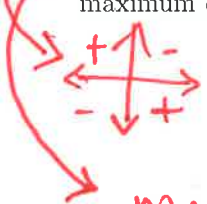


## Short Answer (10 Points)

1. (422) True False: A Decision Tree is more powerful than a perceptron. That is, it can solve more complex problems. If false, briefly explain. If true, give an example.
1. (622) Given that my training data has  $f$  binary features and  $s$  samples, give an expression for the maximum depth a decision tree can have in terms of these two values.


 A DT will get 100% accuracy. Perceptron will not.

$\max(f, s-1)$

2. Why is it that a different ordering of the input data may result in a different  $w$  and  $b$  output from the perceptron?

Because we update  $\vec{w}$  &  $b$  with every wrong prediction and we update  $\vec{w}$  &  $b$  using the  $\vec{x}$  &  $y$  from the wrongly predicted point, we will update differently with each reordering of the data. There are many lines that will separate a set of linearly separable data. These different updates can result in different final  $\vec{w}$  &  $b$ .



## Short Answer (10 Points)

3. (422) True/False: KNN and K-Means behave similarly with noisy training samples. Briefly explain.
3. (622) True/False: Duplicate training samples (same features and label) have no effect on the output of K-Means. For example, if you run K-Means on a dataset with no duplicates you will get the same result as you would if you added 20% duplicates to that same dataset. If true, briefly explain. If false, give an example.

→ They both treat all features equally, so noise causes the same problems for both.



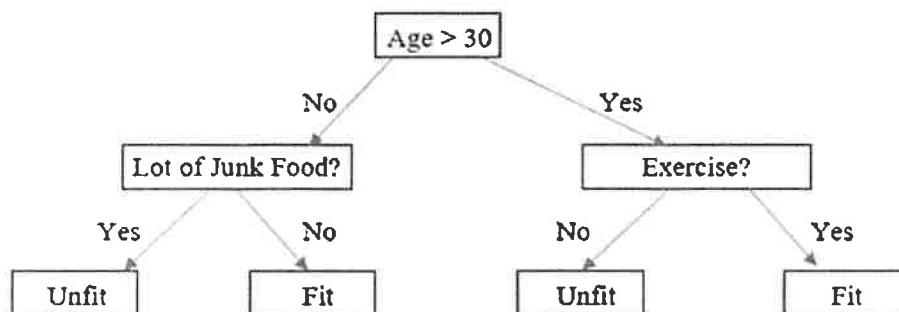
4. "I don't remember all the details of the adaboost algorithm, but I do remember that the final prediction is a weighed sum of the prediction of weak learners." Is this an example of poor recall or precision? Briefly explain.

The student can't recall all the information.  
They have perfect precision but of all the info to remember only 1 thing was remembered!



## Adaboost (10 Points)

5. Assume you have the following decision tree and training data. Give the sample weights for the first and second round of Adaboost (the final two columns of the table) using this decision tree as your weak learner.



Sample	Age	Junk Food	Exercise	Label	$d^{(0)}$	$d^{(1)}$
$s_1$	20	0	0	0	$\frac{1}{8}$	$\frac{1}{6}$
$s_2$	21	0	1	1	$\frac{1}{8}$	$\frac{1}{10}$
$s_3$	27	1	0	0	$\frac{1}{8}$	$\frac{1}{10}$
$s_4$	28	1	1	0	$\frac{1}{8}$	$\frac{1}{10}$
$s_5$	31	0	0	1	$\frac{1}{8}$	$\frac{1}{6}$
$s_6$	33	0	1	1	$\frac{1}{8}$	$\frac{1}{10}$
$s_7$	38	1	0	1	$\frac{1}{8}$	$\frac{1}{6}$
$s_8$	40	1	1	0	$\frac{1}{8}$	$\frac{1}{10}$

Handwritten notes and corrections next to the table:

- Red line through  $s_1$ : pred X
- Red line through  $s_2$ : ✓
- Red line through  $s_3$ : Same ✓
- Red line through  $s_4$ : ✓
- Red line through  $s_5$ : pred X
- Red line through  $s_6$ : ✓
- Red line through  $s_7$ : ✓
- Red line through  $s_8$ : X ✓

$$S_1 + S_5 + S_7 = \frac{1}{2}$$

$$\sum S_2, S_3, S_4, S_6, S_8 = \frac{1}{2}$$

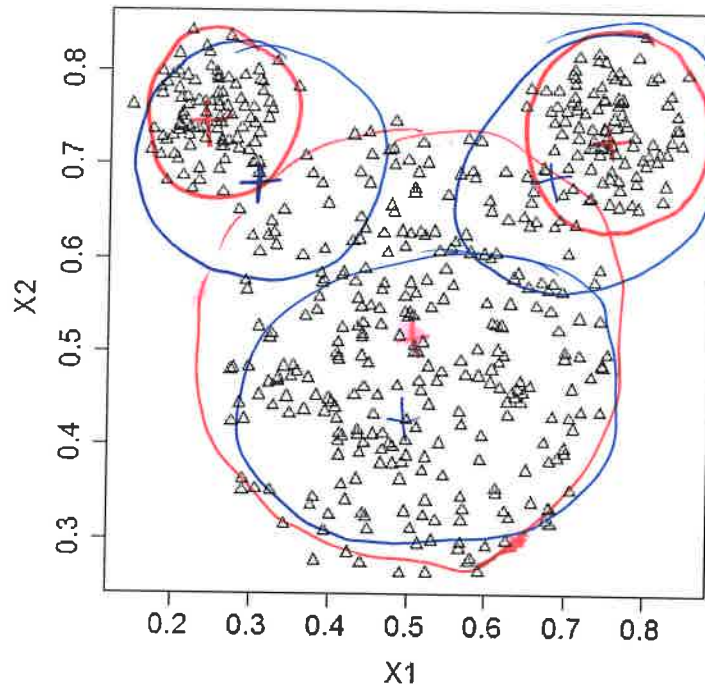
4

1

Handwritten text, possibly a signature or date, located in the lower right quadrant of the page.

## K-Means (10 Points)

6. How would K-means cluster the following data? Indicate the cluster centers and the rough clusters on the graph. Explain your choices briefly.



~~Can one the~~  
Circles

Intuitively we would expect a mickey mouse structure like the red clusters.

But K-means is distance based, so the three clusters would be more equal-sized. Leading to something closer to the blue.





## Linear Classifiers (20 Points)

7. Give the gradient descent update rules for the following regularized loss function:  $L = e^{y\hat{y}} + \lambda|w|$ .

need  $\nabla_w L \approx \frac{\partial L}{\partial b}$

$$L = e^{y\hat{y}} + \lambda|w| = e^{y(\omega \cdot x + b)} + \lambda|w|$$

$$\frac{\partial L}{\partial b} = ye^{y\hat{y}}$$

$$\nabla_w L = \nabla_w e^{y(\omega \cdot x + b)} + \nabla_w \lambda|w|$$

$\downarrow \qquad \qquad \downarrow$

$$xye^{y\hat{y}} + \vec{\lambda}$$

$$b = b - \eta ye^{y\hat{y}}$$

$$w = w - \eta (xye^{y\hat{y}} + \vec{\lambda})$$

$$|w| = |w_1| + |w_2| + \dots$$

$$\frac{\partial |w|}{\partial w_i} = 1$$

so we have  
a vector of 1s.

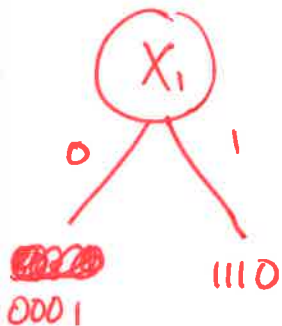


## Decision Tree (20 Points)

8. Build a decision tree using the greedy information gain algorithm from class using the following training data. Given a tie between features  $x_i$  and  $x_j$  choose  $x_i$  such that  $i < j$ . That is, if there is a tie between  $x_2$  and  $x_3$ , choose  $x_2$ .

Sample	$x_1$	$x_2$	$x_3$	$y$
$s_1$	0	0	0	0
$s_2$	0	0	1	1
$s_3$	0	1	0	0
$s_4$	0	1	1	0
$s_5$	1	0	0	1
$s_6$	1	0	1	1
$s_7$	1	1	0	1
$s_8$	1	1	1	0

$H = 1$  (#0 = #1 = 4)



$$H(x_1=0) = -\frac{3}{4} \log \frac{3}{4} - \frac{1}{4} \log \frac{1}{4}$$

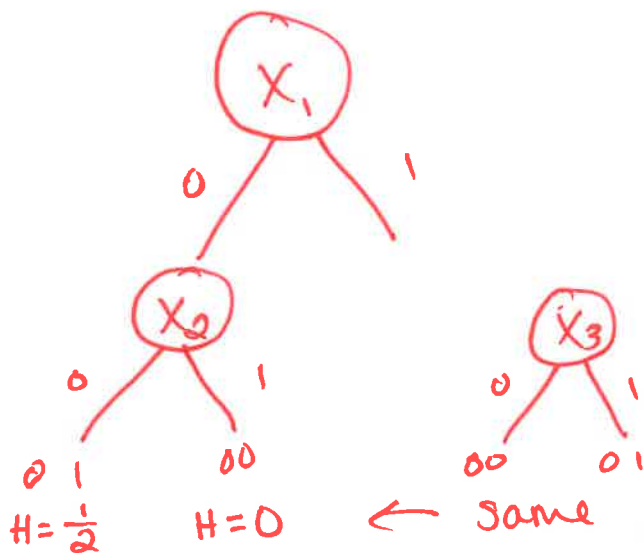
$$0.311 + 0.5 = 0.811$$

$$H(x_1=1) = -\frac{1}{4} \log \frac{1}{4} - \frac{3}{4} \log \frac{3}{4}$$

same.

$$IG(x_1) = 1 - \frac{1}{2}(0.811) - \frac{1}{2}(0.811) = 0.199$$

Choose  $x_1$  to start tree



$$IG(x_2) = 0.811 - \frac{1}{2} \cdot \frac{1}{2} - \frac{1}{2} \cdot 0$$

$$= 0.561$$

choose  $x_2$ .



Same IG as  $x_1$  split

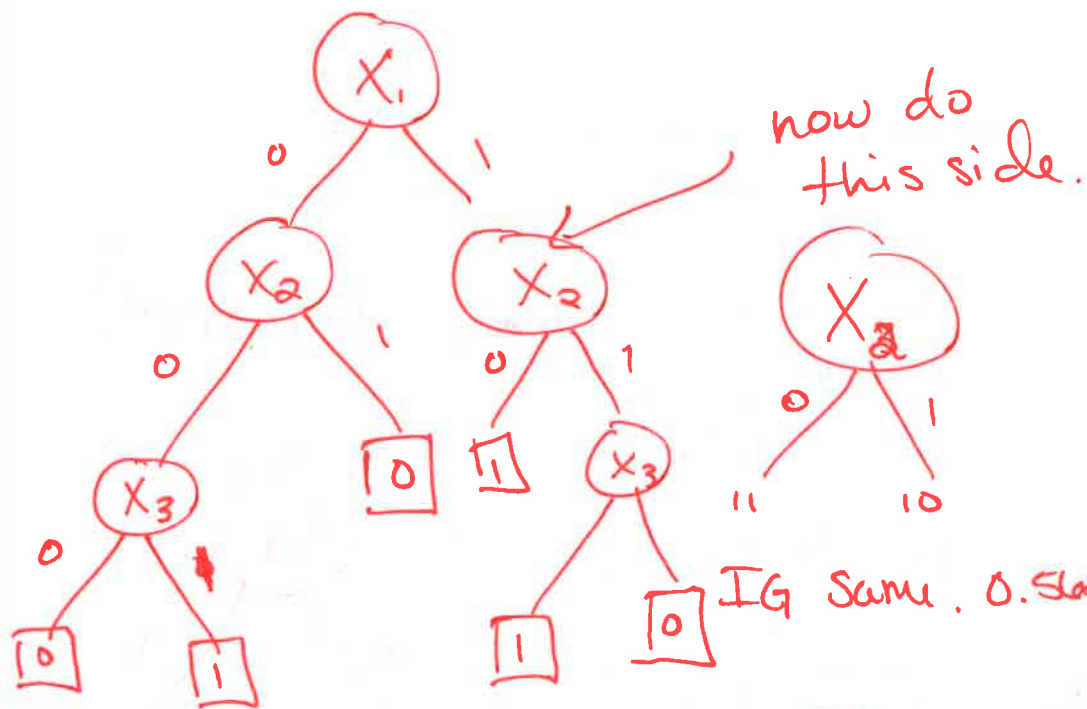


$$H(x_3=0) = -\frac{1}{2} \log \frac{1}{2} - \frac{1}{2} \log \frac{1}{2}$$

$$= 0.5 + 0.5 = 1$$

$$H(x_3=1) = \text{same}$$

$$IG(x_3) = 1 - 1 = 0$$



IG same. 0. Stop

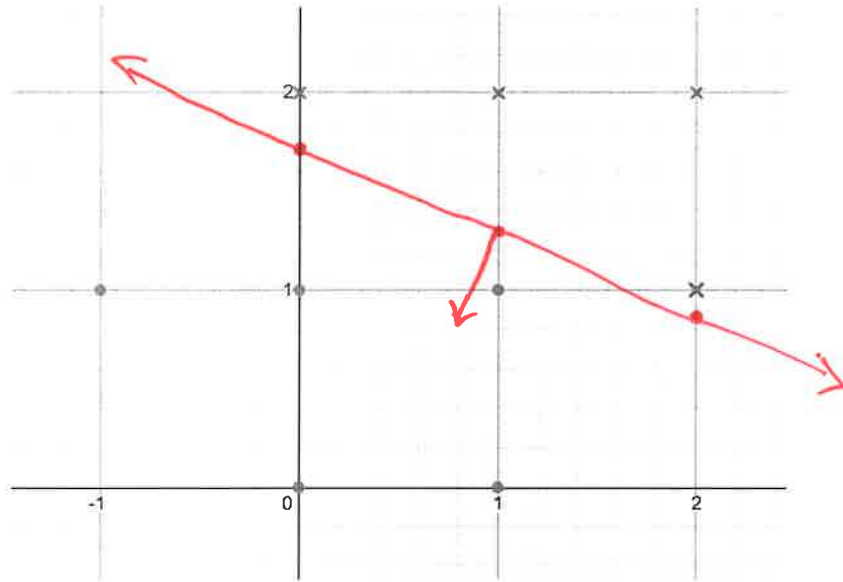
done!



Same IG.

## Perceptron (10 Points)

9. Give a  $w$  and  $b$  for a linear classifier that perfectly classifies the following data ( $x$  = negative, circle = positive). Show/explain how you came up with these values.



Slope of the line = ~~scribbled out~~  $-\frac{4}{10} = -\frac{2}{5}$

$w$  points towards + class (•)  
 $(-2, -5)$

$$b = -\text{intercept} \times w_2 = -1.7 \times -5 = +8.5$$

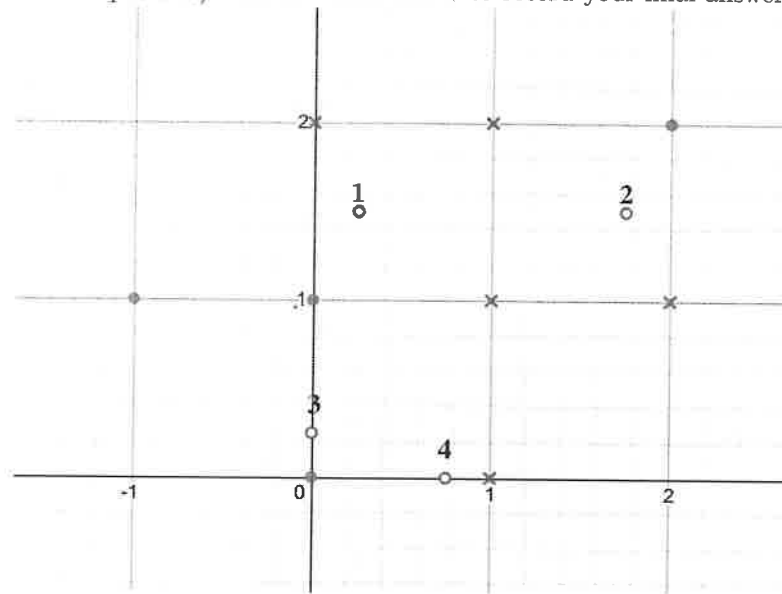
$$\begin{matrix} w & b \\ (-2, -5) & 8.5 \end{matrix}$$

There are many answers to this. I picked a weird line...



## KNN (10 Points)

10. Using  $K=1, 3$  and  $5$ , classify the test data (open circles) using the training data ( $x$  = negative, filled circle = positive). Use the table below to record your final answers.



Sample	$K = 1$	$K = 3$	$K = 5$
1	either	X	X
2	either	X	X
3	●	●	●
4	X	X	X

## Equations

### Entropy and Information Gain

$$H = \sum_{c \in C} -p(c) \log_2(p(c))$$

$$IG = H - \sum_{t \in T} p(t) H(t)$$

$p$	$p \log(p)$
$\frac{1}{8}$	-0.375
$\frac{1}{4}$	-0.5
$\frac{3}{8}$	-0.53
$\frac{1}{2}$	-0.5
$\frac{5}{8}$	-0.423
$\frac{3}{4}$	-0.311
$\frac{7}{8}$	-0.168
1	0

### Adaboost

---

#### Algorithm 32 AdaBoost( $\mathcal{W}, \mathcal{D}, K$ )

---

```

1:  $\mathbf{d}^{(0)} \leftarrow \langle \frac{1}{N}, \frac{1}{N}, \dots, \frac{1}{N} \rangle$  // Initiali
2: for  $k = 1 \dots K$  do
3:    $f^{(k)} \leftarrow \mathcal{W}(\mathcal{D}, \mathbf{d}^{(k-1)})$ 
4:    $\hat{y}_n \leftarrow f^{(k)}(x_n), \forall n$ 
5:    $\hat{\epsilon}^{(k)} \leftarrow \sum_n d_n^{(k-1)} [y_n \neq \hat{y}_n]$ 
6:    $\alpha^{(k)} \leftarrow \frac{1}{2} \log \left( \frac{1 - \hat{\epsilon}^{(k)}}{\hat{\epsilon}^{(k)}} \right)$ 
7:    $d_n^{(k)} \leftarrow \frac{1}{Z} d_n^{(k-1)} \exp[-\alpha^{(k)} y_n \hat{y}_n], \forall n$ 
8: end for
9: return  $f(\hat{x}) = \text{sgn} [\sum_k \alpha^{(k)} f^{(k)}(\hat{x})]$ 

```

---

### Perceptron

$$a = w \cdot x + b$$

$$w = w + xy$$

$$b = b + y$$