K-Means Clustering

Machine Learning Techniques

Dr. Ashish Tendulkar
IIT Madras

Overview

In this course, so far we studied supervised machine learning algorithms, where training data consist of features and labels.

There is another class of ML models where training data contains only features and labels are not available. They are called unsupervised ML algorithms.

Clustering is an example of unsupervised ML algorithm.

Clustering is the process of grouping similar data points or examples in the training set in the same cluster.

You may wonder what do we do with training examples represented with features but without label.

Clustering is widely used in many application such as

- Customer profiling
- Anomaly detection
- Image segmentation
- Image compression
- Geostatistics
- Astronomy

Components of clustering

Just like any ML algorithm clustering also has five components:

1. Training data
2. Model
3. Loss function
4. Optimization
5. Model selection/evaluation

Training data

The training data consists of examples with only features.

$$D = \{\mathbf{x}^{(i)}\}_{i=1}^{n}$$

Each example is represented with m features.

Model

The model of clustering is as follows:

Examples/Clusters	**C1**	**C2**	**…**	**Ck**
x1	1	0		0
x2	0	1		0
…				
xn	0	0		1

- We need to assign each point in the training set to one of the k clusters. This is called **hard clustering**.
- Each point has a probability of membership to k clusters such that the sum of probabilities is 1. This is called **soft clustering**.

In this course we will be focusing on hard clustering.

Examples/Clusters	C1	C2	...	Ck
x1	1	0		0
x2	0	1		0
...				
xn	0	0		1

Cluster $c_r; 1 \leq r \leq k$ is represented by its centroid, which is calculated as average of vector of points in that cluster.

$$\mu^{(r)} = \frac{1}{|\mathbf{x}^{(i)} \in c_r|} \sum_{i=1}^{n} \mathbf{1}(\mathbf{x}^{(i)} \in c_r)\mathbf{x}^{(i)}$$

$$\mu^{(r)} = \frac{1}{|\mathbf{x}^{(i)} \in c_r|} \sum \mathbf{1}(\mathbf{x}^{(i)} \in c_r)\mathbf{x}^{(i)}$$

In this model, there are two unknowns:

- Cluster centroid
- Membership of points to the clusters

The data points are usually assigned to the nearest clusters based on a chosen distance measure.

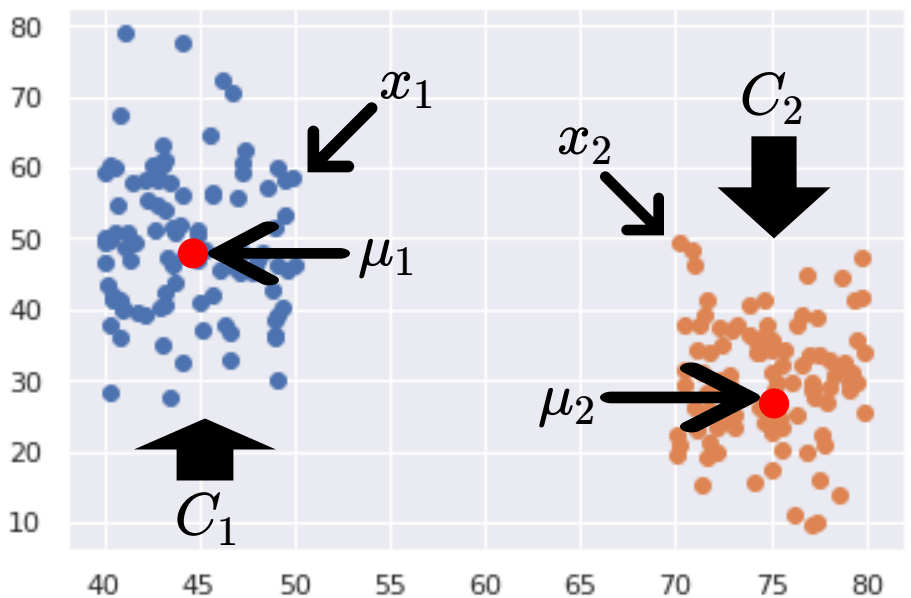
Euclidean distance is one of the commonly used measures in this process.

Euclidean distance between a data point $\mathbf{x}^{(i)}$ and $\mu^{(r)}$ in m dimensions is calculated as

$$d = \sqrt{\sum_{j=1}^{m} \left(\mathbf{x}_j^{(i)} - \mu_j^{(r)} \right)^2}$$

Loss function

$$J(c) = \sum_{r=1}^{k} \sum_{i=1}^{n} \mathbf{1}(\mathbf{x}^{(i)} \in c_r)(||\mathbf{x}^{(i)} - \mu^{(r)}||)^2$$



Optimization

We use k-means algorithm for optimization here.

1. Start off with k initial cluster centers.

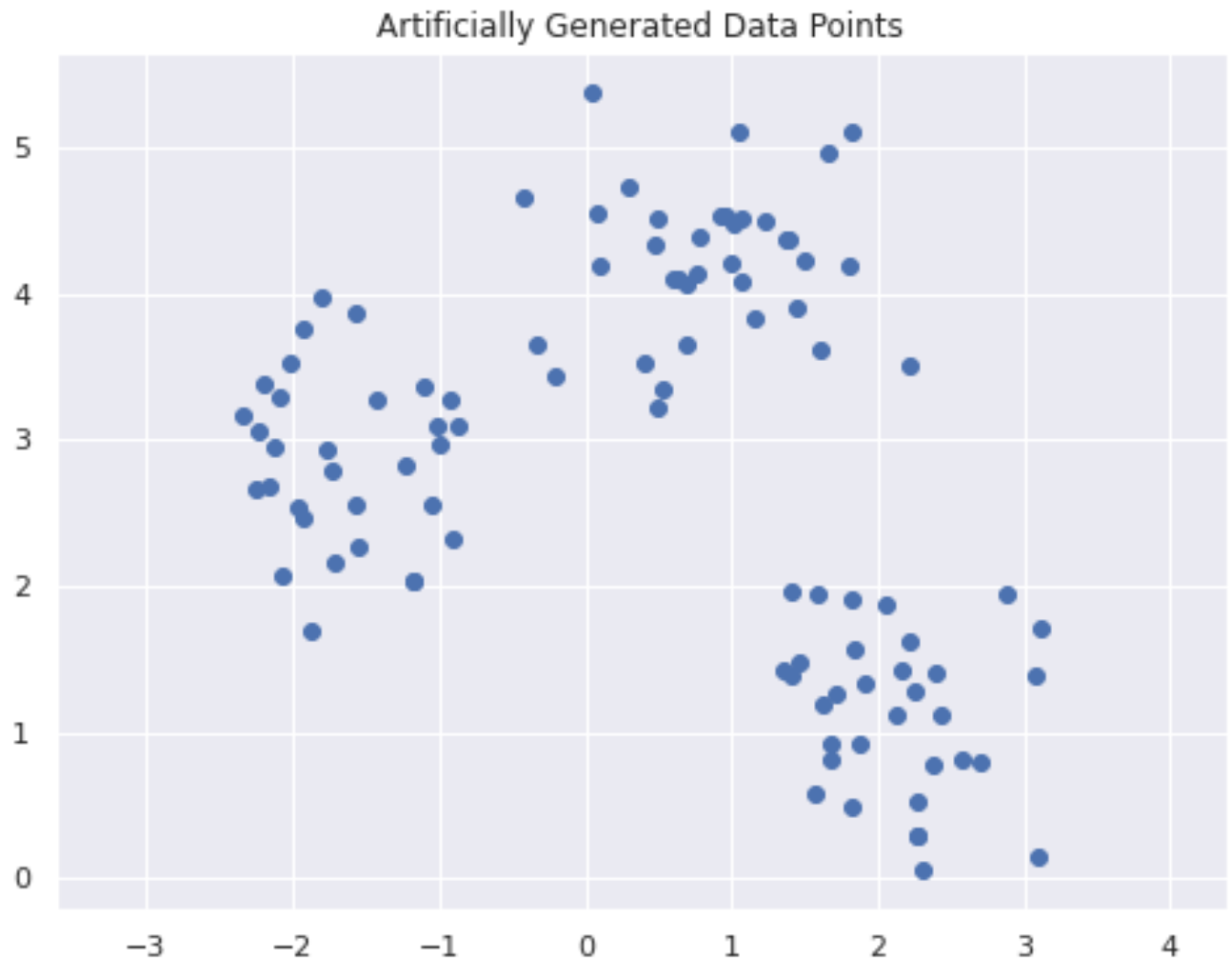
2. Assign each point to the closest cluster center.

3. For each cluster, recompute its center as the average of all its assigned points.

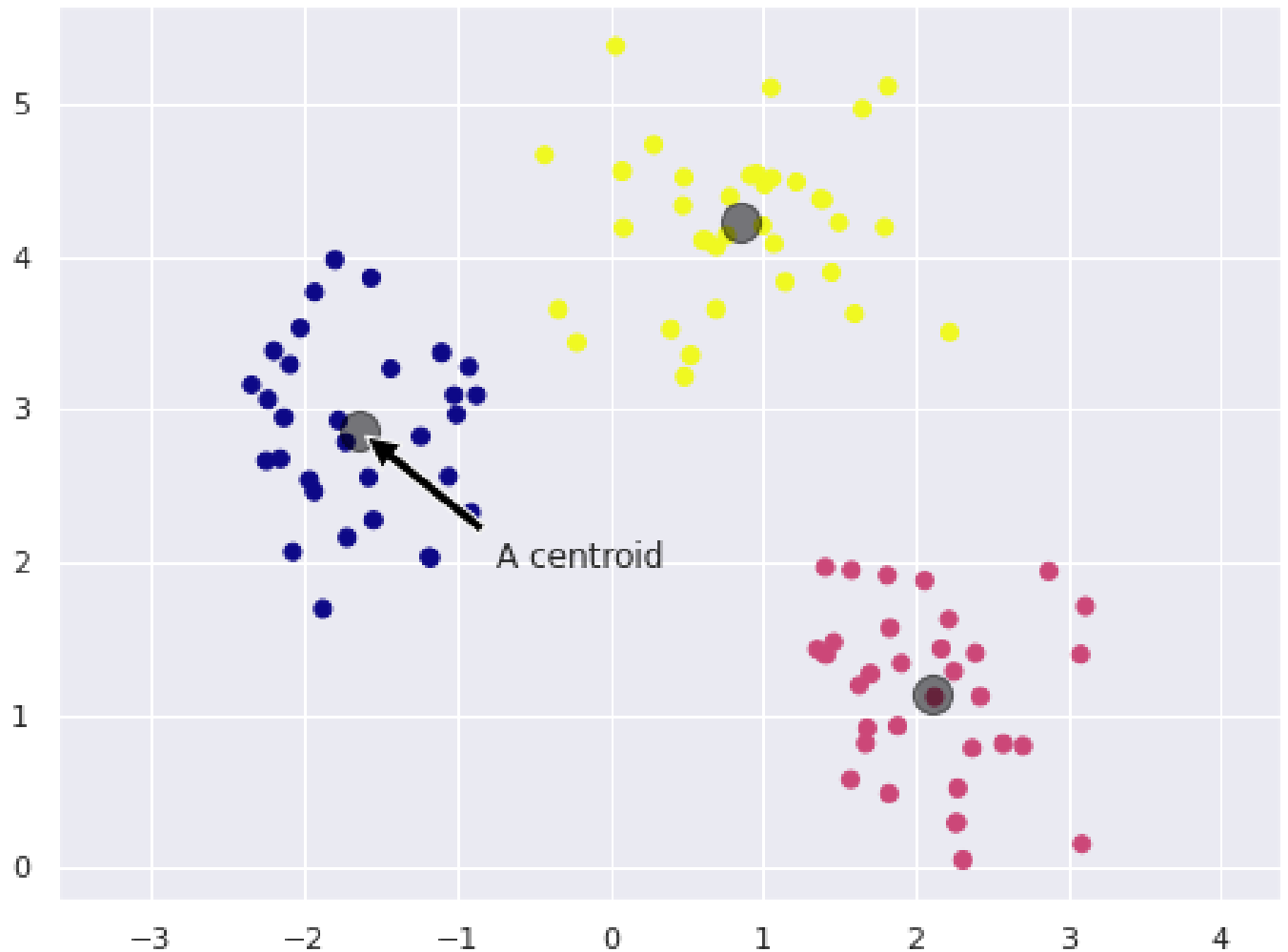
4. Repeat 2 and 3 until centroids don't move or certain number of iterations have been performed.

Data

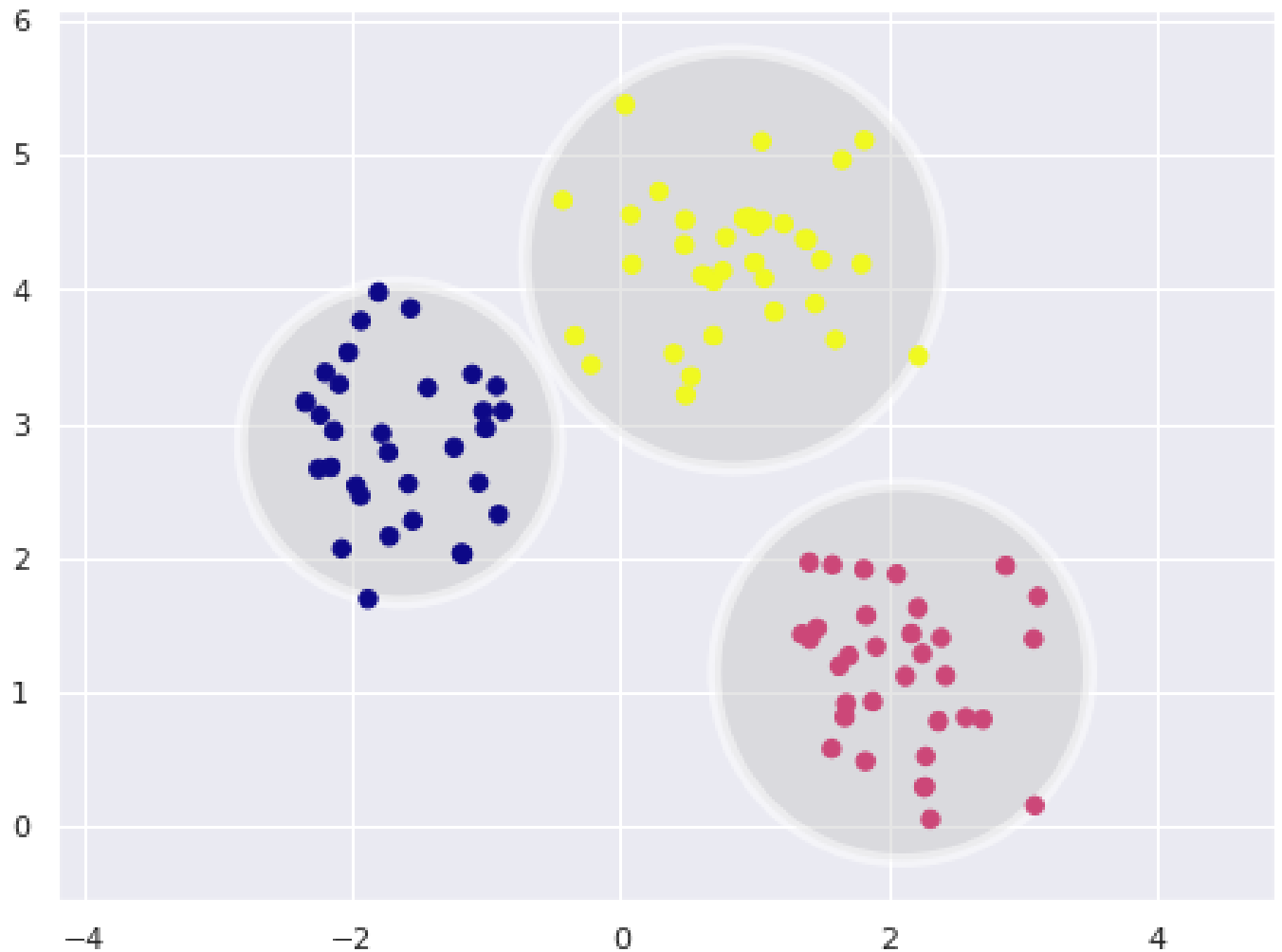
Let's generate points for 3 clusters from sklearn library.



Visualization



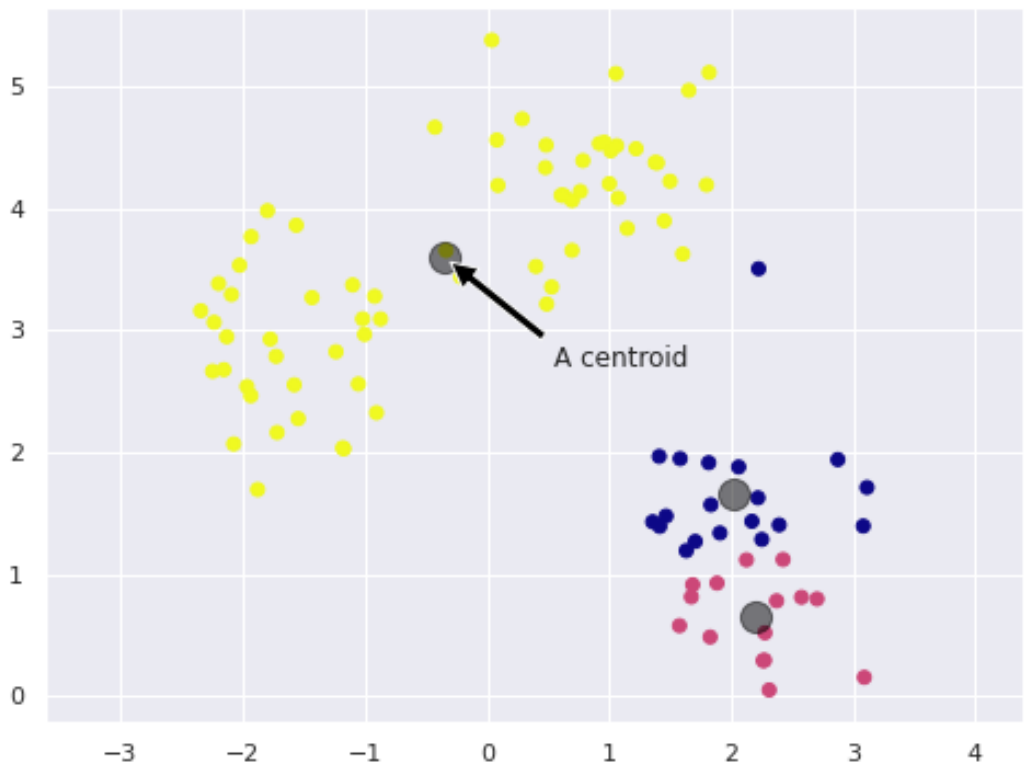
Visualization



Limitations

Local Optima

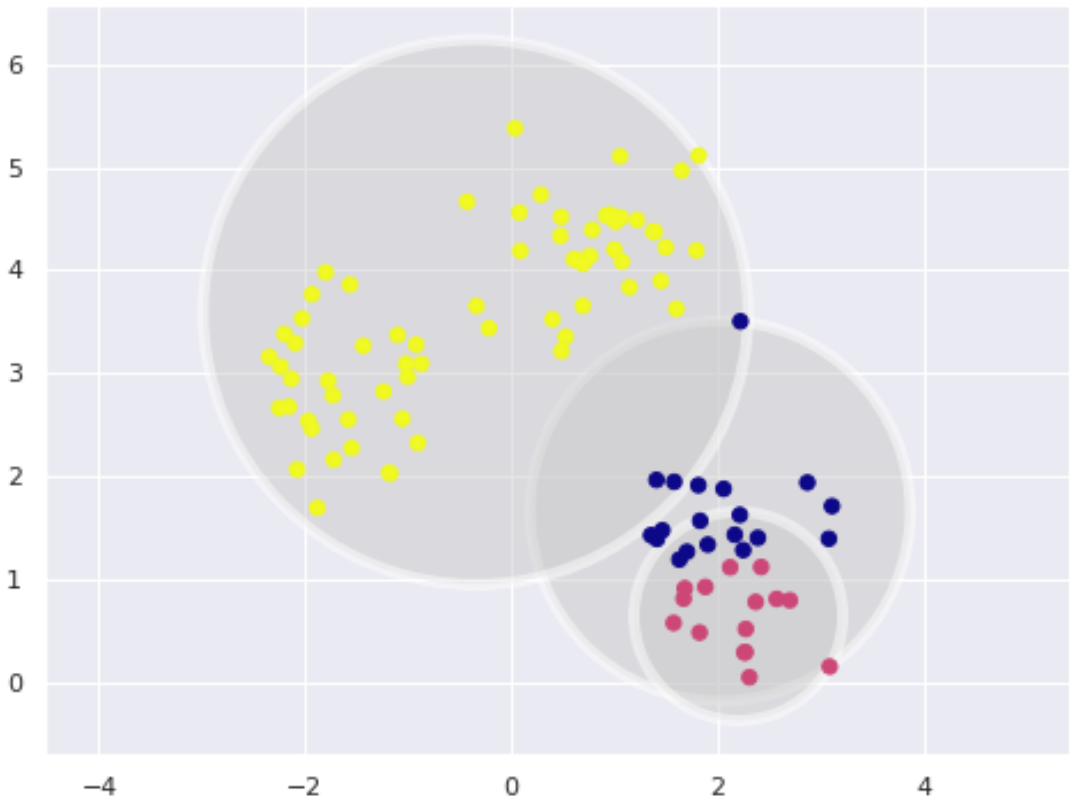
K-means doesn't reach optimal SSE all the times, because the random initialization could cause the algorithm to reach poor clustering.



Limitations

Local Optima

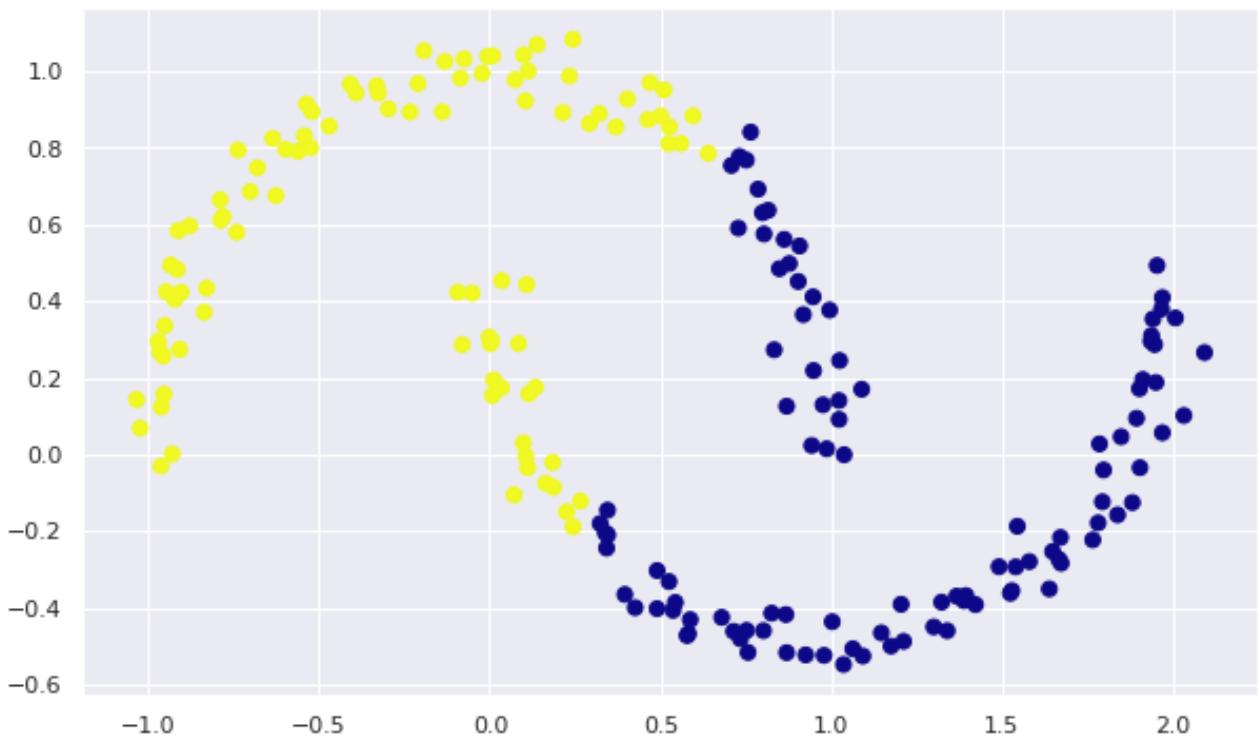
K-means doesn't reach optimal SSE all the times, because the random initialization could cause the algorithm to reach poor clustering.



Limitations

Data is not in spherical shape

Let's look at the following example, where there are two clusters in moon shape. The points in the clusters can not be contained by a sphirical shape.



Limitations

Large datasets

For data sets with large number of data points and features, K-means will be quite slow to converge.

K is unknown at the begining

It is computationally intensive to compute K by elbow method, because it involves running the complete algorithm for many possible values of K.

We evaluate clustering solution by SSE measure
that was defined as a loss function.

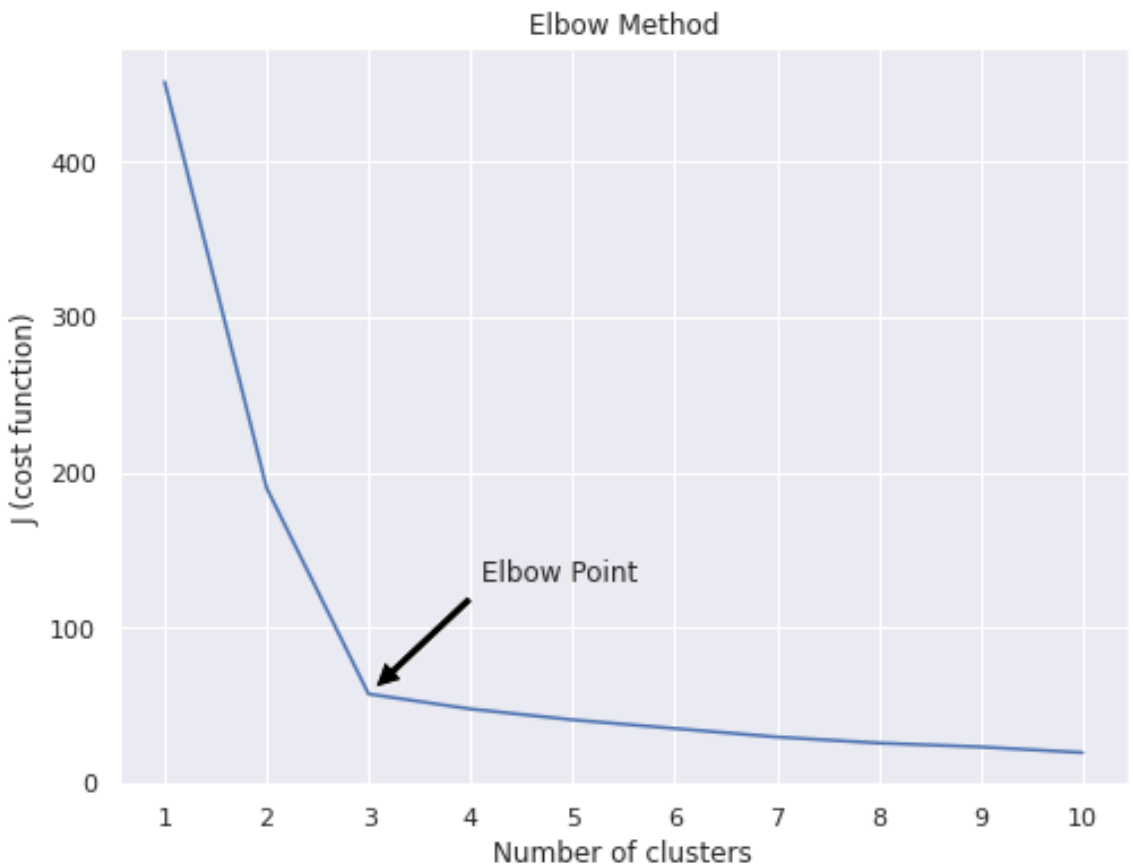
Model selection

How do we find suitable value of k?

- Elbow method
- Silhoutte method

Elbow Method

The Intuition behind elbow method: the cost function does not improve much by increasing K beyond the optimal value.

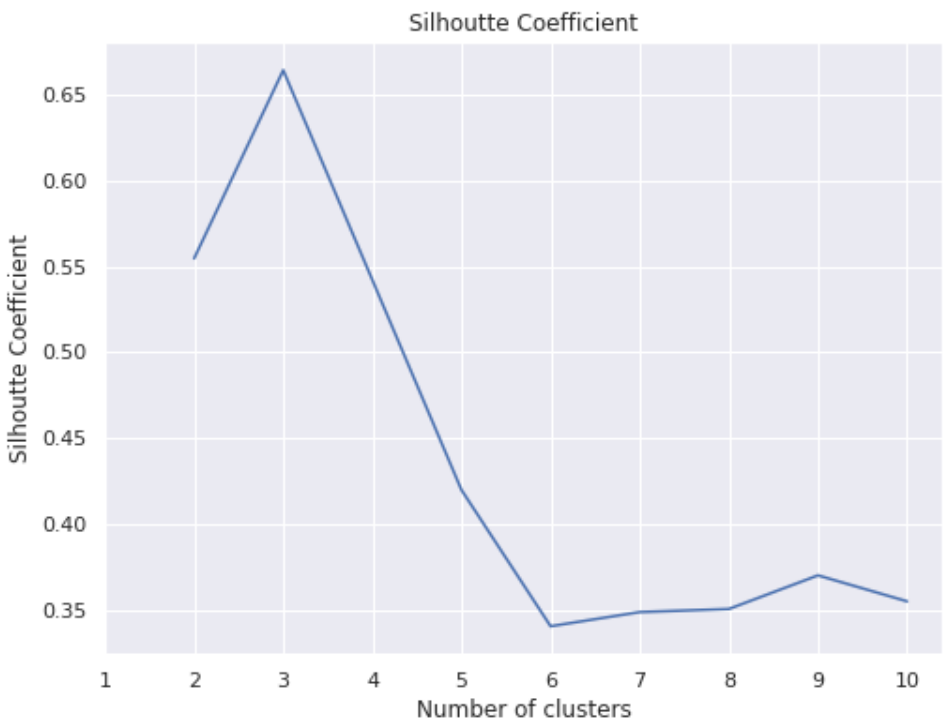


Silhoutte Coefficient

$$s = \frac{ab}{max(a,b)}$$

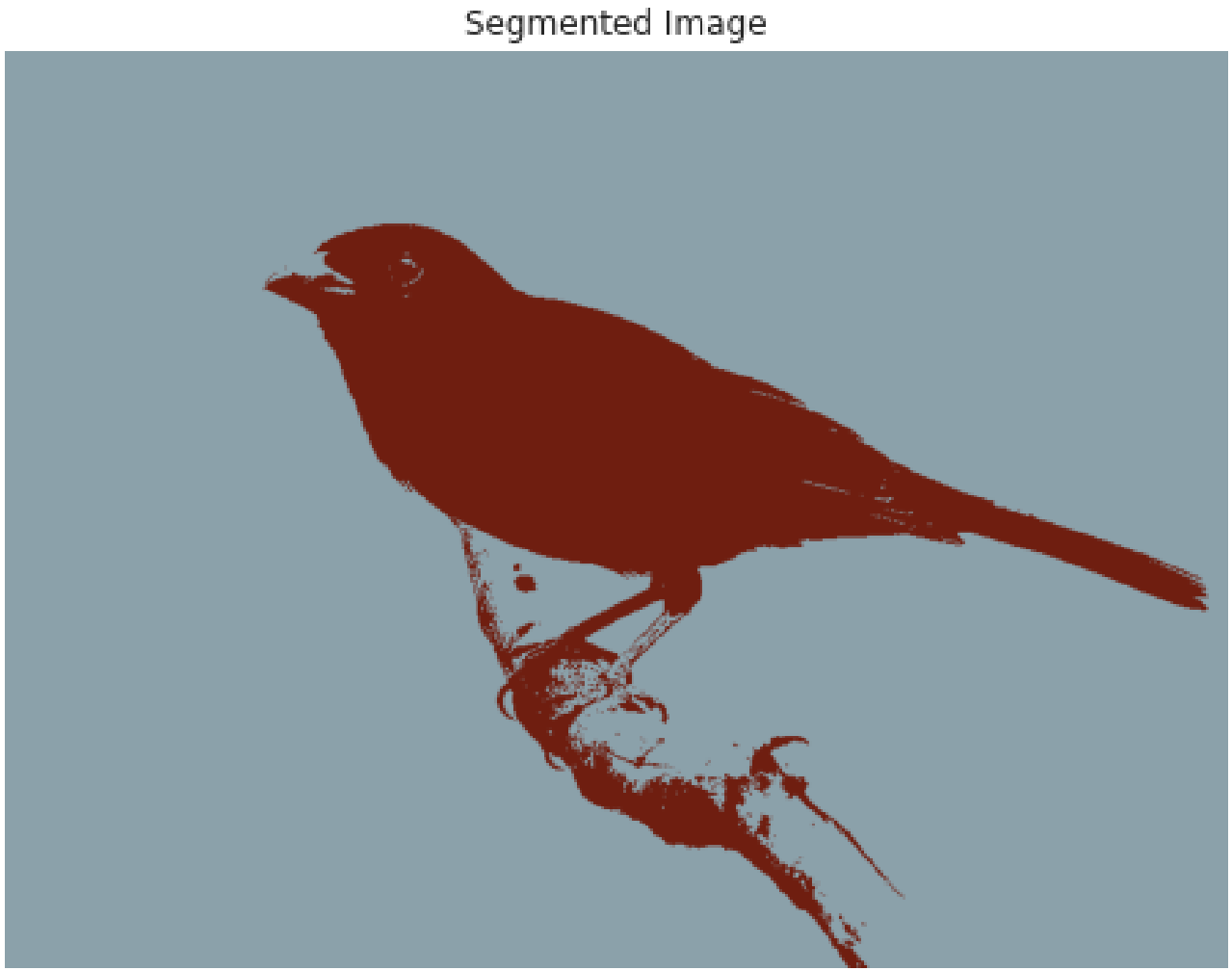
a is the mean distance between the instances in the cluster.

b is the mean distance between the instance and the instances in the next closest cluster.



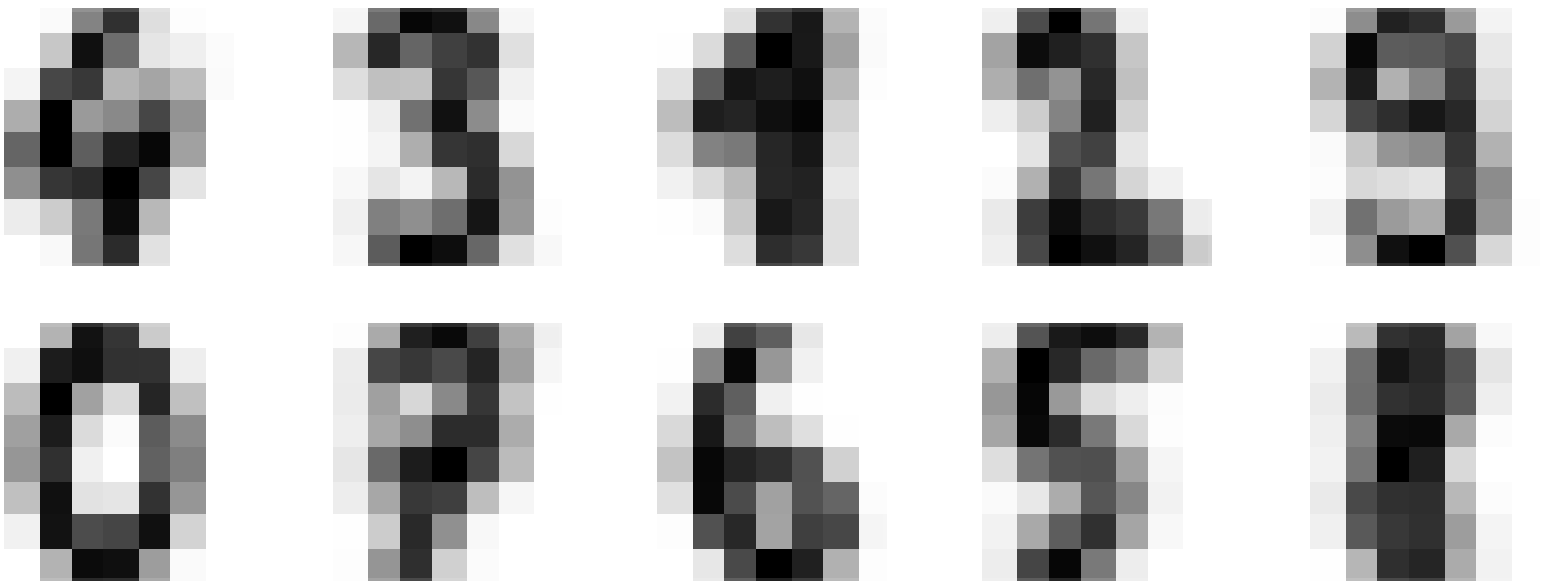
Example 1: Image Segmentation

Example 1: Image Segmentation

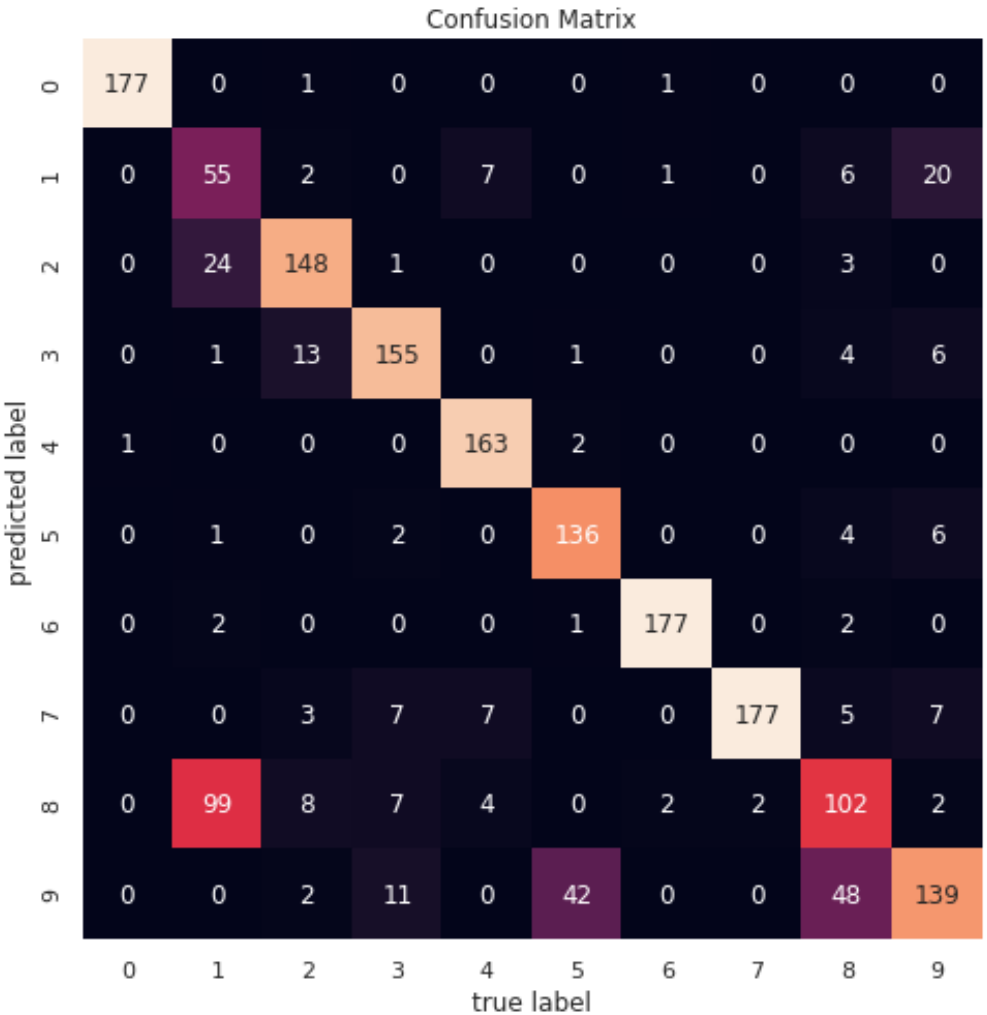


Example 2: Digit Classification

Let us try to classify the digits dataset by K-means clustering.



Example 2: Digit Classification



Clustering is the task of grouping observations so that members of the same group, or cluster, are more similar to each other by some measure than they are to members of other clusters.

K-means is an unsupervised clustering algorithm.

Applications:

- Customer Profiling
- Dimensionality reduction
- Anomaly Detection
- Market segmentation,
- Computer vision (Image segmentation, Image Compression)
- Geo-statistics
- Astronomy