Midterm 1: CS 6375
Fall 2020

- **NAME** ____________________

- **UTD-ID** ____________________

Question	Points	Score
Decision Trees	10	
Poisson Naive Bayes	20	
Support Vector Machines	16	
Linear Classifiers and Neural Networks	13	
AdaBoost	12	
Regression	20	
Short questions	9	
Total:	100	

Question 1: Decision Trees (10 points)

(a) (4 points) Which of the following four are reasonable strategies for handling missing values at test time in a decision tree (namely after a decision tree is learned and you are using it to predict the class of a given example)? Clearly Mark the reasonable strategies (you do not have to explain your choice):

Note: Zero, one or more strategies described below may be reasonable.

1. (Majority class) Let p and n denote the number of leaf nodes in the decision tree that are labeled with positive and negative class respectively. Output positive if $p > n$ else negative.

2. Ignore the features/attributes that are missing and choose any leaf node l such that the path from the root to l is consistent with the assignment of values to the attributes that are not missing. Output the class label associated with l.

3. Let $L = \{l_1, \ldots, l_k\}$ denote the set of leaves of the decision tree such that the path from the root to each leaf $l_i \in L$ is consistent with the assignment of values to the attributes that are not missing. Output the majority class label in L.

4. Use a separate generative model to fill in the most likely values of the missing attributes and then use the decision tree to predict the class.

Solution: 3 and 4 are reasonable strategies.

Draw a decision tree which represents the concept:

If the following CNF formula defined over 5 attributes $\{X_1, \ldots, X_5\}$ evaluates to True then the class is positive, otherwise the class is negative.

$$(X_1 \vee X_2 \vee X_3) \wedge (\neg X_1 \vee X_2 \vee X_3) \wedge (X_4 \vee X_5) \wedge (\neg X_4 \vee X_5)$$

Solution: Many answers possible here.

(b) (3 poin(ts) (3 points) Consider the following approach to learning decision trees. Use a learning algorithm A to learn a CNF representation from data (assume that an efficient algorithm A exists) and then you generalize the method you used in your answer to the previous question (question 1(b)) to convert the CNF to a decision tree. Explain why the above is not a good approach for learning decision trees even if an efficient algorithm A exists.

Solution: Approach is not reasonable because the number of leaf nodes in the decision tree may be exponential in the number of features. Assuming Boolean features, the worst case bound on the number of leaf nodes in a decision tree when the decision tree is learned from data is $O(\min(2^n, N))$ where n is the number of features and N is the number of training examples. When learned using the method described above, the worst case bound is not dependent on data and equals $O(2^n)$.

Question 2: Poisson Naive Bayes (20 points)

In this question, we consider the problem of classifying a variable Y into two categories: good (A), and bad (B). We have two attributes X_1, and X_2 and assume that each attribute (X_i, $i = 1, 2$) is related to each value (A/B) of Y via a Poisson distribution with a particular mean ($\lambda_{A,i}$/$\lambda_{B,i}$). That is

$$Pr[X_i = x|Y = A] = \frac{e^{-\lambda_{A,i}}(\lambda_{A,i})^x}{x!} \quad \text{and} \quad Pr[X_i = x|Y = B] = \frac{e^{-\lambda_{B,i}}(\lambda_{B,i})^x}{x!} \text{ for } i = 1, 2$$

(a) (10 points) Derive a general expression for estimating $\lambda_{A,i}$ and $\lambda_{B,i}$ where $i \in \{1, 2\}$ from data using the maximum likelihood estimation (MLE) approach. Assume that you are given m examples: $\{(x_1^{(1)}, x_2^{(1)}, y^{(1)}), \ldots, (x_1^{(m)}, x_2^{(m)}, y^{(m)})\}$.

X_1	X_2	Y
0	3	A
4	8	A
2	4	A
6	2	B
3	5	B
2	1	B
5	4	B

Table 1: Dataset for Poisson Naive Bayes

Assume that the data given in Table 1 is generated by a Poisson Naive Bayes model.

$\Pr(Y = A) =$	$\Pr(Y = B) =$
$\lambda_{A,1} =$	$\lambda_{B,1} =$
$\lambda_{A,2} =$	$\lambda_{B,2} =$

Table 2: Parameters for Poisson Naive Bayes. Fill in the estimated values (from data) of the six parameters.

(b) (10 points) Using the expression for λs you derived in part (a) and the dataset given in Table 1, compute λs as well as the prior probabilities, namely fill in Table 2.

Question 3: Support Vector Machines (16 points)

Recall that $K(\mathbf{x}, \mathbf{y})$ is a valid kernel where $\mathbf{x}$, $\mathbf{y} \in \mathbb{R}^n$ if there exists a transformation $\phi : \mathbb{R}^n \to \mathbb{R}^k$ where typically $k >= n$ such that:

$$K(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})^T \phi(\mathbf{y})$$

In layman's terms, a Kernel K is valid if there exists a transformation which converts two data points $\mathbf{x}$ and $\mathbf{y}$ from n dimensional space to k dimensional space such that K is equal to the dot product of the transformed data points.

(a) (7 points) Prove that $(225 + \mathbf{x}^T\mathbf{y})^2$ is a valid kernel where $\mathbf{x} = (x_1, x_2)$ and $\mathbf{y} = (y_1, y_2)$ are two dimensional, namely each data point is composed of two attributes. (Hint: Find a transformation that maps the two dimensions to six dimensions).

Consider the dataset given below (x_1, x_2 are the attributes and y is the class variable):

x_1	x_2	y
0	0	-1
$-1/3$	1	$+1$
$1/3$	1	$+1$
0	-1	-1

(b) (9 points) Give the precise primal and dual formulation for linear SVM without slack penalty for the dataset given above.

Primal Formulation (3 points):

Dual Formulation (3 points):

Identify the Support Vectors (3 points):

Solution: Primal problem: $L(w, \lambda) = \frac{1}{2}||w||^2 + \sum_{i=1}^{3} \lambda_i(y_i(w^T.x_i + b) - 1)$ Using data points and differentiating, we get the following equations:

$w_1 + 1/3\lambda_2 - 1/3\lambda_3 = 0$
$w_2 - \lambda_2 - \lambda_3 = 0$
$\lambda_1 - \lambda_2 - \lambda_3 = 0$

Since all three points are support vectors, we have: $y_i(w^T x_i + b) - 1 = 0$ for all i. Therefore,
$b = 1$
$-1/3w_1 + w_2 + b = -1$
$1/3w_1 + w_2 + b = -1$
From this, we see that $w_2 = 0$ and $w_i = 1/3$

Question 4: Linear Classifiers and Neural Networks (13 points)

(a) (3 points) True/False. The Gaussian Naive Bayes Classifier with class independent variances has smaller bias than Logistic Regression. Explain your answer.

(b) (5 points) Consider a function over n Binary features, defined as follows. If exactly k variables are true, then the class is **positive**, otherwise the class is **negative**. Can you represent this function using a linear threshold function, namely a Perceptron. 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.

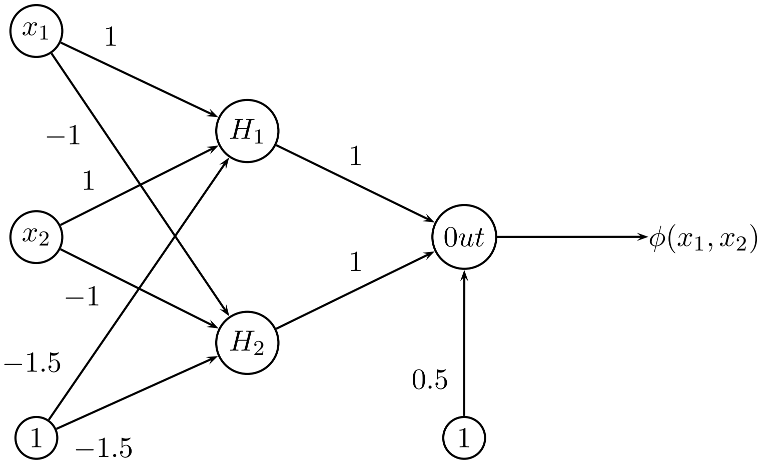
[Recall that a linear threshold function is given by: If $(w_0 + \sum_i w_i x_i) > 0$ then class is positive, otherwise it is negative. Assume that 1 is true and 0 is false.]

(c) (5 points) Draw a neural network that represents the following function.

$$\phi(x_1, x_2) = \begin{cases} -1 & \text{if } x_1 + x_2 = 0 \\ +1 & \text{otherwise} \end{cases}$$

Here x_1 and x_2 are bi-valued discrete variables that take values from the domain $\{-1, +1\}$. Each hidden and output unit that you use must be sign() units. Recall that given inputs $(x_0, \ldots, x_n)$ and weights $(w_0, \ldots, w_n)$, each sign() unit will output a $+1$ if $\sum_{i=0}^{n} w_i x_i \geq 0$ and -1 otherwise. x_0 is the bias input which always equals 1.

Solution:



Many other solutions are possible. H_1, H_2 and Out are simple threshold or sign units (sigmoid units can also be used). Logically, if we consider $+1$ as true and -1 as false, H_1 implements $x_1 \wedge x_2$, H_2 implements $\neg x_1 \wedge \neg x_2$ and O implements $H_1 \vee H_2$.

Question 5: AdaBoost (12 points)

Consult the AdaBoost algorithm given on the class slides. Suppose you have two weak learners, h_1 and h_2, and a set of 17 points.

(a) (2 points) You find that h_1 makes two mistakes 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

$w_2 = 2/17 \times 15 = 30/17$

Case 2: No Error

$w_2 = 2/17$.

(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 6: Regression (20 points)

Consider fitting the model: $y = w_0 + w_1 x$ (x is 1-dimensional) using the squared loss function that we discussed in class:

$$J(w_0, w_1) = \sum_{i=1}^{m} (y_i - (w_0 + w_1 x_i))^2$$

where $[(x_1, y_1), \ldots, (x_m, y_m)]$ are the data points.

Unfortunately we did not keep the original data, but we did store the following five quantities (statistics) that we computed from the data:

1. $\overline{x}^{(m)} = \frac{1}{m} \sum_{i=1}^{m} x_i$
2. $\overline{y}^{(m)} = \frac{1}{m} \sum_{i=1}^{m} y_i$
3. $C_{xx}^{(m)} = \frac{1}{m} \sum_{i=1}^{m} (x_i - \overline{x}^{(m)})^2$
4. $C_{xy}^{(m)} = \frac{1}{m} \sum_{i=1}^{m} (x_i - \overline{x}^{(m)})(y_i - \overline{y}^{(m)})$
5. $C_{yy}^{(m)} = \frac{1}{m} \sum_{i=1}^{m} (y_i - \overline{y}^{(m)})^2$

(a) (5 points) What are the minimal set of statistics that we need to estimate w_1. Namely, which of the above five statistics do you need to estimate w_1. Explain your answer by giving a precise expression for w_1 in terms of your chosen statistics. No credit without correct explanation.

(b) (5 points) What are the minimal set of statistics that we need to estimate w_0. Namely, which of the above five statistics do you need to estimate w_0. Explain your answer by giving a precise expression for w_0 in terms of your chosen statistics. No credit without correct explanation.

(c) (10 points) Suppose a new data point (x_{m+1}, y_{m+1}) arrives, and we want to update our sufficient statistics without looking at the old data, which we have not stored. Give precise expression for the new statistics in terms of the old statistics and the new data point. (This is useful for online learning.)

Question 7: Short questions (9 points)

(a) (5 points) Explain mathematically why the bias of the logistic regression classifier is the same as the bias of the Gaussian Naive Bayes classifier with class independent variances.

(b) (2 points) Let p be the probability of a coin landing heads up when tossed. You flip the coin 8 times and observe 5 tails and 3 heads. Suppose p can only take two values: 0.3 or 0.6. Find the Maximum likelihood estimate of p over the set of possible values $\{0.3, 0.6\}$.

Solution: $L_p = p * (1-p)^2$
$L_{0.3} = 0.3 * (0.7)^2 = 0.147$
$L_{0.6} = 0.6 * (0.4)^2 = 0.096$
Therefore MLE estimate is $p = 0.3$.

(c) (2 points) Suppose that you have the following prior on the parameter p: $P(p = 0.3) = 0.2$ and $P(p = 0.6) = 0.8$. Given that you flipped the coin 8 times with the observations described above (5 tails and 3 heads), find the MAP estimate of p over the set $\{0.3, 0.6\}$, using the prior.