Data mining (TIES445),
Homework 5.12.2007

Perform the following tasks using MATLAB (Octave is ok too!) and write a short report (**use a lot of illustrative figures**). Attach the critical parts of the codes and macros. Deadline for the report is 17.12.2007.

1. Construct a more robust k-medians method (in this case the median refers to the typical coordinate-wise median) by modifying the k-means algorithm (dckmeans). You can also use a spatial median function from the LIBRA package (l1median). Do multiple runs on K6data.csv using both the k-medians and k-means method. The data set in file K6file.csv contains six clusters that are generated from symmetrical bivariate normal distributions with centers at {(-2.2),(1,2),(4,2),(-2,-1),(1,-1),(4,-1)}. Compare and explain the results, for example:
   a. Plot the data so that each cluster is represented with own color and marker (help plot). The clusters can be found by clustering....
   b. Run K-means for several Ks (e.g., 1:10) and plot the SSE curve. Can you find a “correct” number of clusters. Do the same with K-medians.
   c. Perform several runs and compute the errors to the optimal solution. You need to write a function that gives the minimum value (total distance) of the one-to-one mapping between the obtained prototypes and generating distribution centers. Attach also self-implemented parts of code.
   d. Evaluate the dispersion of prototypes using multiple runs (e.g., 100) from different initial points (attach the plots of the obtained prototypes for the both methods to the report). Comment the variance and uniqueness of the prototypes.
2. Do the same as in task 1., but load the data from the file K6dataN.csv. Report the results as in the previous task. If the performance of the two clustering methods differs, try to explain why.
3. Choose at least two clustering method (you can also search these from web and even modify and improve them, but in this case attach the code to the report). Load the Iris data. Compare the classification performance of the methods (external validation) using `total_entropy` and `total_purity` functions (they requires functions `cluster_entropy` and `cluster_purity`). You may need to run the algorithm several times to obtain the best results or use some initialization (e.g., sampling). Explain the results.