CM146: Introduction to Machine Learning **Fall 2018**

Midterm

Nov. 5^{th}, 2018

- This is a closed book exam. Everything you need in order to solve the problems is supplied in the body of this exam.
- This exam booklet contains **Five** problems.
- You have 90 minutes to earn a total of 100 points.
- **Besides having the correct answer, being concise and clear is very important. For full credit, you must show your work and explain your answers.**

Good Luck!

Name and ID: (2 Point) Tianyang Peng 904971546.

Name		/2
True/False Questions		/18
Short Questions		/23
Decision Tree		/15
Perceptron		/23
Regression		/19
Total		/100

1 True/False Questions (Add a 1 sentence justification.) [18 pts]

(a) **(3 pts)** For a continuous random variable x and its probability density function $p(x)$, it holds that $0 \leq p(x) \leq 1$ for all x.

TRUE. Probability function P is $P: X \rightarrow [0, 1]$

(b) **(3 pts)** K-NN is a linear classification model.

FALSE. KNN can be used on non-linear models.

(c) **(3 pts)** Logistic regression is a probabilistic model and we use the maximum likelihood principle to learn the model parameters.

TRUE. Logistic regression returns a probability. We use Gradient Descent to learn the w to maximize the probability.

(d) **(3 pts)** Suppose you are given a dataset with 990 cancer-free images and 10 images from cancer patients. If you train a classifier which achieves 98% accuracy on this dataset, it is a reasonably good classifier.

FALSE. It might not be a good model necessarily. Even if the model is to say all patients does not have cancer, it still gets 99% accuracy in the dataset.

(e) **(3 pts)** A classifier that attains 100% accuracy on the training set is always better than a classifier that attains 70% accuracy on the training set.

FALSE. 100% accuracy might cause overfitting and lead test error to be high.

(f) **(3 pts)** A decision tree is learned by minimizing information gain.

FALSE. It's learned by maximizing it.

2 Short Questions [23 pts]

(a) **(4 pts)** What is the main difference between gradient descent and stochastic gradient descent (in one sentence)? Which one require more iterations to converge, why?

Gradient descent updates all the components of the ~~vector~~ parameters vector by computing the gradient of ~~all~~ the function; Stochastic gradient updates every time on one component of the parameter vector and needs more iterations to converge ~~converge~~. It's because it updates the model frequently, ~~but's but~~ but only updates a little bit in each iteration.

(b) **(3 pts)** What is the motivation to have a development set?

It helps to test which hyperparameter is to use and makes sure the model is not overfitting by the training set.

(c) **(3 pts)** Describe the differences between linear regression and logistic regression (in less than two sentences).

linear regression returns a linear model to predict a continuous value.

logistic regression returns the probability of the ~~target~~ result and predicts based on the ~~threshold~~ threshold of 50% probability.

(d) **(3 pts)** Consider the models that we have discussed in lecture: decision trees, k-NN, logistic regression, Perceptrons. If you are required to train a model that predicts the probability that the patient has cancer, which of these would you prefer, and why?

I would use logistic regression, ~~then~~ because it's preferred when predicting probability.

(e) **(10 pts)** Given n linearly independent feature vectors in n dimensions, show that for any assignment to the binary labels you can always construct a linear classifier with weight vector w which separates the points. Assume that the classifier has the form $sign(w \cdot x)$. Hint: a set of vectors are linearly independent if no vector in the set can be defined as a linear combination of the others.

~~Since each vector V_i is linearly independent,~~

~~Suppose for V_i~~

Suppose $\exists w$ s.t. $y(w \cdot x) \geq 0$

So the must be $y(w \cdot x) < 0$

WLOG, $y = 1$ $w \cdot x < 0$

3 Decision Trees [15 pts]

For this problem, you can write your answers using $\log_2$, but it may be helpful to note that $\log_2 3 \approx 1.6$ and entropy $H(S) = -\sum_{v=1}^{K} P(S = v) \log_2 P(S = v)$. The information gain of an attribute A is $G(S, A) = H(S) - \sum_{v \in Value(A)} \frac{|S_v|}{|S|} H(S_v)$, where S_v is the subset of S for which A has value v.

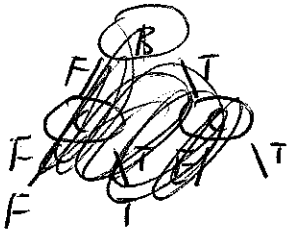
(a) We will use the dataset below to learn a decision tree which predicts the output Y, given by the binary values of A, B, C.

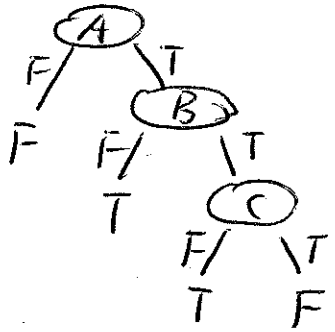
A	B	C	Y
F	F	F	F
T	F	T	T
T	T	F	T
T	T	T	F

i. **(2 pts)** Calculate the entropy of the label y.

$H(y) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = \frac{1}{2} + \frac{1}{2} = 1$

ii. **(5 pts)** Draw the decision tree that will be learned using the ID3 algorithm that achieves zero training error.

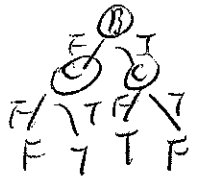


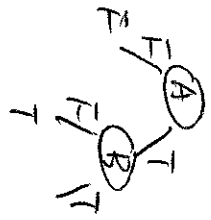


~~$\frac{1}{4}\log\frac{1}{4}$~~

$\frac{1}{4}(-1\log 1$

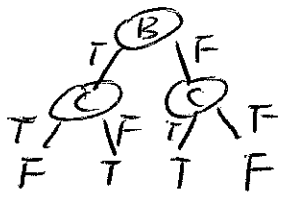
$\frac{3}{4}(-\frac{2}{3}\log\frac{2}{3} - \frac{1}{3}\log\frac{1}{3})$





iii. **(3 pts)** Is this tree optimal (i.e. does it get minimal training error with minimal depth?) explain in two sentences, and if it isn't optimal draw the optimal tree.

It's not. The optimal tree is:



This has one less depth than the previous one.

(b) **(5 pts)** You have a dataset of 400 positive examples and 400 negative examples. Now suppose you have two possible splits. One split results in (300+, 100-) and (100+, 300-). The other choice results in (200+, 400-), and (200+, 0). Which split is most preferable and why?

Split 1 = S_1 Split 2 = S_2

weighted entropy of S_1 $= \frac{1}{2}\left(-\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4}\right) + \frac{1}{2}\left(-\frac{1}{4}\log_2\frac{1}{4} - \frac{3}{4}\log_2\frac{3}{4}\right)$

$= \frac{1}{2}\left(-\frac{3}{4}\times(-2+1.6) - \frac{1}{4}\times -2\right) + \frac{1}{2}\left(-\frac{1}{4}\times -2 - \frac{3}{4}(-2+1.6)\right)$

$= \frac{1}{2}\left(\frac{3}{4}\times\frac{2}{5} + \frac{1}{2}\right) + \frac{1}{2}\left(\frac{1}{2} + \frac{3}{4}\times\frac{2}{5}\right)$

$= \frac{3}{10} + \frac{1}{2} = \frac{7}{10}$

weighted entropy of $S_2 = \frac{3}{4}\left(-\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}\right) + \frac{1}{4}(-1\log 1 - 0\log 0)$

$= \frac{3}{4}\left(-\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}\right)$

$= \frac{3}{4}\left(-\frac{1}{3}\times -1.6 - \frac{2}{3}(-1.6+1)\right)$

$= \frac{3}{4}\left(\frac{1}{3}\times\frac{8}{5} + \frac{2}{3}\times\frac{2}{5}\right)$

$= \frac{2}{10} + \frac{1}{10} = \frac{3}{10}$

$\frac{7}{10} > \frac{3}{10}$, so if $G(H, S_1) < G(H, S_2)$

So Split 2 is more preferred as it has higher information gain.

$\log_2 \frac{3}{4}$

$\log_2 3 \times 2^{-2}$

$2^{-2} = 2^{-2} \cdot 3$

$2^{-2} \cdot 2^{1.6}$

$2^1 \times 2^{-1.6} = \frac{5}{8} \times 2$

$\frac{8}{5}$

$\frac{2}{3} \times \frac{3}{5}$

$\frac{2}{5}$

4 Perceptron Algorithm [23 pts]

(a) **(4 pts)** Assume that you are given training data $(x, y) \in R^2 \times \{\pm 1\}$ in the following order:

Instance	1	2	3	4	5	6	7	8
Label y	+1	−1	+1	−1	+1	−1	+1	+1
Data (x_1, x_2)	(10, 10)	(0, 0)	(8, 4)	(3, 3)	(4, 8)	(0.5, 0.5)	(4, 3)	(2, 5)

We run the Perceptron algorithm on all the samples once, starting with an initial set of weights $w = (1, 1)$ and bias $b = 0$. On which examples, the model makes an update?

On 2, 4, 6, the weight gets updated.

In 2, $w^T \cdot (0,0) = 0 \to 1$, but $y = -1$

4, $w^T (3,3) = 6 \to 1$, but $y = -1$

6, $w^T (0.5, 0.5) = 1 \to 1$, but $y = -1$.

(b) **(8 pts)** Suggest a variation of the Perceptron update rule which has the following property: If the algorithm sees two consecutive occurrences of the same example, it will never make a mistake on the second occurrence. (Hint: determine an appropriate learning rate that guarantees this property). Prove your answer is correct.

The update rule is : make each step the size $-y\frac{w}{x}$

$w \leftarrow w + \underline{\quad\quad}$

$y(w_t^T x) < 0$

$w_{t+1} = w_t + ayx$

$y(w_t + ayx)^T x \geq 0$

$y^2 = 1$

$y w_t^T x + a y^2 x^2 > 0$

$a x^T x \geq |w_t^T x|$

$y(w_t + ayx)^T x$

$y w_t x + a x^T x \geq 0$

$a x^T x \geq -y w x$

$a x^T \geq -w_t^T$

$a \geq -\frac{w_t^T}{x^T}$

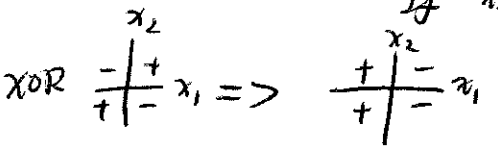
$a \geq -\frac{yw}{x}$

(c) **(3 pts)** Linear separability is a pre-requisite for the Perceptron algorithm. In practice, data is almost always inseparable, such as XOR.

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

Provide a solution to convert the inseparable data to be linearly separable. The XOR can be used for the illustration.

If $x_2 > 0$, $x_1 = -x_1$

XOR  x_1 =>

(d) **(3 pts)** Design (specify w_0, w_1, w_2 for) a two-input Perceptron (with an additional bias or offset term) that computes "OR" Boolean functions. Is your answer the only solution?

x_1	x_2	y
-1	-1	-1
1	-1	1
1	1	1
-1	1	1

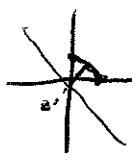
$x_0 = 1$

$w_0 = 1 \quad w_1 = 1 \quad w_2 = 1$

Then $w = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad x = \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix}$

It's not the only solution, $w = \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \end{bmatrix} \quad x = \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix}$.

(e) **(5 pts)** What is the maximal margin γ in the above OR dataset.



γ is the ~~half the~~ distance between $(-1, -1)$ and midpoint of $(1, -1)$ and $(-1, 1)$ => $(0.5, 0.5)$

$\frac{1}{2}\sqrt{(-1-1)^2 + (-1-(-1))^2} = \frac{\sqrt{2}}{2}$

~~The distance is $\frac{\sqrt{2}}{2}$.~~

$\frac{1}{2}\sqrt{(0.5+1)^2 + (0.5+1)^2}$

$= \frac{1}{2}\sqrt{(\frac{3}{2})^2 + (\frac{3}{2})^2} = \sqrt{\frac{18}{4}} = \frac{\sqrt{18}}{2} \cdot \frac{1}{2}$ The ~~dist~~ γ is $\frac{\sqrt{18}}{4}$.

5 Logistic Regression[19 pts]

Considering the following model of logistic regression for a binary classification, with a sigmoid function $\sigma(z) = \frac{1}{1+e^{-z}}$:

$$P(Y = 1|X, w_0, w_1, w_2) = \sigma(w_0 + w_1X_1 + w_2X_2)$$

(a) **(3 pts)** Suppose we have learned that for the logistic regression model, $(w_0, w_1, w_2) = (-\ln(4), \ln(2), -\ln(3))$. What will be the prediction ($y = 1$ or $y = -1$) for the given $x = (1, 2)$?

$$P = \frac{1}{1+e^{-(w_0 - \ln 4 + \ln 2 + -2\ln 3)}}$$

$$= \frac{1}{1+e^{\ln 4 - \ln 2 + 6 \ln 3^2}} = \frac{1}{1+\frac{4}{2} \cdot 9}$$

$$= \frac{1}{1+18} = \frac{1}{19}$$

pred

(b) **(6 pts)** Is logistic regression a linear or non-linear classifier? Prove your answer.

~~It's a non linear classifier. It returns the probability of an event. So if the classifier is not linearly seperable~~

It's a linear classifier. $P(Y=1|x) = 50\%$ threshold is attained when $w^T x = 0$, which is similar to perceptron.

$$\frac{1}{2} = \frac{1}{1+e^{-z}} \Rightarrow e^{-z} = 1 \Rightarrow e^{-z} = 0 \Rightarrow$$

$$-w^T x = 0$$

(c) **(10 pts)** In the hoemwork, we mention an alternative formulation of learning a logistic regression model when $y \in \{1, 0\}$

$$\arg\min_{w} \sum_{i=1}^{m} y_i \log \sigma(w^T x_i) + (1 - y_i) \log(1 - \sigma(w^T x_i))$$

. Derive its gradient.

$$\frac{\partial J}{\partial w_j} = \sum_{i=1}^{m} y_i \frac{\sigma'(w^T x_i)}{\sigma(w^T x_i)} x_j + (1 - y_i) \frac{-\sigma'(w^T x_i)}{1 - \sigma(w^T x_i)} x_j$$

$$\sigma'(w^T x_i) = \sigma(w^T x)(1 - \sigma(w^T x))$$

$$\text{So } \nabla J = \begin{bmatrix} \sum_{i=1}^{m} y_i (1 - \sigma(w^T x_i)) x_i + (1 - y_i)(-\sigma(w^T x_i)) x_i \\ \vdots \\ \sum_{i=1}^{m} y_i (1 - \sigma(w^T x_i)) x_m + (1 - y_i)(-\sigma(w^T x_i)) x_m \end{bmatrix}$$

$$\frac{\partial J}{\partial w_j} = \sum_{i=1}^{m} y_i (1 - \sigma(w^T x_i)) x_j + (1 - y_i)(\sigma(w^T x_i)) x_j$$