Exam Data Mining
Date: 6-11-2014
Time: 13.30-16.30

General Remarks

1. You are allowed to consult 1 A4 sheet with notes written on both sides.
2. You are allowed to use a pocket calculator. Use of mobile phones is not allowed.
3. Always show how you arrived at the result of your calculations. Otherwise you can not get partial credit for incorrect answers.
4. There are six questions.

Question 1 Short Questions

Answer the following questions:

(a) In computing the quality of splits in classification trees, why is the gini-index preferred over resubstitution error as an impurity measure?

(b) Explain why the independence properties of a Tree-Augmented Naive Bayes (TAN) classifier are the same as those of the undirected graphical model obtained by simply dropping the directions of all its edges.

(c) In frequent tree mining, what is the difference between an induced subtree and an embedded subtree?

(d) The following probability model for binary random variables $X_1, \ldots, X_4$ was obtained by removing some of the u-terms from the full log-linear expansion:

$$\begin{aligned}\log P(x) = {} & u_\emptyset + u_1 x_1 + u_2 x_2 + u_3 x_3 + u_4 x_4 + u_{12} x_1 x_2 + u_{23} x_2 x_3 + \\ & + u_{34} x_3 x_4 + u_{14} x_1 x_4 + u_{13} x_1 x_3 + u_{123} x_1 x_2 x_3, \qquad x_i \in \{0, 1\}.\end{aligned}$$

Draw the independence graph of this model, and indicate whether it is:
(1) hierarchical, (2) graphical.

Question 2: Classification Trees

Consider the following data on numeric attribute x and class label y. The class label can take on two different values, coded as A and B.

x	8	8	12	12	14	16	16	18	20	20
y	A	B	A	B	A	A	A	A	A	B

We use the gini-index as impurity measure. The optimal split is the one that maximizes the impurity reduction.

(a) Which candidate split(s) do we have to evaluate to determine the optimal one? (don't list any more splits than strictly necessary)

(b) What is the optimal split on x, and what is the impurity reduction of that split?

Question 3: Frequent Itemset Mining

The table below contains all closed frequent itemsets and their support on a database with transactions on items $\{A, B, C, D, E\}$:

itemset	support
$ABCD$	3
BC	5
ABD	6
B	8

(a) Is ADE frequent? Explain your answer.

(b) Derive the support of all non-empty subsets of ABC (including ABC itself).

(c) Is AC a generator? Explain your answer.

(d) Is AB a generator? Explain your answer.

Question 4: Undirected Graphical Models

The following data concerns an outbreak of food poisoning after the traditional Christmas Lunch of the personnel of the Department of Information and Computing Sciences of our University. This time the theme was Italian. Of the food eaten, interest focused on pizza slices and lasagne. The variables are

1. Pizza slice eaten (1) or not eaten (0) (P)
2. Lasagne eaten (1) or not eaten (0) (L)
3. Sick (1) or not (0) (S)

Questionnaires were completed by 100 of the 114 persons attending. The table of observed counts is given below.

$n(P,L,S)$		S	
P	L	0	1
0	0	22	4
0	1	3	12
1	0	8	1
1	1	12	38

For example, $n(0,1,1) = 12$ is the number of people that did not have a slice of pizza, but did eat the lasagne, and became sick.

(a) Compute $\text{cpr}(P,S)$, the cross-product ratio of P and S, and interpret the outcome.

(b) Draw the undirected independence graph of the graphical model expressing the constraint $P \perp\!\!\!\perp S \mid L$, and state the corresponding independence assumption(s) in words.

(c) Compute the fitted counts $\hat{n}(P,L,S)$ for the model given under (b).

(d) Perform a statistical test to check whether the model you fitted under (c) gives an adequate fit of the data, using $\alpha = 0.05$. To perform the test, you may consult the following table with critical values:

degrees of freedom (ν)	1	2	3	4	5	6	7	8
critical value ($\chi^2_{\nu,0.05}$)	3.84	6.00	7.82	9.50	11.1	12.6	14.1	15.5

Clearly state whether or not the model is rejected, and explain how you made that decision.

Question 5: Bayesian Networks

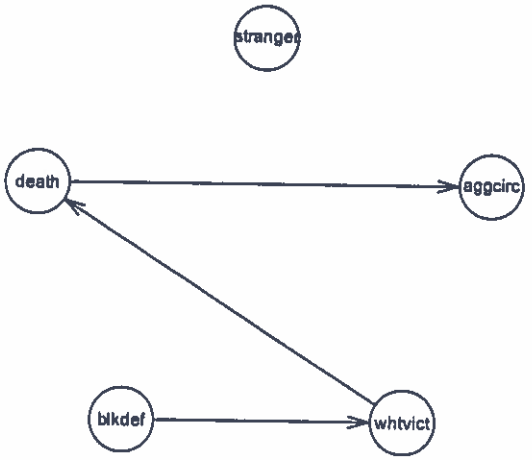
Data were provided by the Georgia Parole Board, the Georgia Supreme Court, and others, on the following variables:

variable	description
death	1 if defendant got death penalty; 0 otherwise
blkdef	1 if black defendant; 0 otherwise
whtvict	1 if white victim; 0 otherwise
aggcirc	1 if more than 2 aggravating circumstances; 0 otherwise
stranger	1 if defendant and victim did not know each other; 0 otherwise

The data is given in Table 1 on the last page. For every possible value combination, the final column of Table 1 specifies how often this combination occurs.
Consider a heuristic search for a Bayesian Network that maximizes the BIC score

$$\mathrm{BIC}(M) = \mathcal{L}(M) - \frac{\ln N}{2} \dim(M).$$

The algorithm performs a hill-climbing search where the neighbors of the current model are obtained by either: removing an arrow from the current model, adding an arrow to the current model, or turning an arrow of the current model around.
The current model in the search is:



The search process started from the empty graph, and up to this point can be summarized as follows:

iteration	action
1	add blkdef → whtvict
2	add whtvict → death
3	add death → aggcirc

(a) In iteration 4 of the search, for which operations (additions/removals/reversals of arcs) does the algorithm need to *compute* the score? Note: assume that scores of operations computed in previous iterations that are still valid are not recomputed.

(b) Would the addition of death → stranger in iteration 4 improve the BIC score? Show the necessary calculations. Use the *natural* logarithm in your computations.

The following is a general question, not restricted to this specific data set:

(c) Does the hill-climbing structure learning algorithm for Bayesian networks (the simple structure learning algorithm discussed in the lecture notes and slides) always terminate after a finite number of iterations? If your answer is "yes", explain why. If your answer is "no", give a scenario in which the algorithm does not terminate.

Question 6: Frequent Sequence Mining

Consider the following variation of frequent pattern mining. Let Σ denote an alphabet of symbols. A *sequence* is defined as an ordered list of symbols $S = s_1 s_2 \ldots s_k$ where $s_i \in \Sigma$ is a symbol at position i. Let $S = s_1 s_2 \ldots s_n$ and $R = r_1 r_2 \ldots r_m$ be two sequences over Σ. We say R is a subsequence of S, denoted $R \preceq S$, if there exists a one-to-one mapping $\phi : [1, m] \rightarrow [1, n]$, such that $r_i = s_{\phi(i)}$ and for any two positions i, j in R, if $i < j$ then $\phi(i) < \phi(j)$. In other words, each position in R is mapped to a different position in S and the order of symbols is preserved, even though there may be intervening gaps between consecutive elements of R in the mapping. If R is a subsequence of S, we also say that S *contains* R, or R *occurs in* S. Now consider the following frequent sequence mining settings and corresponding definitions of support:

(1) Given a database $D = \{S_1, S_2, \ldots, S_N\}$ of N sequences, and some sequence R, the support of R is defined as the total number of sequences in D that contain R.

(2) Given a single data sequence S and some sequence R, the support of R is defined as the number of different occurrences of R in S, that is, the number of distinct one-to-one mappings that satisfy the conditions for a subsequence.

Consider the following property: if $R \preceq S$ then support$(R) \geq$ support(S).

(a) Does the stated property hold for support definition (1)? If yes, explain why. If not, give a counterexample.

(b) Does the stated property hold for support definition (2)? If yes, explain why. If not, give a counterexample.

(c) How could the stated property be exploited in an algorithm for finding all frequent subsequences?

	death	blkdef	whtvict	aggcirc	stranger	count
1	0	0	0	0	0	0
2	1	0	0	0	0	0
3	0	1	0	0	0	5
4	1	1	0	0	0	0
5	0	0	1	0	0	7
6	1	0	1	0	0	1
7	0	1	1	0	0	1
8	1	1	1	0	0	0
9	0	0	0	1	0	0
10	1	0	0	1	0	0
11	0	1	0	1	0	5
12	1	1	0	1	0	2
13	0	0	1	1	0	9
14	1	0	1	1	0	12
15	0	1	1	1	0	5
16	1	1	1	1	0	2
17	0	0	0	0	1	0
18	1	0	0	0	1	0
19	0	1	0	0	1	1
20	1	1	0	0	1	1
21	0	0	1	0	1	2
22	1	0	1	0	1	2
23	0	1	1	0	1	1
24	1	1	1	0	1	1
25	0	0	0	1	1	1
26	1	0	0	1	1	1
27	0	1	0	1	1	8
28	1	1	0	1	1	2
29	0	0	1	1	1	2
30	1	0	1	1	1	10
31	0	1	1	1	1	4
32	1	1	1	1	1	15

Table 1: Data set for Question 5. Each row contains a different value combination of the 5 variables. The final column specifies how many times this value combination occurs in the data set. The total number of observations is $N = 100$.