

Midterm

Nov. 5th, 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) *Ziqing Jiang* *204917652*

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 .

~~This is True.~~

False. - $P(x) = 2$ for $x \in (0, \frac{1}{2})$. as domain.

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

False. KNN does not generate a linear model. in the form of, $y = W^T x$.

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

True. $\sigma(W^T x) = \frac{1}{e^{-W^T x} + 1}$ would return the probabilities of $y=1$.

when the parameters of w is set so that the probability of generating the given result is greatest. we have a optimal.

- (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. Too little cancer images. Still easy to

- (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. May overfit the training set.

100% accurate classifier.

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

False. It's learned by maximizing info gain.

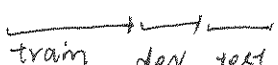
2 Short Questions [23 pts]

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

stochastic gradient descent updates after looking at each instance.
(Batch) gradient descent updates after going over the whole data set.
Stochastic gradient descent requires more iterations to converge.
since in 1 iteration it goes over only 1 instance.

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

check which hyperparameter is optimal for a model.

Eg.  for k-NN.

for each k , use train/dev. to find best k with smallest error.

for the best k , train with original train+dev, then test with "test" set.

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

linear regression returns a predicted real value. / linear model

logistic regression - - - probability of a certain classification. / sigmoid function.

- (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?

logistic regression.

It returns the probability for a certain classification.

decision tree / k -NN / perceptron returns only the classification, not 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 $\text{sign}(w \cdot x)$. Hint: a set of vectors are linearly independent if no vector in the set can be defined as a linear combination of the others.

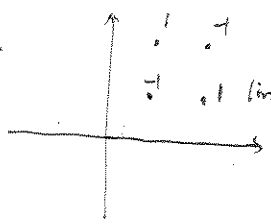
$S = \{v_1, \dots, v_n\}$ is linearly independent. $\therefore v_i = \begin{pmatrix} v_{i1} \\ \vdots \\ v_{in} \end{pmatrix}$ $y_i = 0 \text{ or } 1$, randomly assigned.

$\forall v_i \in S$. $a_1 v_1 + \dots + a_n v_n = 0$ if and only if $a_1 = \dots = a_n = 0$.

for w that separates the data.

$$\begin{cases} y_i = 1 & \text{if } w^T v_i \geq 0 \\ y_i = 0 & \text{if } w^T v_i < 0 \end{cases}$$

~~If the data.~~



If data are linearly dependent, then there are labels that there is no separator.

	x_1	x_2	y
v_1	1	1	-1
v_2	2	1	1
v_3	1	2	1
v_4	2	2	-1

$w^T v_i \geq 0$ if $y_i = \dots$

~~Assume linear independence.~~

$v_1 = v_2 + v_3 - v_4 \leftarrow$ linearly dependent.

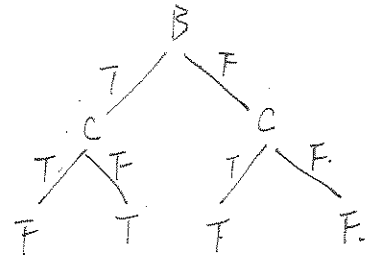
XOR: Not separable.

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 $G(S, A) = H(S) - \sum_{v \in \text{Value}(A)} \frac{|S_v|}{|S|} H(S_v)$, 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 .

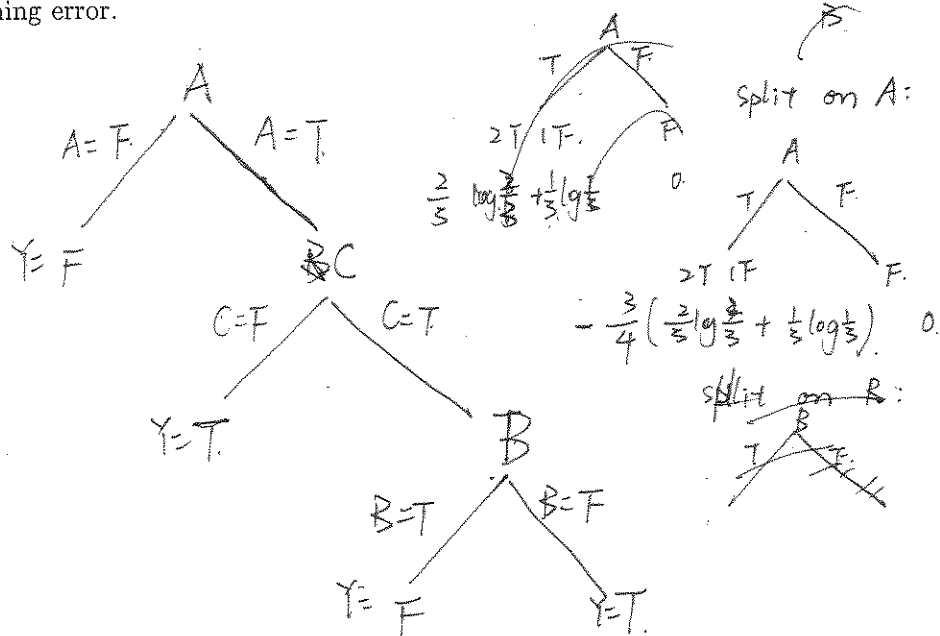
A	B	C	Y
F	F	F	F
T	F	T	T
T	T	F	T
T	T	T	F



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

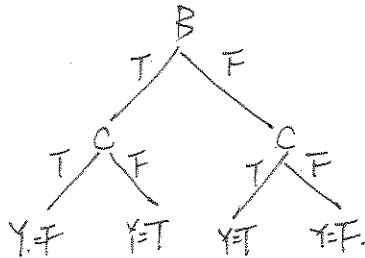
$$\begin{aligned}
 H(S) &= -(P(F) \log_2(P(F)) + P(T) \log_2(P(T))) \\
 &= -\left(\frac{1}{2} \log_2\left(\frac{1}{2}\right) + \frac{1}{2} \log_2\left(\frac{1}{2}\right)\right) \\
 &= -\left(\log_2\left(\frac{1}{2}\right)\right) \\
 &= 1.
 \end{aligned}$$

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



- 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.

No, we can have a tree with 2 level that makes no mistakes



- (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?

~~Info gain of 1~~

$$\begin{aligned}
 H(S) &= -\left(\frac{400}{800} \cdot \log \frac{400}{800}\right) \times 2 \\
 &= -2 \left(\frac{1}{2} \log \frac{1}{2}\right) \\
 &= 1
 \end{aligned}$$

Info gain (1):

$$\begin{aligned}
 H(S_A) &= \left[\frac{400}{800} \cdot \left(-\frac{300}{400} \log \frac{300}{400} - \frac{100}{400} \log \frac{100}{400} \right) + \frac{400}{800} \cdot \left(-\frac{100}{400} \log \frac{100}{400} - \frac{300}{400} \log \frac{300}{400} \right) \right] \times 2 \\
 &= \frac{3}{4} \log \frac{3}{4} + \frac{1}{4} \log \frac{1}{4} \\
 &= \frac{3}{4} (\log 3 - \log 4) \\
 &= \left(-\frac{3}{4} \times (-0.4) + \frac{1}{4} \times (-2) \right) \\
 &= -(-0.3 - 0.5) \\
 &= 0.8
 \end{aligned}$$

$$\text{Info gain} = 1 - 0.8 = 0.2$$

2nd is preferred

(2)

$$\begin{aligned}
 & \frac{600}{800} \cdot \left(-\frac{200}{600} \log \frac{200}{600} - \frac{400}{600} \log \frac{400}{600} \right) + 0 \\
 &= \frac{3}{4} \cdot \left(-\frac{1}{3} \log \frac{1}{3} - \frac{2}{3} \log \frac{2}{3} \right) \\
 &= \frac{3}{4} \cdot \left(-\frac{1}{3} \times (-1.6) + \frac{2}{3} \times (-1.6) \right) \\
 &= \frac{3}{4} \cdot \left(\frac{16}{30} + \frac{14}{30} \right) \\
 &= \frac{3}{4}
 \end{aligned}$$

Info gain: 0.25

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)

~~(10, 10)~~ ~~(0, 0)~~ ~~(8, 4)~~ ~~(3, 3)~~ ~~(4, 8)~~ (1.5, 1.5)

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?

the 4th, 5th, 6th

(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 :

$$w \leftarrow w + \frac{2(-w^T x)}{x} y_i x$$

Want to have.

$$\text{if } y_i (w^T x) < 0$$

$$\text{then } w + \eta y_i x = w_{\text{new}}$$

$$\text{such that } y_i (w_{\text{new}}^T x) > 0$$

$$y_i (w^T + \eta y_i x) > 0$$

$$y_i w^T x + \eta y_i^2 x > 0$$

$$y_i w^T x + \eta x > 0$$

$$(y_i w + \eta) x > 0$$

$$y_i (w^T + \eta y_i x) > 0$$

$$y_i w^T + \eta x > 0$$

$$\eta > \frac{y_i}{x} w^T$$

S.T.

$$y_i$$

$$y_i w^T$$

$$y_i (w^T + \eta y_i x) > 0$$

$$y_i w^T x + \eta y_i x^2 > 0$$

$$\eta y_i x^2 > -y_i w^T x$$

$$\eta y_i x^2 > -y_i w^T x$$

$$\eta x > -w^T$$

$$\eta > -\frac{w^T}{x}$$

$$\text{choose } \eta = 2 \left(\frac{w^T}{x} \right)$$

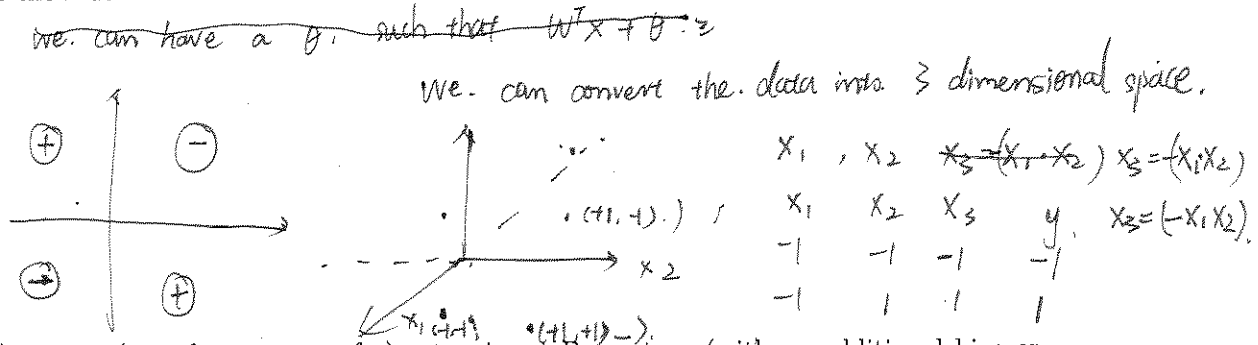
This term is positive.

2 x this term greater than 1 x this term

- (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.



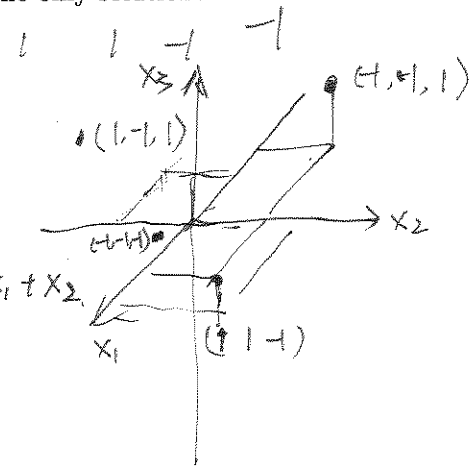
- (d) (3 pts) Design (specify w_0, w_1, w_2 for) a two-input Perceptron (with an additional bias or offset term) that computes "OR" Boolean functions. Is your answer the only solution?

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

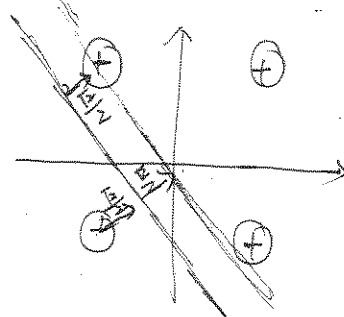
$w_0 = 1, w_1 = 1, w_2 = 1.$

$y_i = 1 + x_1 + x_2$

No, not the only one



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



$\gamma = \frac{\sqrt{2}}{2}$

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 $x = (1, 2)$?

$$W^T x = -\ln 4 + \ln 2 \cdot 1 \rightarrow \ln 3$$

$$e^{-W^T x} = \ln 4 - \ln 2 + 2 \ln 3$$

$$e^{-W^T x} = 4 - 2 + 12 = 14.$$

$$\sigma = \frac{1}{1+e^z} = \frac{1}{15}.$$

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

A linear classifier.
 logistic regression is essentially a ~~non-linear~~ ^{linear} classifier.

$\sigma(z)$ is essentially determined by a linear model $W^T x$.

if $\sigma(z)$ just calculate the probability based on $W^T x$.

if $W^T x > 0$, $P > 0.5$, more likely to be + than -.

$W^T x < 0$, $P < 0.5$ less likely to be + than -.

- (c) (10 pts) In the homework, 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.

$$\begin{aligned} \text{gradient} : \frac{\partial J}{\partial w_i} &= \frac{\partial}{\partial w_i} \sum_{i=1}^m y_i \log \sigma(w^T x_i) + (1 - y_i) \log(1 - \sigma(w^T x_i)) \\ &= \sum_{i=1}^m \left(y_i \cdot \frac{1}{\sigma(w^T x_i)} \cdot \sigma' + (1 - y_i) \frac{1}{1 - \sigma} (-\sigma') \right) \cdot x_i \end{aligned}$$

$$\sigma' = \sigma(1 - \sigma)$$

$$= \sum_{i=1}^m \left(y_i \cdot (1 - \sigma) + (1 - y_i) \cdot (-\sigma) \right) x_i$$

$$= \sum_{i=1}^m \left(y_i - y_i \sigma - \sigma + y_i \sigma \right) x_i$$

$$= \left((1 - \sigma) y_i + \sigma (1 - y_i) \right) x_i$$

$$= \sum_{i=1}^m \sigma x_i$$

$$= \sum_{i=1}^m (y_i - \sigma) x_i$$

