

# 6.034 Quiz 3

## 10 November 2010

Name	
email	

Circle your TA and recitation time **(for 1 point)**, so that we can more easily enter your score in our records and return your quiz to you promptly.

TAs
Martin Couturier
Kenny Donahue
Daryl Jones
Gleb Kuznetsov
Kendra Pugh
Mark Seifter
Yuan Shen

Thu	
Time	Instructor
1-2	Bob Berwick
2-3	Bob Berwick
3-4	Bob Berwick

Fri	
Time	Instructor
1-2	Randall Davis
2-3	Randall Davis
3-4	Randall Davis

Problem number	Maximum	Score	Grader
1	50		
2	50		
Total	100		

**There are 8 pages in this quiz, including this one, but not including blank pages and tear-off sheets. Tear-off sheets are provided at the end with duplicate drawings and data. As always, open book, open notes, open just about everything, including a calculator, but no computers.**

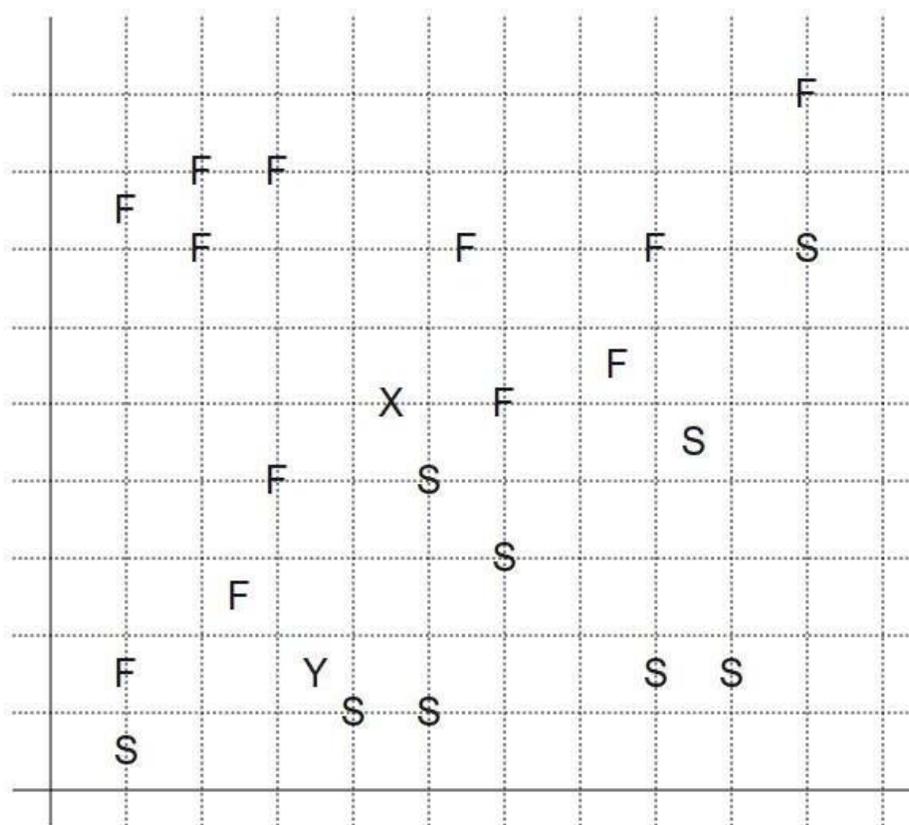
# Problem 1: Learning (50 points)

Alice and Bob, a pair of 6.034 students, traveled to DC last weekend for a rally. Over Saturday night, they attended a cocktail party for rally-goers. They knew almost everyone there, but there were a couple of really interesting party crashers.

Alice and Bob decided to use their 6.034 skills to figure out whether the party crashers were at the rally to promote Fear, or restore Sanity.

## Part A: Nearest Neighbors (25 points)

During the party, Bob suggests that they look at who the party crashers were spending their time with, given what Bob and Alice know about their friends' reasons to attend the rally (with either Fear/"F" or Sanity/"S").



### A1: Draw KNN (10 points)

Alice sketches the above drawing on a napkin, indicating the party crashers, X and Y, and the leanings of their friends, indicated by "F" or "S." She then draws nearest neighbor decision boundaries.

**On the above graph, draw the decision boundaries produced by k-nearest-neighbors where k=1 and distance measure is Euclidean distance.**

## A2: More KNN (15 points)

Based on Alice's decision boundaries, **what are the classifications for X and Y?**

X =

Y =

Alice changes her mind and decides that those boundaries aren't quite right, and tells Bob they should switch to using  $k=3$ . “Why? That's so hard to draw!” Says Bob. “I think  $k=1$  boundaries are too specific,” says Alice. **What's the name for the problem with  $k=1$  decision boundaries?**

She decides to classify the party goers using  $k=3$ . **If  $k=3$ , what are the classifications for X and Y?**

X =

Y =

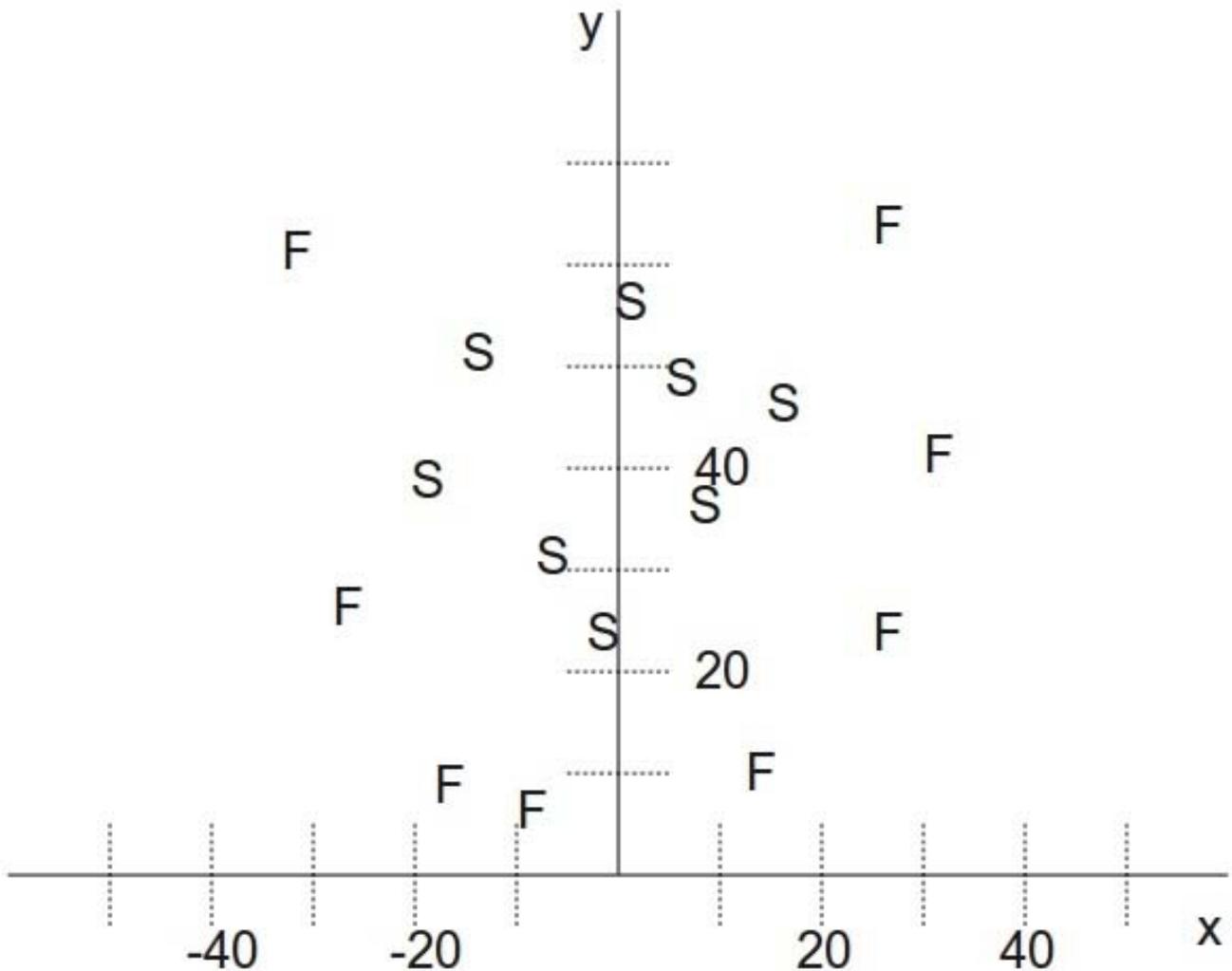
“Okay okay! Based on what you just said, how about  $k=21$ ?” Alice says “I don't think that'd be a good classifier either.” **What's the problem Alice has with  $k=21$ ?**

## Part B: ID Trees (25 points)

Alice and Bob give up on classifying the party goers using who they stand near at the party. “Why not look at where they stood during the event?” says Bob. He then pulls up a high-resolution satellite image of the event on his smart phone, zooms and enhances, picks out his and Alice's friends, and sketches all their relative positions on a separate napkin.

Here's the picture he gets. He and Alice argue about the distance their friends were spread out over the event, so he puts in distance from the stage, as well as spread from the center of the mall:

NOTE: lowercase  $x$  and  $y$  are axes, measuring distance from the stage( $y$ ) and the center of the mall( $x$ ). There are 16 friends total.



**B1: ID Trees (15 points)**

Using the greedy heuristic, determine the decision boundaries Bob draws for ID trees. **Ties are broken by: vertical lines before horizontal lines, lesser values before greater values.** Draw them on the picture, above, and write the equations in the box below. The numbers in your equations need only be approximate values; we know you cannot produce exact values from the diagram.

What's the disorder associated with the **first** of the decision boundaries? You may express your answer in terms of logarithms.

Draw the resulting decision tree in the space below. Order your branches such that the less-than-threshold branch is left of the other branch.

**B2: A Better Way (10 points)**

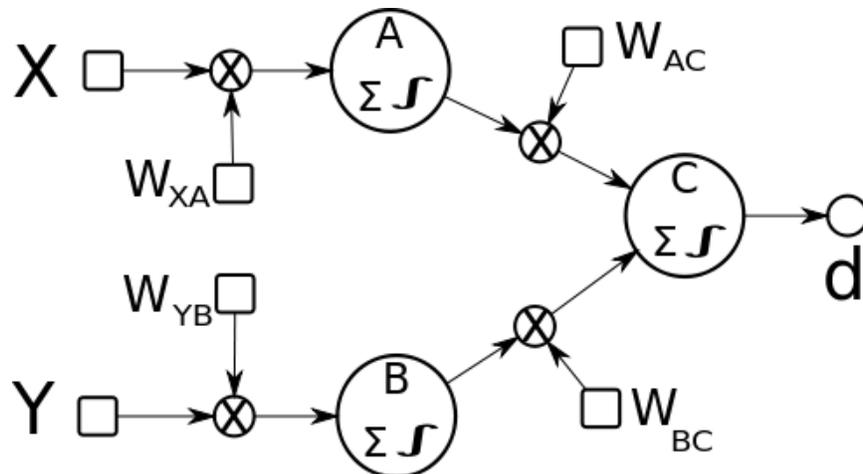
Alice suggests that if they change their representation of the data, she and Bob may have an easier time creating decision boundaries. **Briefly describe how you would change how this data is represented.**

**Based on your representation, what are the new decision boundaries and associated disorder?**

# Problem 2: Neural Nets (50 points)

## Part A: Warmup (25 points)

For the network below, answer the following questions:



**A1: Simulate Forward Propagation. (15 points)** Compute and fill in the values in the table below. Leave numerical answers to 2-decimal precision. You may use the sigmoid table to help with your calculations. **Note that there are no threshold weights in this network.**

X	Y	$W_{XA}$	$z_A$	$o_A$	$W_{YB}$	$z_B$	$o_B$
60	70	1			-1		

		$W_{AC}$	$W_{BC}$	$z_C$	$o_C$	d	$d - o_C$
		1	0			1	

Table of relevant values of the sigmoid function

x	s(x)	x	s(x)
< -50	0.00	0	0.5
-10	$4.5 \times 10^{-5}$	1	0.73
-5	0.01	2	0.88
-3	0.05	3	0.95
-2	0.12	5	0.99
-1	0.27	> 50	1.00

**A2: Back Propagation: (10 points)**

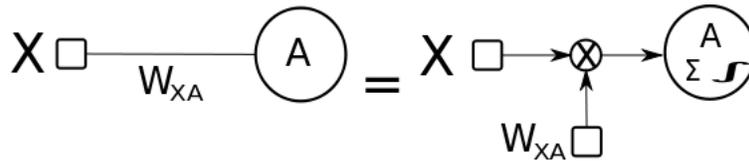
Compute numerical values for weight updates for back propagation. Write out the full expressions you are calculating for partial credit. Assume that the learning rate  $\alpha = 1$ .

$\Delta W_{AC}$

$\Delta W_{XA}$

**Part B: Multi-class Output (25 points)**

**NOTE: For Networks from this point on, we will adopt the abbreviated network notation.**



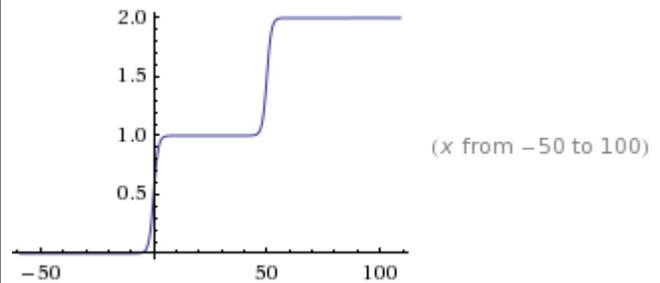
One possible method for making neural nets capable of multi-class classification is to change the sigmoid function. Inspired by the 6.034 GPA function, Yuan decides to adopt a 2-step sigmoid function as the output of the sigmoid unit, creating neural nets that can output roughly 3 values, 0, 1, and 2. The 2-step sigmoid  $S_2(x)$  has the following equation.

New Output function:

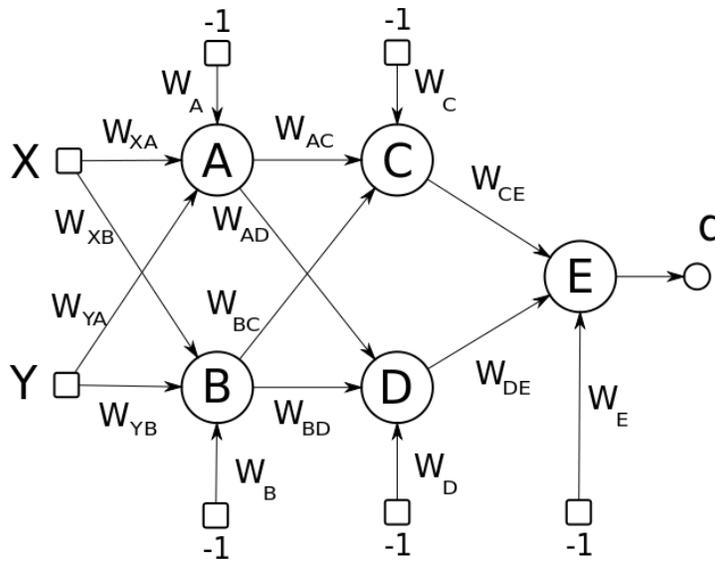
$$s_2(x) = \frac{1}{1 + e^{-x}} + \frac{1}{1 + e^{-(x-k)}} = s(x) + s(x-k)$$

New performance Function:

$$P_2(o) = -\frac{1}{2} \left( \frac{d-o}{2} \right)^2$$



For instance, when  $k = 50$ , the sigmoid  $S_2(x)$  would have the graph shown at the right. Thus, the output is roughly, 0 when  $x$  is  $< 0$ , 1 when  $0 < x < 50$ , and 2 when  $x > 50$ . Changing the sigmoid function triggers a similar change in the performance function in order to normalize range of values of the error.



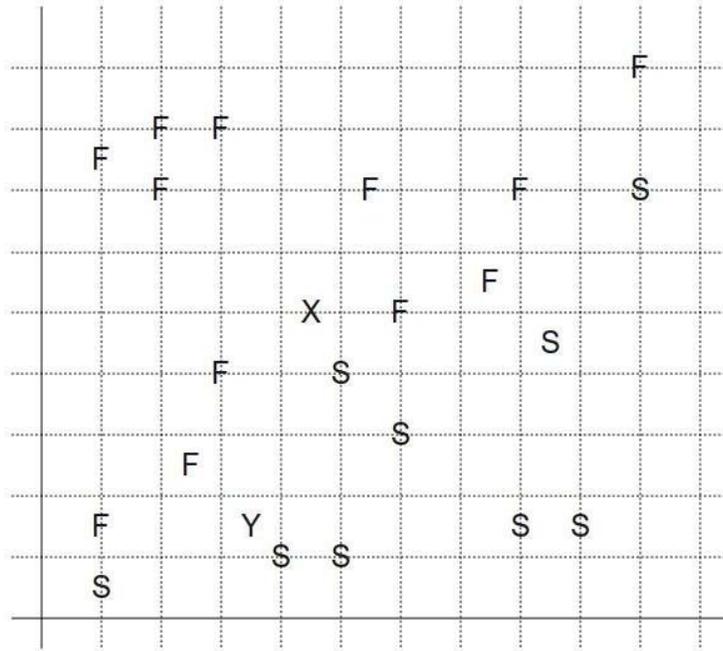
**B1. (9 points)** For the neural network, given above, where the sigmoid units use  $S_2$  and the performance function is  $P_2$ , write out the equation for  $\delta_E = \frac{\partial P_2}{\partial o_E} \frac{\partial o_E}{\partial z_E}$ . Express your answer in terms of  $d$ ,  $k$ ,  $s()$ , and  $z_E$  (the sum of weights times the inputs at node E). Hint:  $o_E = s(z_E) + s(z_E - k)$ .

**B2. (8 points)** Write out the equation for  $\delta_C$ . Express your answer in terms of  $d$ ,  $k$ ,  $s()$ ,  $z_C$ , and any weights in the network or any answer you've computed before.

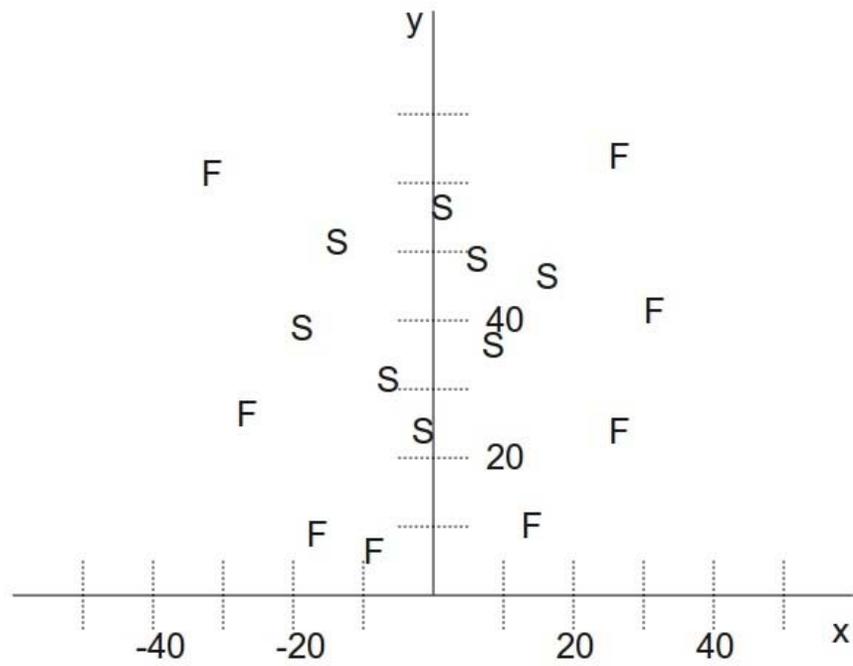
**B3. (8 points)** List all the weights that would be used in the **fully expanded calculation** for  $\mathbf{W}_B'$  (the new value of weight  $W_B$ ).

Tear off sheet, you need not hand this in.

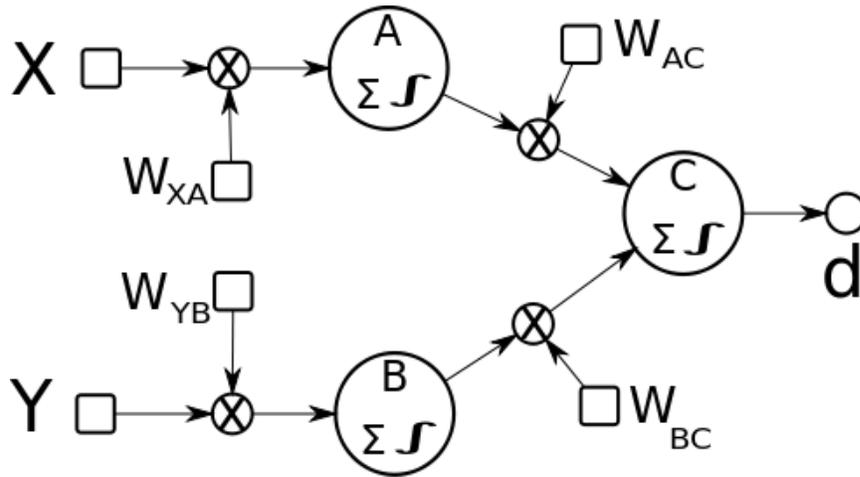
1A



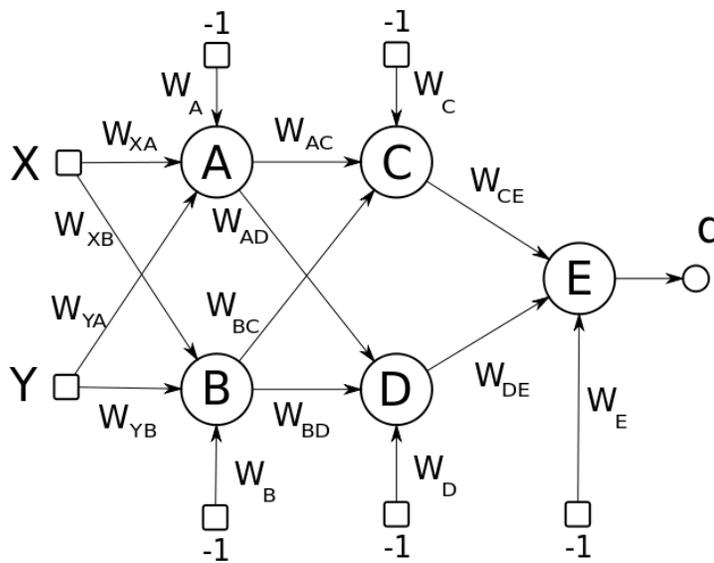
1B



2A



2B



MIT OpenCourseWare  
<http://ocw.mit.edu>

6.034 Artificial Intelligence  
Fall 2010

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.