Hand-in 4

Mads-Peter Verner Christiansen, au616397
Zeyuan Tang, au597881
Douwe Tjeerd Schotanus, au600876

December 7, 2018

1 Implementing the Algorithms

A visual comparison of the two algorithms have been carried out on the iris dataset, which is shown in the figure below. Both algorithms find the left most cluster easily, the remaining data is difficult to cluster as is shown by the ground truth. For this data set a visual inspection favours K-means.

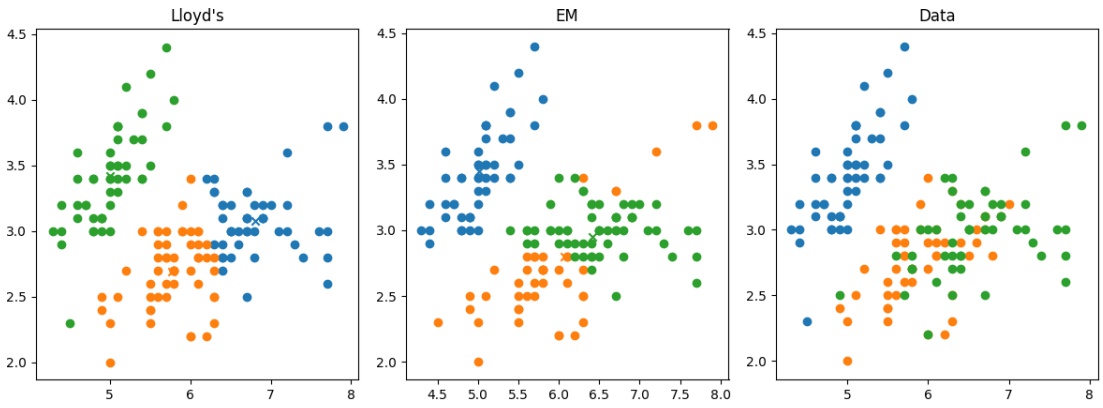

Figure 1: Comparison of Lloyds algorithm and EM-clustering

2 Evaluating clusterings

2.1 Silhouette Coefficient

Differences between Lloyds and EM for calculating Silhouette Coefficient are shown in Table 2. For both algorithms, k=2 gives the best Silhouette Coefficient.

k	Lloyds	EM
2	0.471	0.455
3	0.453	0.276
4	0.425	0.328
5	0.408	0.219
6	0.394	0.157
7	0.413	0.106
8	0.417	0.110
9	0.420	0.055

Table 1: Silhouette Coefficient

2.2 F1 Score

The table below shows the F1-score for the the algorithms at different cluster sizes. For EM the best score is obtained with only 2 clusters, considering the looks like two tilted ellipsoidal clusters. Lloyd's algorithm achieves the best score with three clusters as it needs three cluster centers to separate the green data points on the right in Figure 1 from the orange.

k	EM	Lloyds
2	0.826	0.751
3	0.703	0.812
4	0.626	0.650
5	0.501	0.583
6	0.510	0.531
7	0.397	0.467
8	0.422	0.418
9	0.354	0.391

Table 2: F1 score

2.3 Differences between Silhouette Coefficient and F1 Score

Silhouette Coefficient is an internal/unsupervised measure which can be obtained without labels, while F1 score can only be calculated with labels available.

3 Compressing images

The compression algorithm works by clustering the colors of the image. Clustering with only two clusters will thus turn the image approximately black and white which makes it possible for the jpeg format to compress the size of the image, even if the same amount of pixels are present. Intuitively one can imagine an image format where each pixel only has one color channel and the value corresponds to a predefined color, i.e. 1 = white, 2 = black. Limiting the amount of colors present in the image will thus lead to a higher compression

ratio. When we use four clusters, we get a compression ration of 1.81944 (original size: 127983, compressed size: 70342), with original figure and compressed figure shown below.

Figure 2: Original figure

Figure 3: Compressed figure

4 Sampling from MNIST

The EM-algorithm learns an approximate distribution from the training images, in the form of a mean and a co-variance for each cluster. The training data consists of images represented as 28^2 dimensional vectors, thus the learned Gaussian distributions will be of the same dimensionality. Knowing the mean and the co-variance samples can be generated from the distributions, which is why the EM-algorithm can be used as a generator of images. This is exactly the same as drawing samples from a one-dimensional Gaussian with some mean/standard deviation.