Final: CS 7301
Spring 2016

The exam is closed book (2 cheat sheets allowed). If you run out of room for an answer, use an additional sheet (available from the instructor) and staple it to your exam. Questions marked with $**$ are the hardest, followed by questions marked with a $*$. Unmarked questions are relatively easy and a good strategy might be to attempt them before the questions marked with $*$ and $**$.

- **NAME** ____________________
- **UTD-ID if known** ____________________
- **Time:** 2 hours 40 minutes.

Question	Points	Score
Decision Trees	10	
Linear versus Non-linear functions	10	
Neural Networks	10	
Maximum Likelihood Estimation	8	
Computational Learning Theory	10	
AdaBoost	12	
Bayesian Networks: Inference	10	
Bayesian networks: Learning	10	
Support Vector Machines	10	
K-means	10	
Total:	100	

Question 1: Decision Trees (10 points)

(a) (6 points) **(True/False) You are given a propositional formula in conjunctive normal form (CNF). Recall that a formula is in CNF if it is a conjunction of clauses where a clause is a disjunction of literals and a literal is a propositional variable or its negation. For example, the following formula over four variables $\{X_1, X_2, X_3, X_4\}$ is in CNF:

$$(X_1 \vee \neg X_3) \wedge (X_3 \vee X_4) \wedge (\neg X_4 \vee X_2 \vee X_1)$$

Let f be a CNF formula having m clauses such that the length of each clause is at most k (also called k-CNF). Is the following statement true or false? f can be represented by a decision tree having height k and m leaf nodes. Explain your answer. If your answer is True, describe an algorithm for constructing the decision tree given f. If your answer is False, give a counter-example or provide a reasonable argument.

Solution: False. Take a k-CNF such as the one given above and try to find an optimal decision tree for it and you will realize that it is not possible.

(b) (4 points) Which of the following statements are true or false? Explain your answer in one or two sentences.

- Reduced-Error pruning reduces the bias but increases the variance.

  Solution: False. Actually, it is the opposite. Reduced error pruning decreases the number of nodes (namely parameters) in the decision tree and thus it yields simpler models having higher bias and smaller variance.

- The Entropy impurity is always preferable (in the sense that it will give better generalization accuracy) to the Gini impurity because the former is based on sound information-theoretic principles.

  Solution: False. It depends on the dataset. As we saw in homework 1, sometimes Gini impurity gives better generalization. Moreover, the no-free lunch theorem says that no algorithm/impurity will dominate all others and thus the statement is clearly false.

Question 2: Linear versus Non-linear functions (10 points)

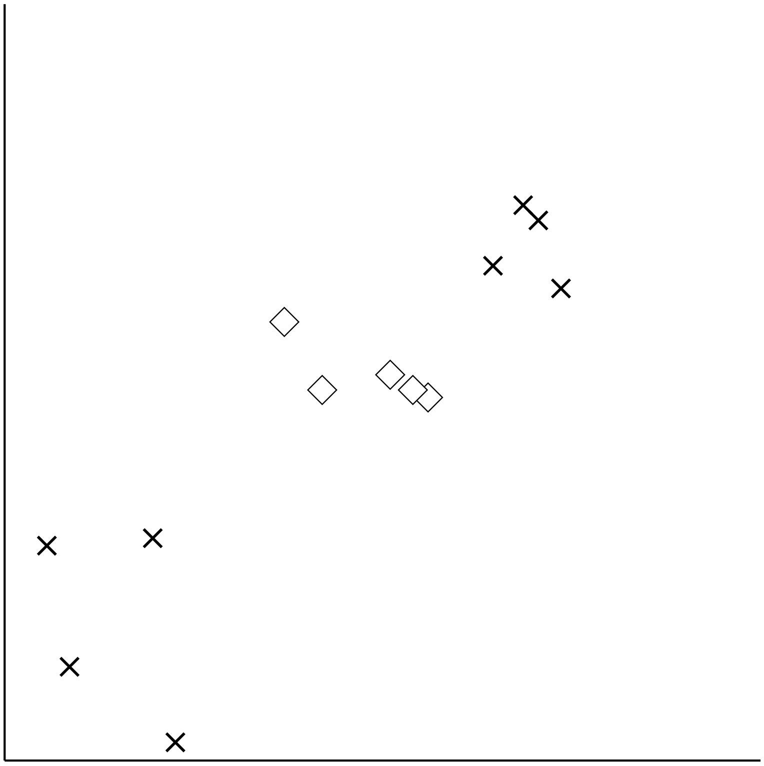
Recall that a linear threshold function or a linear classifier is given by: If $(w_0 + \sum_i w_i x_i) > 0$ then the class is positive, otherwise it is negative. Here, $x_1, \ldots, x_n$ are the attribute values and $w_0, \ldots, w_n$ are the weights (w_0 is the bias term). Assume that 1 is true and 0 is false.

(a) (5 points) * Consider the Boolean function given below. x_1, x_2 and x_3 are the attributes and y is the class variable.

x_1	x_2	x_3	y
0	0	0	1
0	0	1	-1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	-1
1	1	0	-1
1	1	1	-1

Can you represent the function using a linear threshold function. If your answer is **YES**, then give a precise numerical setting of the weights. Otherwise, clearly explain, why this function cannot be represented using a linear threshold function. No credit will be given if the explanation is incorrect.

Solution: No. The function is not linear. You can simply plot it in 3-D and see for yourself. The Boolean function is given by: $(\neg x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge x_2)$.



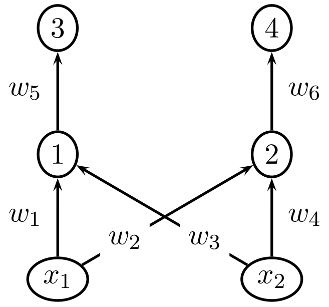
(b) (5 points) Which of the following will classify the data perfectly and why? Draw the decision boundary.

- Logistic Regression
- SVMs using a quadratic kernel.

Solution: SVMs with a quadratic kernel. There will be a circle around the squares or a parabola depending on the quadratic kernel you use.

Question 3: Neural Networks (10 points)

Consider the Neural network given below.



Assume that all internal nodes and output nodes compute the $tanh$ function. In this question, we will derive an explicit expression that shows how back propagation (applied to minimize the least squares error function) changes the values of w_1, w_2, w_3, w_4, w_5 and w_6 when the algorithm is given the example (x_1, x_2, y_1, y_2) with y_1 and y_2 being outputs at 3 and 4 respectively (there are no bias terms). Assume that the learning rate is η. Let o_1 and o_2 be the output of the hidden units 1 and 2 respectively. Let o_3 and o_4 be the output of the output units 3 and 4 respectively.

Hint: Derivative of $tanh(x) = 1 - tanh^2(x)$.

(a) (3 points) Forward propagation. Write equations for o_1, o_2, o_3 and o_4.

Solution:
$o_1 = tanh(w_1x_1 + x_3x_3)$
$o_2 = tanh(w_2x_1 + w_4x_2)$
$o_3 = tanh(w_5o_1)$
$o_4 = tanh(w_6o_2)$

(b) (3 points) Backward propagation. Write equations for δ_1, δ_2, δ_3 and δ_4 where δ_1, δ_2 , δ_3 and δ_4 are the values propagated backwards by the units denoted by 1, 2, 3 and 4 respectively in the neural network.

Solution:
$\delta_1 = (1 - (o_1)^2)\delta_3 w_5$
$\delta_2 = (1 - (o_2)^2)\delta_4 w_6$
$\delta_3 = (1 - (o_3)^2)(y_1 - o_3)$
$\delta_4 = (1 - (o_4)^2)(y_2 - o_4)$

(c) (4 points) Give an explicit expression for the new (updated) weights w_1, w_2, w_3, w_4, w_5 and w_6 after backward propagation.

Solution:
$w_1 = w_1 + \eta\delta_1 x_1$
$w_2 = w_2 + \eta\delta_2 x_1$
$w_3 = w_3 + \eta\delta_1 x_2$
$w_4 = w_4 + \eta\delta_2 x_2$
$w_5 = w_5 + \eta\delta_3 o_1$
$w_6 = w_6 + \eta\delta_4 o_2$

Question 4: Maximum Likelihood Estimation (8 points)

(a) (8 points) *You are given a collection of n documents, where the word count of the i-th document is x_i. Assume that the word count is given by an Exponential Distribution with parameter λ. In other words, for a non-negative integer x,

$$P(wordcount = x|\lambda) = \frac{\lambda^x e^{-\lambda}}{x!}$$

Compute λ such that the likelihood of observing $\{x_1, x_2, \ldots, x_n\}$ is maximized.

Solution:

The log likelihood is given by:

$$\begin{aligned} LL &= \ln\left(\prod_{i=1}^{n} P(x_i|\lambda)\right) && (1) \\ &= \ln\left(\prod_{i=1}^{n} \frac{\lambda^{x_i} e^{-\lambda}}{x_i!}\right) && (2) \\ &= \sum_{i=1}^{n} x_i \ln(\lambda) - \lambda \ln(e) - \ln(x_i!) && (3) \\ &= \left\{\sum_{i=1}^{n} x_i \ln(\lambda) - \ln(x_i!)\right\} - (n\lambda) && (4) \end{aligned}$$

Taking derivative w.r.t. λ and setting it to zero, we get

$$\sum_{i=1}^{n} \frac{x_i}{\lambda} - n = 0$$

Thus,

$$\lambda = \frac{1}{n} \sum_{i=1}^{n} x_i$$

Question 5: Computational Learning Theory (10 points)

Useful formulas for this section:

$$m \geq \frac{1}{\epsilon}\left(\ln(1/\delta) + \ln(|H|)\right)$$

$$m \geq \frac{1}{2\epsilon^2}\left(\ln(1/\delta) + \ln(|H|)\right)$$

$$m \geq \frac{1}{\epsilon}\left(4\log_2(2/\delta) + 8VC(H)\log_2(13/\epsilon)\right)$$

(a) (5 points) Consider the class C of concepts of the form $(a \leq x_1 \leq b) \wedge (c \leq x_2 \leq d) \wedge (e \leq x_3 \leq f)$. Let a, b be integers in the range $[0, 199]$ and c, d, e, f be integers in the range $[0, 99]$. Give an upper bound on the number of training examples sufficient to assure that for any target concept $c \in C$, any **consistent learner** using $H = C$ will, with probability 0.99, output a hypothesis with error at most 0.05.

Solution: Substitute $|H| = \frac{(100)(101)}{2}\frac{(200)(201)}{2}\frac{(100)(101)}{2}$, $\epsilon = 0.05$ and $\delta = 0.01$ in the following expression:

$$m \geq \frac{1}{\epsilon}\left(\ln(1/\delta) + \ln(|H|)\right)$$

(b) (5 points) Consider the class C of concepts of the form $(a \leq x_1 \leq b) \wedge (c \leq x_2 \leq d)$. Suppose that a, b, c, d take on real values instead of integers. Give an upper bound on the number of training examples sufficient to assure that for any target concept $c \in C$, a learner will, with probability 0.99, output a hypothesis with error at most 0.05.

Solution: The VC-dimension of axis-parallel rectangles in d dimensions is $2d$. Here $d = 2$. Therefore the VC-dimension is 4.

Substitute $VC(H) = 4$, $\epsilon = 0.05$ and $\delta = 0.01$ in the following expression:

$$m \geq \frac{1}{\epsilon}\left(4\log_2(2/\delta) + 8VC(H)\log_2(13/\epsilon)\right)$$

Question 6: AdaBoost (12 points)

Consult the AdaBoost algorithm given on the last page for this question. Suppose you have two weak learners, h_0 and h_1, and a set of 17 points.

(a) (2 points) You find that h_1 makes one mistake and h_2 makes four mistakes on the dataset. Which learner will AdaBoost choose in the first iteration (namely $m = 1$)? Justify your answer.

Solution: Will choose h_1.

(b) (2 points) What is α_1?

Solution: $\epsilon_m = 1/17$. $\alpha_m = \ln\{(1 - 1/17)/1/17\} = \ln(16)$

(c) (2 points) Calculate the data weighting co-efficients w_2 for the following two cases: (1) the points on which the chosen learner made a mistake and (2) the points on which the chosen learner did not make a mistake.

Solution: Case 1: Error made

$w_2 = 1/17 \times 16 = 16/17$

Case 2: No Error

$w_2 = 1/17$.

(d) (6 points) Consider a simple modification to the AdaBoost algorithm in which we normalize the data weighting co-efficients. Namely, we replace $w_n^{(m+1)}$ by $w_n^{(m+1)}/Z^{(m+1)}$ where $Z^{(m+1)} = \sum_{n=1}^{N} w_n^{(m+1)}$. Prove that $Z^{(m+1)} = 2(1 - \epsilon_m)$.

Hint: Notice that if the weights are normalized, then $\epsilon_m = \sum_{n=1}^{N} w_n^{(m)} I(y_m(\mathbf{x}_n) \neq t_n)$.

Solution:

$$Z^{(m+1)} = \sum_{n=1}^{N} w_n^{(m+1)}$$

Simplifying:

$$Z^{(m+1)} = \sum_{n=1;I(y_m(\mathbf{x}_n)=t_n)}^{N} w_n^{(m)} + \sum_{n=1;I(y_m(\mathbf{x}_n)\neq t_n)}^{N} w_n^{(m)} \frac{1-\epsilon_m}{\epsilon_m}$$

The term $\frac{1-\epsilon_m}{\epsilon_m}$ can be factored out from the summation:

$$Z^{(m+1)} = \sum_{n=1;I(y_m(\mathbf{x}_n)=t_n)}^{N} w_n^{(m)} + \frac{1-\epsilon_m}{\epsilon_m} \sum_{n=1;I(y_m(\mathbf{x}_n)\neq t_n)}^{N} w_n^{(m)}$$

Note that:

$$\epsilon_m = \sum_{n=1;I(y_m(\mathbf{x}_n)\neq t_n}^{N} w_n^{(m)}$$

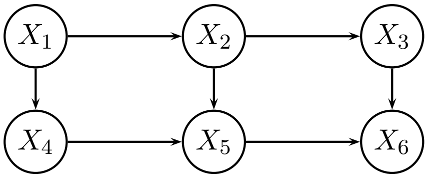
$$1 - \epsilon_m = \sum_{n=1;I(y_m(\mathbf{x}_n)=t_n}^{N} w_n^{(m)}$$

Substituting these in the Equation for Z given above.

$$Z^{(m+1)} = (1-\epsilon_m) + \frac{1-\epsilon_m}{\epsilon_m}\epsilon_m = 2(1-\epsilon_m)$$

Question 7: Bayesian Networks: Inference (10 points)

Consider the Bayesian network given below:



(a) (5 points) Let X_6 be the evidence variable. Trace the operation of **Variable elimination** for computing $\Pr(X_6 = x_6)$ along the order $(X_1, X_2, X_3, X_4, X_5)$ where x_6 is a value in the domain of X_6. Precisely show the various factors (functions) that will be generated. What is the time and space complexity of the algorithm along the ordering.

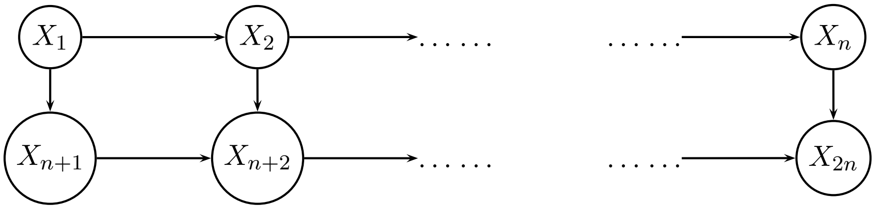
Solution: Factors after instantiating the evidence.
$\phi_1(X_1) = P(X_1)$; $\phi_2(X_1, X_2) = P(X_2|X_1)$; $\phi_3(X_1, X_4) = P(X_4|X_1)$; $\phi_4(X_2, X_3) = P(X_3|X_2)$; $\phi_5(X_2, X_4, X_5) = P(X_5|X_2, X_4)$; and $\phi_6(X_3, X_5) = P(X_6 = x_6|X_3, X_5)$

- Eliminating X_1: $\sum_{X_1} \phi_1(X_1)\phi_2(X_1, X_2)\phi_3(X_1, X_4) = \phi_7(X_2, X_4)$
- Eliminating X_2: $\sum_{X_2} \phi_7(X_2, X_4)\phi_4(X_2, X_3)\phi_5(X_2, X_4, X_5) = \phi_8(X_3, X_4, X_5)$
- Eliminating X_3: $\sum_{X_3} \phi_8(X_3, X_4, X_5)\phi_6(X_3, X_5) = \phi_9(X_4, X_5)$
- Eliminating X_4: $\sum_{X_4} \phi_9(X_4, X_5) = \phi_{10}(X_5)$
- Eliminating X_5: $\sum_{X_5} \phi_{10}(X_5) = \Pr(X_6 = x_6)$

Time and space complexity: $O(5\exp(4))$ and $O(5\exp(3))$ respectively.
Common mistakes:

- $\phi(X_2, X_4)$ is expressed as $P(X_2)P(X_4)$ (-3 points). This means that the student does not understand how to take product of functions.

Consider a generalization of the Bayesian network given on the previous page:



(b) (5 points) Let X_{2n} be the evidence variable. Assume that all variables have exactly d values in their domain. What is the best (optimal) time and space complexity of computing the probability of evidence using the **Variable elimination algorithm** on the network given above. Be sure to mention the **best elimination order**, briefly explaining why it is optimal.

Solution:
A possible optimal order is $(X_1, X_{n+1}, X_2, X_{n+2}, \ldots, X_{n-1}, X_{2n-1}, X_n)$.

- Time complexity: $O(nd^3)$
- Space complexity $O(nd^3)$. Space complexity of $O(nd^2)$ or $O(d^2)$ is acceptable (here the student would have only counted the space required by the new factors).

Students can give various explanations including but not limited to the following:

- The number of variables in the new factors never exceeds 2 along the order. Since the maximum number of variables in the original factors is 3, the elimination order is optimal.
- The Bayesian network is not a tree. Only trees have treewidth 1 and as a result the width of the optimal order is ≥ 2. Since the width of the order is 2, it is optimal.

Common mistakes:

- Insufficient explanation for optimality (-3)
- Wrong complexity (-4 total: -2 for time and -2 for space)
- Wrong order (-3)

Question 8: Bayesian networks: Learning (10 points)

Consider a Bayesian network with edges $A \rightarrow B$ and $A \rightarrow C$, and parameters which are given below:

- $P(A = 1) = 0.9$
- $P(B = 1|A = 1) = 0.1$, $P(B = 1|A = 0) = 0.6$
- $P(C = 1|A = 1) = 0.7$, $P(C = 1|A = 0) = 0.3$

Consider the dataset given below:

A	B	C
0	1	?
0	1	1
?	0	1
1	1	?
1	0	?
0	0	0
1	1	1

Assume that the CPTs are the CPTs at some iteration of EM. What you are going to do is derive the new set of parameters after running one iteration of EM.

(a) (5 points) Show the calculations involved in the E-step. Recall that in the E-step, you make the dataset bigger (by considering all possible completions) and weigh each new data point appropriately.

Solution: The possible completions are:

A	B	C	weight
0	1	0	w_1
0	1	1	w_2
0	1	1	1.0
0	0	1	w_3
1	0	1	w_4
1	1	0	w_5
1	1	1	w_6
1	0	0	w_7
1	0	1	w_8
0	0	0	1.0
1	1	1	1.0

where:

$$w_1 = (1 - 0.9)(0.6)(1 - 0.3); w_2 = (1 - 0.9)(0.6)(0.3); \text{ such that } w_1 + w_2 = 1$$

$$w_3 = (1 - 0.9)(1 - 0.6)(0.3); w_4 = (0.9)(1 - 0.1)(0.7); \text{ such that } w_3 + w_4 = 1$$

$$w_5 = (0.9)(0.1)(1 - 0.7); w_6 = (0.9)(0.1)(0.7); \text{ such that } w_5 + w_6 = 1$$

$$w_7 = (0.9)(1 - 0.1)(1 - 0.7); w_8 = (0.9)(1 - 0.1)(0.7); \text{ such that } w_7 + w_8 = 1$$

(b) (5 points) Show the calculations involved in the M-step. What are the new parameters?

Solution:

$$P(A=1) = \frac{w_4 + w_5 + w_6 + w_7 + w_8 + 1}{7}$$

$$P(B=1|A=1) = \frac{w_5 + w_6}{w_4 + w_5 + w_6 + w_7 + w_8 + 1}$$

$$P(B=1|A=0) = \frac{w_1 + w_2 + 1}{w_1 + w_2 + 1 + w_3 + 1}$$

$$P(C=1|A=1) = \frac{w_4 + w_6 + w_8 + 1}{w_4 + w_5 + w_6 + w_7 + w_8 + 1}$$

$$P(C=1|A=0) = \frac{w_2 + 1 + w_3}{w_4 + w_5 + w_6 + w_7 + w_8 + 1}$$

Question 9: Support Vector Machines (10 points)

Consider the following 2-D dataset (x_1 and x_2 are the attributes and y is the class variable).

Dataset:

x_1	x_2	y
0	0	$+1$
0	1	-1
1	0	-1
1	1	$+1$

(a) (5 points) Precisely write the expression for the dual problem. Let $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ be the lagrangian multipliers associated with the four data points.

Solution:

$$\text{Maximize:}\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 - \frac{1}{2}\left((\alpha_2 - \alpha_4)^2 + (\alpha_3 - \alpha_4)^2\right)$$

$$\text{subject to:}\alpha_1, \alpha_2, \alpha_3, \alpha_4 \geq 0,\ \ \alpha_1 - \alpha_2 - \alpha_3 + \alpha_4 = 0$$

(b) (5 points) **Solve the dual problem. What is the value of α_1, α_2, α_3 and α_4.

Solution: The problem is not linearly separable and therefore all alphas equal ∞.

Question 10: K-means (10 points)

(a) (10 points) Give a formal proof that K-means always converges to local minima. Recall that K-means minimizes the following objective function:

$$\phi(\{x_i\}, \{a_i\}, \{c_j\}) = \sum_{i=1}^{n} dist(x_i, c_{a_i})$$

where $\{x_i\}$ is the set of n points, $a_i \in \{1, \ldots, k\}$ gives the cluster to which the i-th point is assigned to, c_j is the mean of the j-th cluster and $dist$ denotes the Euclidean distance.

Solution: See class slides. Show that each step of the co-ordinate descent scheme can only improve the solution or keep it the same.

Page left Blank

Page left Blank

Page left Blank

AdaBoost

1. Initialize the data weighting coefficients $\{w_n\}$ by setting $w_n^{(1)} = 1/N$ for $n = 1, \ldots, N$.
2. For $m = 1, \ldots, M$:
   (a) Fit a classifier $y_m(\mathbf{x})$ to the training data by minimizing the weighted error function

$$J_m = \sum_{n=1}^{N} w_n^{(m)} I(y_m(\mathbf{x}_n) \neq t_n) \qquad (14.15)$$

   where $I(y_m(\mathbf{x}_n) \neq t_n)$ is the indicator function and equals 1 when $y_m(\mathbf{x}_n) \neq t_n$ and 0 otherwise.

   (b) Evaluate the quantities

$$\epsilon_m = \frac{\sum_{n=1}^{N} w_n^{(m)} I(y_m(\mathbf{x}_n) \neq t_n)}{\sum_{n=1}^{N} w_n^{(m)}} \qquad (14.16)$$

   and then use these to evaluate

$$\alpha_m = \ln\left\{\frac{1-\epsilon_m}{\epsilon_m}\right\}. \qquad (14.17)$$

   (c) Update the data weighting coefficients

$$w_n^{(m+1)} = w_n^{(m)} \exp\left\{\alpha_m I(y_m(\mathbf{x}_n) \neq t_n)\right\} \qquad (14.18)$$

3. Make predictions using the final model, which is given by

$$Y_M(\mathbf{x}) = \operatorname{sign}\left(\sum_{m=1}^{M} \alpha_m y_m(\mathbf{x})\right). \qquad (14.19)$$