CS178 Final Exam

Machine Learning & Data Mining: Winter 2011

Tuesday March 15th, 2011

Open book, open notes: total time is 1h 50m.

Your name: SOLUTIONS

Name of the person in front of you (if any):

Name of the person to your right (if any):

- READ THE EXAM FIRST and organize your time; don't spend too long on any one problem.
- Please **write clearly** and **show all your work.**
- If you need clarification on a problem, please raise your hand and wait for myself or the TA to come over.
- Turn in any scratch paper with your exam.

(This page intentionally left blank)

Problem 1: Bayes classifiers

Consider the following table of measured data:

x_1	x_2	x_3	y
0	0	0	0
0	0	0	1
0	1	1	0
1	1	0	0
1	1	0	1
1	0	1	1
1	1	1	1

We will use the three observed features x_1, x_2, x_3 to predict class y. In the case of a tie, we will prefer to predict class $y = 0$.

(a) Write down the probabilities necessary for a naïve Bayes classifier:

$p(y=1) = 4/7$

$p(x_1=1 | y=0) = 1/3$ $\quad$ $p(x_2=1 | y=0) = 2/3$ $\quad$ $p(x_3=1 | y=0) = 1/3$

$p(x_1=1 | y=1) = 3/4$ $\quad$ $p(x_2=1 | y=1) = 1/2$ $\quad$ $p(x_3=1 | y=1) = 1/2$

(b) Using your naïve Bayes model, what value of y is predicted given observation $(x_1, x_2, x_3) =$ (000).

$$p(y=1 | x_1=x_2=x_3=0) = \frac{p(y=1)\, p(x_1=0|y=1)\, p(x_2=0|y=1)\, p(x_3=0|y=1)}{(\;''\;) + p(y=0)\, p(x_1=0|y=0) \cdots}$$

$$= \frac{(4/7)(1/4)(1/2)(1/2)}{(4/7)(1/4)(1/2)(1/2) + (3/7)(2/3)(1/3)(2/3)}$$

$$= \frac{(1/4)}{(1/4)+(4/9)} = \frac{1}{1+16/9} = \frac{9}{25} \le 1/2 \Rightarrow \text{predict } \hat{y} = \emptyset.$$

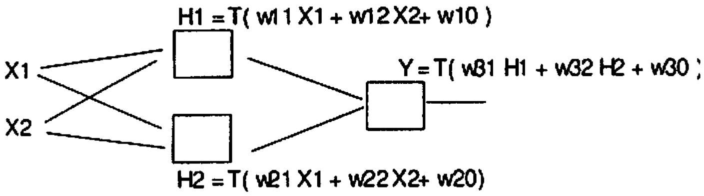
(c) Describe a problem in which we might prefer to use a naïve Bayes classifier, and why.

Classification using text data is the classic example (eg spam)

⇒ the # of features is very high, so a joint model is impossible, but they are "reasonably independent" given the type of email.

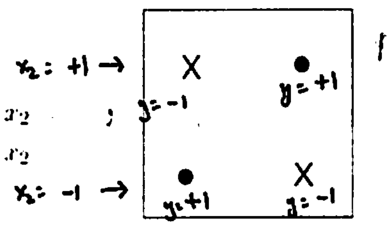
Problem 2: Neural networks

Consider each of the following neural network structure and specified target output function for binary valued input features $x_i \in \{-1, +1\}$ and hidden nodes h_1, h_2. For the purposes of this problem, use a hard threshold sigmoid function, $T(x) = sign(x)$, so that h_i are also binary valued.



Give the weights w_{ij} for the given neural network to produce the desired outputs of an XOR-like function, or write "none exist" if they cannot be produced by the given structure. (Hint: first try to construct a zero-error classifier of the right "shape", then try to deduce the weights.)

Targetoutput : $f(x_1, x_2) = \begin{cases} +1 & x_1 = x_2 \\ -1 & x_1 \neq x_2 \end{cases}$;



Easiest is to design H_1 to predict one point & H_2 the other:

$H_1 = \begin{cases} +1 & x_2 + x_1 > 1/2 \quad (\Rightarrow \text{upper right point}) \\ -1 & \text{otherwise} \end{cases}$ $\Rightarrow w_{11} = +1 \quad w_{12} = +1 \quad w_{10} = -1/2$

$H_2 = \begin{cases} +1 & x_2 + x_1 < -1/2 \quad (\Rightarrow \text{lower left point}) \\ -1 & \text{otherwise} \end{cases}$ $\Rightarrow w_{21} = -1 \quad w_{22} = -1 \quad w_{20} = -1/2$

$H_3 = \begin{cases} +1 & \text{if } H_1 \text{ or } H_2 = +1 \\ -1 & \text{otherwise} \end{cases}$ – ~~[illegible]~~

$\Rightarrow w_{31} = +1 \quad w_{32} = +1 \quad w_{30} = +1/2$

Problem 3: Decision Trees

We plan to use a decision tree to predict an outcome y using four features, $x_1, \ldots, x_4$. We observe eight training patterns, each of which we represent as $[x_1, x_2, x_3, x_4]$ (so, "0101" means $x_1 = 0$, $x_2 = 1$, $x_3 = 0$, $x_4 = 1$). We observe the training data,

$y = 0$: [0010], [1010], [0100], [1111]

$y = 1$: [1011], [0000], [0011], [1101]

You may find the following values useful (although you may also leave logs unexpanded):

$\log_2(1) = 0$ $\log_2(2) = 1$ $\log_2(3) = 1.59$ $\log_2(4) = 2$

$\log_2(5) = 2.32$ $\log_2(6) = 2.59$ $\log_2(7) = 2.81$ $\log_2(8) = 3$

(a) What is the entropy of y?

$p(y=1) = 1/2 \Rightarrow H(y) = 1$ bit.

(b) Which variable would you split first? Justify your answer.

If we split on: $p(x_i=1)$

$x_1 \Rightarrow (1/2\ 1/2)\ \&\ (1/2\ 1/2),\ 1/2$

$x_2 \Rightarrow (2/5\ 3/5)\ \&\ (2/3\ 1/3),\ 3/8$

$x_3 \Rightarrow (3/5,\ 2/5)\ \&\ (1/3\ 2/3),\ 3/8$

$x_4 \Rightarrow (3/4,\ 1/4)\ \&\ (1/4\ 3/4),\ 1/2$

The most "skewed" (least uniform) of these is x_4, although to be sure you could calculate

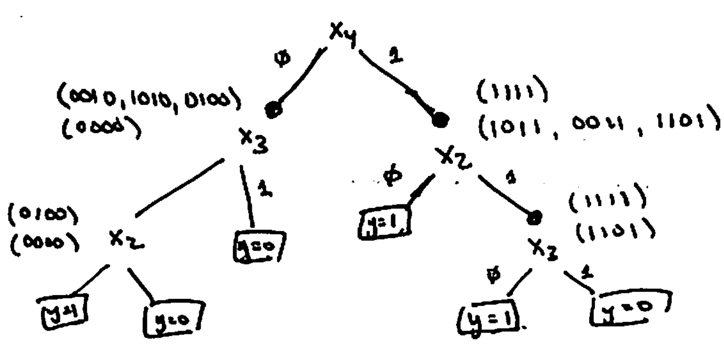
$IG = p(x_i=1)\left(H(y) - H(y|x_i=1)\right) + (1-p(x_i=1))\left(H(y) - H(y|x_i=0)\right)$

(c) What is the information gain of the variable you selected in part (b)?

$IG = \text{above} \Rightarrow \frac{1}{2}\left(1 - \left[\frac{3}{4}\log(4/3) + \frac{1}{4}\log(4/1)\right]\right) + \frac{1}{2}(\ldots)$

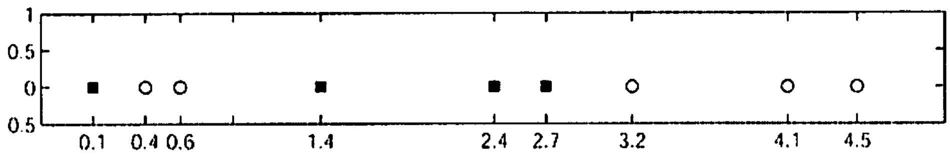
$=$

(d) Draw the rest of the decision tree learned on these data.



Problem 4: Cross validation

Consider the following one-dimensional training data and optimally trained (minimum classification error) classifiers of two types: (1) a fully trained decision tree, and (2) a one-level decision stump, i.e., $sign(x > a)$ for some a. (Note: you do not have to calculate any entropies to answer this question.)



(a) Calculate the training error for the full decision tree

ϕ

Decision tree will train until all data classified correctly

(b) Calculate the leave-one-out cross-validation error for the full decision tree

Errors: × ✓ ✓ × ✓ ✓ × ✓ ✓

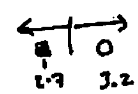
All training data correct, splits at midpoints ⇒ like a nearest neighbor decision boundary.

⇒ 3/9.

(c) Calculate the training error for the decision stump

2/9.

Stump will pick



(d) Calculate the leave-one-out cross-validation error for the decision stump

Errors: ✓ × × $\underbrace{✓ \quad ✓}$ ✓ × ✓ ✓

depends on tie-breaking.

I assumed we prefer to get circles wrong than squares in the case of a tie.

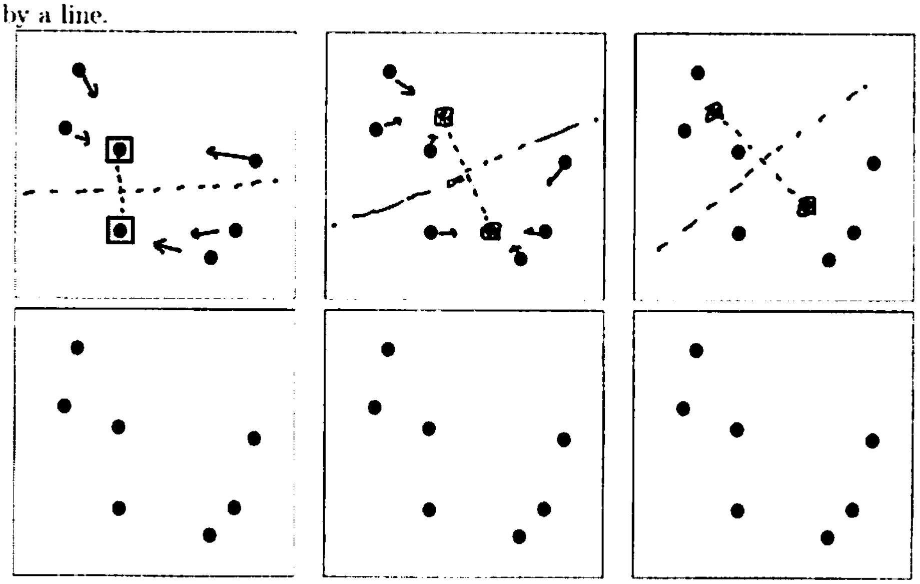
3/9.

Problem 5: Clustering

Consider the two-dimensional data points plotted in each panel. In this problem, we will cluster these data using two different algorithms, where each panel is used to show an iteration or step of the algorithm.

k-means

(a) Starting from the two cluster centers indicated by squares, perform k-means clustering on the data points. In each panel, indicate (somehow) the data assignment, and in the next panel show the new cluster centers. Stop when converged, or after 6 steps, whichever is first. It may be helpful to recall from our nearest-neighbor classifier that the set of points nearer to A than B is separated by a line.



stable.

(b) Write down the cost function optimized by the k-means algorithm, and explain your notation.

Centers μ_c $\quad c = 1 \ldots \#$ clusters $= k$.

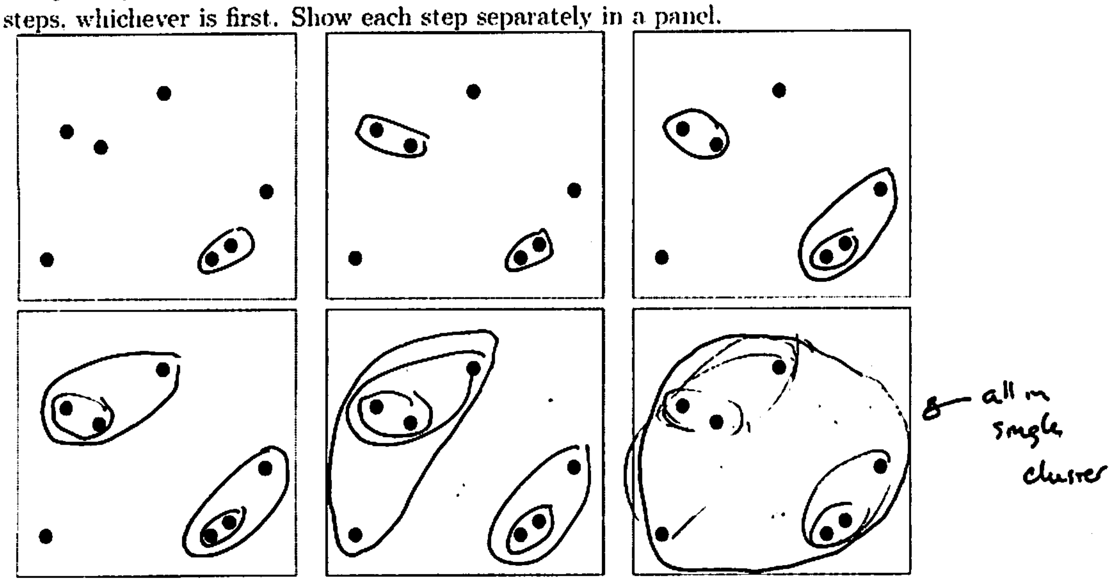
assignment z_i $\quad z_i \in \{1 \ldots \# \text{ clusters}\}$, $\quad i = 1 \ldots \#$ data $= N$.

x_i $\quad$ data point.

$$C(\mu, z) = \frac{1}{N} \sum_{i=1}^{N} (x_i - \mu_{z_i})^2 .$$

Linkage

(a) Now execute the hierarchical agglomerative clustering (linkage) algorithm on these data points, using "single linkage" (minimum distance) for the cluster scores. Stop when converged, or after 6 steps, whichever is first. Show each step separately in a panel.



→ judgement call at this point.

(b) What is the algorithmic (computational) complexity of the hierarchical clustering algorithm? Briefly justify your answer.

$O(n^2)$, n = # of data

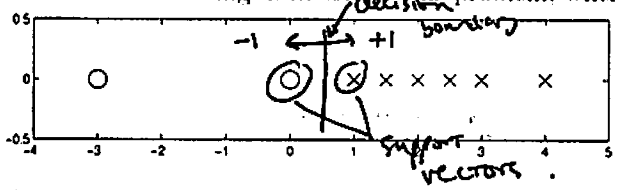
Costs $O(n^2)$ to calculate all pairs of distances

each of n steps costs $O(n)$ (or less) to update distances.

$\Rightarrow O(n^2) + n\,O(n) = O(n^2)$.

Problem 6: Support Vector Machines

Consider the following data set for this problem, where "x" is class +1 and "o" is class -1:



(a) Sketch the solution (decision boundary) of a linear SVM on the data, and identify the support vectors.

(b) Give the solution parameters w and b, where the linear form is $wx + b$.

$wx+b > 0$ for $x > 1/2$, and

$w(x=0) + b = -1 \Rightarrow b = -1$

$w(x=1) + b = +1 \Rightarrow w = 2$

$w(x=1/2) + b = \phi$

(c) Calculate the training error:

ϕ.

(d) Calculate the leave-one-out cross-validation error for these data

✓ × ✓ ✓ ✓ ✓ ✓ ✓

$1/8$.

Problem 7: VC Dimension

Give the VC dimension for each of the following learners. In each case, state what you think is the VC dimension, and justify why it must be at least this value. We use the convention that the data features are x_1, x_2 and the learner's parameters are a, b, c.

(a) $sign(x_1^2 + a)$ (be careful!)

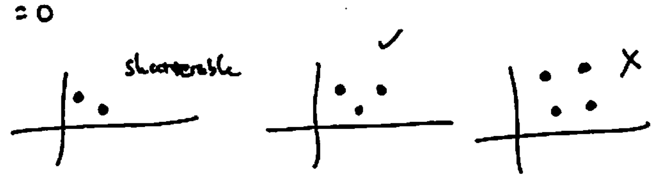
1

(b) A Gaussian Bayes classifier with equal covariances

→ linear decision boundary $ax_1 + bx_2 + c = 0$

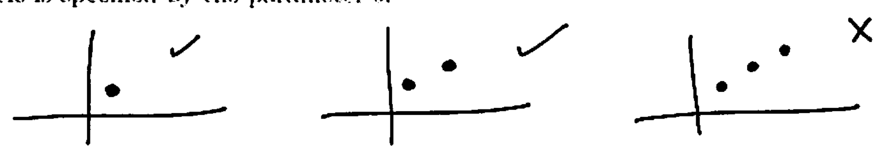
3

(argument from class)



(c) Decision boundaries that are circles centered at the origin, of radius a and where the class value we predict inside the circle is specified by the parameter b.

2



Circle one choice:

(a) The VC dimension of a linear SVM on x_1, x_2 is (greater equal less than) the VC dimension of a perceptron classifier on the same features

(b) The VC dimension of a decision stump on x_1, x_2 is (greater equal less than) the VC dimension of a perceptron classifier on the same features

(c) The VC dimension of a decision stump on x_1 alone is (greater equal less than) the VC dimension of a decision stump on both features x_1, x_2.

Problem 8: Linear regression

Consider a linear regression problem $\hat{y} = ax + b$, with training features $x^{(1)} \dots x^{(m)}$ and targets $y^{(1)} \dots y^{(m)}$. Suppose that we wish to minimize the *mean fourth-degree* error,

$$C = \frac{1}{N} \sum_i (y^{(i)} - ax^{(i)} - b)^4$$

(a) Calculate the gradient with respect to the parameter a

$$\frac{\partial}{\partial a} C = \frac{1}{N} \sum (4)(y^i - ax^i - b)^3 (-x^i)$$

(b) Write down pseudocode for online gradient descent on a for this problem. (You do not need to include the equations for b.)

Init a, b, step size α.

While (! done) {
  for $i = 1..N$,
    $a \leftarrow a + \alpha \, (y^i - ax^i - b)^3 x^i$
    $b \leftarrow b + \alpha \, (\dots)$
  endfor
} check for convergence (change in MSE, change in a, b values, # iterations, etc).

(c) Give one reason in favor of online gradient descent compared to batch gradient descent, and one reason in favor of batch over online.

Online - often faster, easy to use with very large datasets

Batch - less randomness ⇒ easier to check for convergence,
often easier to set step size α,