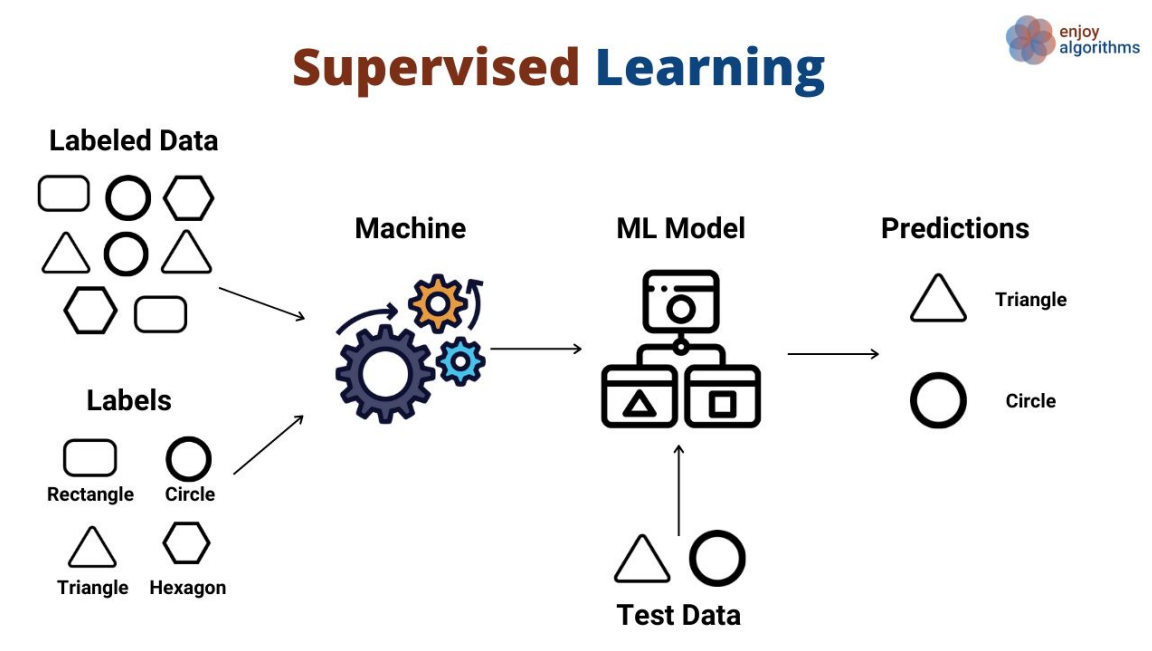
Semi-Supervised Learning

Gabriel e Thiago

Definição - Supervisionado

- Entrada de dados tem rótulo conhecido
- Algoritmos utilizam rótulos conhecidos e mais parâmetros para criar regras
- Regras definidas são utilizadas para definir valores para dados desconhecidos

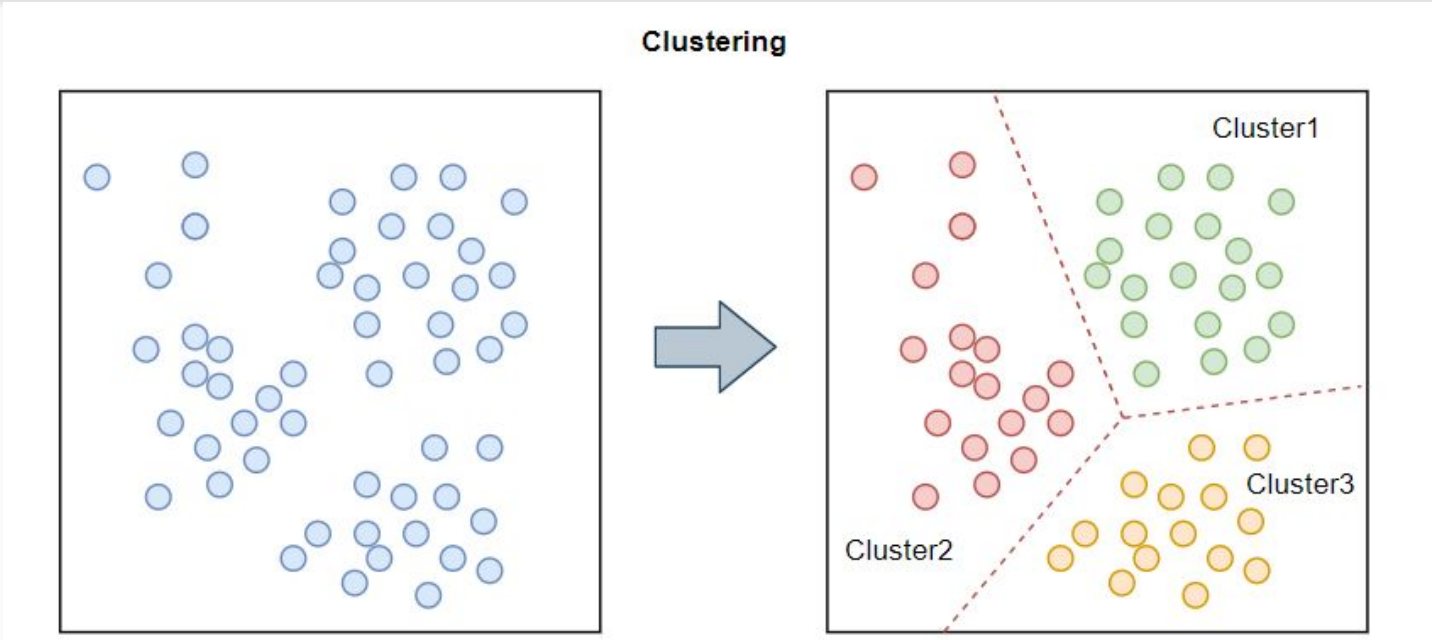
Definição - Supervisionado



Definição - Não supervisionado

- Entrada de dados não tem rótulo conhecido
- Algoritmos utilizam apenas parâmetros conhecidos para definir regras
- É possível diferenciar dados utilizados mas não definir o seu rótulo

Definição - Não supervisionado



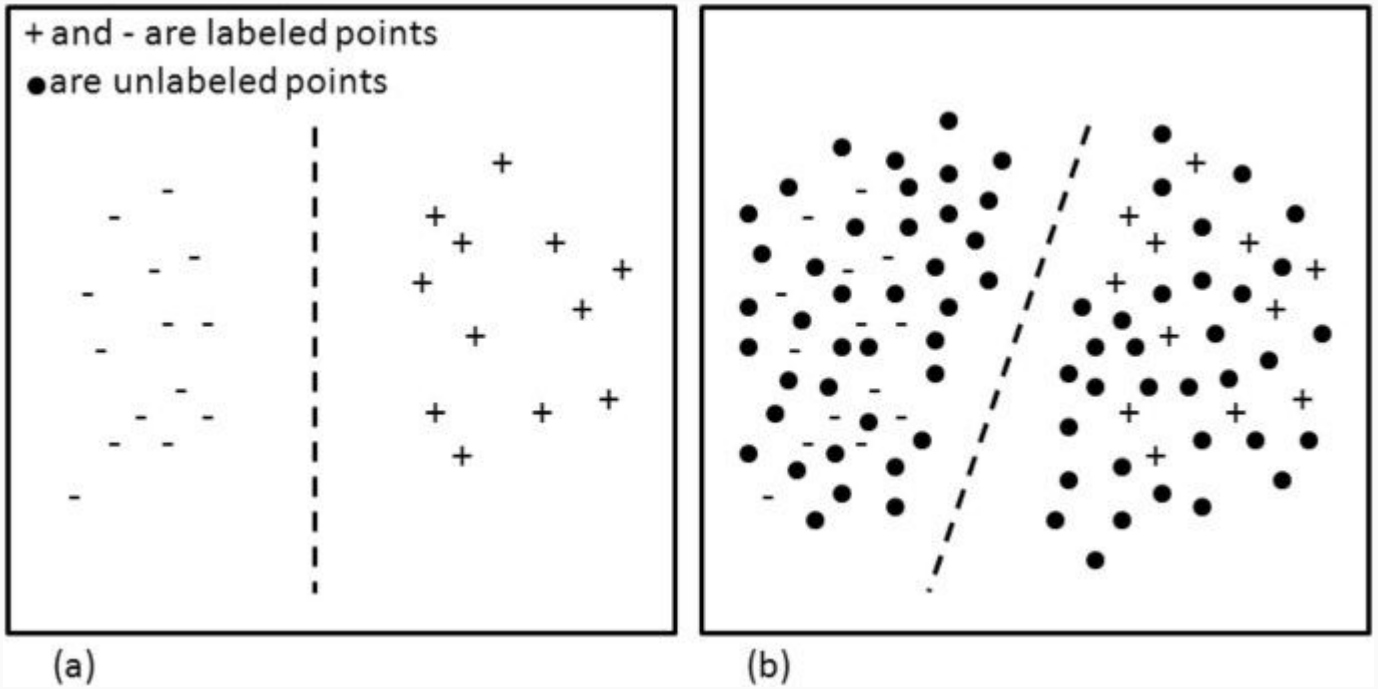
Definição - Semi Supervisionado

- Trabalha com dados rotulados e não rotulados
- Regras podem ser traçadas de acordo com os dados rotulados
- Dados não rotulados ajudam a melhorar acurácia

Definição - Semi Supervisionado

- Generative Models
- Low Density Separation
- Graph Based Methods

Definição - Semi Supervisionado



Histórico

- Self Training 1960s
- Transductive Inference - Vapnik - 1970s
- Natural Language related 1990s
- “Semi-Supervised” para classificação - 1992 - Mertz et al - SMART2

Histórico - Exemplos

- Co-Learning
- COP Kmeans
- Seeded-K-means
- Usado em áreas com vasta quantidade de dados não rotulado
  - Imagens
  - Texto de websites
  - Sequencias Proteicas
  - Linguagem natural

COP Kmeans - Introdução

- Clusterização de dados rotulados e não rotulados
- É uma variante do algoritmo Kmeans
- Usa relações já conhecidas entre dados para criação de clusters

COP Kmeans - Algoritmo

COP-k-means (data set D, must-link constraints $Con_{=} \in D \times D$, cannot-link constraints $Con_{\neq} \in D \times D$)
1: Let $C_1, \ldots, C_k$ be the initial cluster centers.
2: For each point d_i in D, assign it to the closest cluster C_i such that VIOLATE-CONSTRAINTS(d_i, C_j, $Con_{=}$, $Con_{\neq}$) is false.
3: If no such cluster exists, fail and return.
4: For each cluster C_i, update its center by averaging all of the points d_i that have been assigned to it.
5: Iterate between step 2 and step 4 until convergence.
6: Return $C_1, \ldots, C_k$.

VIOLATE-CONSTRAINTS(d, C, $Con_{=}$, $Con_{\neq}$)
7: For each $(d, d_m) \in Con_{=}$: If $d_m \notin C$, return true.
8: For each $(d, d_c) \in Con_{=}$: If $d_c \in C$, return true.
9: Otherwise, return false.

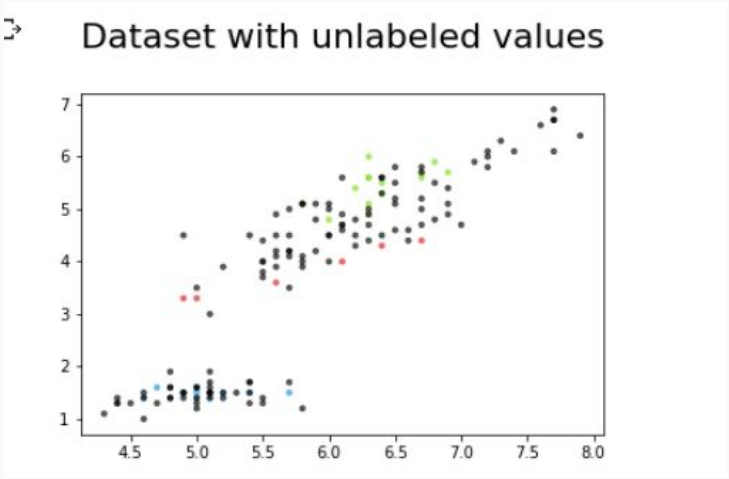
COP Kmeans - Implementação

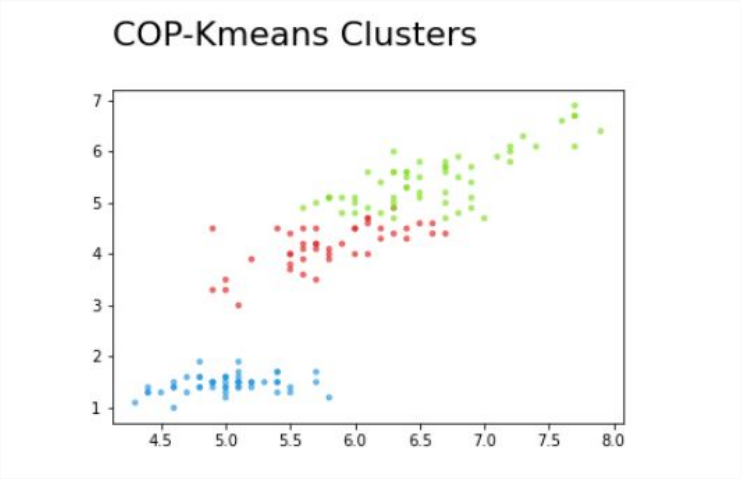
```
## Rotina principal
centroids, clusters = create_clusters()
must_link, not_link = create_retrictions(dataset)
max_iter = 100
for iter in range(max_iter):
  clusters = reset_clusters(clusters)

  ## Assign Clusters
  for i in range(len(dataset)):
    d = dataset[i]
    cluster_idx = get_closest_cluster_index(centroids, d, i, must_link, not_link)
    if cluster_idx is None:
      print("ERROR: Falhou")
      break
    if i not in clusters[cluster_idx]:
      clusters[cluster_idx].append(i)

  ## Recalculate Centroids
  for cluster_idx in range(len(clusters)):
    cluster_data = get_cluster_data(cluster_idx)
    centroids[cluster_idx] = np.mean(cluster_data[:,0:4], axis=0)
```

COP Kmeans - Resultados





Link

https://github.com/Suniaster/COP-Kmeans/blob/main/FSI_COP_Kmeans.ipynb

Referências

Semi Supervised Learning - Adaptive Computation and Machine Learning - Olivier Chapelle, Bernhard Schölkopf e Alexander Zien - 2006

Aprendizado de máquina semi-supervisionado: proposta de um algoritmo para rotular exemplos a partir de poucos exemplos rotulados - Marcelo Kaminski Sanches - 2003