Machine Learning hand-in 4

This handin is about implementing and using representative-based clustering algorithms.

The handin is mandatory, and should be done groups of 2-3 students. Each group must prepare a report in PDF format as outlined below. Please submit all your Python files in a zip file, and your PDF report outside the zip file, to BlackBoard no later than Monday, December 14 at 9:00 AM.

All data and Python files are available below.

For questions and issues regarding this hand-in, please use the course discussion forum. If you have problems that for some reason cannot be shared on the discussion forum, contact the teaching assistant, Mathias Rav, either by coming to his office, Nygaard 334, or by sending an email to rav@cs.au.dk.

Data set

The Iris data set is included in SciKit-Learn as the load_iris function in the sklearn.datasets module. The file load_and_show_iris.py provides two functions, load_iris and load_iris_pca, which load the 4-dimensional Iris data set from SK-Learn and optionally applies principal component analysis (PCA) to produce a 2-dimensional data set.
In addition, you are provided with two images of AU buildings for the image compression exercise.

Files

two PNG image files AU_main_back_small, DAIMI_AU_small
load_and_show_iris.py
plotmatrix.py
kmeans.py
em.py
f1.py
silhouette.py
run.py
image_compression.py

Tasks

Implementing EM and the K-means algorithm

You must implement the K-means algorithm and the Gaussian Mixture Expectation Maximization algorithm as discussed on p. 349 and p. 335 of the textbook [ZM]. Your implementations may assume that k=3 if that makes it easier. For the EM algorithm, use the multivariate_normal class from scipy.stats. Read the documentation at http://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.multivariate_normal.html. Specifically, you should use the pdf method of the multivariate_normal class to compute the probability density of each data point.

You are provided with Python code for loading and displaying data. The IRIS data that was discussed at the lecture is available, both in a 2d version, load_iris_pca, based on the principle components (PCA), and in the original 4d version, load_iris. There are functions for displaying the PCA Iris data in a 2d plot and the 4d version as a plot matrix.

Use the 2d Iris data to validate your algorithms (compare the results you get with the results in the textbook on the same data), and run your algorithms on the 4d data and compare.

Cost function for the K-means problem

To ensure that your implementation of the K-means algorithm is correct, you should implement the cost function (objective function / optimization goal) for the K-means problem. Print the cost after each iteration of the K-means algorithm -- the cost should be decreasing.

Evaluating clusterings

Implement the F1 score (build the contingency table p.426, measure p. 427-428) and the silhouette coefficient (p. 444-445), and compare the quality of several runs of your algorithms with different values for k.

Initializing EM

In order to determine an initial set of cluster centers for EM, one can utilize the best centroids determined by k-means. For this, run k-means several times, and choose the best one. You are asked to do this for the 2d and 4d IRIS data set and to compare with your earlier results.

Image compression

You are provided with two images that you are asked to subject to clustering in order to find the most representative colors for the two images. Use these results to display a compressed version of the images. You can relate to the comments in the Python files to see how to go about this.

Report

Your report should be no more than 3 pages and clearly state who is in the group. It must cover:

- The status of the work, i.e., does it work, if not, then why.
- A discussion of plots of at least two runs of your algorithm implementations detailing what you can see. Make sure that you relate this to the discussion in the lecture or textbook about the strengths and weaknesses of the algorithms.
- A discussion of plots of the evaluation measures F1 and silhouette coefficient, detailing what you can learn from them. Include an explanation of what the evaluation measures reflect.
- Describe how you can use one of the clustering algorithms for image compression, and demonstrate the results for at least one algorithm on both images, discussing their quality and giving a reasoning for the differences.