encontrar grupos (ou ***clusters***) em um conjunto de dados.

- ***Cluster***: Uma coleção de objetos similares entre si e diferentes dos objetos pertencentes a outros clusters.

- Métodos de *clustering* podem ser divididos em:
  - Particionamento
  - Hierárquico
  - Densidade

Particionamento

k-Means

- Separar os dados em um número pré-determinado de clusters
- k-Means Clustering

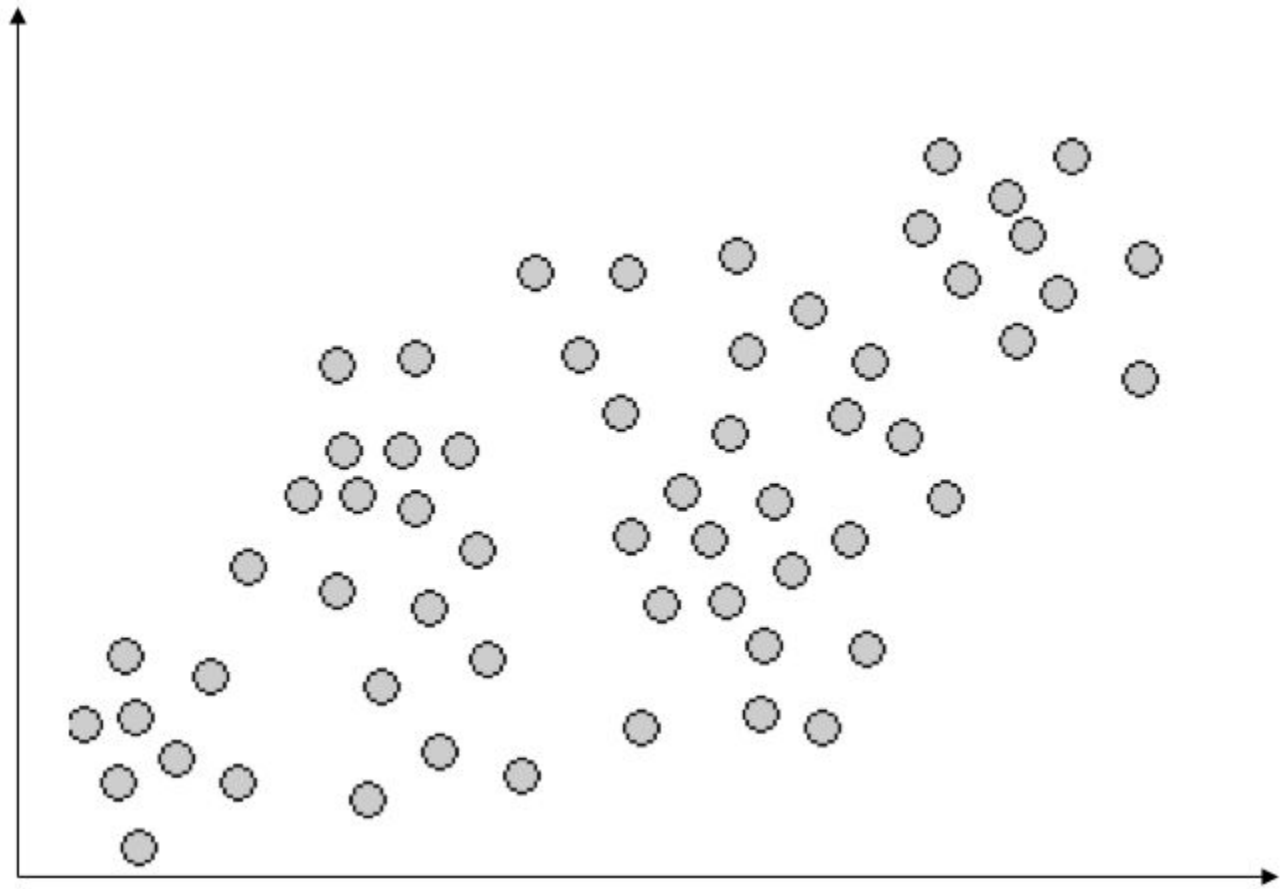
Algoritmo

Entrada: k, dados (**X**)

Saída: dados agrupados em k grupos

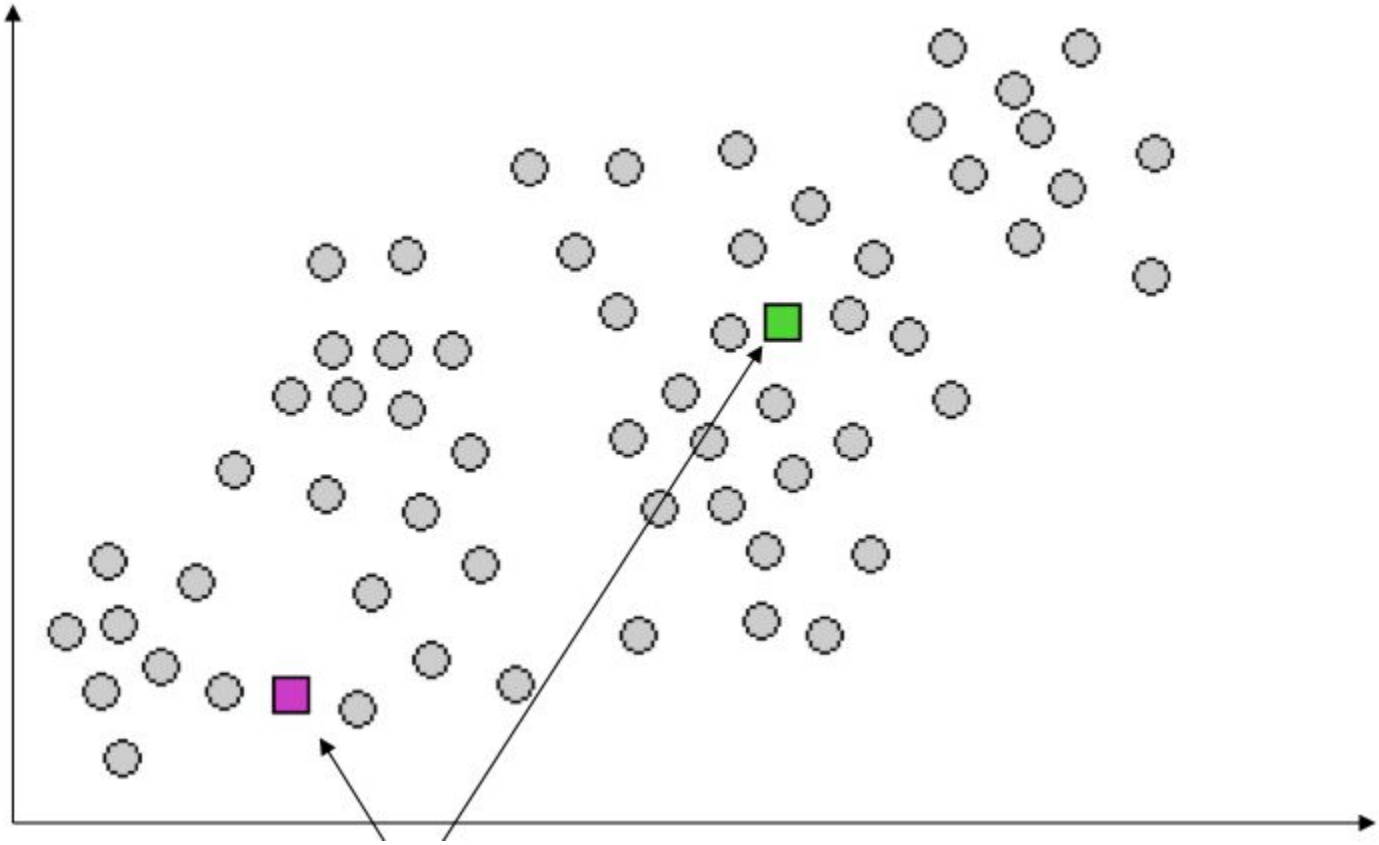
1. Determinar os k centróides
2. Atribuir a cada objeto do grupo o centróide mais próximo.
3. Após atribuir um centróide a cada objeto, recalcular os centróides
4. Repetir os passos 2 e 3 até que os centróides “não sejam“ modificados

k-Means



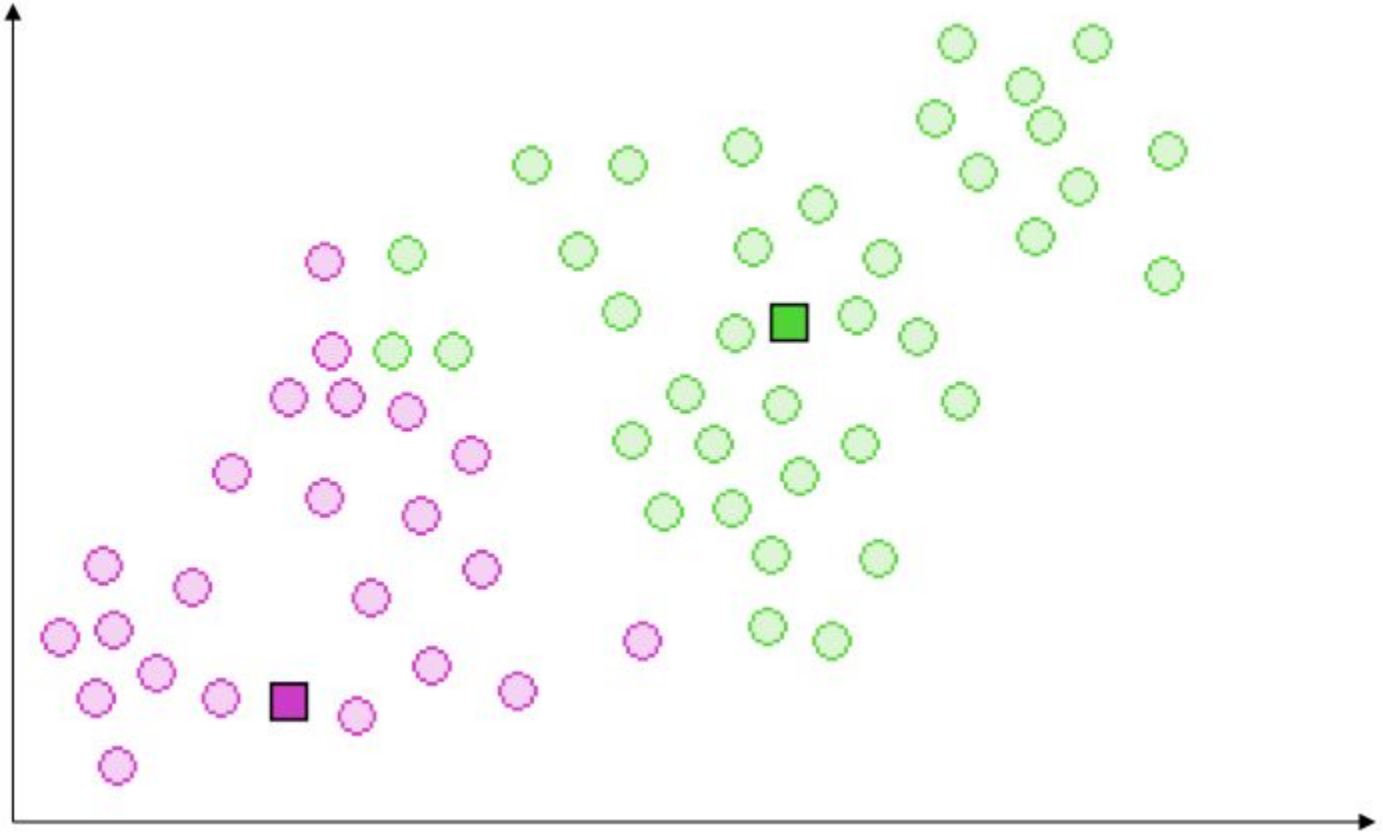
Dados em duas dimensões

k-Means



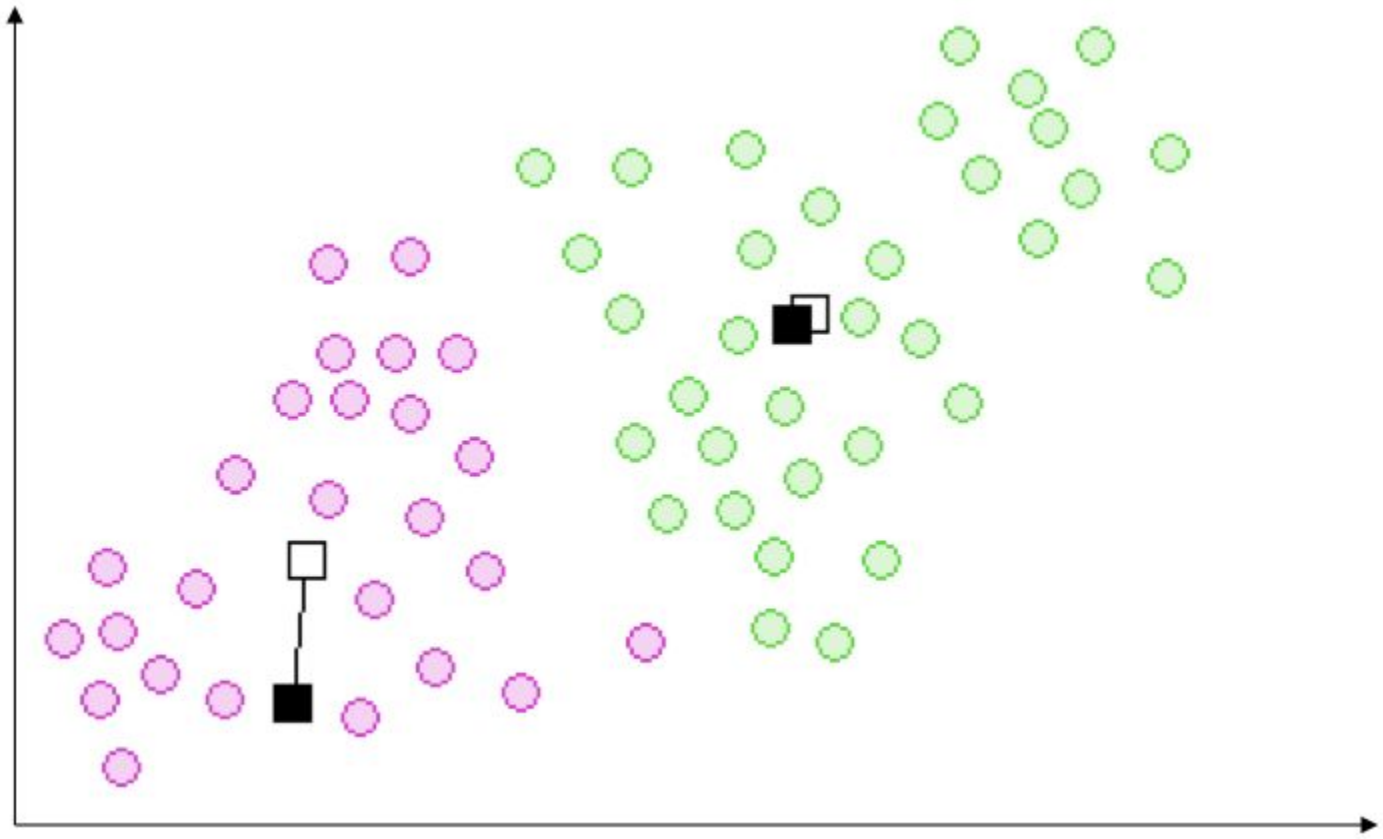
Passo 1: Centróides definidos aleatoriamente

k-Means



Passo 2: Atribuir a cada objeto o centróide mais próximo

k-Means



Passo 3: Recalcular os centróides

k-Means

Obsevações

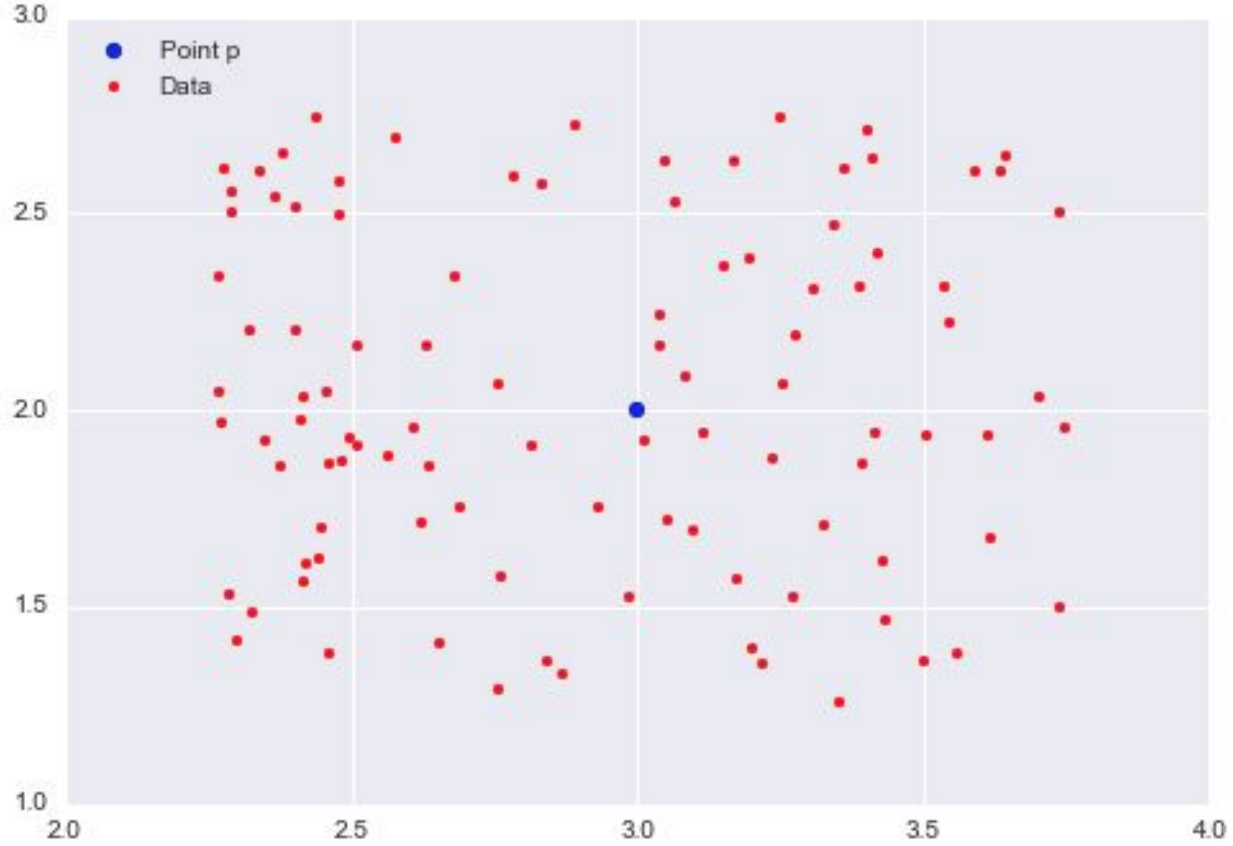
- Relativamente eficiente (escalável)
- Necessidade de definir o número de clusters *a priori*.
- Ineficiente para lidar com ruídos e/ou *outliers*.
  - Todo ponto será atribuído a um cluster mesmo que ele não pertença a nenhum.
  - Em alguns domínios de aplicação (detecção de anomalias) isso causa problemas.
- Inadequado para descobrir *clusters* com formato **não convexo**.

Métodos baseados em Densidade

- Diferentemente do k-Means, os métodos baseados em densidade identificam regiões densas, permitindo a formação de clusters com diferentes formatos.
- Resistentes a presença de ruídos.
- Baseado no conceito de ε-vizinhança.
- Em um espaço bi-dimensional, a ε-vizinhança de um ponto *p* is o conjunto de pontos contido num círculo de raio ε, centrado em *p*.

ε-vizinhança

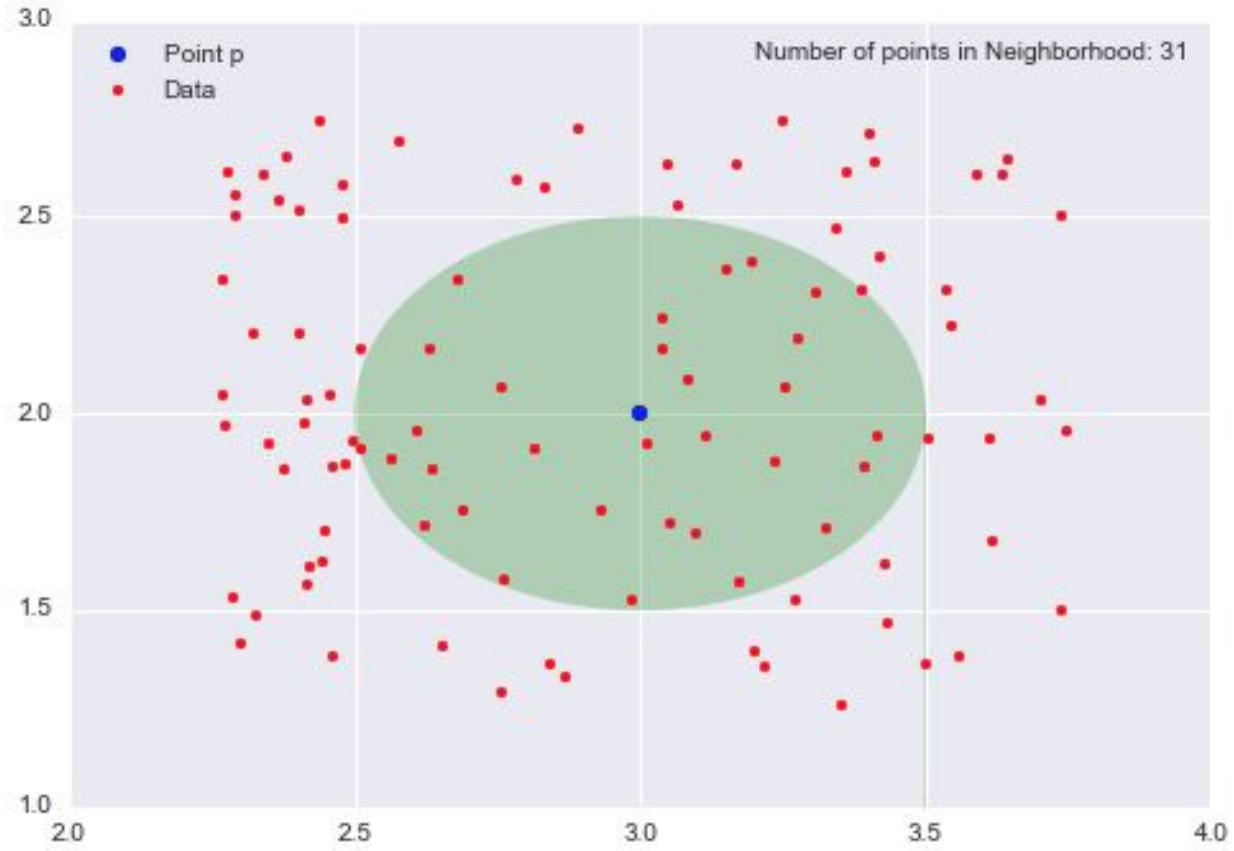
Exemplo



100 pontos no intervalo [1,3] x [2,4] e *p* = (3,2)

ε-vizinhança

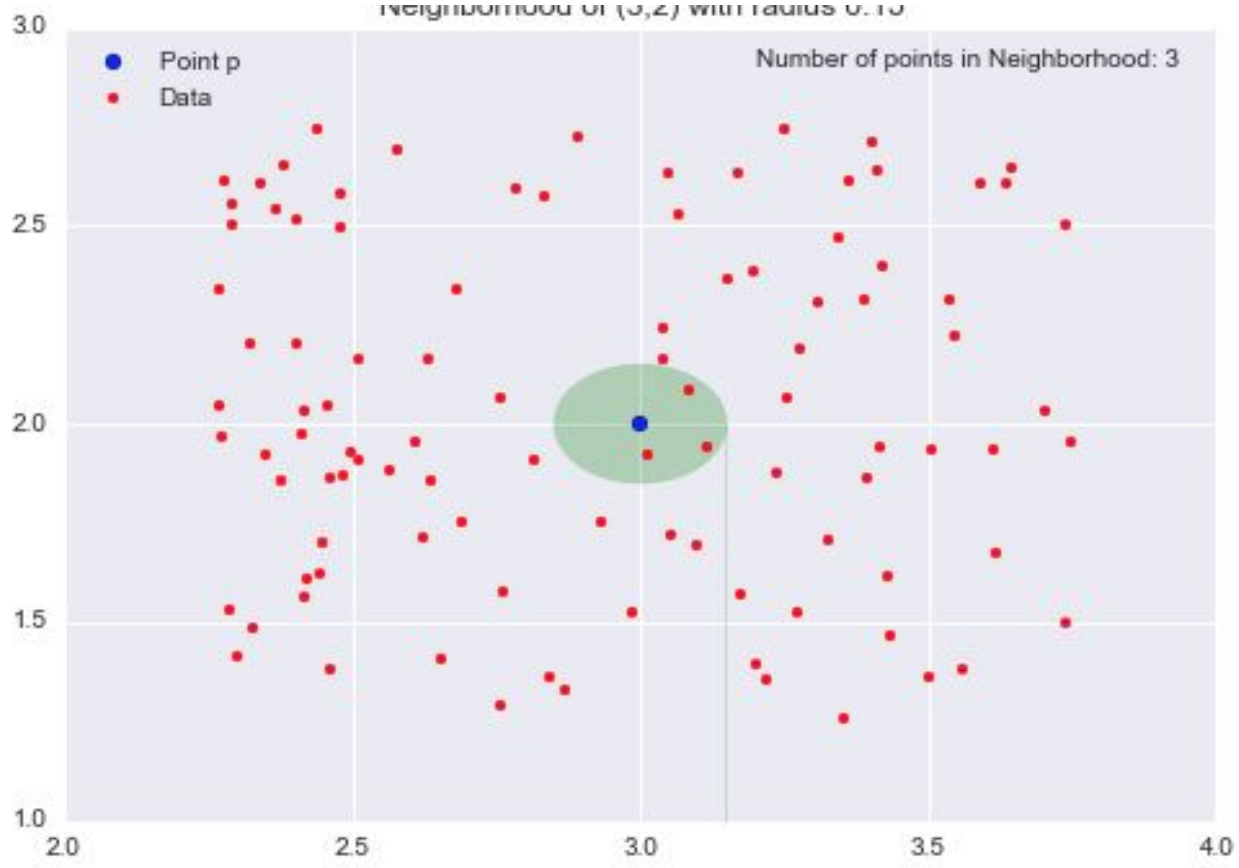
Exemplo



Vizinha de *p* com raio 0,5 (ε = 0,5) e 31 pontos de 100

ε-vizinhança

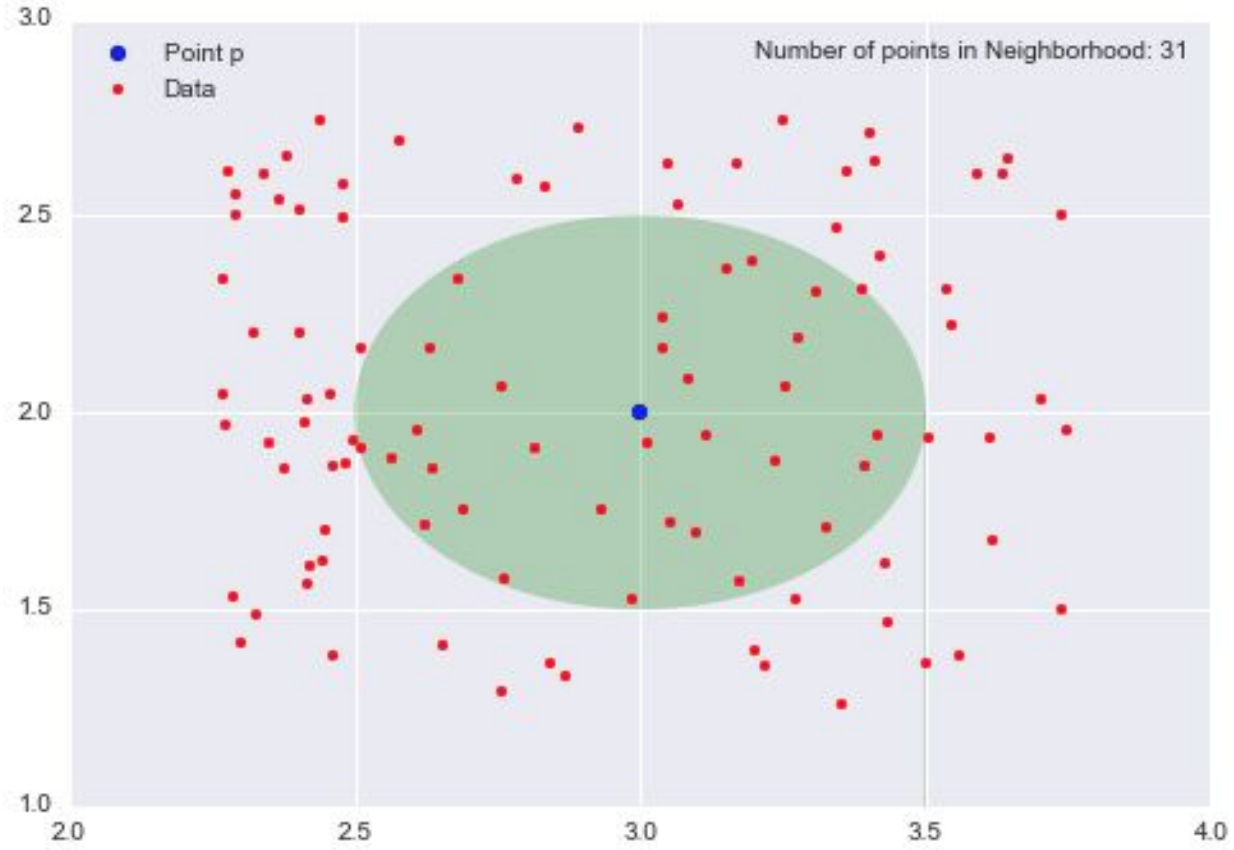
Exemplo



Vizinha de *p* com raio 0,15 (ε = 0,15) e apenas 3 vizinhos.

Densidade

Densidade = Massa / Volume

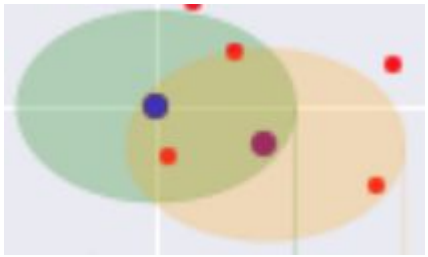


Para o caso de ε = 0,15, temos o volume como a área do círculo, logo $\pi(0,5)^2 = \pi/4$.
Desta forma, massa/volume = 31 pontos / (π/4) ≅ 39,5

Densidade

- A ideia é usar o valor de densidade para agrupar aqueles pontos que possuem valores similares
- Identificar **vizinhanças** densas nas quais a maioria dos pontos está contida.
- Essa é a ideia do algoritmo DBSCAN

DBSCAN



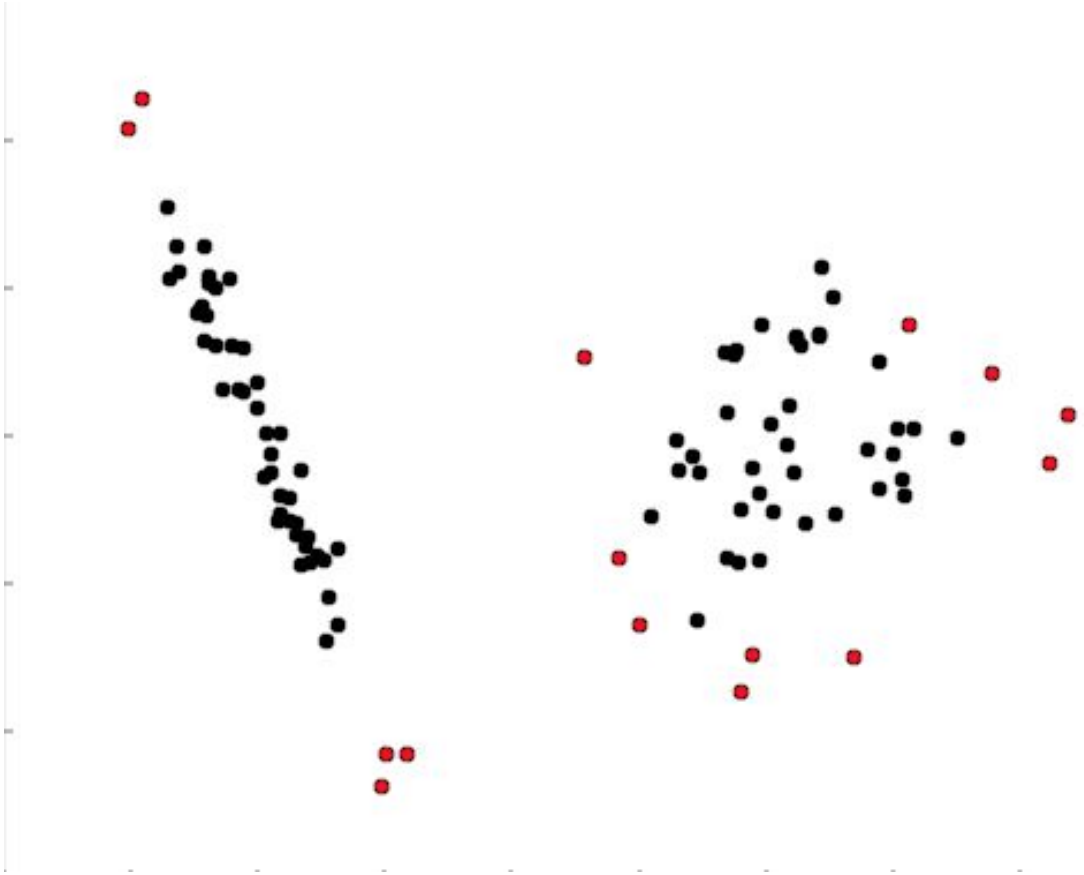
- Possui dois parâmetros:
  - ε: Raio da vizinhança
  - **MinPts**: Número mínimo de pontos na vizinhança para definir um *cluster*.
- Com bases nesses dois parâmetros, DBSCAN classifica os **pontos** em três categorias:
  - ***core point***: Se a vizinhança de p com raio ε contém pelo menos **MinPts**.
  - ***border point***: Se a vizinhança de q com raio ε contém menos de **MinPts**, mas pode ser alcançado por um *core point* p.
  - ***outlier***: Nenhum dos casos acima.

DBSCAN

1. Selecione um ponto aleatoriamente que não tenha sido atribuído a nenhum *cluster* e que não seja um ***outlier***. Calcule sua ε-vizinhança e determine se ele é um ***core point***.
   - Se sim, comece um cluster ao redor desse ponto.
   - Senão, rotule como ***outlier***.
2. Depois de ter encontrado ***core point***, expanda-o adicionando todos os pontos alcançáveis.
3. Repita os dois passos acima até que todos os pontos sejam atribuídos a um cluster ou sejam rotulados como ***outliers***.

DBSCAN

Exemplo



Outliers marcados em **vermelho**

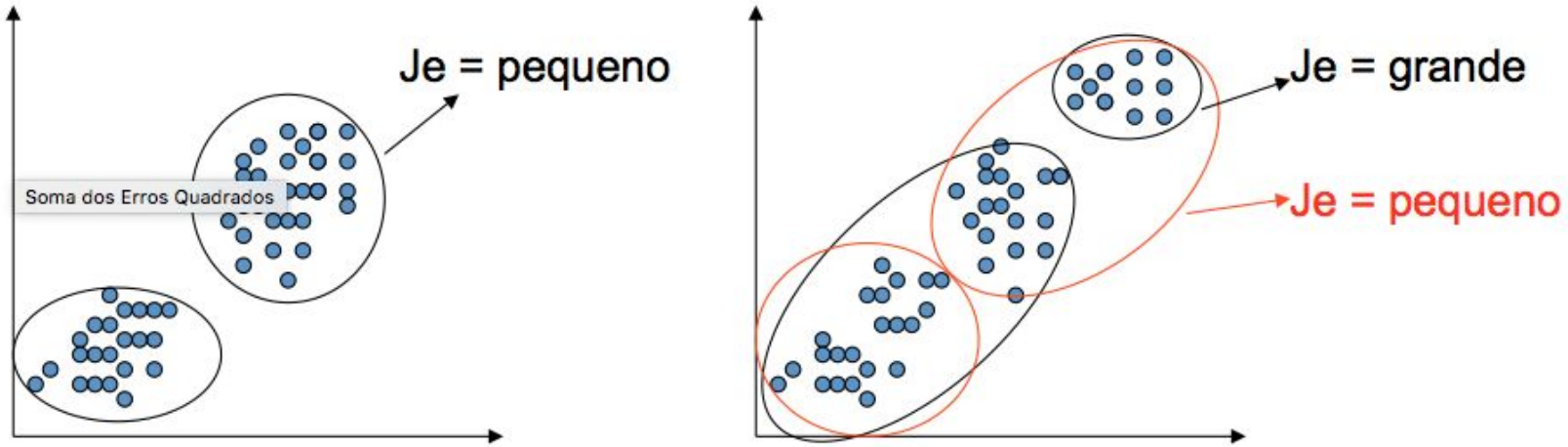
Avaliação em Aprendizagem Não Supervisionada

- Diferentemente da aprendizagem supervisionada, na qual podemos utilizar diferentes métricas de avaliação **objetivas**, a avaliação em aprendizagem não supervisionada é mais subjetiva.

- Porque é importante avaliar *clusters*?
  - Comparar algoritmos de *clustering*.
  - Comparar *clusters* gerados por mais de um algoritmo.

Avaliação em Aprendizagem Não Supervisionada

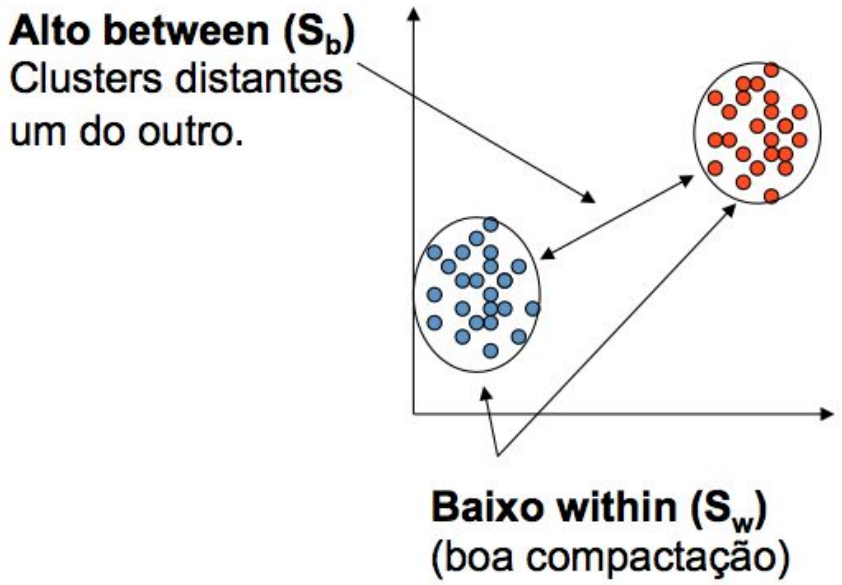
- O que é um bom cluster ? x Coesão e separação.
- A média (centróide) de cada cluster D_i
  - $m_i = \frac{1}{n_i} \sum_{x \in D_i} x$
- Erro^2 $J_e = \sum_{x=1}^{c} \sum_{x \in D_i} (x - m_i)^2$



- ***Outliers*** podem afetar bastante os vetores médios.

Avaliação em Aprendizagem Não Supervisionada

- A coesão (*within-cluster*) pode ser medida pela soma dos erros quadrados dentro de cada cluster
  - $S_w = \sum_{x \in D_i} (x - m_i)^2$
- A separação (between-cluster) é medida pela soma dos quadrados entre clusters
  - $S_b = \sum_{x=1}^{c} n_i (m_i - m)^2$
  - Em que m é o vetor médio total (centróide geral)



Avaliação em Aprendizagem Não Supervisionada

Coeficiente Silhouete

- O coeficiente Silhouette combina coesão e separação e pode ser calculado seguindo quatro passos:
  a. Para um dado objeto do cluster i, calcule a distância média (a_i) para todos os outros objetos de i.
  b. Seja b_i a menor distância média de i para todos os outros *clusters*, dos quais i não seja membro.
  c. $s_i = (b_i - a_i)/\max(a_i, b_i)$
  d. O coeficiente s_k para todos os objetos I numa partição de k *clusters* é dados por $s_k = \frac{1}{I}\sum_{i=1}^{I} s_i$

Avaliação em Aprendizagem Não Supervisionada

Coeficiente *Silhouete*

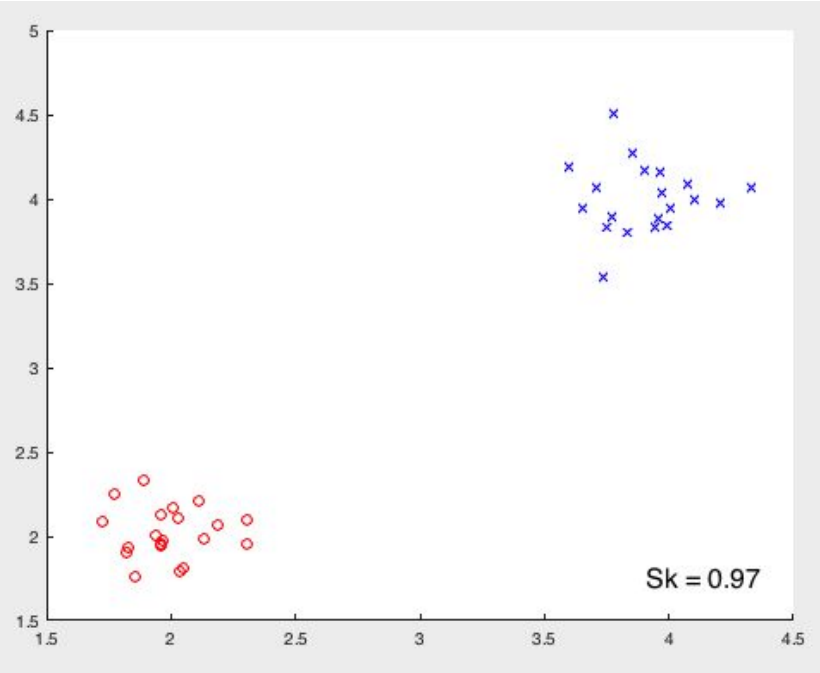
- Coesão e separação

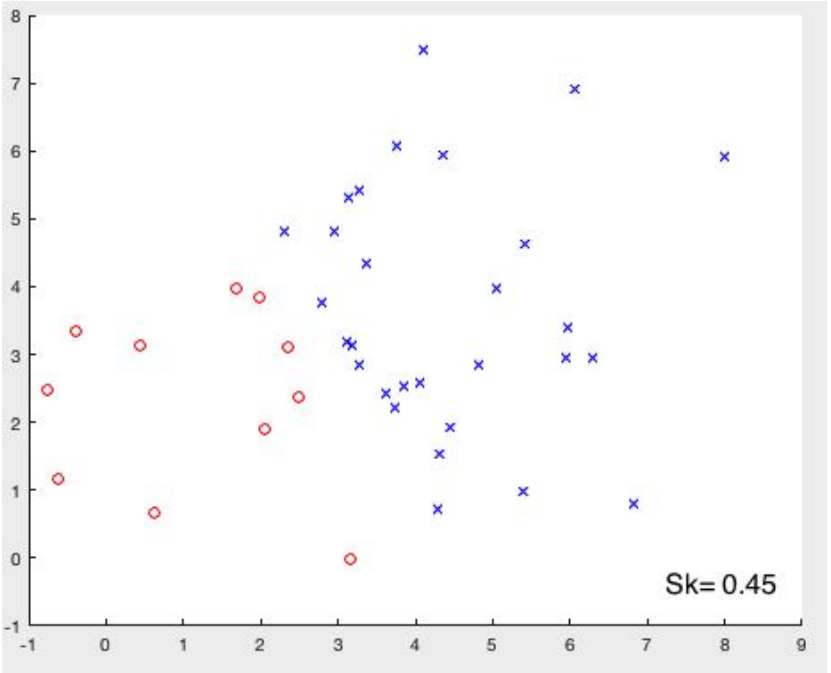
$$s_k = \frac{1}{I} \sum_{i=1}^{I} s_i \qquad\qquad s_i = (b_i - a_i) / \max(a_i, b_i)$$

- O valor do coeficiente pode variar entre -1 e 1.
- Um valor negativo não é desejável pois nesse caso o valor de a_i seria maior do que b_i
- É desejável que o coeficiente seja positivo ($a_i < b_i$) o que indica clusters compactos (a_i próximo de zero).

Avaliação em Aprendizagem Não Supervisionada

Coeficiente *Silhouete*





Avaliação em Aprendizagem Não Supervisionada

Coeficiente Silhouete

- Prática comum: Utilize mais de um índice de clustering
- Voto entre os índices pode ser um bom indicativo de qual número de clusters você deve escolher.

```
*****************************************************************
* Among all indices:
* 3 proposed 2 as the best number of clusters
* 4 proposed 3 as the best number of clusters
* 19 proposed 4 as the best number of clusters
* 1 proposed 5 as the best number of clusters

                 ***** Conclusion *****

* According to the majority rule, the best number of clusters is  4
```

Referências

- Charrad et al.,
  NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set,
  Journal of Statistical Software, 61, 2014.

- Luiz E. S. Oliviera,
  Aprendizado Não-supervisionado,
  Notas de Aulas, DInf / UFPR, 2017.