Homework Assignment #7
Due: 04/27/2010 (11:55pm, through Oncourse)

Question 1 (40 points)
From class web site (week 14) download the data package and run Apriori program for several (at least 3) different settings of parameters (min_support, correlation_hi, correlation_lo, max_level). These parameters represent: minimum support for a rule, lower and upper thresholds for the correlation measure that describes interestingness of the rule and the maximum number of levels the algorithm should expand to when searching for frequent itemsets. Remember that correlation measure is 1 for the least interesting rules and the closer it is to 0 or the more it is above 1, the more interesting the rule is (negative and positive rules, respectively). The only function in which you should make changes is apriori.m, lines 5-8, which is the main function you should run from Matlab’s command line. Describe what parameters you used, why you used them and how the algorithm behaved for these parameters. Report on three rules (from the census data) that you find interesting and also three rules that you find to be obvious.

Question 2 (30 points)
Write function distance_matrix.m that for a given set of data points $X$ finds Euclidean distances between every pair of points.

```matlab
function [DM] = distance_matrix (X)
```

Assume that data points in $X$ are represented by the rows of that matrix (there is no class label or target value in the last column). Note that you can build this function from your code knn.m with minor modifications. Example output you can use to test your code:

```matlab
X = [1 2 3 4; 5 6 7 8; 9 10 11 12];

distance_matrix(X)
```

```matlab
ans =

    0     8    16
    8     0     8
   16     8     0
```

In the above example, $X$ was a data set of three 4-dimensional data points. Thus, the distance matrix is 3-by-3 and it is also symmetric with all main diagonal elements being 0.
Extra Points: (30 points)
Write function centroid.m that for a given set of data points $X$ finds a centroid for those data points.

\[
\text{function } [c] = \text{centroid} \ (X)
\]

Assume that data points in $X$ are represented by the rows of that matrix (again, no class label or target column). See the class code for z-score normalization as a guide how to calculate the centroid. Example output you can use to test your code:

\[
X = \begin{bmatrix}
1 & 2 & 3 & 1; \\
5 & 6 & 7 & 12; \\
9 & 10 & 11 & 5
\end{bmatrix};
\]

\[
\text{centroid}(X)
\]

\[
\text{ans} = \\
5 \quad 6 \quad 7 \quad 6
\]

In the above example, $X$ was again a data set of three 4-dimensional data points. Centroid over the three data points is the mean value over every column (e.g. mean of 1, 12, and 5 is 6 etc.).

What to Submit? A text file (.doc or similar) responding to Q1, as well as two Matlab files, named distance_matrix.m and centroid.m responding to Q2 and Extra Credit Question, respectively.

Homework Policy
Homework assignments are to be submitted through Oncourse on or before the specified due date. Unless there are legitimate circumstances, late assignments will be accepted up to 5 days after the due date and graded using the following rule:

- on time: your score $\times 1$
- up to 1 day late: your score $\times 0.9$
- up to 2 days late: your score $\times 0.7$
- up to 3 days late: your score $\times 0.5$
- up to 4 days late: your score $\times 0.3$
- up to 5 days late: your score $\times 0.1$

For example, this means that if you submit 3 days late and get 80 points for your answers, your total number of points will be $80 \times 0.5 = 40$ points (that is, 50% of the credit you receive for your answers).

All assignments are individual, except when collaboration is explicitly allowed. All the sources used for problem solution must be acknowledged, e.g. web sites, books, research papers, personal communication with people, etc. Academic honesty is taken seriously; for detailed information see Indiana University Code of Student Rights, Responsibilities, and Conduct.

Good Luck!