Name:_______________________________________

1	2	3	4	5	6	7	total
40	*20*	*45*	*45*	*20*	*15*	*15*	*200*

UMBC CMSC 671 Final Exam

December 20, 2009

Please write all of your answers on this exam. The exam is closed book and has seven problems that add up to 230 points. You have the two hours to work on this exam. Good luck.

1. True/False (40 points)

T F An important advantage of support vector machines (SVMs) is that they can directly implement classifiers with a large number of classes. False

T F SVMs are a good choice for machine learning problems with a large number of features. True

T F SVMs require features that have a finite number of values. False

T F The ID3 decision tree learning algorithm only works for binary classification problems. False

T F Naive Bayes can't capture interdependencies between variables. True

T F The ID3 decision tree learning algorithm always finds an optimal decision tree, i.e., one that minimizes the number of questions needed to classify a case. False

T F In a well formed Bayesian Belief Network, a node is always conditionally independent of its non-descendants given its parents. True

T F *Information gain* is used to maximize the margin the network structure in an SVM. False

T F Random variables A and B are independent if p(A^B) = p(A|B)*p(B). False

T F Partial order planners search though a space formed by possible situations. False

T F PDDL is a standard language for representing planning graphs. False

T F The situation calculus is an approach to reasoning about changes in random variables using first order logic. False

T F Overfitting can result when a machine learning hypothesis spaces has too few dimensions. False

T F An advantage of using decision trees for machine learning is that the classifiers produced can be easily implemented with rules. True

T F If the Blackbox planner finds a plan, it is guaranteed to be an optimal one, i.e., there is no other plan that has fewer steps. True

T F The Blackbox planner encodes the operators as Horn clauses. False

T F Bayesian belief networks are powerful because they can reason from causes to symptoms and also from symptoms to clauses. True

T F In theory, a decision tree with N Boolean variables can represent any Boolean function over those N variables. True

T F Determining whether a propositional formula is satisfiable is NP-Complete. True

T F A common way to classify text is to use every word in the text as a feature. True

2. Decision trees I (20 points)

Given the following decision tree, show how the new examples in the table would be classified by filling in the last column in the table. If an example cannot be classified, enter *UNKNOWN* in the last column.

```
                 Color
                   |
           -------------------
       Blue|      Red|    Green|
           |         |         |
         Width       NO     Height
     -----------          ---------
Thin|       Fat|    Short|     Tall|
    |          |         |         |
   NO         YES        NO       YES
```

Example	Color	Height	Width	Class
A	Red	Short	Thin	NO
B	Blue	Tall	Fat	YES
C	Green	Short	Fat	NO
D	Green	Tall	Thin	YES
E	Blue	Short	Thin	NO

3. Decision trees II (45 points)

log₂ approx.	
log2(1/8)	= -3.0p
log2(1/4)	= -2.00
log2(1/3)	= -1.58
log2(3/8)	= -1.42
log2(3/7)	= -1.22
log2(1/2)	= -1.00
log2(4/7)	= -0.81
log2(5/8)	= -0.68
log2(2/3)	= -0.58
log2(3/4)	= -0.42
log2(7/8)	= -0.19

We would like to predict the sex of a person based on two binary attributes: leg-cover (pants or skirts) and facial-hair (some or none). We have a data set of 2,000 individuals, half male and half female. 75% of the males have no facial hair. Skirts are worn by 50% of the females. All females are barefaced and no male wears a skirt.

(a) What is the initial entropy in the system? [5]

initial entropy = I(1/2,1/2) = -(1/2*log(1/2) + 1/2*log(1/2)) = 1

(b) Compute the information gain of initially choosing the attribute leg-cover and for initially choosing facial-hair. [15]

Entropy after leg-cover = ¼*I(0,1) + ¾*I(1/3,2/3) ≈ 0 + 0.69 = 0.69
Information gain for leg cover = 0.31

Entropy after facial-hair = 1/8*I(0,1) + 7/8*I(3/7,4/7) ≈ 0 + 0.86
Information gain for facial hair = 0.14

(c) Based on yours answers, which attribute should be used as the root of a decision tree? [10]

Leg cover

(d) How can the information gain heuristic that was originally designed for symbolic attributes be generalized to cope with continuous attributes? [5]

Quantize the variable into a fixed number of ranges. For example, if T represents the temperature and is assumed to be a real number ranging from -50 to +120, we might quantize it into ranges like (<0, 0-32, 32-90, >90)

(e) A friend notes that in some countries men do wear skits or skirt-like garments. To ensure a more accurate decision tree, he suggests adding another variable: the person's ID number represented as an integer between 0 and one billion. Is this a good idea? Why or why not, assuming you are using ID3 and the simple information gain heuristic. [10]

Using this variable would produce the largest information gain, since it would split all 2000 observations into 200 categories, each with just one instance and thus a known class. So ID# would choose that variable and then stop. Of course, this would be useless as a decision tree, since any new examples, would most likely not match any of the IDs in the decision tree and would not the classified.

4. Resolution theorem proving (45 points)

Suppose we have the following predicates and a universe where all of the objects are people.

sat(X) --- X is satisfied with life
doc(X) --- X is a doctor
psy(X) --- X is a psychiatrist
child(X,Y) --- X is a child of Y
Bob --- an individual person

(a) Express he following statements in FOL using the predicates and objects above. [10]

S#	English	FOL
1	A person is satisfied with life if all his/her children are doctors.	**Ax (Ay child(y,x) => doc(y)) => sat(x)**
2	All of Bob's children are psychiatrists	**Ax child(x, Bob) => pys(x)**
3	Psychiatrists are doc-tors.	**Ax psy(x) => doc(x)**
4	Bob is satisfied with life.	**Sat(Bob)**

(b) Convert your expressions to a set of clauses in CNF, i.e., into a set of disjunctions of atomic literals. For each clause, the S# column should hold the statement number (1-4 from the table above) it was derived from. Use as many of the rows in the table as needed. [15]

C#	S#	Clause (in conjunctive normal form)
1	**1**	**child(sk0(x), x)) v sat(x)** ***; note that sk0 is a skolem function***
2	**1**	**~ doc(sk0(x)) v sat(x)**
3	**2**	**~child(x,Bob) v psy(x)**
4	**3**	**~psy(x) v doc(x)**
5	**4**	**sat(Bob) ; don't use this one in the proof!**
6	**--**	**~sat(Bob) ; well add the negation of what we want to prove**

(c) Are all of the clauses in your table Horn clauses? Explain your answer. [5]

No. The first clause has two non-negative literals, so it is not a Horn clause

(d) Using the clauses in your table, give a resolution proof that Bob is satisfied with life. Show all of the resolutions and the substitutions. Draw the proof as a graph or give it as a list of steps, showing for each step the clauses resolved, the new clause and substitution used. [15]

Add C#6: ~sat(Bob) ; negation of our goal

Resolve C#6 and C#1 producing

C#7: child(sk0(Bob),Bob) ; sk0(Bob) is random child of Bob

Resolve C#7 and C#3 producing

C#8: psy(sk0(Bob)) ; who must be a psychiatrist

Resolve C#8 and C#4 producing

C#9: doc(sk0(Bob)) ; and therefore a doctor

Resolve C#9 and C#2 producing

C#10: sat(Bob) ; so Bob is happy

Resolve C#10 and C#6 producing

empty clause

5. Probabilistic Reasoning (20)

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}.$$

Bayes Rule

Let Boolean random variables B stand for "has breast cancer" and M stand for "mammography test is positive." A research study has produced the following three observations.

- The prior probability of having breast cancer is 0.01.
- The probability of testing positive when you have breast cancer is 90%.
- The probability of testing negative when you do not have breast cancer is 89.9%.

(a) What is the prior probability of having a positive mammography test, i.e., what is P(M)? [5]

given: p(B) = 0.01; p(M|B) = 0.90; p(~M|~B) = 0.899

so: p(~B) = 0.99; p(~M|B) = 0.10, p(M|~B) = 0.101

p(M) = p(M^B) + p(M^~B)
= p(M|B)*p(B) + p(M|~B)*p(~B)
= 0.9*0.01 + 0.101*0.99 = 0.10899

while we are at it, let's compute the probability of a negative mammography test

P(~M) = 1 - p(M) = 0.89

(b) If a patient has a positive mammography test, what is the probability that she has breast cancer? That is, compute P(B | M). [5]

p(B|M) = (p(M|B)*p(B)) / p(M) ; use Bayes rule
= 0.90 * 0.01 / 0.10899
= 0.0826

(c) If a patient gets a negative mammography test, what is the probability that she has breast cancer? That is, compute P(B | ~M). [5]

p(B|~M) = p(~M|B)*p(B)/p(~M) ; use Bayes rule
= 0.10 * 0.01 / 0.89
= 0.0011

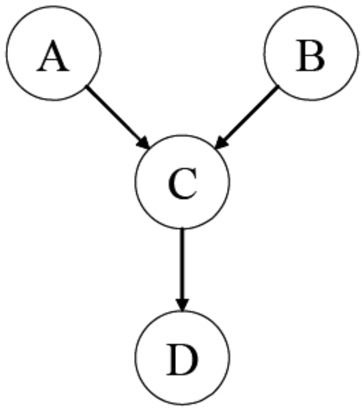
(d) Draw a Bayesian Belief Network for these two variables. [5]

We can use our knowledge of the world here to see that B is a condition and M a symptom of the condition. From the above statistics we know that M depends on B. So the structure of the network is: B → M

6. Bayesian Networks I (15 points)
Use the network structure and conditional probability tables below to calculate P(C=false).

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}.$$

Bayes Rule



A	P(A)
False	0.6
True	0.4

B	P(B)
False	0.2
True	0.8

A	B	C	P(C|A,B)
False	False	False	0.75
False	False	True	0.25
False	True	False	0.75
False	True	True	0.25
True	False	False	0.75
True	False	True	0.25
True	True	False	0.75
True	True	True	0.25

C	D	P(D|C)
False	False	0.75
False	True	0.25
True	False	0.9
True	True	0.1

C depends only on A and B, which are independent

From the table for p(C|A,B) we can select all of the rows where C=false.
Since A and B are independent of each other and of C, we can compute the probabilities for each row and then multiply by the p(C|A,B) column for that row.

The answer is 0.75

7. Partial Order Planning (15 points)

The partial order planning algorithm discussed in class and in the text uses three kinds of links: ordering constraints, causal links and threats. Briefly describe what each kind of link represents and how it is used.

An *ordering constraint link* from step A to step B means that step A must be done before step B in the plan. Ordering constraints are added to resolve threats where one step will undo a prerequisite of another.

A *causal link* exists from step A to step B when step A is done in order to make a prerequisite of step B true. The causal link is actually connects an effect of step A and a prerequisite for step B. This results in the unification of the effect and precondition, which may bind some variables. A causal link also implies a temporal ordering. In this case, step A must be done before step B to ensure that the associated prerequisite is satisfied.

A *threat link* goes from a step A to a causal link between steps B and C. It indicates that the step A will interfere or undo C's precondition associated with the causal link. The threat can be resolved by adding an ordering link from A to B (meaning A must be done before B) or from C to A, meaning that A can only be done after C has finished.