Quiz 1 for 6.034

Name:

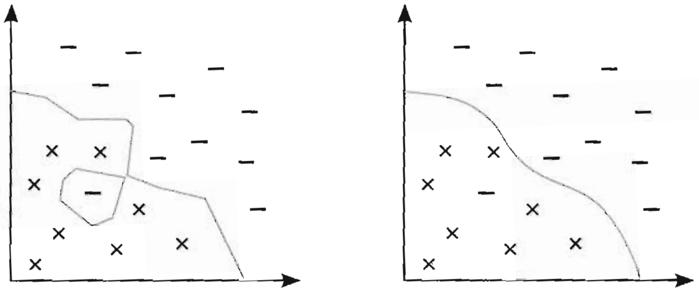
20%	20%	20%	20%

Good luck!

Question #1 30 points

1. Figure 1 illustrates decision boundaries for two nearest-neighbour classifiers. Determine which one of the boundaries belongs to the 1-nearest neighbour classifier and which one belongs to the 3-nearest neighbour classifier.

1NN



3NN

Decision Boundary 1

Decision Boundary 2

Figure 1: Nearest-neighbour decision boundaries

2. Consider the OR function defined over three binary variables: $(f(x_1, x_2, x_3) = x_1 \vee x_2 \vee x_3)$. Would it be possible to learn this function using the perceptron? Why or why not?

yes because the data set has only 1 point for the class y=0. In 3d, this 1 point can be separated from the others with a plane

3. Given a set of linearly separable training examples, we train the perceptron algorithm twice, initializing the weights differently for each run. The two training procedures traverse the data points in the same order and are run until convergence.

   - Would the two resulting classifiers have the same performance on the training set?

     yes

   - Would the two resulting classifiers have the same performance on the test set?

     no

4. All the classifiers we have studied in class were developed for binary classification (i.e., the label belongs to one of the two classes -1,1.) We would like to expand our classifiers for ternary classification. Which of the following algorithms can be used with no modifications:

   - decision trees
   - [✓ k-nearest neighbours]
   - perceptron
   - naive Bayes

5. We never test the same attribute twice along one path in a decision tree. Why not?

using the value again will not
↓ entropy

6. In class, we defined Laplace correction for a binary classifier that operates over binary features. For instance, $R_j(1,1)$ is computed as $\frac{\#(x_j^i=1 \wedge y^i=1)+1}{\#(y^i=1)+2}$. In this question, we consider two modifications to this classifier. In both cases, write a corresponding expression for $R_j(1,1)$.

- Assume a ternary classifier with binary features.

$$R_j(1,1) = \frac{\#(x_j^i=1 \wedge y^i=1)+1}{\#(y^i=1)+2}$$

- Assume a binary classifier with features that can take three values (e.g., 0,1,2).

$$R_j(1,1) = \frac{\#(x_j^i=1 \wedge y^i=1)+1}{\#(y^i=1)+3}$$

7. Is the decision tree algorithm presented in class guaranteed to find a tree with the minimal entropy?

no, the algorithm is greedy

8. Consider the following transformation of the feature space

$$\phi(x) = (3x + 5)$$

Compute the kernel $\mathbf{K}(\mathbf{x}, \mathbf{y})$ defined by this transformation.

$9 x \cdot y$

9. We apply the transformation described above to the XOR dataset. Would the perceptron converge in the new space?

no because the kernel is linear

10. We apply backward feature selection to eliminate n out of m features ($n < m$). How many times would a classifier be retrained in the process of this selection?

We start with m features

train m times with $m-1$ features in each set

then drop 1 feature + train $m-1$ times

etc.

so we got $m + (m-1) + (m-2) + \ldots + (m-n+1)$

$$= \frac{(m + (m-n+1)) \cdot n}{2}$$

Question #2 — 30 points

For each of the learning situations below, say what learning algorithm to use and why.

- You are developing a spam filter. Since spammers constantly change their strategy, you want your classifier to adjust accordingly. Every hour, your training set is augmented with millions of emails classified as spam or not spam. You assume that recent examples are more indicative of current spam patterns.

naive Bayes

- You want to build a binary classifier that identifies review sentiment, classifying it either as positive or negative. As features for this classifier, you use all the words in English vocabulary. It is known that your training data contains many erroneously labeled examples.

perceptron can work

knn not bad but high dimensionality is a problem for it

- You are aiming to develop a classifier that identifies the disease pseudititis based on the results of several tests. Any single test is a not sufficient indicator of the disease. It is the combination of the test results that can signal the presence of pseudititis.

Question #3

Consider a Bayesian approach to a real medical diagnosis problem. Many people who your urban health clinic in Lima, Peru, have tuberculosis (TB). You want to identify patients who might have TB, but it's too costly to test everyone who comes in. However, if doctors notice that you are coughing, bringing up phlegm on each cough and you have had this cough for more than two weeks, then there is a pretty high chance that in fact you may be suffering from TB. For example, one recent study by the US Centers for Disease Control and Prevention found that 12% of such patients actually have the disease.

These investigators were interested in what demographic factors predict the risk of a patient's having TB, and compiled the following table:

	minibus		**commute**		**family TB**		**TB history**		**not home**		**crowded**	
	Yes	No	≥ 1 hr	$<$ 1hr	Yes	No	Yes	No	Yes	No	Yes	No
TB	14	2	8	8	7	10	4	13	16	1	6	11
no TB	51	36	20	67	40	85	24	101	87	38	50	75

Note: not all people answered all questions. The features are: (1) whether the person commutes to work by **minibus**, (2) their **commute** time, (3) previous contact with **TB** cases in the **family**, (4) a **history** of pulmonary **tuberculosis**, (5) occupation away from **home**, and (6) over**crowded** conditions.

This table actually lists six *contingency tables* showing the number of times that, say for the first table, having TB is associated with riding a minibus to work or not, compared to similar numbers for those not having TB. This is a more compact representation than the *feature tables* you saw in lecture, which give the values for each feature and the outcome for each (in this case) patient. That table would have 142 rows and one column for each of the above features plus one for TB; the entries in each cell would only be 0 or 1.

Nevertheless, from these tables, we can easily compute the R's we described in lecture. For example, $R_{\text{minibus}}(1,1) = 14/16$, $R_{\text{minibus}}(0,1) = 2/16$, $R_{\text{minibus}}(1,0) = 51/87$, and $R_{\text{minibus}}(0,0) = 36/87$. This gives us the basis for using the prediction algorithm to suggest for each patient, depending on his or her answers to these questions, whether they have TB or not.

(a) In using Naive Bayesian classification, which of the six features above would give the greatest contribution in the prediction algorithm for a patient with TB?

(b) For a new patient for whom the prediction algorithm based on the first five features computes $S(1) = 0.004$ and $S(0) = 0.01$, what would be the resulting $S(0)$ and $S(1)$ after updating for the fact that the patient actually does *not* live in an overcrowded condition?

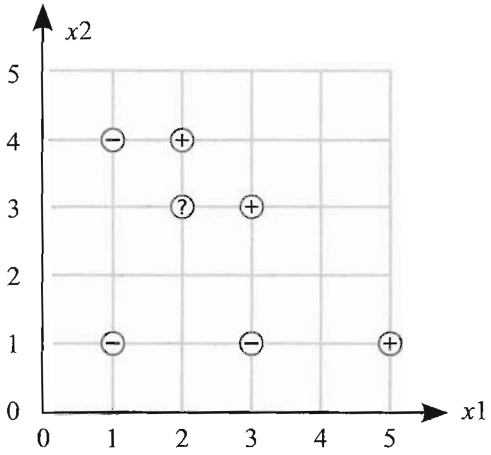
(c) Is your result in (b) consistent with the intuition that overcrowding is a risk factor for TB? Why?

(d) Suggest a set of numerical entries for the **crowding** table that would be consistent with the intuition suggested above, and would "tip the balance" in favor of a prediction of TB in the case described in (b).

(e) The prediction algorithm we have described does not explicitly take the *prior probability* of TB into account. How could we modify the prediction algorithm to make sure that in the absence of any other evidence, $S(1)/(S(0)+S(1)) = P_0(\text{TB})$, the prior probability of TB, which in our case is 12%?

Question #4

Perceptron



Data points are: Negative: (1, 1) (3, 1) (1, 4) Positive: (2, 4) (3, 3) (5, 1). Data points are classified as either +1 or -1. An unknown point is located at (2, 3)

1. Assume that the points are examined in the order given above, starting with the negative points and then the positive points. Simulate one iteration of the perceptron algorithm with a learning rate of 0.5 and an initial weight vector of (-15 5 3).

$W_{start} = (-15 \quad 5 \quad 3)$

y	x0	x1	x2	**W** (new values)
-1	1	1	1	same
-1	1	3	1	(-15.5 3.5 2.5)
-1	1	1	4	same
+1	1	2	4	same
+1	1	3	3	same
+1	1	5	1	same

2. What is the equation of this line using the final weights from part 1.

$-15.5 + 3.5 x_1 + 2.5 x_2 = 0$

3. Is this line a linear separator of the data?

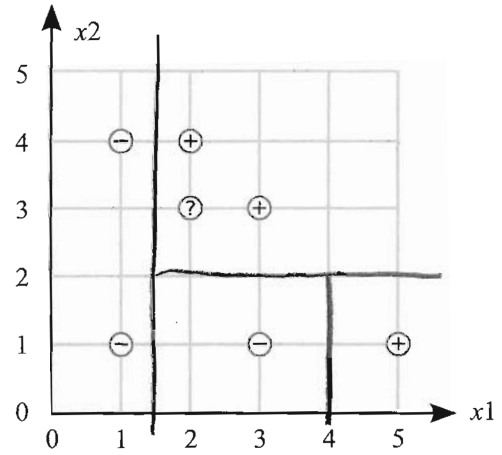
yes

4. Using this line, what would be the predicted class of the unknown point (2, 3)? What is the margin of this point using the predicted class?

⊖

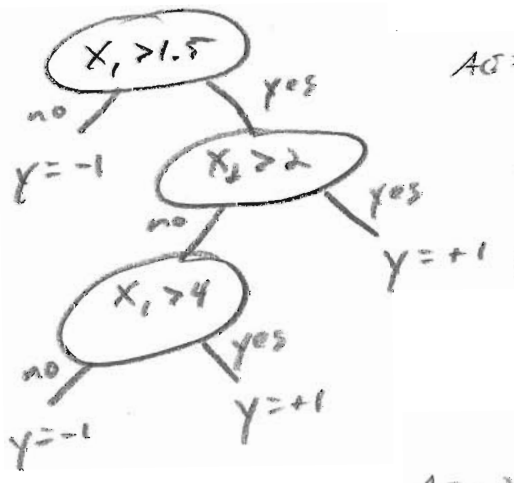
$-15.5 + 3.5 \times 2 + 2.5 \times 3 = -1$

Decision Trees



Data points are: Negative: (1, 1) (3, 1) (1, 4) Positive: (2, 4) (3, 3) (5, 1). Data points are classified as either +1 or -1. An unknown point is located at (2, 3)

1. Draw the decision tree boundaries on the graph above that would be used in a decision tree based on average entropy. Show below the average entropy of each decision tree boundary drawn on the graph above. (A table of $-x/y * log(x/y)$ values is given on the next page)



$X_1 > 1.5$

$$AE = \frac{2(0) + 4\left(-\frac{3}{4}\log\frac{3}{4} - \frac{1}{4}\log\frac{1}{4}\right)}{6}$$

$$= \frac{2}{3}(0.31 + 0.50)$$

$$= 0.54$$

$X_2 > 2$

$$AE = \frac{2\cdot(0) + 2\left(-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2}\right)}{4}$$

$$= 0.5$$

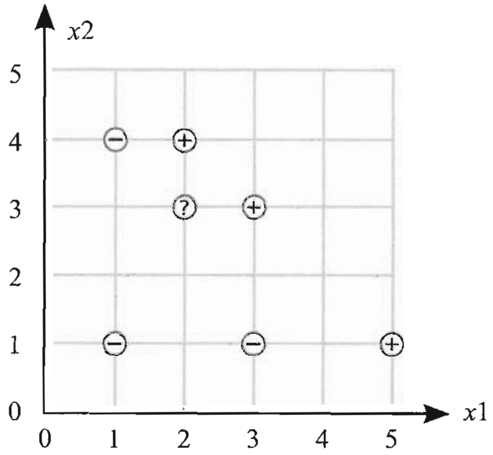
2. How would this decision tree classify the unknown point (2, 3).

+

x	y	-(x/y)*lg(x/y)
1	2	0.50
1	3	0.53
2	3	0.39
1	4	0.50
3	4	0.31
1	5	0.46
2	5	0.53
3	5	0.44
4	5	0.26
1	6	0.43
2	6	0.53
5	6	0.22
1	7	0.40
2	7	0.52
3	7	0.52
4	7	0.46
5	7	0.35
6	7	0.19

x	y	-(x/y)*lg(x/y)
1	8	0.38
3	8	0.53
5	8	0.42
7	8	0.17
1	9	0.35
2	9	0.48
4	9	0.52
5	9	0.47
7	9	0.28
8	9	0.15
1	10	0.33
3	10	0.52
7	10	0.36
9	10	0.14

Nearest Neighbor



Data points are: Negative: (1, 1) (3, 1) (1, 4) Positive: (2, 4) (3, 3) (5, 1). Data points are classified as either +1 or -1. An unknown point is located at (2, 3)

1. Draw the 1-NN decision boundaries on the graph above.

2. How would 1-NN classify the unknown point (2, 3).