02450 Project 2

Report

by Karol Dzitkowski
Marco Becattini

Technical University of Denmark
Department of Applied Mathematics and Computer Science
Introduction to Machine Learning and Data Mining
Tue Herlau
28th November 2014

Abstract

The objective of this second report is to apply the methods you have learned in the second section of the course on „Supervised learning: Classification and regression" in order to solve both a relevant classification and regression problem for your data.

Clustering

We clustered our data containing letters with a set of their attributes using Gaussian Mixture Model as well as with Hierarchical clustering. We used cross-validation to estimate the optimal parameters for both algorithms. At the end we estimated the quality of both algorithms in terms of our true labels which are letters.

0.1 Gaussian Mixture Model (GMM)

For GMM algorithm we used double cross-validation to estimate the best number of components in internal loop and calculating quality score in external loop as an average of scores. We also applied Principal Component Analysis to visualize the results in more human friendly way. We used only a subset of our dataset for performance reasons, using random sampling. We can see the results in the figure **??** In an average the most optimal number of components

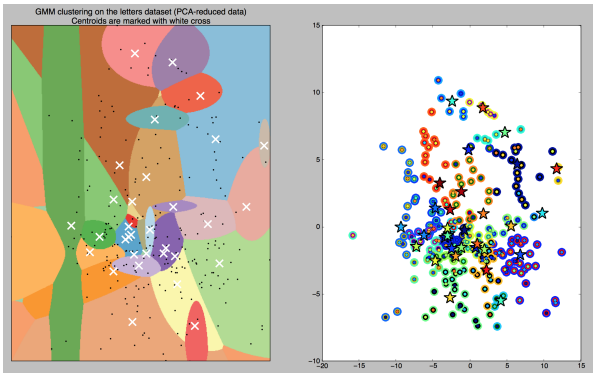

Figure 0.1: GMM - clustered (data reduced using PCA)

choosed in internal cross-validation loop was 30. Which is a close to the number of actual classes. We can see how the quality of GMM clustering in terms of actual labels depends on the number of components in the figure **??** Cluster centers should be in ideal case (especially if we consider number of components equal to the number of actual classes - 26) an average vector

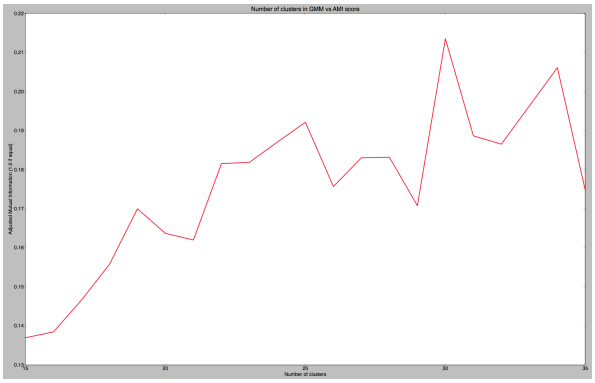

Figure 0.2: GMM - quality vs components count

from the attribute vectors corresponding to particular letters. So the centroids should correspond to some „unified" letters.

0.2 Hierarchical Clustering

We also used single cross-validation technique in Hierarchical Clustering and choose the best metric and linkage function from all available options. Score results are mesured for best performing option sets. We also used PCA to help visualize results (Figure **??** and Figure **??**). Even using first two principal

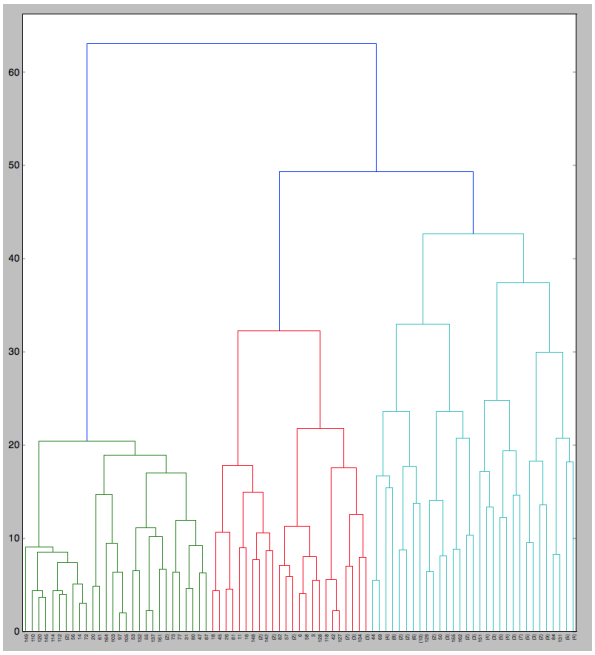

Figure 0.3: Hierarchical clustering (without PCA reduction)

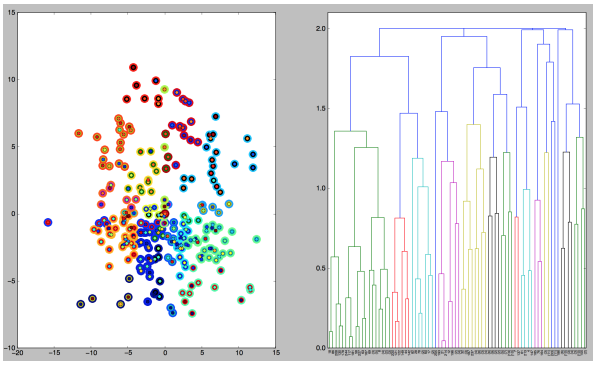

Figure 0.4: Hierarchical clustering (data reduced using PCA)

components explaining the most variance, results are dyrastically worse than when we use all data dimensions. The most commonly choosed options by the program were ('ward', 'euclidean') which means that the ward linkage function and euclidean metric were most suitable for our data.

0.3 Evaluation

We choosed an adjusted mutual info score metric as our quality messure for clustering methods. For two clusterings U and V, the AMI is given as:

$$AMI(U,V) = [MI(U,V) - E(MI(U,V))]/[max(H(U),H(V)) - E(MI(U,V))]$$

And has following properties:

1. Perfect labeling is scored 1.0
2. Independent labelings have non-positive scores
3. No assumption is made on the cluster structure
4. It is symmetric
5. It is independent of the absolute values of the labels: a permutation of the class or cluster label values won't change the score value in any way
6. It accounts for the fact that the MI is generally higher for two clusterings with a larger number of clusters, regardless of whether there is actually more information shared

Reaserch has shown that Hierarchical clustering performs better for our data than GMM and also that both methods have better scores while having more data. However it was impossible for performance reasons to run this methods on more than 3000 records of our data, where results would be even better. It is visualized in the figure **??**

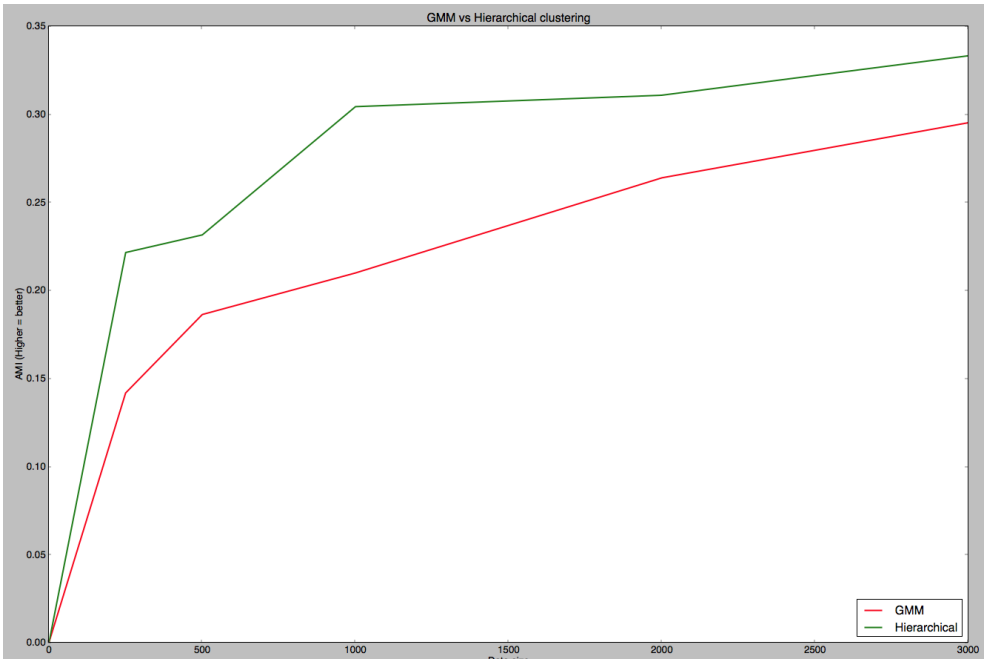

Figure 0.5: GMM vs Hierarchical clustering quality

Association Mining

Association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. We divide values of features into three categories: small values, medium values and high values and than we are going to do association mining for this three categories separately. So we have three different dataset and attributes are in set when they belong to that category. For instance if we consider the item set of smal values (from 1 to 4) we binarize our data by transforming the value of each attribute to 1 if it belong to the interval [1-4], and 0 otherwise. The medium values are defined from 5 to 7 while high values are defined from 8 to 16 and we binarized the data in the same way described above. For each item set we apply Apriori algorithm in order to find possible associations among the attributes.

The table below shows the best association rules found by the algorithm including its confidence percentage.

RULE	CONFIDENCE
Assiociations for HIGH values of attributes (from 8 to 16)	
xegvy → xy2br	62.96 %
xy2br → xegvy	74.62 %
yegvx → xy2br	72.00 %
xy2br → yegvx	72.64 %
yegvx → xegvy	76.39 %
xegvy → yegvx	65.03 %
Assiociations for MEDIUM values of attributes (from 4 to 7)	
x2ybr → width	67.53 %
width → x2ybr	60.37 %
width → x-box	63.67 %
x-box → width	82.51 %
high → width	80.93 %
width → high	69.55%
Assiociations for SMALL values of attributes (from 1 to 4)	
x-ege → onpix	76.01 %
onpix → x-ege	72.54 %
y-ege → onpix	78.07 %
onpix → y-ege	60.95 %
onpix → x-box	76.15 %
x-box → onpix	81.39 %
x-ege → x-box	68.88 %
x-box → x-ege	70.25 %

In the first report we computed correlation between attributes and we found out that the strongest correlation was between x-box and width (corr=0.85). Thus the results of Apriori confirms the findings from the our previous analysis ,in fact, the association for medium values of attributes x-box → width has the highest confidece.