6.034 Final Examination
December 18, 2007

Name	
EMail	

Quiz number	Maximum	Score	Grader
1	100		
2	100		
3	100		
4	100		
5	100		

There are 34 pages in this quiz, including this one. 10 pages of tear-off sheets are provided at the end with duplicate drawings and data. As always, open book, open notes, open just about everything.

Quiz 1, Question 1, Rules (50 points)

The primaries are coming upon us in New England, and the two frontrunners for the democrats, Barack and Hillary, are in a dead heat. Mike has been canvasing for Barack this fall and wants to know whether he will win the election. He decides to write an impartial set of rules to determine whether or not Barack will win the election. Below is a list of these rules:

```
P1:   (IF     ((?  x) is in the Midwest)
       THEN   ((?  x) is not important))

P2:   (IF     ((?  x) is in New England)
       THEN   ((?  x) is important))

P3:   (IF     ((?  x) is a senator near (?  l))
       THEN   ((?  x) has the advantage in (?  l)))

P4:   (IF     (AND  ((?  x) has the advantage in (?  l1))
                    (OR    ((?  x) has the advantage in (?  l2))
                           ((?  l2) is not important)))
       THEN   ((?  x) will win the primary))
```

Mike goes to the internet and finds a few facts about the candidates, which he subsequently codes up as follows:

A1: (Hillary is a senator near New Hampshire)
A2: (Barack is a senator near Iowa)
A3: (Hillary is not a senator near Iowa)
A4: (Barack is not a senator near New Hampshire)
A5: (Iowa is in the midwest)
A6: (New Hampshire is in New England)

Goal Trees

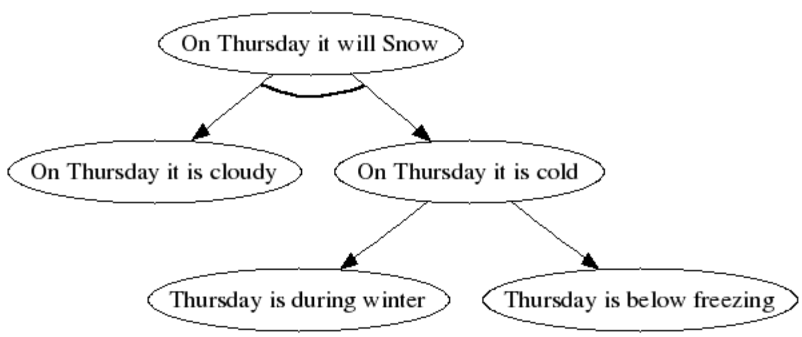
An example goal tree is shown below using the following rules:

P1: (IF (AND (On (? x) it is cloudy)
(On (? x) it is cold))
THEN (On (? x) it will snow))

P2: (IF ((? x) is during winter)
THEN (On (? x) it is cold))

P2: (IF ((? x) is below freezing)
THEN (On (? x) it is cold))

Don't forget that facts are nodes and edges indicate either an AND or an OR relationship. A goal tree for the proposition *(On Thursday it will snow)* will look like:



Part A, Backward Chaining (30 pts)

Essential Assumptions for backward chaining:

- When working on a hypothesis, the backward chainer tries to find a matching assertion in the database. If no matching assertion is found, the backward chainer tries to find a rule with a matching consequent. In case none are found, then the backward chainer assumes the hypothesis is false.
- The backward chainer never alters the database, so it can derive the same result multiple times.
- Rules are tried in the order they appear.
- Antecedents are tried in the order they appear.

Mike first wants to determine whether or not Barack will win the campaign. Draw the goal tree for the statement *Barack will win the primary.* Partial credit will be given for partial completion of the goal tree.

Draw the Goal Tree

(Barack will win the primary)

Part B Forward Chaining (20 pts)

Essential assumptions for forward chaining:

- Assume rule-ordering conflict resolution
- New assertions are added to the bottom of the dataset
- If a particular rule matches assertions in the database in more than one way, the matches are considered in the order corresponding to the top-to-bottom order of the matched assertions. Thus, if a particular rule has an antecedent that matches both A1 and A2, the match with A1 is considered first.

Mike then performs forward chaining on the original facts. Circle the rules that match the initial database.

P1	P2	P3	P4

What fact is *first added or deleted* from the database in this step?

.

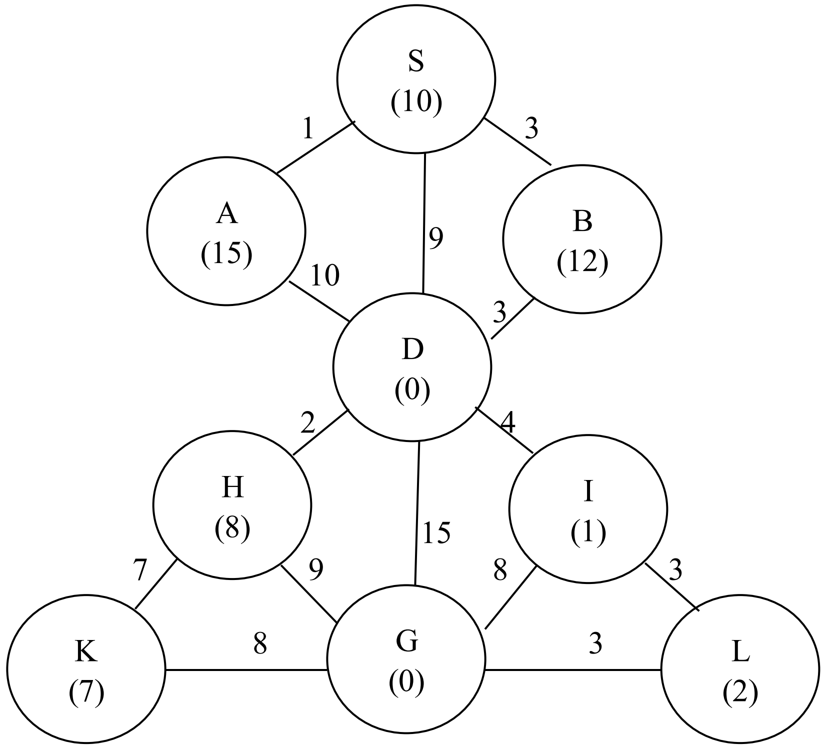
In the second step, circle the rules that **match** at the second iteration of the algorithm.

P1	P2	P3	P4

What new fact is *added or deleted* from the database in the second step?

.

Quiz 1, Question 2, Search (50 points)



You and some friends are lost in a strange city . You get into a big fight about the best way to find a path from your start position (S) to your goal (G), and so you split up and each try a different search technique. You and your friends always break ties in lexicographic order.

Part 2A (16 points)

Meredith decides to try Hill-Climbing using an extended list, because it seems like the easy way out. She asks for your help. Remember that we will never create a cycle by adding a node to the path that is already part of the path somewhere earlier and that you and your friends always break ties in lexicographic order.

a. What is the final path that Meredith will find?

b. How many times will Meredith need to backtrack?

Part 2B (22 points)

Christine suggests that you use an A* search using both the heuristic and an extended list—that way you will find the shortest path to your destination. She also asks for your help, and again, she suggests a search tree to help visualize the process, and to help provide you partial credit if you make a mistake. Remember that you and your friends always break ties in lexicographic order.

a. The tree

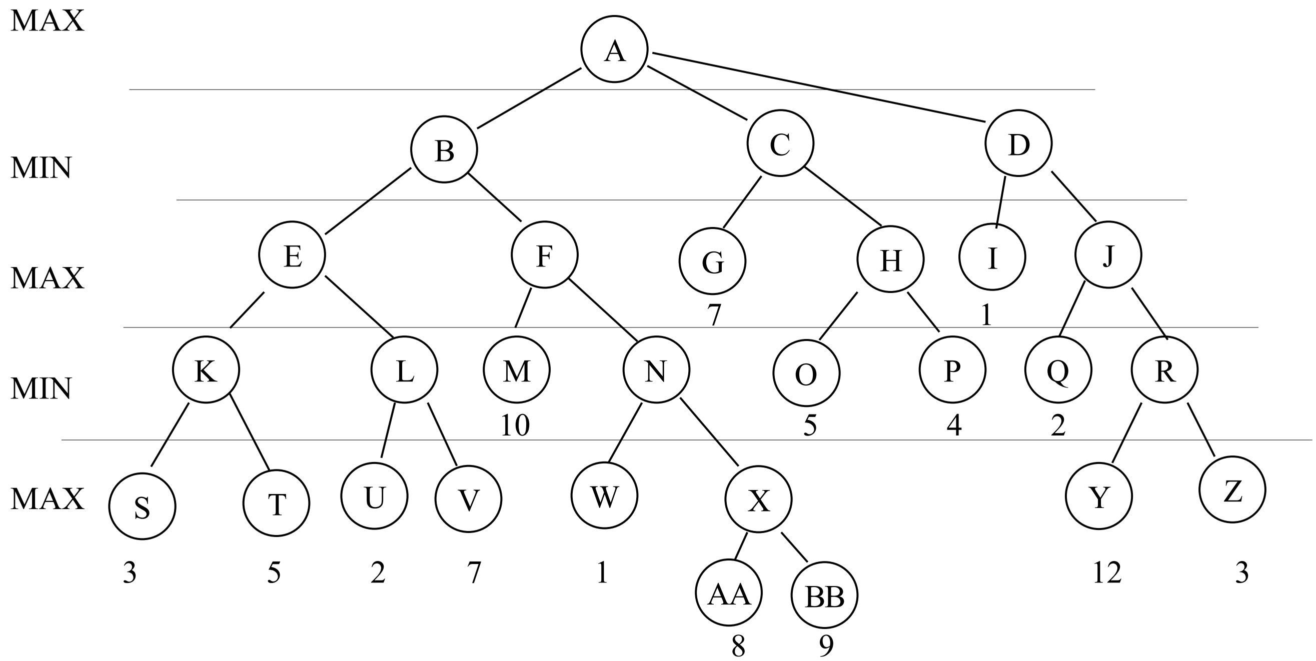
b. What is the final path that Christine will find?

c. How many paths will Christine extend during her search (remember to count the right thing—when she extends the path 'S' and retrieves the three paths 'SA', 'SB', 'SC', that is still only one path extended!). Also, do ***not*** count the final path to the goal as being extended.

Part 2C (12)

Kendra is still skeptical about the results so far. She proposes a branch-and-bound search instead. Will Kendra's search produce the same path as Christine's? If so, explain why. If not, explain why not.

Quiz 2, Question 1, Games (50 points)



In the game tree above, the values below the nodes are the static evaluations at those nodes. MAX next to a horizontal line of nodes means that the maximizer is choosing on that turn, and MIN means that the minimizer is choosing on that turn.

Part A (16)

Which of the three possible moves should the maximizer take at node A?

What will be the final minimax value of node A?

Part B (34)

Perform a minimax search with alpha-beta pruning, and list all the nodes that you statically evaluate, in the order of evaluation, below:

Quiz 2, Question 2, Constraint Satisfaction (50 points)

You've successfully completed 6.034 and are now working as a UROP with Professor Winston. One day, you come across a cryptogram puzzle, in which each of the letters represents a distinct number, in the lounge area:

```
  VS
+ NC
----
 HIE
```

Bored, you decide to use your 6.034 knowledge to solve the puzzle by treating it as a constraint satisfaction problem with each of the letters as a variable. There are some notes scribbled on the paper with the puzzle, from which you conclude that I = 3 and E = 0 and you learn that the carry digits, R1 and R2, are both 1.

```
R2 R1
  VS
+ NC
----
 H30
```

From the rest of the notes, you construct the following description of the domains of the variables.

V	1, 2, 5
N	4, 5, 7
S	3, 5, 8
C	1, 2, 7
H	0, 1

From the structure of the problem, you have the following constraints:

- S + C = E + 10
- V + N + 1 = I + 10
- H = 1

You also know that none of the values for V, S, N, C, H, I, E, can be the same.

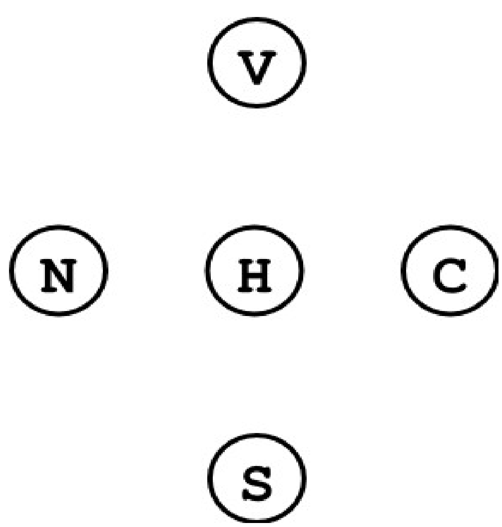
Part A (5 points)

Rewrite the constraints using the information you have about the values of the variables E and I and the other variables listed in the table.

- S + C =
- V + N =
- H =

Part B (5 points)

Draw lines in the graph below between nodes whose values are dependent on each other according to the 3 constraints above. **Do not draw in the no-two-have-same-value constraint, of course, because then every pair would be connected.**



Part C (26 points)

Fill in the tree for a backtracking search with forward-checking. That is, check the domains of the neighbors of the just-assigned variable. Remember that each letter in the original puzzle must have a different value.

•

V

N

S

C

H

Part D (14 points)

What are the final assignments in the puzzle?

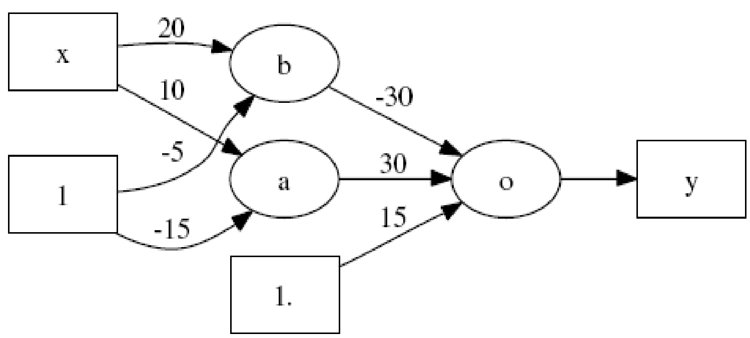
V	S	N	C	H	I	E

Quiz 3, Question 1, NN and NN (60 points)

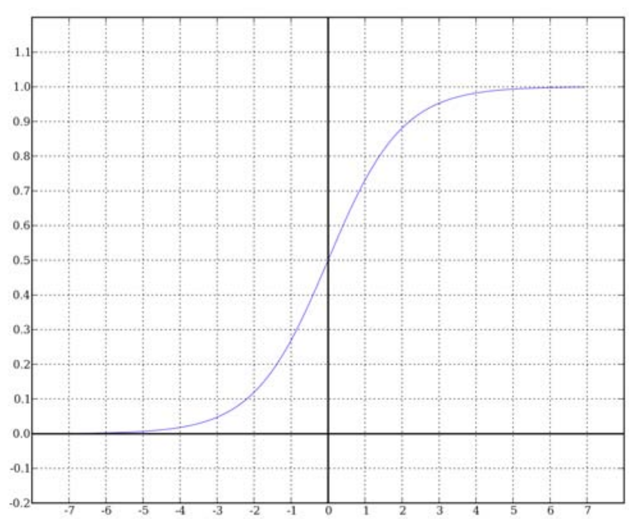
Part A, Neural Networks (40 pts)

David wants to determine how much Larry will like a movie based on its length. Like or dislike of a movie is quantified as a scalar between 0 and 1 (0 being a really bad movie and 1 being a great movie). David expects that really short movies (e.g. Youtube) will often get high ratings, middle length movies will be disliked (because characters don't have enough time to be developed), normal length movies will be acceptable, and really really long movies will be highly disliked.

He gets the following three [x, y] sample pairs: [0,1], [1, 0], and [2, 1] (x is the length of the movie in hours and y is the movie rating).



Nodes a, b, and o are all sigmoid. Use the following graph for reference:



Part A1 (10 points)

Compute the feed-forward output for x=0, 0.25, 1, and 2.

x=0

.

x=0.25

.

x=1

.

x=2

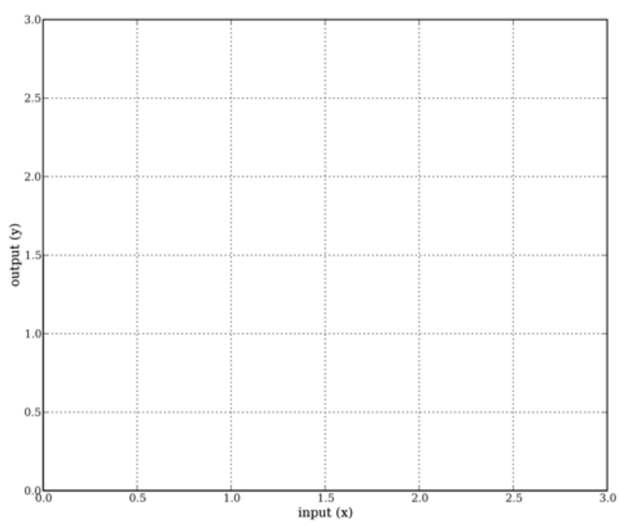
.

Does this approximate the training data well?

.

Part A2 (10 points)

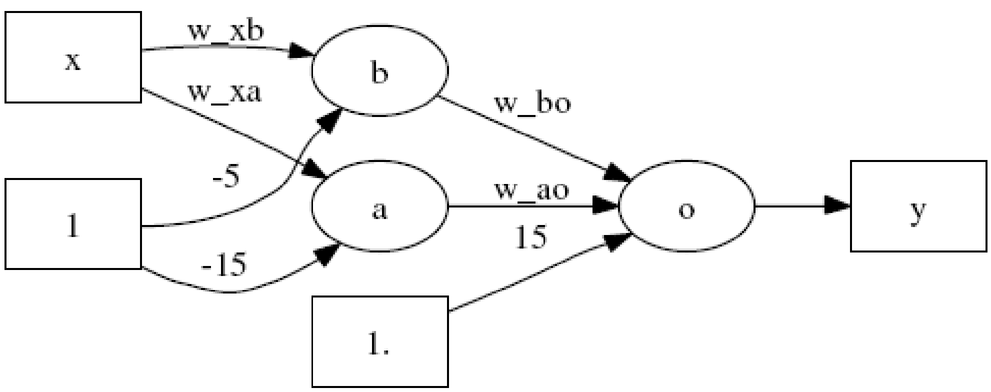
Sketch the resulting (continuous) function that the neural network has approximated below:



David has a test dataset of a single point: *x*=3. For this value, we know the output should be *y*=0 (he really doesn't like long movies). Will the neural net give a good approximation for this test dataset?

Part A3 (20 points)

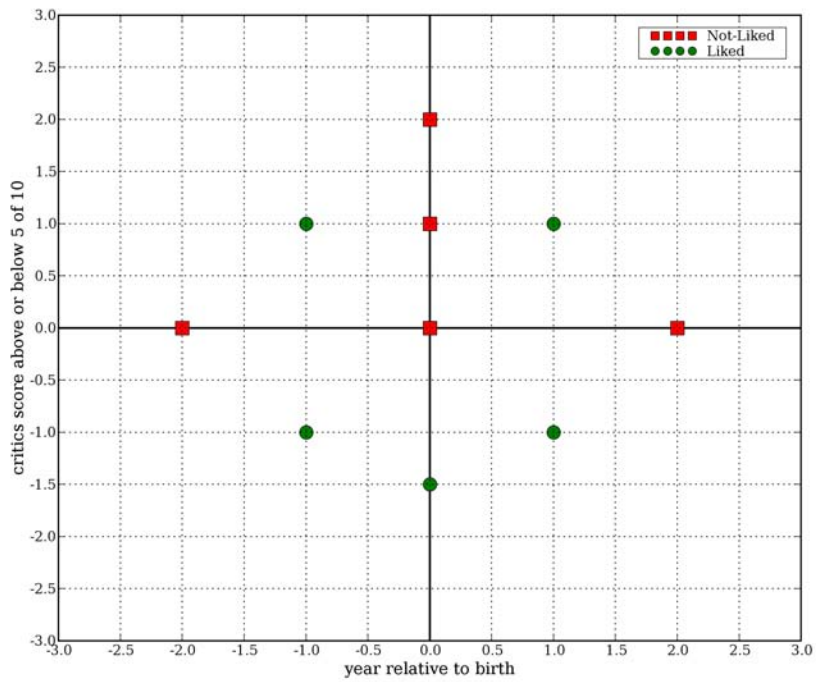
For *x*=0.25, perform backpropagation using the fact that the output should have been 1.0. Give the updated weights for the entire network excluding bias terms; that is, calculate values for each of the four weights, listed as w_xa, w_xb, w_bo, and w_ao. Assume that the learning rate is 1.0.



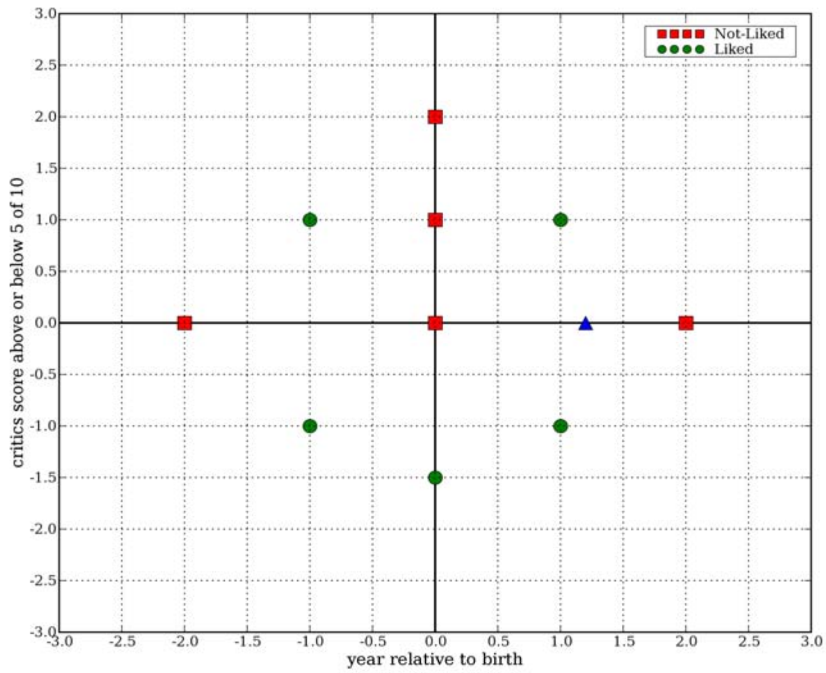
Part B, Nearest Neighbors (20 pts)

Part B1 (14 points)

David decides to use nearest neighbors to solve this problem and plots the liked/not-liked data. Squares are not-liked and circles are liked. **Sketch the resulting decision boundary** on the figure below.



It is the end of the month and David's friend Bob comes over with new data hot off the presses (the triangle). He wants David to analyze whether or not Larry would like the new movie.

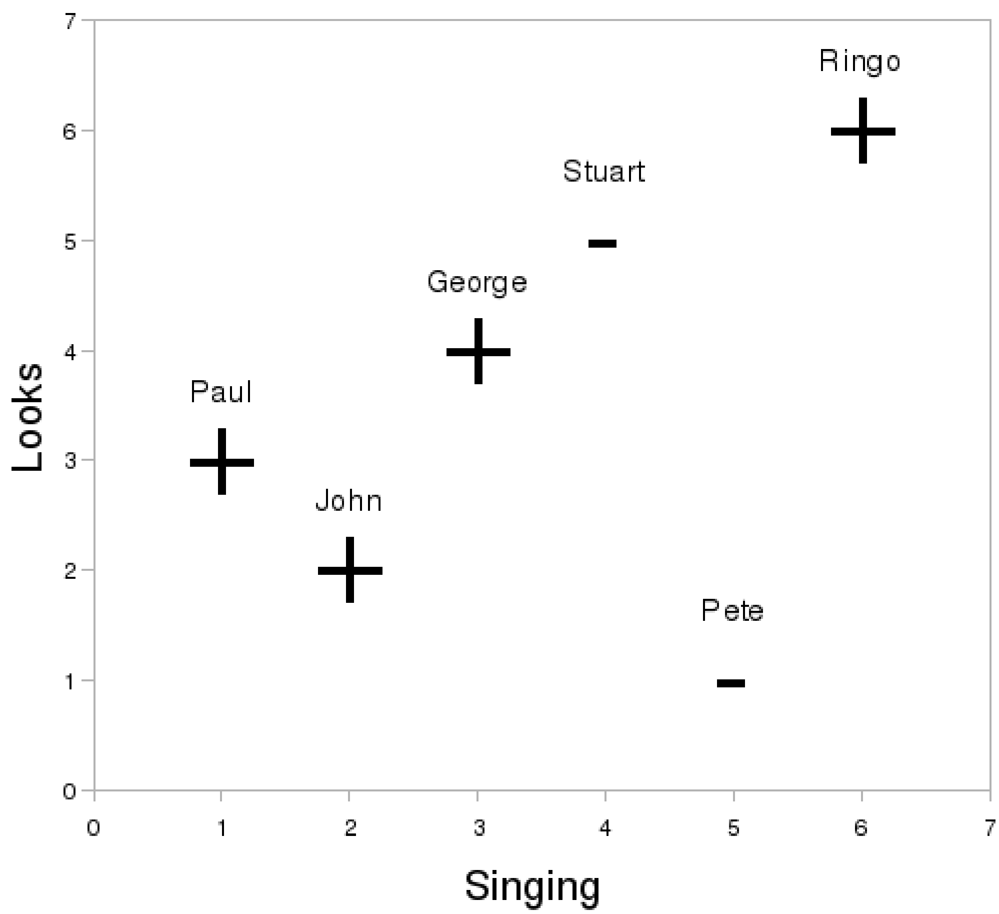


Part B2 (2 points) What is the nearest neighbor classification of the new movie?

Part B3 (4 points) He's not too sure about this classification and decides to rerun it using k-nearest neighbors for k=3 and then for k=5. Will Larry like the movie for these values of k?

Quiz 3, Question 2, ID Trees (40 points)

The Beatles had 6 members, but only 4 stayed with the band long enough to become famous. You are curious why the other 2 didn't make it big, so you create the table below. It ranks the musicians on their singing ability and looks in the rank diagram below.



Name	Singing rank	Looks rank	Famous?
John Lennon	2	2	Yes
Paul McCartney	1	3	Yes
George Harrison	3	4	Yes
Ringo Starr	6	6	Yes
Pete Best	5	1	No
Stuart Sutcliffe	4	5	No

You decide to flex your 6.034 muscles and use an ID-tree to understand why some Beatles became famous and some didn't. You warm up by practicing your minimum entropy heuristic.

Part A (10 points):

What is the disorder for the boundary at Singing=5.5? (You may use $\log_2$ in your answer.)

What is the disorder for the boundary at Looks=3.5? (You may use $\log_2$ in your answer.)

Part B (15 points):

Draw the full ID tree using the minimum entropy heuristic. **Note that the boundaries in Part A may or may not be used.**

Part C (15 points):

If Pete or Stuart had become famous instead of one of the other Beatles, the ID tree you just built would have been smaller. **Circle who you would swap out, (same in both cases) and draw the 2 new ID trees.**

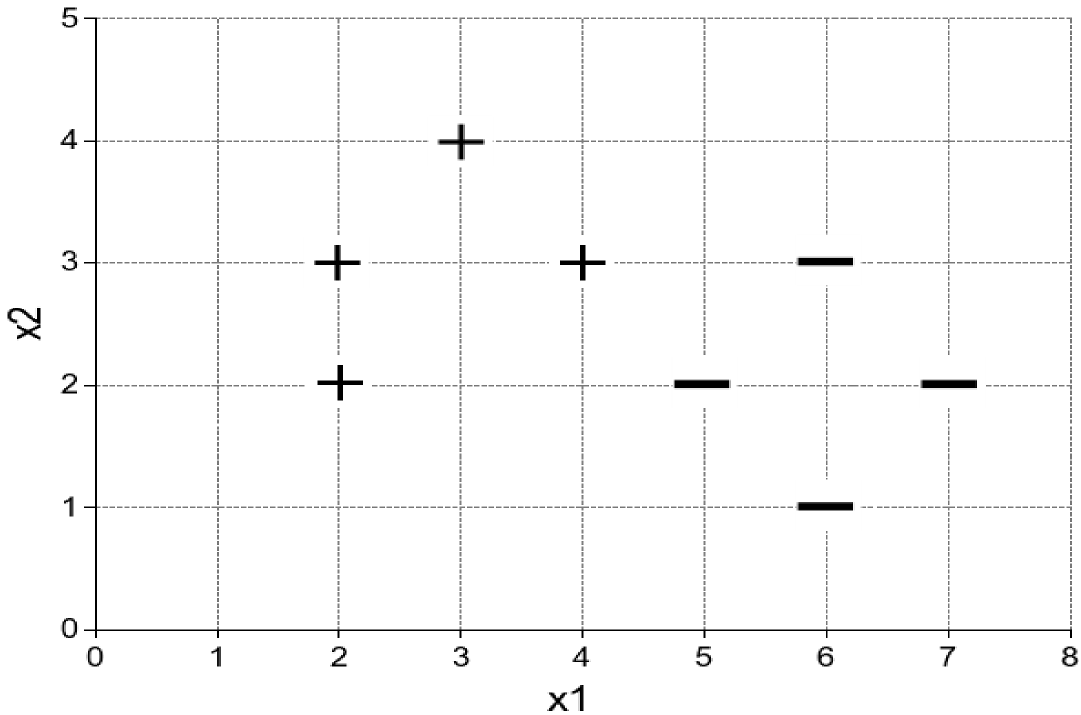
John Paul George Ringo

Pete becomes famous	Stuart becomes famous

Quiz 4, Question 1, SVMs (50 points)

Part A (38 points)

You want to use support vector machines to help detect steganography. That is, you want to classify digital images based on whether they have been altered to hide a message inside the bits. Based on several features of each image, you train a **linear** classifier on regular images and altered images. The altered images are marked below with '+' and the regular images are marked with '**-**':



Part A1 (4 points)

On the above plot, draw the **decision boundary** found by the linear SVM,

Part A2 (4 points)

Three vectors lie in the gutters defined by your decision boundary. Two of them are support vectors and one is not (that is, one has an alpha of zero). Circle the support vectors.

Part A3 (15 points)

What is the final classifier? Your answer is to be an arithmetic expression in terms of x1, x2, and constants.

What are the weights α for the two support vectors?

What is the width of the margin, between the two gutters?

Part A4 (15 points)

Suppose you found that one of the images in your training data had been mislabeled. If the point at (5,2) had a positive label instead of a negative label, what would the classifier be?

How many support vectors would there be?

What would be the width of the margin?

Part B (12 points)

There is no penalty for wrong answers, so it pays to guess in the absence of knowledge.

Part B1 (4 points)

Circle the **best** answer:

A support vector machine finds a classifier that:

a. has the most support vectors
b. has the maximum margin
c. minimized disorder
d. none of the above
e. all of the above

Using SVMs with kernels:

a. allows you to divide data into sets with the least disorder
b. increases your error rate on training data
c. can allow you to separate data that is not linearly separable
d. all of the above
e. none of the above

Part B2 (8 points)

True/False:

☐ SVMs are more resistant to overfitting than boosting classifiers.

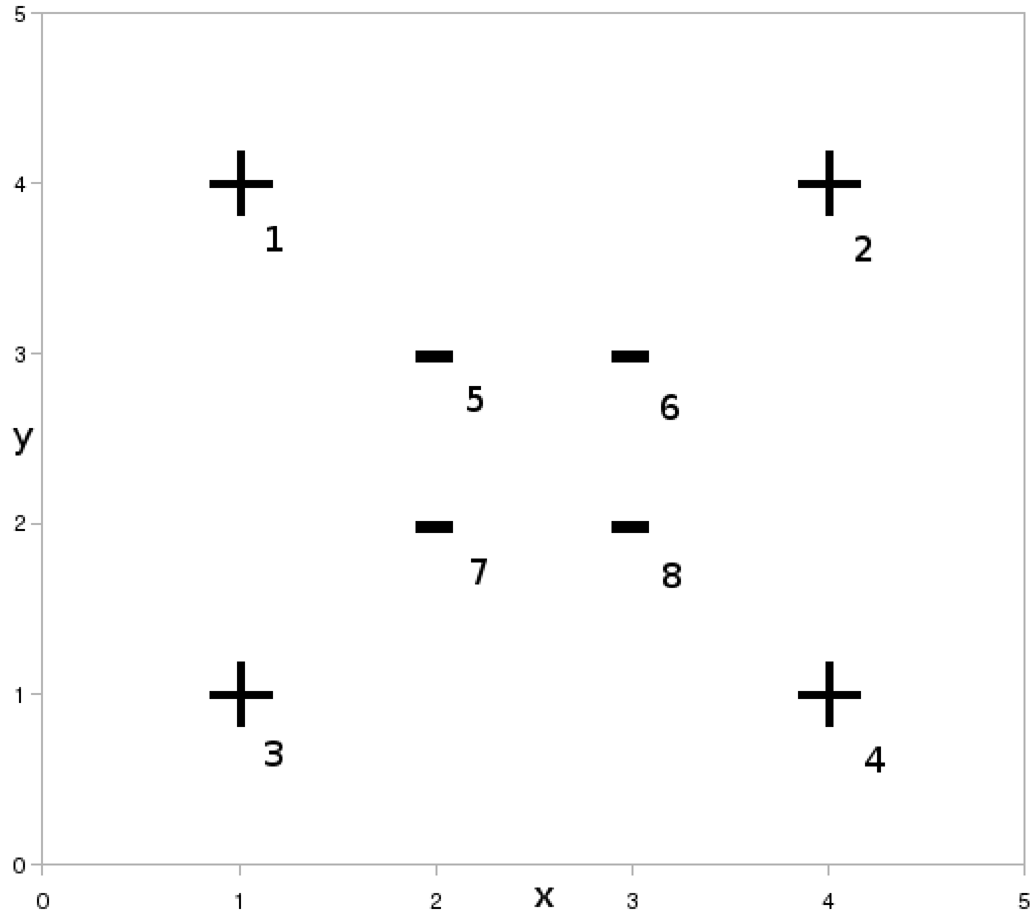
☐ SVMs are prone to getting stuck in local optima.

☐ A polynomial kernel of degree greater than 1 will always find a different decision boundary than a linear kernel will.

☐ SVM classifiers with more support vectors are less sensitive to noise in the training data.

Quiz 4, Question 2, Boosting (50 points)

In this problem you'll use Adaboost to classify this data set:



In all parts of the problem, use *only* these stubs as possible classifiers.

- **h1** = sign(y - 3.5)
- **h2** = sign(3.5 - y)
- **h3** = sign(y - 2.5)
- **h4** = sign(2.5 - y)
- **h5** = sign(y - 1.5)
- **h6** = sign(1.5 – y)

When choosing classifiers, break ties in order.

Part A (12 points)

Circle the first classifier that Adaboost picks.

h1 h2 h3 h4 h5 h6

What is the first classifier's α ? (You may use ln in your answers.)

What are the new weights?

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8

Part B (12 points)

Circle the second classifier that Adaboost picks.

h1 h2 h3 h4 h5 h6

What is the second classifier's α ?

What are the new weights?

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8

Part C (12 points)

Circle the third classifier that Adaboost picks.

h1 h2 h3 h4 h5 h6

What is the third classifier's α ?

What are the new weights?

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8

Part D (14 points)

As the number of terms in the Adaboost classifier goes to infinity, the sample weights converge upon limits. **What values do the weights approach?**

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8

Circle the next classifier Adaboost would pick based on these limit weights.

h1 h2 h3 h4 h5 h6

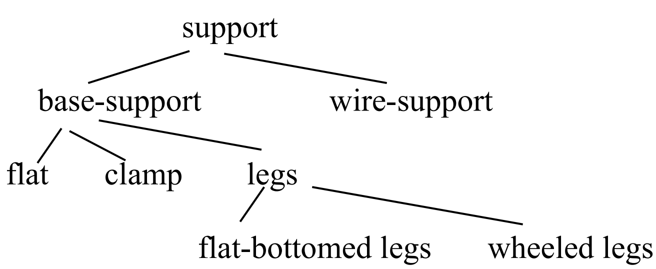
What is this final classifier's α ?

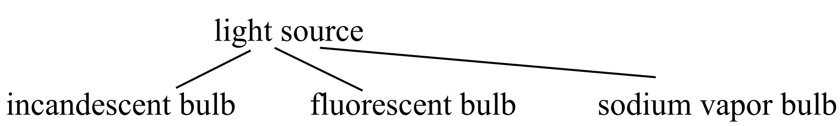
Quiz 5, Question 1, Near-Miss (50 points)

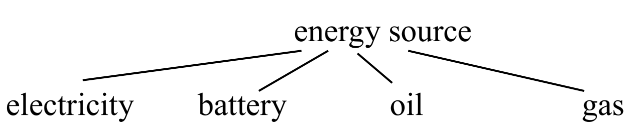
Part A (45 points)

You wish to teach Alvin the Robot the concept of TABLE LAMP using the samples below and the arch learning system. Given the samples; the knowledge below, represented as class hierarchies; and the starting model (specified on the next page), identify the heuristic used from the official list, **if any**, and provide a description of the model Alvin builds after being presented with each image. Your descriptions should be in the same form as the starting model.

Knowledge:







Official heuristics list includes forbid link, require link climb tree, drop link, and close interval. It is also possible for a sample to trigger no heuristic.

See the tear off sheet for explanations of what the heuristics do, but note that enlarge set is not to be used in this problem.

Starting model: incandescent bulb, height = 24", flat base, electricity, shade

Example 1, a hit (example of concept)

Incandescent bulb, height =11", flat base, electric, shade:

Heuristic:

New model:

Example 2, a hit (example of concept)

Incandescent bulb, height =11.5", flat base, electricity

Heuristic:

New model:

Example 3, a near miss

Incandescent bulb, height = 60", flat base, electricity, shade

Heuristic:

New model: _

Example 4, a hit (example of concept)

Fluorescent bulb, height =13", flat base, electricity, shade

Heuristic:

New model:

Example 5, a hit (example of concept)

Fluorescent bulb, height =14", wheeled legs base, electricity

Heuristic:

New model:

Part B (5 points)

Which model would you present to Alvin in order to teach that a table lamp requires a base?

1. Incandescent bulb, height = 6", flat base, electricity
2. Incandescent bulb, height = 8", battery
3. Incandescent bulb, height = 12", wire-support, electricity
4. Fluorescent bulb, height = 21", clamp base, electricity
5. Incandescent bulb, height = 12", flat-bottomed legs, electricity

3

1

2

4

5

Quiz 5, Problem 2, (25 points)

Circle the best answer for each of the following question. There is no penalty for wrong answers, so it pays to guess in the absence of knowledge.

Yip and Sussman set out to develop a theory of how:

- Syntactic rules can be learned using near-miss examples.
- Distinctive features can be identified in sound streams.
- Boosting can be used to learn tense information.
- A* search can be used to generate bird songs
- None of the above.

The Yip and Sussman learning algorithm

- Specializes from a pattern that matches almost nothing.
- Generalizes from a pattern that matches almost nothing.
- Constructs a seed pattern using arch learning.
- Does not use a seed pattern.
- None of the above.

Yip and Sussman

- Discovered boosting worked better than search
- Discovered search worked better than boosting
- Used radial-basis kernels with a support-vector machine
- Exploited the representation of phonemes as distinctive feature vectors
- None of the above

Winston focused on dogs and cats because

- The words have the same root in Sanskrit
- The words end with different phonemes
- The words both start with voiced phonemes
- The words have no voiced phonemes
- None of the above

The phonemes of each human language

- Almost fill up distinctive-feature space
- Occupy almost none of distinctive-feature space
- About half-fill distinctive-feature space
- Occupy about a thousand points in distinctive-feature space
- None of the above

Quiz 5, Problem 3, (25 points)

Circle the best answer for each of the following question. There is no penalty for wrong answers, so it pays to guess in the absence of knowledge.

Disorientation experiments with rats, children, and adults demonstrate

- Rats cannot find food in the presence of loud music
- Children become confused if each wall of a room has a different texture
- Rats cannot combine information about geometry with information about color
- Children and adults cannot reorient themselves as well after eating
- None of the above

Chomsky believes the distinguishing characteristic of human intelligence is

- A doubling of brain size relative to other primates
- An ability to connect concepts using causal relations
- An ability to tolerate aural ambiguity
- An ability to imagine events that never occur
- None of the above

Brooks subsumption architecture is best described as

- A way of building robots using interchangeable parts
- An approach to programming using recursion rather than loops
- An approach to building robots using layers specialized to behavior
- An approach to programming using short-term memory
- None of the above

Winston believes that to develop an account of human intelligence we must

- Find new ways to exploit boosting and support vector machines
- Focus on creating robots with serious application potential
- Understand tightly coupled loops connecting language with imagined events
- Collect commonsense knowledge using programs that read children's books
- Study differences between chimps and children
- None of the above

Among the **essential** qualities of a good representation, from an AI perspective is that

- It is compact
- It supports fast access
- It contains only logically consistent information
- It exposes constraint
- None of the above