MACHINE LEARNING AND PATTERN RECOGNITION

K-means and Spectral Clustering

Year: 2017/2018

Dao Quang Hoan 871510

Outline

1. K-means & Spectral Clustering
2. Elbow method
3. Apply to datasets
    a. Moons dataset
    b. Iris dataset
    c. Seeds dataset
4. Conclusion

Introduction

K-means:

Step 1: Pick K random points as cluster centers

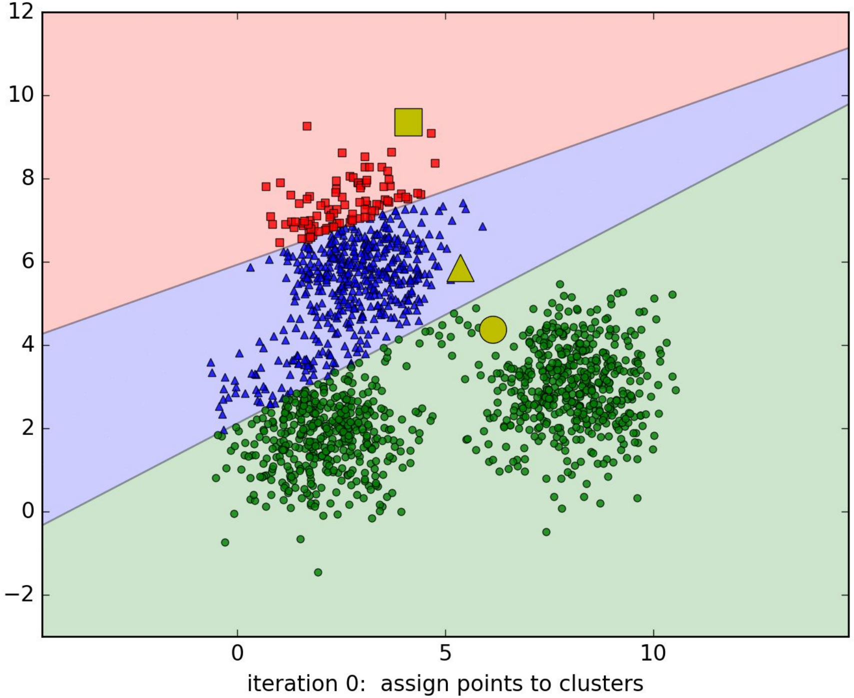
Step 2: Assign data points to the closest centers

Step 3: Change the cluster center to the average of the assigned points

Step 4: Repeat step 2 and 3 until until the centers don’t change

$$\mathrm{y}_i = \arg\min_{\mathrm{y}_i} \sum_{j=1}^{K} y_{ij} \|\mathrm{x}_i - \mathrm{m}_j\|_2^2$$

$$\text{subject to: } y_{ij} \in \{0,1\} \ \forall j; \quad \sum_{j=1}^{K} y_{ij} = 1$$



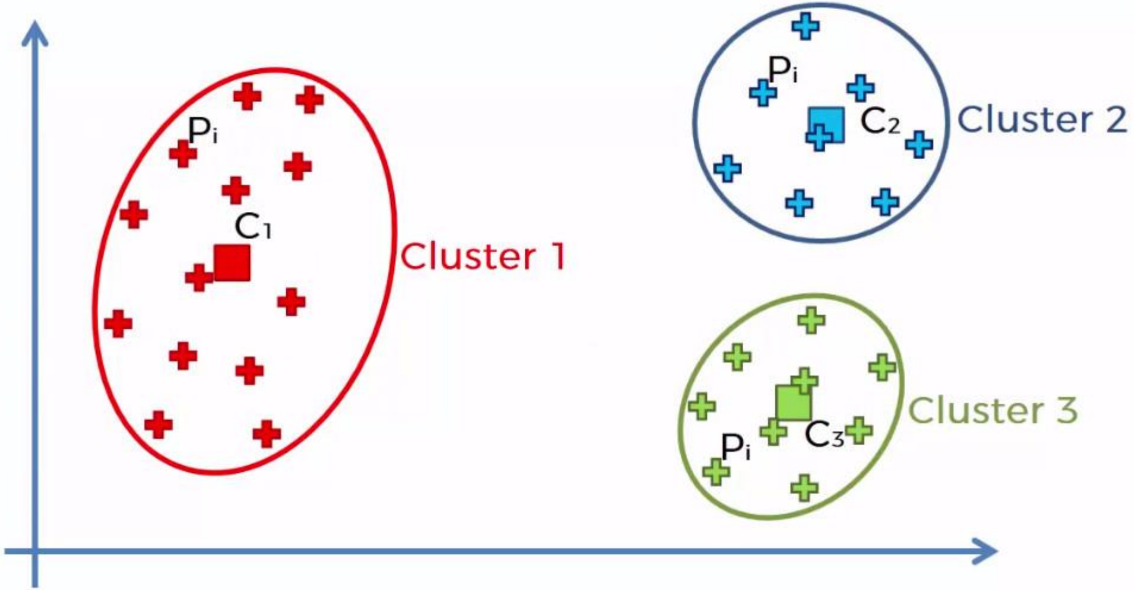
Spectral clustering:

Step 1: Construct a similarity graph (KNN graph)

Step 2: Embed the data points in low dimensional space in which the clusters are

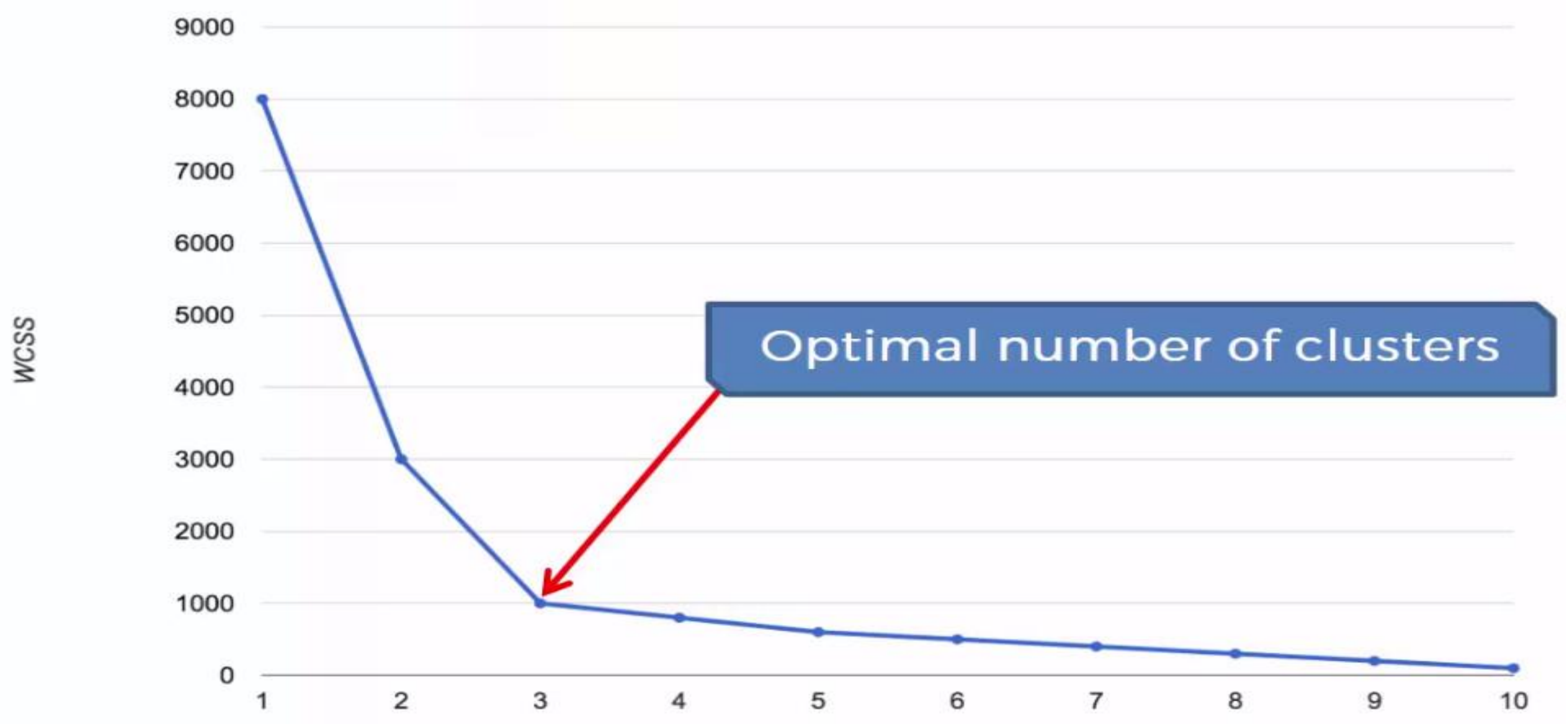
Step 3: Use the lowest eigenvalue in order to choose the eigenvector for clusters

Elbow method



$$\mathrm{WCSS} = \sum_{\mathrm{P_i\ in\ Cluster\ 1}} \mathrm{distance}(\mathrm{P_i}, \mathrm{C_1})^2 + \sum_{\mathrm{P_i\ in\ Cluster\ 2}} \mathrm{distance}(\mathrm{P_i}, \mathrm{C_2})^2 + \sum_{\mathrm{P_i\ in\ Cluster\ 3}} \mathrm{distance}(\mathrm{P_i}, \mathrm{C_3})^2$$

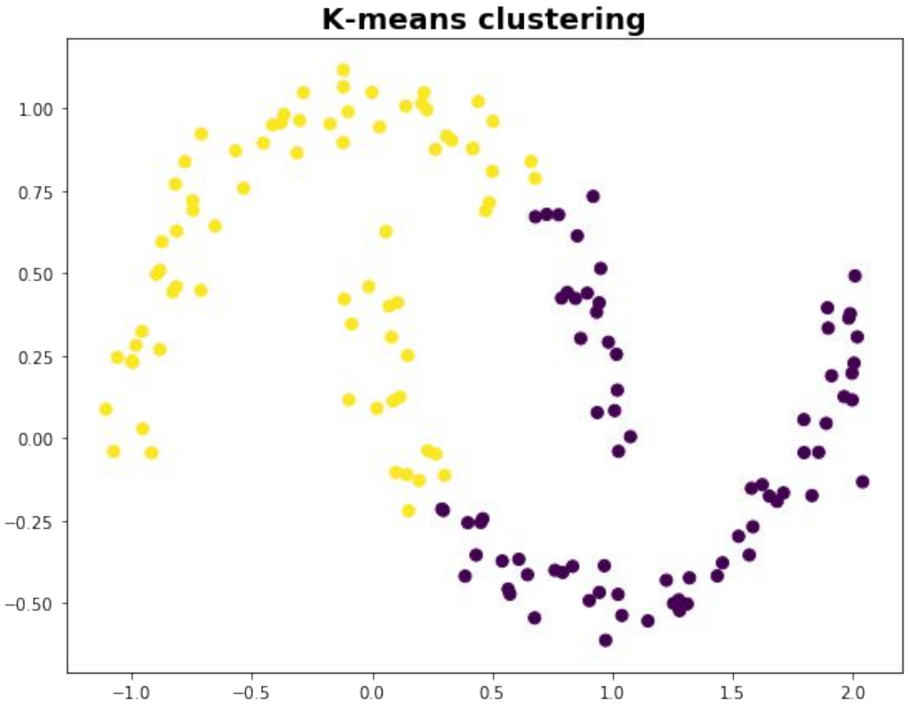
The Elbow Method

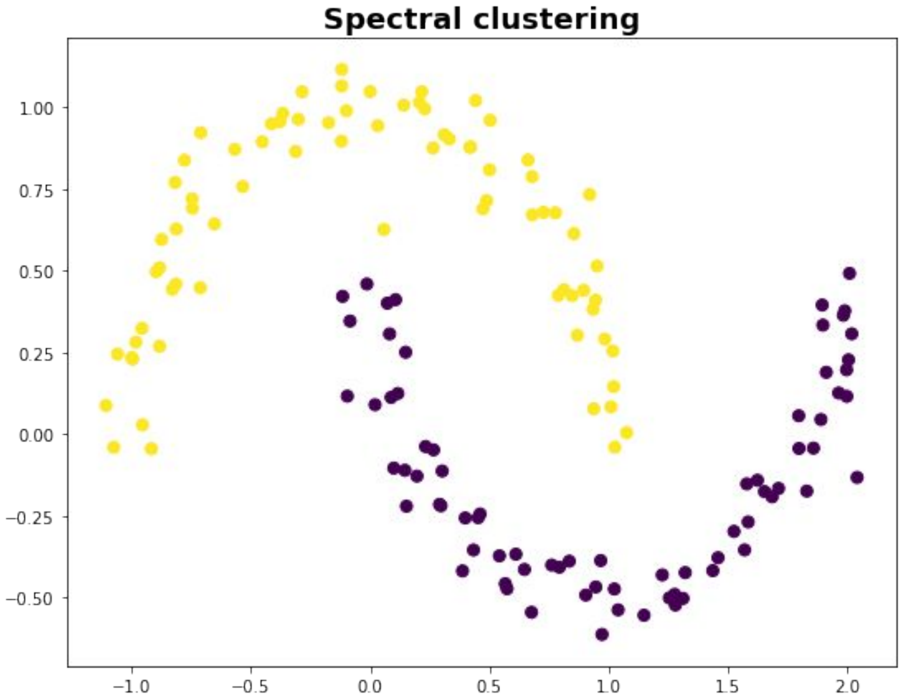


Datasets

Dataset	Num of instances	Num of attributes	Num of clusters
Moons	150	2	2
Iris	150	4	3
Seeds	210	7	3

Moons dataset





Iris dataset

1. sepal length in cm
2. sepal width in cm
3. petal length in cm
4. petal width in cm
5. Class:

-- Iris Setosa

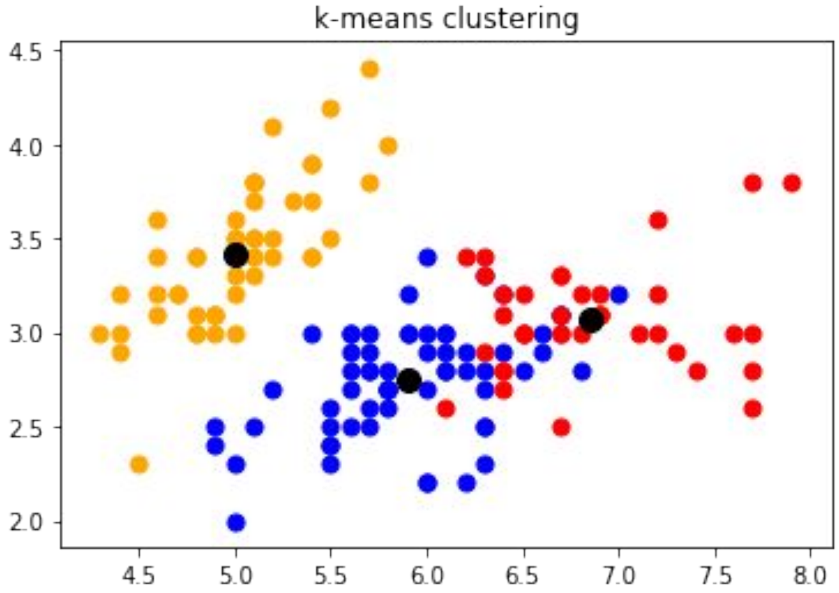
-- Iris Versicolour

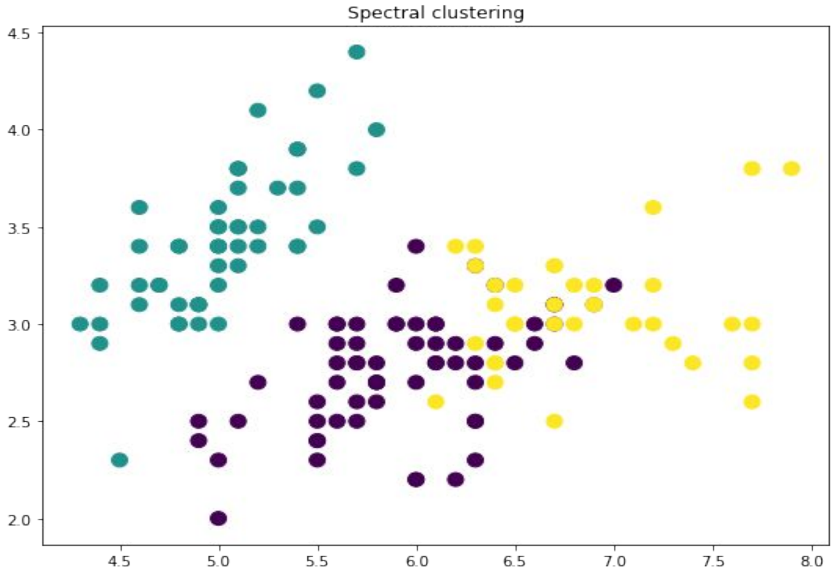
-- Iris Virginica

```
dataset.head()
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

Iris dataset





Seeds dataset

0. area A,

1. perimeter P,

2. compactness C = 4*pi*A/P^2,

3. length of kernel,

4. width of kernel,

5. asymmetry coefficient

6. lngth of kernel groove.

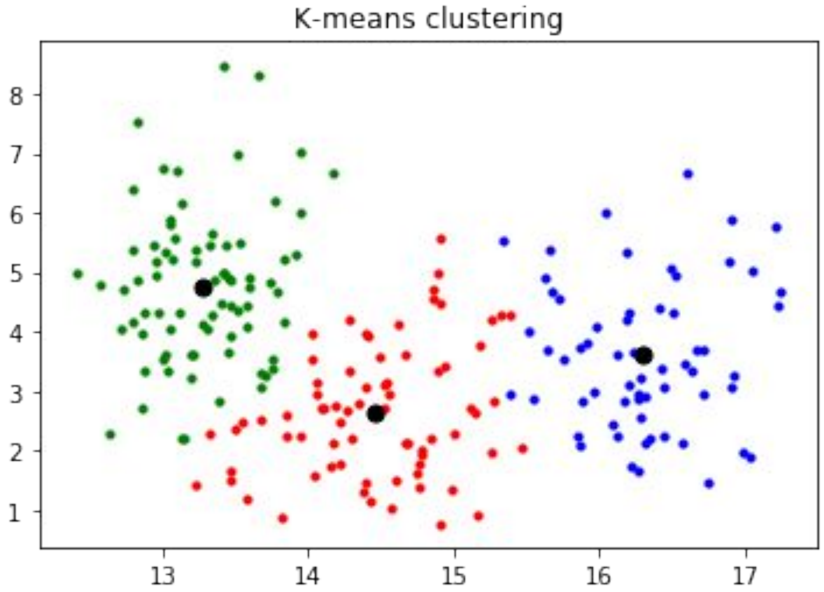
7. Labels

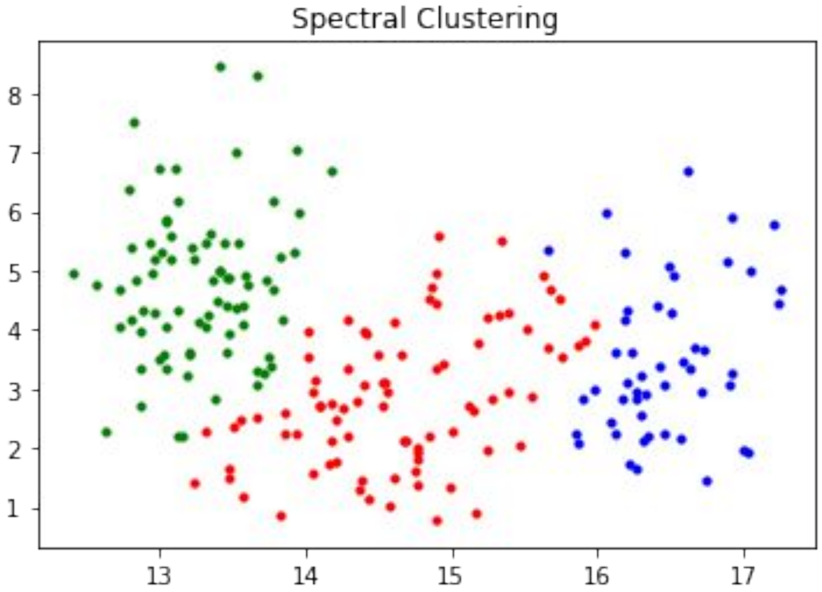
All of these parameters were real-valued continuous.

```
data.head()
```

	0	1	2	3	4	5	6	7
0	15.26	14.84	0.8710	5.763	3.312	2.221	5.220	1
1	14.88	14.57	0.8811	5.554	3.333	1.018	4.956	1
2	14.29	14.09	0.9050	5.291	3.337	2.699	4.825	1
3	13.84	13.94	0.8955	5.324	3.379	2.259	4.805	1
4	16.14	14.99	0.9034	5.658	3.562	1.355	5.175	1

Seeds dataset





Spectral Clustering: Pros and Cons

- Elegant, and well-founded mathematically
- Works quite well when relations are approximately transitive (like similarity)
- Very noisy datasets cause problems

– “Informative” eigenvectors need not be in top few

– Performance can drop suddenly from good to terrible

- Expensive for very large datasets

– Computing eigenvectors is the bottleneck

Conclusion

K-Means

– Fast and Simple

– “Embarrassingly parallel”

– Not very useful on anisotropic data

Spectral clustering

– Excellent quality under many different data forms

– Much slower than KMeans