# Methodologies for Intelligent Systems
# (Homework 15-06-2010)

*Matteo Matteucci, Davide Eynard, Simone Tognetti*

***Note:*** homework should be turned in by 05/07/2010 as a digital document (you can scan your handwritten assignment into a pdf) through email. Please, not to flood my mailbox, send me the link to the document and I will acknowledge its receipt to you. For ***any*** doubt on the text ask by email the teachers!

## Exercise 1: Classifiers & Co. (6 Points)

You are the manufacturer of the world's finest widgets and you are currently looking for a way to focus your marketing efforts to more closely entice your target demographic. Here is a table containing a sampling of the information you collected during an email survey:

| ***Age*** | ***Education*** | ***Income*** | ***Marital Status*** | ***Purchase?*** |
|---|---|---|---|---|
| 36-55 | master's | high | single | will buy |
| 18-35 | high school | low | single | won't buy |
| 36-55 | master's | low | single | will buy |
| 18-35 | bachelor's | high | single | won't buy |
| < 18 | high school | low | single | will buy |
| 18-35 | bachelor's | high | married | won't buy |
| 36-55 | bachelor's | low | married | won't buy |
| > 55 | bachelor's | high | single | will buy |
| 36-55 | master's | low | married | won't buy |
| > 55 | master's | low | married | will buy |
| 36-55 | master's | high | single | will buy |
| > 55 | master's | high | single | will buy |
| < 18 | high school | high | single | won't buy |
| 36-55 | master's | low | single | will buy |
| 36-55 | high school | low | single | will buy |
| < 18 | high school | low | married | will buy |
| 18-35 | bachelor's | high | married | won't buy |
| > 55 | high school | high | married | will buy |
| > 55 | bachelor's | low | single | will buy |
| 36-55 | high school | high | married | won't buy |

*Table 1: The survey dataset*

By using the results of your survey you now have to:

- Compute a full decision tree out of the dataset of Table 1
- Compute a classifier for the *will buy* class using the sequential covering algorithm
- Compute a Naïve Bayes Classifier out the same data (use m-estimates if needed with your preferred m)
- Classify the record: {19,high-school,high,???} with the previous classifiers

## Exercise 2: Markov Chains (4 Points)

Snakes and Ladders, sometimes called `Chutes and Ladders', is a classic children's board game played between 2 or more players on a playing board with numbered grid squares. On certain squares on the grid are drawn a number of "ladders" connecting two squares together, and a number of "snakes" also connecting squares together.

Each player starts with a token in the starting square and takes turns to roll a single die to move the token by the number of squares indicated by the die roll, following a fixed route marked on the gameboard which usually goes from the bottom to the top of the playing area, passing once through every square.

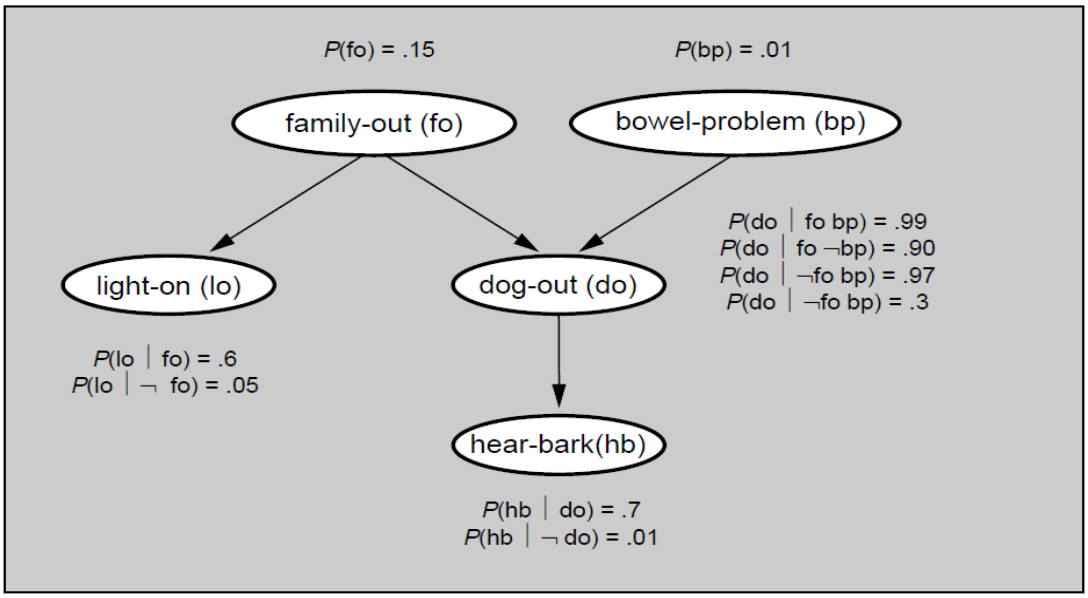
If, on completion of this move, they land on the lower-numbered end of the squares with a "ladder", they can move their token up to the higher-numbered square (known as "climbing the ladder"). If they land on the higher-numbered square of a pair with a "snake'', they must move their token down to the lower-numbered square (known as 'sliding down the snake'').


- Model the Snakes and Ladders game for a single player starting in cell ``14'' as described so far (i.e., write down the transition matrix). Notes are written on the board that affect the movement and should be taken into account. Are all the 25 states of the game reachable? What is the exact number of states for the game (starting at cell 1)?
- A variation exists where a player must roll the exact number to reach the final square (hence winning). Depending on the particular variation, if the roll of the die is too large the token remains where it is, or the token may proceed to the final square and then go backwards until it has transversed the same number of squares as the die shows. How these two variations change your transition model?
- Is this game an ergothic markov chain? Is it possible to give an estimate duration of a game?
- In the real version, a player who rolls a 6 with the die may, after moving, immediately take another turn; otherwise, the play passes to the next player in turn. If a player rolls three 6's on the die, he returns to the beginning and may not move until he rolls another 6. Is this still a markov chain?

## Exercise 3: Bayesian Networks (4 Points)

Suppose when I go home at night, I want to know if my family is home
before I try the doors. (Perhaps the most convenient door to enter is double locked when nobody is home.) Now, often when my wife leaves the house, she turns on an outdoor

light. However, she sometimes turns on this light if she is expecting a guest. Also, we have a dog. When nobody is home, the dog is put in the back yard. The same is true if the dog has bowel troubles. Finally, if the dog is in the backyard, I will probably hear her barking (or what I think is her barking), but sometimes I can be confused by other dogs barking.




A Bayesian Network for the family-out Problem.

- Describe independences and conditional independences in the network by using d-separation
- Compute the probability distribution of **light-on** given that I **hear bark**
- Compute the probability of **dog-out** given the **light-on**
- Compute previous probabilities by using both standard Monte Carlo simulation amd likelihood weighting (use provided random numbers).

***Random Numbers***

0.7937, 0.9992, 0.1102, 0.6226, 0.1326, 0.3100, 0.1348, 0.2233, 0.3965, 0.1351, 0.2411, 0.9275, 0.3911, 0.5113, 0.0929, 0.0217, 0.1595, 0.8445, 0.8792, 0.1870, 0.9913, 0.7120, 0.8714, 0.4796, 0.4960, 0.2875, 0.0609, 0.2625, 0.1863, 0.9171, 0.4869, 0.8175, 0.6416, 0.3063, 0.6609, 0.3580, 0.9382, 0.4877, 0.0910, 0.6738, 0.5149, 0.2216, 0.7250, 0.0682, 0.9641, 0.2077, 0.1611, 0.6382, 0.0002, 0.3356, 0.2751, 0.0445, 0.0939, 0.4100, 0.8169, 0.8705, 0.0226, 0.7272, 0.8480, 0.7286, ...

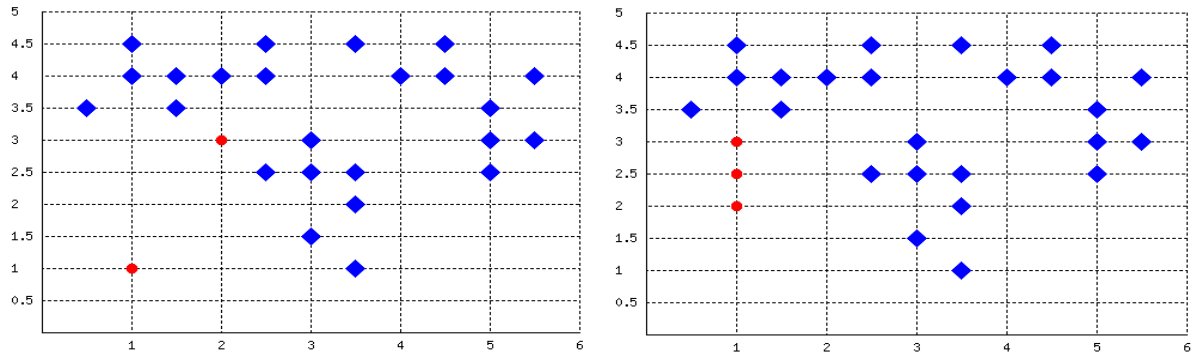
## Exercise 4: Chi-square test of independence (2 Points)

Consider the following hypothetical experiment: a drug that decreases anxiety is given to one group of subjects before they attempted to play a game of chess against a computer. The control group was given a placebo. The contingency table is shown below; is the effect of the drug significant?

| Condition | Win | Lose | Draw | Total |
|---|---|---|---|---|
| Drug | 12 (14.29) | 18 (14.29) | 10 (11.43) | 40 |
| Placebo | 13 (10.71) | 7 (10.71) | 10 (8.57) | 30 |
| Total | 25 | 25 | 20 | 70 |

## Exercise 5: Clustering (8 Points)

Given the two starting positions shown in figure (where blue dots are data points and red ones the centroids starting points)

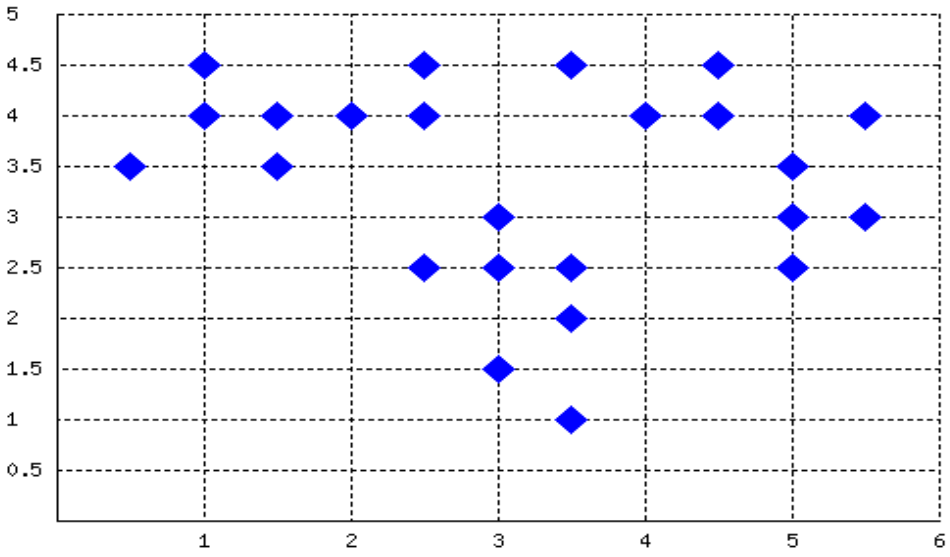



calculate and show the different steps of the K-Means algorithm for both of the cases in the following way:

- at each step, specify the initial positions of the centroids
- without actually calculating it (unless it is needed to verify distances you cannot tell apart at a glance - if you are not sure do the calculation!), for each step specify which centroid the various dataset points belong to
- after you have assigned points to the different centroids, calculate their new positions and proceed to next step.

Tell how many iterations the algorithm needs to converge in the different cases and answer the following questions:

- does the 2 centroids execution perform better than the 3 centroids one?
- which execution ends with the lower number of iteration? Is the number of centroids related to the number of iterations?
- finally, starting from the same dataset, execute the steps of a hierarchical (agglomerative) algorithm using the single linkage technique, showing the new links you create at each step of the algorithm (labeling them with a number) and stopping when you obtain two clusters. How is the result you obtain, compared with the two-clusters k-means? Comment the results motivating your answer.



## Exercise 5: Features Projection and Crossvalidation (8 Points)

We want to compare the performance (classification accuracy) of a classifier when it is applied, on a given dataset, after two feature projection algorithms: Fisher's Linear Discriminant Analysis(LDA) and Principal Component Analysis (PCA).

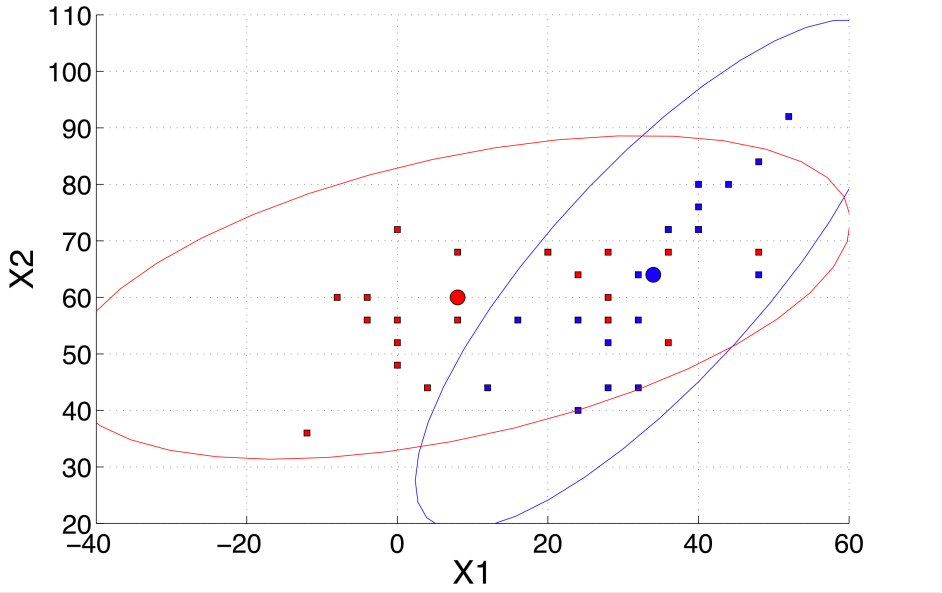
Consider the following dataset X having 2 features X1, X2 and n=40 instances.

*X1 = [0, 24, 0, 28, 0, 32, 28, -12, 44, 48, 28, 40, 0, -4, 28, -8, 40, 24, -4, 36, 36, 8, 24, 48, 48, 52, 40, 20, 12, 32, 36, 16, 8, 32, 40, 32, 4, 28, 36, 20];*

*X2 = [72, 56, 52, 56, 56, 64, 60, 36, 80, 84, 52, 80, 48, 56, 68, 60, 76, 64, 60, 52, 68, 68, 40, 64, 68, 92, 72, 68, 44, 56, 72, 56, 56, 64, 72, 44, 44, 44, 72, 68];*

*C = [2, 1, 2, 2, 2, 1, 2, 2, 1, 1, 1, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 1, 1, 2, 1, 1, 2, 1, 1, 1, 1, 2, 1, 1, 1, 2, 1, 1, 2];*

Where C is the class vecor representing the class of each instance.




- Compute the projection matrix corresponding to all the possible principal components (PCA) and choose the best principal component.
- Compute the projection matrix given by the Fisher's Linear Discriminant Analysis.
- Compute the confidence interval of a K-NN (K=1) classifier built on the principal component obtained at step 1, when a K-fold cross-validation (K=10) is performed.
- Compute the confidence interval of a K-NN (K=1) classifier built on the projected dimension obtained by LDA at step 2, when a K-fold cross-validation (K=10) is performed.
- Are the classifier built after PCA (obtained at step 3) and LDA (obtained at step 4) equal? Perform a Student's Paired t-test to establish this.

**Suggestion:** you might want to use the matlab code provided by the teacher on the course website ;-)