# Final Examination

CS 540: Introduction to Artificial Intelligence

August 10, 2006

**LAST NAME: __________SOLUTION______________________________**

**FIRST NAME: ________________________________________________**

| Problem | Score | Max Score |
|---|---|---|
| 1 | ________ | 15 |
| 2 | ________ | 10 |
| 3 | ________ | 15 |
| 4 | ________ | 15 |
| 5 | ________ | 15 |
| 6 | ________ | 15 |
| Total | ________ | 85 |

1. [15] **Perceptrons**

(a) [4] Can a Perceptron learn the "SAME" function of three binary inputs, defined to be 1 if all inputs are the same value and 0 otherwise? Either argue/show that this is impossible or construct a Perceptron that correctly represents this function.

```
No.  SAME is the complement of XOR.  It is not linearly separable
and therefore cannot be represented by a Perceptron.  Proof is by
a figure showing the 3D input space.
```

(b) [4] Can a Perceptron learn to correctly classify the following data, where each consists of three binary input values and a binary classification value: (111,1), (110,1), (011,1), (010,0), (000,0)? Either argue/show that this is impossible or construct such a Perceptron.

```
Yes.  Output is 1 if at least 2 of the 3 inputs are 1.  Therefore
a Perceptron with all three weights equal to 0.5 will work.
```

(c) [4] Consider a Perceptron with 3 inputs and one output unit that uses a linear threshold activation function with threshold 0.7, learning rate 0.2, and initial weights W1=0.2, W2=0.7, W3=0.9.

(i) [1] What is the output of the Perceptron given the inputs I1=1, I2=0, I3=1?

```
O=1 because (.2)(1) + (.7)(0) + (.9)(1) = 1.1 ≥ 0.7
```

(ii) [3] What are the weights' values after applying the Perceptron Learning Rule with the above input and desired output 0?

```
T=0 and O=1, so update the weights using Wj = Wj +
(.2)(0-1)Ij.  Thus, the new weights are W1 = 0.2 + (-.2)(1) =
0.0, W2 = 0.7 + (-.2)(0) = 0.7, and W3 = 0.9 + (-.2)(1) = 0.7
```

(d) [3] Briefly describe a good way to determine when to stop the Perceptron Learning algorithm.

```
It is not a good idea to iterate until a local minimum on the
training set is achieved because this will often cause over-
fitting of the training data.  Better is to use a Tuning Set,
which is separate from both the Training Set and the Testing Set,
to determine when to stop.  That is, stop when the Tuning Set's
error is minimized while using the Training Set to iteratively
update the weights.  Performance is then evaluated using the
Testing Set.
```

2. [10] **Probabilistic Reasoning**
A barrel contains many plastic eggs. Some eggs are painted red and some are blue. 40% of the eggs in the barrel contain pearls, and the rest contain nothing. 30% of eggs containing pearls are painted blue, and 10% of eggs containing nothing are blue. What is the probability that a blue egg contains a pearl? Use Boolean random variables $B$ for a blue egg, and $P$ for contains a pearl. Show your work.

Given: $P(P) = 0.4$, $P(B \mid P) = 0.3$, and $P(B \mid \neg P) = 0.1$

$$P(P|B) = P(B \mid P)P(P) \,/\, P(B)$$

$$= 0.12 \,/\, [P(B \mid P)\ P(P) + P(B \mid \neg P)\ P(\neg P)]$$

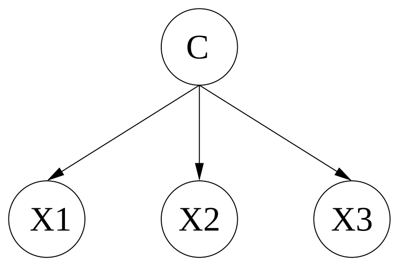
$$= 0.12 \,/\, [(0.3)(0.4) + (0.1)(0.6)]$$

$$= 0.67$$

So, a 67% probability.

3. [15] **Naive Bayes**
You are given some documents, each specified by a feature vector, $(X1, X2, X3)$, where each component has a binary value. There are two possible classifications of a document, $C$ = 0 or 1. It is 3 times more likely that a document is in class 1 than in class 0. You also know: $P(X1 = 1 \mid C = 0) = 0.25$, $P(X2 = 1 \mid C = 0) = 0.5$, $P(X3 = 1 \mid C = 0) = 0.4$, $P(X1 = 1 \mid C = 1) = 0.5$, $P(X2 = 1 \mid C = 1) = 0.25$, and $P(X3 = 1 \mid C = 1) = 0.3$.

(a) [3] Draw the Bayesian network (with CPTs) that represents this as a Naive Bayes model.




(b) [9] Given a new document with feature vector $(X1 = 0, X2 = 1, X3 = 1)$, determine the classification of this document using the Naive Bayes model represented in (a). Show your work.

Compute $P(C=0 \mid \mathbf{X})$ and $P(C=1 \mid \mathbf{X})$ and select the class with the larger value. Using Bayes's rule, $P(C \mid \mathbf{X}) = [P(\mathbf{X} \mid C)\ P(C)] / P(\mathbf{X})$. Since the denominator is the same for both C=0 and C=1, we can drop this term and just compute the numerator. $P(011 \mid C=0) = P(X1=0 \mid C=0)\ P(X2=1 \mid C=0)\ P(X3=1 \mid C=0)\ P(C=0)$ since the Xi values are conditionally independent given C=0. $P(C=1) = 3\ P(C=0)$ and $P(C=0) + P(C=1) = 1$, so $P(C=0)$ = 0.25 and $P(C=1)$ = 0.75. So, $P(011 \mid C=0) = (.75)(0.5)(0.4)(0.25) = 0.0375$ Similarly, $P(011 \mid C=1)$ = $P(X1=0 \mid C=1)\ P(X2=1 \mid C=1)\ P(X3=1 \mid C=1)\ 3\ P(C=0)$ since $P(C=1) = 3\ P(C=0)$ So, $P(011 \mid C=1) = (0.5)(0.25)(.3)(0.75) = 0.028$ Thus the document is in class 0.

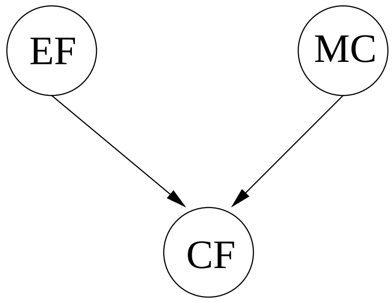
(c) [3] Which of the following express the conditional independence assumption that is used in defining Naive Bayes? Give your answer by listing 0 or more of these three.

(i) $P(C \mid X1, X2, X3) = [P(X1, X2, X3 \mid C)\ P(C)] / P(X1, X2, X3)$
(ii) $P(X1, X2, X3 \mid C) = P(X1 \mid C)\ P(X2 \mid C)\ P(X3 \mid C)$
(iii) $P(X1, X2, X3) = P(X1)\ P(X2)\ P(X3)$

(ii)

4. [15] **Bayesian Networks**
We want to design a troubleshooting advisor for PCs. Let $CF$ be a Boolean random variable representing whether the Computer Fails ($CF$ = true) or not. Assume there are two possible causes of failure: Electricity-Failure and Malfunction-of-the-Computer, represented using the Boolean random variables $EF$ and $MC$, respectively. Let $P(EF) = 0.1$, $P(MC) = 0.2$, $P(CF \mid \neg EF, \neg MC) = 0.0$, $P(CF \mid \neg EF, MC) = 0.5$, $P(CF \mid EF, \neg MC) = 1.0$, and $P(CF \mid EF, MC) = 1.0$.

(a) [3] Draw the Bayesian Network (with CPTs) for this problem.




(b) [4] Compute $P(CF, \neg EF, MC)$

`Using chain rule,`

$$P(CF, \neg EF, MC) = P(CF \mid MC, \neg EF)\, P(MC \mid EF)\, P(\neg EF)$$

$$= P(CF \mid MC, \neg EF)\, P(MC)\, P(\neg EF)$$

$$= (0.5)(0.2)(0.9) = 0.09$$

(c) [4] Compute $P(MC \mid EF)$

$P(MC \mid EF) = P(MC)$ `because MC and EF are independent.  So,` $P(MC \mid EF)$ `= 0.2`

(d) [4] Compute $P(EF \mid CF)$

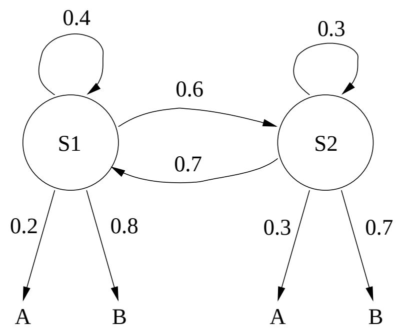
$$P(EF \mid CF) = P(EF, \neg MC \mid CF) + P(EF, MC \mid CF)$$

$$= [P(CF, EF, \neg MC) \,/\, P(CF)] + [P(CF, EF, MC) \,/\, P(CF)]$$

$$= (0.08 + 0.02) \,/\, 0.19 = 0.5263$$

5. [15] **Hidden Markov Models**

Consider the following HMM with 2 hidden states, S1 and S2, and two possible observation values, A and B, at each time in an observation sequence. Also assume the initial state probability is given by $\pi(s1) = 0.2$ and $\pi(s2)=0.8$.




(a) [4] Compute the probability of the hidden state sequence q1=S1, q2=S1, q3=S2, q4=S1.

```
(0.2)(0.4)(0.6)(0.7) = 0.0336 or about a 3.4% chance
```

(b) [9] What is the probability of the observation sequence o1=A, o2=B?

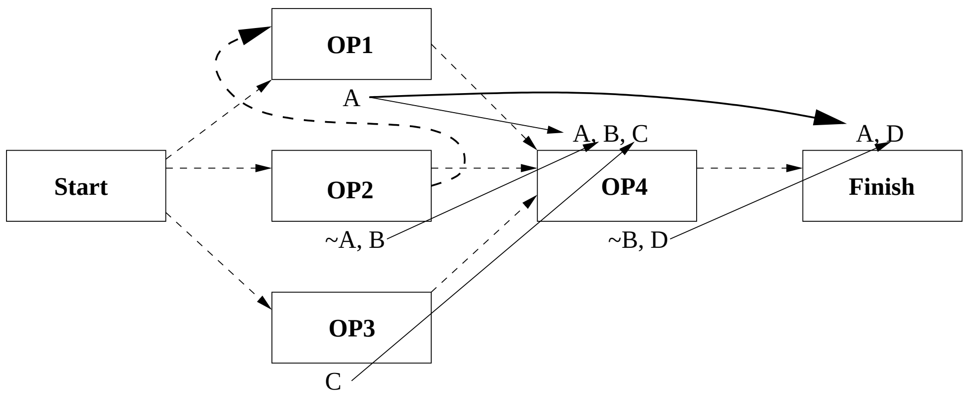
The desired probability is determined by summing over all possible state transition sequences that could produce the given observation sequence. For each possible initial starting state, we have to compute all possible state transition sequences that could produce the given observations o1=A and o2=B. If we start in S1, there are two possible state transition sequences that could produce observations A,B: (S1, S1) and (S1, S2). If we start in S2, there are two possible state transition sequences that could produce the observations: (S2, S2) and (S2, S1). Now, computing the probabilities of each of these we get $P(O \mid HMM)$ = (0.2)(0.2)(0.4)(0.8) + (0.2)(0.2)(0.6)(0.7) + (0.8)(0.3)(0.3)(0.7) + (0.8)(0.3)(0.7)(0.8) = 0.0128 + 0.0168 + 0.0504 + 0.1344 = 0.2144

(c) [2] State a problem for this HMM that would be appropriately solved using the Viterbi algorithm.

Find the sequence of hidden states, $Q$, that maximizes $P(O, Q \mid HMM)$ where $O$ is a sequence of observation values

6. [15] **Partial-Order Planning**

Consider the following intermediate state of a partial-order planner that uses four operators, OP1, OP2, OP3, and OP4. Each operator's preconditions are listed above it's box, and its effects are listed below. Causal links are solid arcs; temporal links are dashed arcs.




(a) [10] Add causal links and temporal links to the above partial-order plan so that there are no open preconditions and no threats. In the case of a threat, also state what is threatening what.

```
One open precondition, A, at step Finish can be solved by Simple Establishment using existing step OP1, creating a causal link shown as a bold solid arc in the modified partial plan above. After solving the open precondition there are two threats: (1) step OP2 threatens causal link from OP1 to OP4, and (2) step OP2 also threatens causal link from OP1 to Finish. Both of these threats can be eliminated by demotion of the threatening step, adding a temporal link from OP2 to OP1 which forces OP2 to occur before OP1, shown as a bold dotted arc above.
```

(b) [3] Give all possible solution plans that are consistent with the final partial-order plan you produced in (a).

```
With the fixes given in (a), there are three linearizations resulting in the solution plans: (OP2, OP1, OP3, OP4), (OP3, OP2, OP1, OP4), and (OP2, OP3, OP1, OP4)
```

(c) [2] True or False: In general (i.e., not just the plan above), every partial-order plan with no open preconditions and no threats has a linearization that is a correct solution plan.

```
True
```