[AIML2015] Homework 3: K-Means, GMM+EM and Clustering Evaluation

Francesco Cucari - 1535743

December 19, 2015

1 Dataset Description

The MNIST dataset is composed of about 70000 images divided into 10 classes of handwrittend digits. The digits have been size-normalized and centered in a fixed-size image. My dataset is composed by 1000 images, 200 per class. The chosen examples belong to classes {0,1,2,3,4}

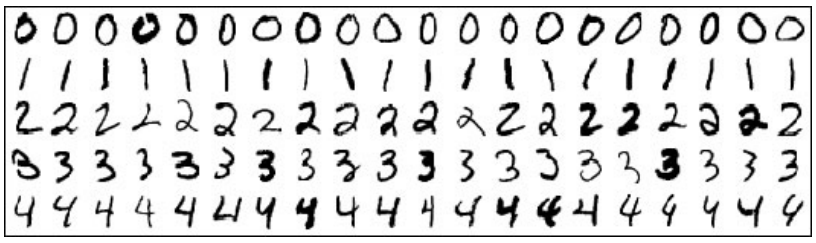
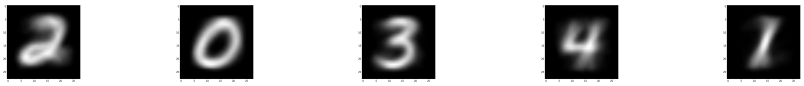


Figure 1: Subset of MNIST Dataset

2 Clustering with K-Means

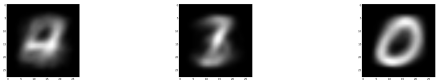
Clustering is the process of examining a collection of "points" and grouping them into "classes" (called *clusters*) according to some distance measure. Clustering is characterized by an high intra-class similarity (points in the same cluster have a small distance from one another) and by a low inter-class similarity (points in different clusters are at a large distance from one another).
K-Means is one of the unsupervised learning algorithms that solve the well known clustering problem. The algorithm classifies a given data set through a certain number (K) of clusters fixed a priori. The main idea is to define k centroids, one for each cluster. It is important that centroids are placed in a proper way because different locations causes different results. So, the better choice is to place them as much as possible far away from each other.

The cluster centroid is the middle of a cluster. It is a vector containing one number for each variable, where each number is the mean of a variable for the observations in that cluster. In order to plot the centroids as image, I transform each vector in a 28x28 matrix. Firstly, by computing the clustering of dataset using K-Means with K=5, the obtained cluster centroids, plotted as images, are the following:



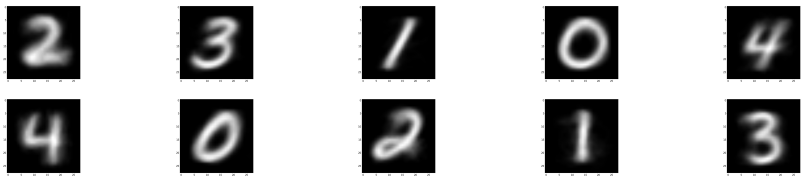
As you can see, the images are well defined and we have a clear image for each centroid.

Then, by computing the clustering of dataset using K-Means with K=3, the obtained cluster centroids are the following:



In this case, you can notice that the number of classes is greater than the number of clusters. As you can see, we have some unclear images. In fact, the most similar classes will result in a fuzzy centroid. There is an overlap of digits (digit 4 and 2 in the first image, and digit 1 and 3 in the second) due to the fact that the algorithm cannot capture the underlying trend of the data.

Finally, by computing the clustering of dataset using K-Means with K=10, the obtained cluster centroids are the following:



In this case, we have some clear images but there are some repetitions for each value, due to fact that the number of cluster is two times than the number of classes. The centroids capture some differences within the examples belonging to the same class of data, as you can see for example in the digits 1 and 2.

3 Clustering with GMM and Performance Evaluation

Purity is an external evaluation criterion of cluster quality. It is the percent of the total number of objects (data points) that were classified correctly. Values of purity are included

in the range {0..1}.

$$Purity(C, \Omega) = \frac{1}{N} \sum_{i=1}^{k} max_{i \in (1,..,J)} \{c_i \cap t_j\}$$

Where $C = \{c_1, .., c_K\}$ is the set of cluster, $\Omega = \{\omega_1, .., \omega_J\}$ is the set of classes (J=5), N is the number of objects (1000 in our case), K is the number of clusters (it varies in {2,3,..,10}), c_i is a cluster in C and t_j is the classification which has the max count for cluster c_i.

Bad clustering has purity values close to 0, a perfect clustering has a purity of 1.

The task of homework is to evaluate the clusters with purity measure, varying the number of clusters k in the range 2,3,..,10. I consider 200 examples for the 5 indicated classes. The results are the following:

	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9	k=10
Purity	27.7%	30.8%	48%	63.5%	65.5%	64.3%	61.8%	66.7%	74%

Fig.2 is the plot of the numbers of cluster againts the purity. You can notice that the purity increases with the number of classes. High purity is easy to achieve when the number of clusters is large. In particular, the highest growth of purity is when the number of clusters is equal to the number of classes, that is k=5.

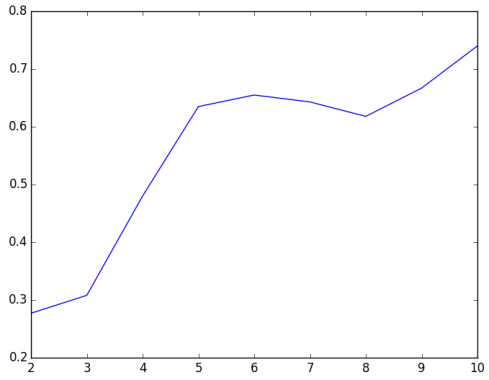


Figure 2: Plot of number of clusters against the purity

4 Classifying with GMM/EM

For each class a GMM is fitted by varying the number of the components between 2 and 5. By doing so, for each class I obtain the vector of "logprob", which represents the Log

probabilities of each data point in the dataset. Then, I construct a matrix combining these vector for each point and I extract the maximum value in correspondence with an index representing the label predicted by the classifier. So, I compare the original y_test and the vector of label predicted by the classifier. Finally, I can count the misclassification and I can compute the accuracy of the classifier using the testing set.

These are the results:

N°of components	Accuracy
2	77.8%
3	78.9%
4	89.6%
5	89.1%

4.1 Optional

Then, I repeat the previous procedure but now I use the validation set to find the number of components (the best k). The validation set is derived from the splitting of the training set. After the validation the best number of components k is 4 and the accuracy is:

Accuracy	90%

4.2 Optional

Non-Linear SVM performs better than GMM. In fact, SVM is a supervised training method and it requires less patterns to estimate the model. While, GMM is a non-supervised training algorithm that requires a larger number of training patterns to achieve a good estimation and its convergence is slower than those of SVM.

	Accuracy
SVM	91.1%
GMM	90%