Final Examination

CS 540-2: Introduction to Artificial Intelligence

May 7, 2017

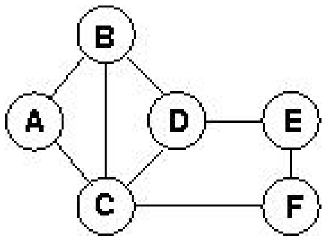
LAST NAME: SOLUTIONS

FIRST NAME: ______________________

Problem	Score	Max Score
1		14
2		10
3		6
4		10
5		11
6		9
7		8
8		12
9		12
10		8
Total		100

Question 1. [14] Constraint Satisfaction Problems

You are given the following constraint graph representing a map that has to be colored with three colors, red (R), green (G) and blue (B), subject to the constraints that no adjacent regions, which are connected by an arc in the graph, are assigned the same color.



(a) [4] Assuming variable **C** has already been assigned value R, apply the **Forward Checking** algorithm and *cross out* all values below that will be eliminated for other variables.

A	B	C	D	E	F
R G B	R G B	R	R G B	R G B	R G B

(b) [4] Assuming variable **A** has already been assigned value B and variable **C** has already been assigned value R, apply the **Arc-Consistency** algorithm (AC-3) (but no backtracking search) and *cross out* all values below that will be eliminated for other variables.

A	B	C	D	E	F
B	R G B	R	R G B	R G B	R G B

(c) [3] Consider the set of assignments below after variable **B** was selected and assigned value G, and constraint propagation performed. Of the remaining five variables, which *variable(s)* will be selected using the **Most-Constraining Variable Heuristic** (aka **Degree Heuristic**) as the best candidate(s) to consider next? Circle the variable(s) selected.

C

A	B	C	D	E	F
R B	G	R B	R B	R G B	R G B

(d) [3] Consider the set of inconsistent assignments below. Variable **C** has just been selected to be assigned a new value during a local search for a complete and consistent assignment. What new *value* will be selected for variable **C** by the **Min-Conflicts Heuristic**?

R

A	B	C	D	E	F
B	G		G	G	B

Question 2. [10] Support Vector Machines

(a) [3] Consider an SVM with decision boundary $\mathbf{w}^T\mathbf{x} + b = 0$ for a 3D feature space, where ^{T} stands for vector transpose, $\mathbf{w} = (1\ 2\ 3)^T$ (i.e., **w** is a column vector) and $b = 2$. **w** and **x** are both column vectors. What will be the **classification**, +1 or -1, of a test example defined by $\mathbf{x} = (1\ \ -1\ \ -1)^T$?

$\mathbf{w}^T\mathbf{x} + b$ = (1)(1) + (2)(-1) + (3)(-1) + 2 = -2 < 0 so classification is -1

(b) [2] True or False: In an SVM defined by the decision boundary $\mathbf{w}^T\mathbf{x} + b = 0$, the *support vectors* are those examples in the training set that lie *exactly* on this decision boundary.

```
False
```

(c) [2] True or False: An SVM that uses *slack variables* with a penalty parameter *C* that determines how much to penalize a misclassified example, can result in *either overfitting or underfitting* the training data depending on the value of the parameter *C*.

```
True
```

(d) [3] Consider a Linear SVM trained using *n* labeled points in a 2-dimensional feature space, resulting in an SVM with $k = 2$ support vectors. After adding *one* more labeled training example (so there are now a total of $n + 1$ training examples) and retraining the SVM, what is the *maximum* number of support vectors possible in the new SVM?

(i) $k = 2$
(ii) $k + 1 = 3$
(iii) $k + 2 = 4$
(iv) $n + 1$

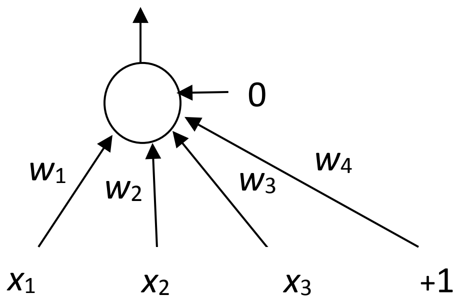
(iv) $n + 1$ (For example, if the current $n = 5$ points are: negative {(0, 1), (1, 1)}; positive {(2, 0), (3, 0), (4, 0)} then these have only 2 support vectors: (1, 1) and (2, 0). But, if we add one new negative point (2, 1), then all 6 points become support vectors.)

Question 3. [6] Perceptrons

(a) [2] True or False: A Perceptron that uses a *fixed threshold value* of 0 and *no extra input with a bias weight* can learn *any linearly-separable* function.

```
False
```

(b) [4] Consider a Perceptron with 3 inputs, x_1, x_2, x_3, and one output unit that uses a linear threshold unit (LTU) as its activation function. The initial weights are $w_1 = 0.2$, $w_2 = 0.7$, $w_3 = 0.9$, and bias $w_4 = -0.7$ (the LTU uses a fixed threshold value of 0). Hence we have:



Given the inputs $x_1 = 0$, $x_2 = 0$, $x_3 = 1$ and the above weights, this Perceptron outputs 1. What are the four (4) *updated weights' values* after applying the **Perceptron Learning Rule** with this input, actual output = 1, teacher (aka target) output = 0, and learning rate = 0.2?

T=0 and O=1, so update the weights using $w_i = w_i + (.2)(0-1)x_i$. Thus, the new weights are

$w_1 = 0.2 + (-.2)(0) = 0.2$

$w_2 = 0.7 + (-.2)(0) = 0.7$

$w_3 = 0.9 + (-.2)(1) = 0.7$

$w_4 = -0.7 + (-.2)(1) = -0.9$

Question 4. [10] Back-Propagation in Neural Networks

(a) [3] Which *one* (1) of the following *best* describes the process of ***learning*** in a multilayer, feed-forward neural network that uses *back-propagation learning*?

(i) Activation values are propagated from the input nodes through the hidden layers to the output nodes.
(ii) Activation values are propagated from the output nodes through the hidden layers to the input nodes.
(iii) Weights on the arcs are modified based on values propagated from input nodes to output nodes.
(iv) Weights on the arcs are modified based on values propagated from output nodes to input nodes.
(v) Arcs in the network are modified, gradually shortening the path from input nodes to output nodes.
(vi) Weights on the arcs from the input nodes are compared to the weights on the arcs coming into the output nodes, and then these weights are modified to reduce the difference.

```
(iv)
```

(b) [2] True or False: The back-propagation algorithm applied to a multilayer, feed-forward neural network using sigmoid activation functions is guaranteed to achieve 100% correct classification for any set of training examples, given a sufficiently small learning rate.

```
False (because it may converge to a local minimum error)
```

(c) [2] True or False: A multilayer, feed-forward neural network that uses as its activation function the *Identity function*, i.e., $g(x) = x$, and trained using the back-propagation algorithm can *only* learn *linearly-separable* functions.

```
True
```

(d) [3] Which one (1) of the following activation functions is the best one to use in a multilayer, feed-forward neural network that uses the back-propagation algorithm and that will most likely have the fastest training time?

(i) LTU
(ii) ReLU
(iii) Sigmoid

```
(ii) ReLU
```

Question 5. [11] Deep Learning

(a) [5] A Convolutional Neural Network (CNN) has a 100 x 100 array of feature values at one layer that is connected to a Convolution layer that uses 10 x 10 filters with a stride of 5 (i.e., the filter is shifted by 5 horizontally and vertically over different 10 x 10 regions of feature values in the layer below). Only filters that are *entirely inside* the 100 x 100 array are connected to a unit in the Convolution layer. In addition, each unit in the Convolution layer uses an ReLU activation function to compute its output value.

(i) [2] How many *units* are in the Convolution layer?

```
19 x 19 = 361
```

(ii) [3] How many *weights* must be learned for this layer, *not* including any bias weights needed?

```
Each unit in the Convolution layer is connected to 10 x 10 =
100 units in the layer below.  These weights are shared across
all units in this layer, so the total is 100.
```

(b) [3] What is the *main* purpose of using **Dropout** during training in a deep neural network? Select the *one* (1) best answer of the following:

(i) To speed up training
(ii) To avoid underfitting
(iii) To avoid overfitting
(iv) To increase accuracy on the training set

```
(iii)  To avoid overfitting
```

(c) [3] Answer True or False for *each* of the following statements about a **Max Pooling** layer that uses a 3 x 3 filter with a stride of 3 in a CNN.

(i) Exhibits local translation invariance of the features detected at the previous layer

```
True
```

(ii) Reduces the number of units in this layer compared to the layer below it

```
True
```

(iii) Shared weights mean only 9 weights have to be learned for this layer

```
False (there are no weights associated with a Pooling layer)
```

Question 6. [9] Probabilistic Reasoning

A barrel contains many balls, some of which are red and the rest are blue. 40% of the balls are made of metal and the rest are made of wood. 30% of the metal balls are colored blue, and 10% of the wood balls are colored blue. Let Boolean random variables B mean a ball is blue (so $\neg B$ means a ball is red), and M means a ball is metal (and $\neg M$ means it is wood). Hence, $P(M) = 0.4$, $P(B \mid M) = 0.3$ and $P(B \mid \neg M) = 0.1$

(a) [3] What is the probability that a wood ball is red, i.e., $P(\neg B \mid \neg M)$?

$P(\neg B \mid \neg M) = 1 - P(B \mid \neg M) = 1 - 0.1 = 0.9$

(b) [3] What is the prior probability that a ball is blue, i.e., $P(B)$?

$P(B) = P(B|M)P(M) + P(B|\neg M)P(\neg M) = .3 * .4 + .1 * (1 - .4) = 0.18$

(c) [3] What is the posterior probability that a blue ball is metal, i.e., $P(M \mid B)$?

$P(M \mid B) = P(B \mid M)\ P(M)\ /\ P(B) = .3 * .4 / .18 = 0.67$

Question 7. [8] Naïve Bayes

Consider the problem of detecting if an email message contains a Virus. Say we use four random variables to model this problem: Boolean class variable V indicates if the message contains a virus or not, and three Boolean feature variables: A, B and C. We decide to use a Naïve Bayes Classifier to solve this problem so we create a Bayesian network with arcs from V to each of A, B and C. Their associated CPTs are created from the following data: $P(V) = 0.2$, $P(A|V) = 0.8$, $P(A|\neg V) = 0.4$, $P(B|V) = 0.3$, $P(B|\neg V) = 0.1$, $P(C|V) = 0.1$, $P(C|\neg V) = 0.6$

(a) [4] Compute $P(\neg A, B, C \mid V)$

Use the conditionalized version of the chain rule plus conditional independence to get:

$$P(\neg A,B,C \mid V) = P(\neg A \mid B,C,V)\ P(B \mid C,V)\ P(C \mid V)$$

$$= P(\neg A \mid V)\ P(B \mid V)\ P(C \mid V)$$

$$= P(1 - P(A|V))\ P(B|V)\ P(C|V)$$

$$= (.2)(.3)(.1)$$

$$= 0.006$$

(b) [4] Compute $P(A, \neg B, \neg C)$

Use the conditioning rule, then the conditionalized version of the chain rule as in (a):

$$P(A, \neg B, \neg C) = P(A, \neg B, \neg C \mid V)P(V) + P(A, \neg B, \neg C \mid \neg V)P(\neg V))$$

$$= P(A \mid V)\ P(\neg B \mid V)\ P(\neg C \mid V)\ P(V)$$

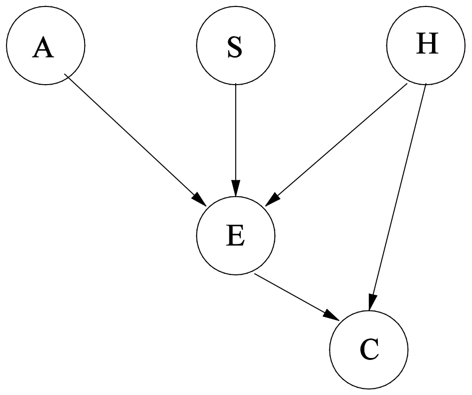
$$+ P(A \mid \neg V)\ P(\neg B \mid \neg V)\ P(\neg C \mid \neg V)\ P(\neg V)$$

$$= (.8)(.7)(.9)(.2) + (.4)(.9)(.4)(.8)$$

$$= 0.216$$

Question 8. [12] Bayesian Networks

Consider the following Bayesian Network containing 5 Boolean random variables:



(a) [3] Write an expression for computing $P(A, \neg S, H, E, \neg C)$ given *only* information that is in the associated CPTs for this network.

$P(A)\ P(\neg S)\ P(H)\ P(E \mid A, \neg S, H)\ P(\neg C \mid E, H)$

(b) [2] True or False: $P(A, \neg C, E) = P(A)\ P(\neg C)\ P(E)$

False. This is only true when *A* and *C* and *E* are independent, which they are not here by the structure of the Bayesian network

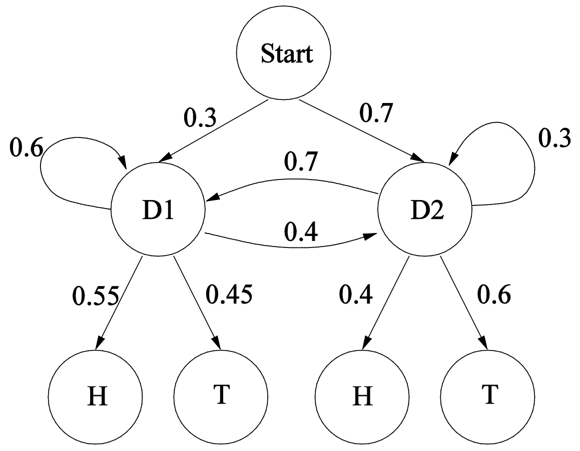
(c) [3] How many numbers must be stored in *total* in all CPTs associated with this network (excluding numbers that can be calculated from other numbers)?

$1 + 1 + 1 + 2^3 + 2^2 = 15$

(d) [4] *C* is conditionally independent of ____A and S____ given ____E and H_____

Question 9. [12] Hidden Markov Models

I have two coins, D1 and D2. At each turn I pick one of the two coins but don't know which one I picked. Which coin is selected is assumed to obey a first-order Markov assumption. After I pick a coin, I flip it and observe whether it's a Head or a Tail. The two coins are not "fair," meaning that they each don't have equal likelihood of turning up heads or tails. The complete situation, including the probability of which coin I pick first is given by the following HMM, where D1 and D2 are the two hidden states, and H and T are the two possible observable values.



(a) [4] Compute $P(q_1 = \text{D1}, q_2 = \text{D1}, q_3 = \text{D2}, q_4 = \text{D2})$

$$
\begin{aligned}
&P(q_1=\text{D1}, q_2=\text{D1}, q_3=\text{D2}, q_4=\text{D2}) \\
&\quad = P(q_4=\text{D2} \mid q_3=\text{D2})\, P(q_3=\text{D2} \mid q_2=\text{D1})\, P(q_2=\text{D1} \mid q_1=\text{D1})\, P(q_1=\text{D1}) \\
&\quad = (.3)(.4)(.6)(.3) \\
&\quad = 0.0216
\end{aligned}
$$

(b) [4] Compute $P(o_1 = \text{H}, o_2 = \text{T}, q_1 = \text{D2}, q_2 = \text{D2})$

$$
\begin{aligned}
&P(o_1=\text{H}, o_2=\text{T}, q_1=\text{D2}, q_2=\text{D2}) \\
&\quad = P(q_1=\text{D2})\, \text{P}(\text{q}_2=\text{D2} \mid \text{q}_1=\text{D2})\, P(o_1=\text{H} \mid q_1=\text{D2})\, \text{P}(o_2=\text{T} \mid q_2=\text{D2}) \\
&\quad = (.7)(.3)(.4)(.6) \\
&\quad = 0.0504
\end{aligned}
$$

(c) [4] Compute $P(q_1 = \text{D1} \mid o_1 = \text{H})$

$$
\begin{aligned}
&P(q_1=\text{D1}|o_1=\text{H}) = P(o_1=\text{H}|q_1=\text{D1})\, P(q_1=\text{D1}) \,/\, P(o_1=\text{H}) \text{ by Bayes' rule} \\
&\quad = P(o_1=\text{H}|q_1=\text{D1})\, P(q_1=\text{D1}) \,/\, [P(o_1=\text{H}|q_1=\text{D1})\, P(q_1=\text{D1}) \\
&\qquad\quad + P(o_1=\text{H}|q_1=\text{D2})\, P(q_1=\text{D2})] \text{ by conditioning rule} \\
&\quad = ((.55)(.3))/[(.55)(.3) + (.4)(.7)] \\
&\quad = 0.37
\end{aligned}
$$

Question 10. [8] AdaBoost

(a) [2] True or False: In AdaBoost, if an example in the training set is *classified correctly* by a chosen weak classifier, we should *eliminate* it from the training set used in the next iteration to compute the next weak classifier.

```
False
```

(b) [2] True or False: During each iteration of AdaBoost after the next weak classifier is determined, the weights of the *misclassified* examples will *all* be updated (before normalization) by the *same* factor.

```
True
```

(c) [2] True or False: The decisions of the individual weak classifiers (+1 for class 1 and -1 for class 2) selected by Adaboost are combined by choosing the *majority class*.

```
False (The sign of the linear combination of their decisions is
used, which is the weighted majority class)
```

(d) [2] True or False: In a 2-class classification problem where each example is defined by a *k*-dimensional feature vector, if each weak classifier is defined by a threshold on a single attribute (defining a linear decision boundary), AdaBoost can only learn a strong classifier that has 0 misclassifications on the training set when the training set is *linearly separable,* no matter how many weak classifiers are combined.

```
False
```