CS 188 Introduction to AI
Spring 2005 Stuart Russell
Final

You have 2 hours and 50 minutes. The exam is open-book, open-notes. 100 points total. Panic not.

Mark your answers ON THE EXAM ITSELF. Write your name, SID, and section number at the top of each page.
For true/false questions, CIRCLE *True* OR *False*.
For multiple-choice questions, CIRCLE *ALL* CORRECT CHOICES (in some cases, there may be more than one).

If you are not sure of your answer you may wish to provide a *brief* explanation.

For official use only

Q. 1	Q. 2	Q. 3	Q. 4	Q. 5	Q. 6	Total
/12	/10	/20	/20	/20	/18	/100

1. (12 pts.) Some Easy Questions to Start With

(a) (2) *True/False*: HMMs with continuous evidence variables cannot be represented by DBNs.

(b) (2) *True/False*: If two objects A and B are on flat, level ground, and A appears completely above B in an upright camera image, then B must be closer to the camera.

(c) (2) *True/False*: Let α and β be any propositional CNF sentences, each containing the same n proposition symbols. Then either $\alpha \models \beta$ or $\alpha \models \neg\beta$.

(d) (2) *True/False*: Let α and β be any propositional CNF sentences each containing the same n proposition symbols, where α consists only of unit clauses. Then either $\alpha \models \beta$ or $\alpha \models \neg\beta$.

(e) (2) *True/False*: Every search problem can be represented as an MDP whose optimal solutions are the same as for the search problem.

(f) (2) *True/False*: A language model is useful for transcribing handwritten text.

2. (10 pts.) Propositional Logic and CSPs
In this question, we will consider representing propositional satisfiability (SAT) problems as CSPs.

(a) (2) Draw the constraint graph corresponding to the SAT problem
$(\neg X_1 \lor X_2) \land (\neg X_2 \lor X_3) \land \ldots \land (\neg X_{n-1} \lor X_n)$ for $n = 4$.

(b) (2) How many solutions are there for this general SAT problem as a function of n?
(i) 0 (ii) 1 (iii) 2 (iv) n (v) $n+1$ (vi) 2^n

(c) (2) Suppose we apply backtracking (p.142) to find *all* solutions to a SAT CSP of the type given in (a). (To find *all* solutions to a CSP, we simply modify the basic algorithm so it continues searching after each solution is found.) Assume that variables are ordered $X_1, \ldots, X_n$ and $false$ is ordered before $true$. How much time will the algorithm take to terminate?
(i) constant time (ii) linear in n (iii) quadratic in n (iv) exponential in n

(d) (1) *True/False*: Every Horn-form SAT problem can be solved in time linear in the number of variables.

(e) (1) *True/False*: Every tree-structured binary CSP with discrete, finite domains can be solved in time linear in the number of variables.

(f) (2) *True/False*: (d) is a consequence of (e).

3. (20 pts.) First-Order Logic
In this question, we will use the following vocabulary: $IsA(x, c)$ means that object x is a member of product category c; $Weight(x, w)$ means that object x weighs w grams; $SKU1286$ is the name of a particular category of aluminium bolts.

(a) (2) Using this vocabulary and first-order logic, write down the fact that bolts of this type weigh $18g$.

(b) (2) Suppose that a Horn KB contains one such sentence for each of m product categories, as well as the particular sentence $IsA(Bolt_3, SKU1286)$. What is the time complexity of answering the query $Weight(Bolt_3, w)$ using backward chaining? (Assume constant-time retrieval of matching facts and rules for atomic queries.)
(i) constant time (ii) linear in m (iii) exponential in m

(c) (2) How about using forward chaining, beginning with the fact $IsA(Bolt_3, SKU1286)$?
(i) constant time (ii) linear in m (iii) exponential in m

(d) (2) Suppose that we add sentences to the KB defining $n-1$ new properties for each product category, in addition to weight (e.g., price, diameter, material, coating, head type, etc.). What is the new time complexity of forward chaining for the same query?
(i) constant time (ii) linear in m (iii) linear in n (iv) linear in mn

(e) (3) Write down a first-order Horn clause stating that any two objects of the same product category have the same weight.

(f) (6) Given your sentence in part (e), and the sentences
$IsA(Bolt_3, SKU1286), IsA(TypicalSKU1286Instance, SKU1286), Weight(TypicalSKU1286Instance, 18g)$,
show in graphical form a resolution proof of the fact that $Weight(Bolt_3, 18)$. For each step, indicate which clauses are resolved, the unifier, and the resulting clause. You may abbreviate the predicates and constants by their first letters.

(g) (3) Suppose the KB from (d) is rewritten to contain the following, in addition to $IsA(Bolt_3, SKU1286)$:

- one sentence naming a typical instance for each of m categories;
- one sentence of the kind given in (e) for each of n properties;
- mn sentences specifying the value of each property for each typical instance.

Now what is the complexity of solving the query $Weight(Bolt_3, w)$ using backward chaining?
(i) constant time (ii) linear in m (iii) linear in n (iv) linear in mn

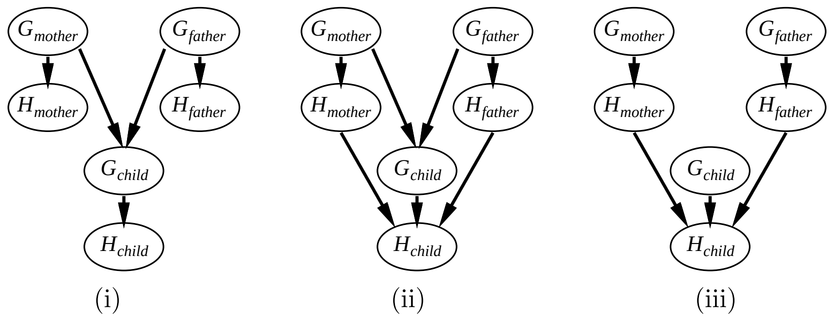


Fig. 1: Three possible structures for a Bayesian network describing genetic inheritance.

4. (20 pts.) Bayesian networks

Let H_x be a random variable denoting the handedness of an individual x, with possible values l or r. A common hypothesis is that left- or right-handedness is inherited by a simple mechanism; i.e., perhaps there is a gene G_x, also with values l or r, and perhaps actual handedness turns out the same (with some probability s) as the gene an individual possesses. Furthermore, perhaps the gene itself is equally likely to be inherited from either of an individual's parents, with a small non-zero probability m of a random mutation flipping the handedness.

(a) (3) Which of the three network structures in Fig. 1 claim that
$\mathbf{P}(G_{father}, G_{mother}, G_{child}) = \mathbf{P}(G_{father})\mathbf{P}(G_{mother})\mathbf{P}(G_{child})$? (i) (ii) (iii)

(b) (3) Which of the three networks make independence claims that are consistent with the hypothesis?
(i) (ii) (iii)

(c) (2) Which of the three networks is the best description of the hypothesis? (i) (ii) (iii)

(d) (4) Draw the CPT for the G_{child} node in networks (i) or (ii) and fill in the values

(e) (4) Suppose that $P(G_{father} = l) = P(G_{mother} = l) = x$. In network (i) or (ii), derive an expression for $P(G_{child} = l)$ in terms of m and x only, by conditioning on its parent nodes.

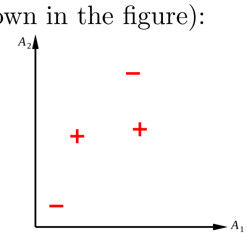
(f) (4) Under conditions of genetic equilibrium, we expect the distribution of genes to be the same across generations. Use this to calculate the value of x, and, given what you know about handedness in humans, explain why the hypothesis described at the beginning of this question must be wrong.

5. (20 pts.) Learning

In this question we consider decision trees with *continuous* input attributes $A_1, \ldots, A_n$ and a *Boolean* output attribute Y. In such trees, the test at each internal node is an inequality of the form $A_i > c$, where c, the *split point*, may be any real number (to be chosen by the learning algorithm). The value at each leaf is *true* or $false$. In a *test-once* tree, each attribute may be tested at most once on any path in the tree; in a *test-many* tree, each attribute may be tested more than once on a path.

Suppose we are given the following four examples (also shown in the figure):

A_1	A_2	Y
3	3	$false$
6	13	$true$
15	14	$true$
14	22	$false$



(a) (4) Draw a test-once decision tree that classifies the examples correctly.

(b) (2) Write down the information gain of your root test (your answer may contain logs; numerical evaluation not required).

(c) (4) Suppose we sort the examples by their value for attribute A_i. (See, e.g., the values for A_2 in the table above.) Explain why the best (i.e., highest information gain) split point for A_i never falls directly between two examples with the same output classification. (E.g., for A_2 in the table, it could not be between 13 and 14.)

(d) (2) *True/False*: Every non-noisy training set can be correctly classified by a test-once decision tree.

(e) (4) Can every non-noisy training set be classified correctly by a test-many tree? Why (not)?

(f) (2) Now consider the four training sets in Fig. 2 below.
Which are correctly classifiable by *low-depth* test-many decision trees? (i) (ii) (iii) (iv)

(g) (2) Which are correctly classifiable by a perceptron? (i) (ii) (iii) (iv)

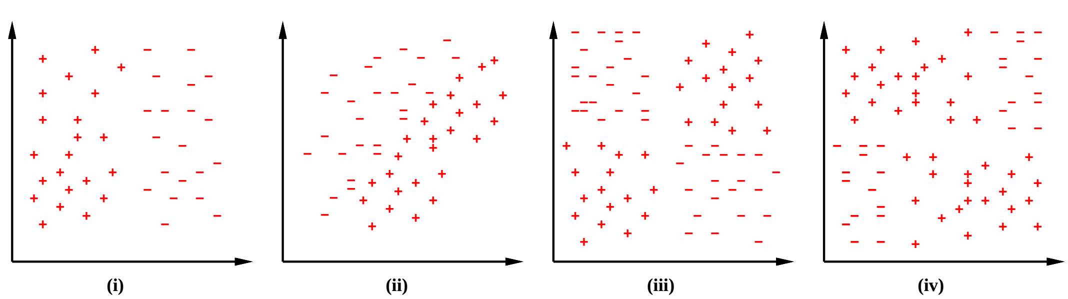


Fig. 2: Four different training sets with two continuous input attributes.

6. (18 pts.) Natural language

Consider the following context-free grammar (where X* means zero or more occurrences of X):

S → NP VP
S → first S then S
NP → Determiner Modifier Noun | Pronoun | ProperNoun
Determiner → a | the | every
Pronoun → she | he | it | him | her
Modifier → Adjective* | Noun*
Adjective → red | violet | fragrant
Noun → rose | dahlia | violet
VP → Verb NP
VP → IntransitiveVerb
VP → Copula Adjective
Verb → smelled | watered | was
IntransitiveVerb → smelled | rose
Copula → was | seemed | smelled
ProperNoun → Spike

(a) (6) Which of the following sentences are generated by the grammar?
(i) first first Spike smelled fragrant then he smelled then he watered the violet violet
(ii) the red red rose rose rose
(iii) she was a violet violet violet

(b) (4) Show the parse tree for the sentence, "First she watered the rose then it smelled"

[*continued overleaf ...*]

(c) (2) How many ways can the sentence “First the violet violet rose then the violet violet violet smelled” be parsed?
(i) 0 (ii) 1 (iii) 2 (iv) 4 (v) more than 4

(d) (1) What type of ambiguity is causing this multiplicity?
(i) lexical (ii) semantic (iii) referential

(e) (4) In English, one can say “first A then B then C” whereas nested constructions such as “first first A then B then first C then D” are not usually allowed. Show how to replace the rule “S $\rightarrow$ first S then S” by one or more new rules to reflect this.

(f) (1) *True/False*: A sentence that has exactly one parse tree is unambiguous.