Exemplo de Agrupamento

1) Identificação do Problema

Nome da base: Iris
Qtde instâncias: 150
Atributos de entrada: 4 valores numéricos, a saber:

1. sepal length in cm
2. sepal width in cm
3. petal length in cm
4. petal width in cm

Atributo alvo: tipo da flor Iris (Setosa, Versicolour, Virginica)
Obs: No caso da tarefa de agrupamento utilizaremos o atributo alvo apenas para verificar os resultados, pois a aprendizagem neste caso é não supervisionada.

Descrição do problema: as instâncias da base íris serão consideradas sem o rótulo (saída ou resposta) e os atributos de entrada serão utilizados para encontrar grupos por similaridade.
Link para a base de dados e descrição detalhada: https://archive.ics.uci.edu/ml/datasets/iris

2) Descrição dos experimentos

Objetivo deste exemplo é apresentar a tarefa de agrupamento e como podemos executá-la usando Python e sua biblioteca de Machine Learning (Scikit Learn). Utilizaremos o algoritmo KMeans (ou KMédias) e assumiremos a existência de 3 clusters assim poderemos comparar com a situação real.

- Protocolo Experimental
  - Toda a base será utiliza no agrupamento;
  - Número de clusters: 3
- Medida de desempenho: Silhouette score

3) Script Python Utilizado

```
# Carrega as bibliotecas necessárias
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
from sklearn import datasets
```

```
np.random.seed(5)

# carrega a base Iris
iris = datasets.load_iris()
X = iris.data
y = iris.target

# Realiza o agrupamento considerando 3 grupos ou cliusters
n_clusters=3
cluster=KMeans(n_clusters);
cluster.fit(X)
cluster_labels = cluster.fit_predict(X)

# Calcula o Silhouette_score o qual dá uma perspectiva da densidade e separação dos clusters
silhouette_avg = silhouette_score(X, cluster_labels)
print("\n\n For ", n_clusters,
      " clusters, the average silhouette_score is :", silhouette_avg)

# Plota o resultado (visualização)
fig = plt.figure("Figura 1", figsize=(4, 3))
ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=134)
labels = cluster.labels_
ax.scatter(X[:, 3], X[:, 0], X[:, 2],
           c=labels.astype(np.float), edgecolor='k')

ax.w_xaxis.set_ticklabels([])
ax.w_yaxis.set_ticklabels([])
ax.w_zaxis.set_ticklabels([])
ax.set_xlabel('Petal width')
ax.set_ylabel('Sepal length')
ax.set_zlabel('Petal length')
ax.set_title("Resultado para 3 clusters")
ax.dist = 12



# Plota o resultado real (ground truth) para verificação
fig = plt.figure("Figura 2", figsize=(4, 3))
ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=134)
```

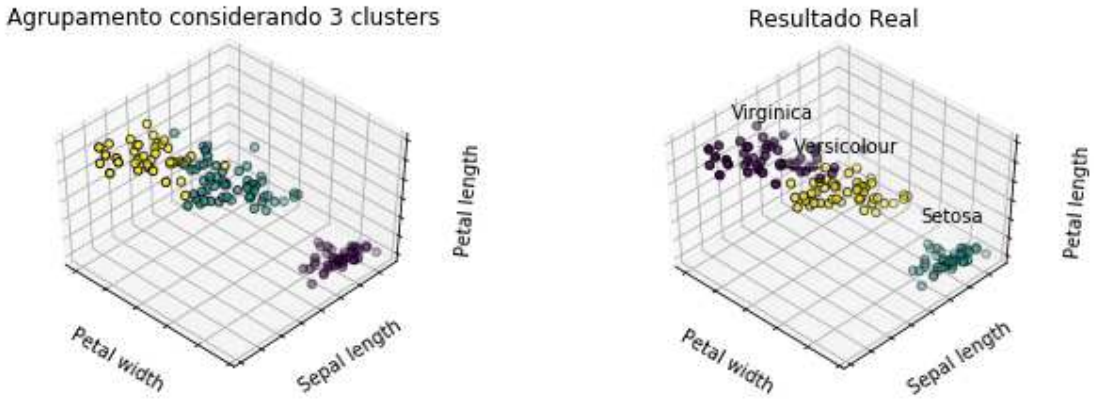
```
for name, label in [('Setosa', 0),
                    ('Versicolour', 1),
                    ('Virginica', 2)]:
    ax.text3D(X[y == label, 3].mean(),
              X[y == label, 0].mean(),
              X[y == label, 2].mean() + 2, name,
              horizontalalignment='center',
              bbox=dict(alpha=.2, edgecolor='w', facecolor='w'))

y = np.choose(y, [1, 2, 0]).astype(np.float)
ax.scatter(X[:, 3], X[:, 0], X[:, 2], c=y, edgecolor='k')

ax.w_xaxis.set_ticklabels([])
ax.w_yaxis.set_ticklabels([])
ax.w_zaxis.set_ticklabels([])
ax.set_xlabel('Petal width')
ax.set_ylabel('Sepal length')
ax.set_zlabel('Petal length')
ax.set_title('Resultado Real')
ax.dist = 12
fig.show()
```

4) Resultados do Processo de Classificação

4.1)Agrupamento considerando 3 clusters



Explicaremos o KMeans nas próximas aulas. Conforme é possível observar o agrupamento considerando 3 clusters visualmente apresenta um resultado bem próximo da realidade, com erros apenas entre os tipos de flores Virginica e Versicolour. Vale ressaltar que em situação real dificilmente teremos os dados rotulados, neste caso foi possível pois este problema é originalmente utilizado para a tarefa de classificação.

4.2) Resultado para 3 clusters

Silhouette score: 0.55

O índice silhouette nos dá uma perspectiva da densidade e separação dos clusters. O melhor valor é 1 e o pior valor é -1. Valores próximos a 0 indicam clusters sobrepostos. Valores negativos geralmente indicam que uma amostra foi atribuída ao cluster errado, pois um cluster diferente é mais semelhante. Espera-se que os cluster sejam bem compactos (baixa dispersão dentro de cada cluster) e bem separados (alta dispersão entre clusters).

Observações importantes:

i) A análise dos resultados é importante para definirmos os próximos passos. Uma vez identificados os grupos estes podem ser utilizados em um processo futuro de classificação.