CS273 Midterm Exam

Introduction to Machine Learning: Winter 2015

Tuesday February 10th, 2014

Your name: SOLUTIONS

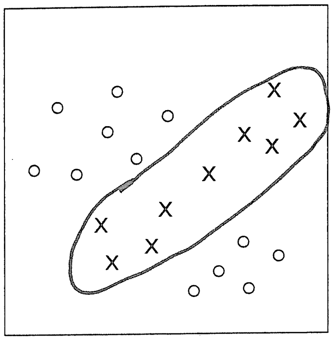
Your UCINetID (e.g., myname@uci.edu):

Your seat (row and number):

- Total time is 80 minutes. 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 the instructor to come over.
- Turn in any scratch paper with your exam.

Problem 1: (8 points) Separability and Features

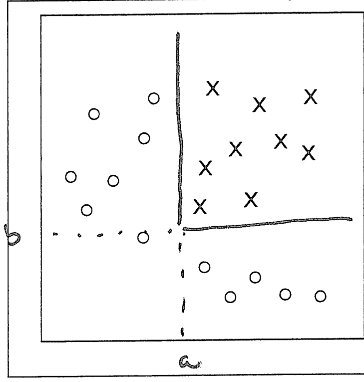
For each of the following examples of training data, **(1)** sketch a classification boundary that separates the data; **(2)** state whether or not the data are linearly separable, and if not, **(3)** give a set of features that would allow the data to be separated. (Your features do not need to be minimal, but should not contain any obviously unneeded features.)



No - not linearly separable

Could use features such as

$[1 \quad x_1 \quad x_2 \quad x_1x_2 \quad x_1^2 \quad x_2^2]$



No, not linearly separable

The same quadratic features would work, or eg.

$[1 \quad x_1 > a \quad x_2 > b]$

also works.

Problem 2: (8 points) Under- and Over-fitting

Suppose that I am training a neural network classifier to recognize faces in images. Using cross-validation, we discover that my classifier appears to be overfitting the data. Give two ways I could improve my performance – be specific.

Lots of possible answers:

Regularize
Use fewer hidden nodes
Use fewer layers

Get more data (if possible!)
Use "early stopping"
Use fewer input features / feature selection

After following some of your advice, we now think that the resulting classifier is underfitting. Give two ways, **other than** reversing the methods you mentioned above, that we could improve performance; again, be specific.

Just reverse two ideas in the other part

eg: generate new features (polynomials, etc)

or increase the # of hidden nodes.

Problem 3: (9 points) Bayes Classifiers and Naïve Bayes

Consider the table of measured data given at right. We will use the two observed features x_1, x_2 to predict the class y. In the case of a tie, we will prefer to predict class $y = 0$.

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

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

$p(y=1) = 5/8$

$p(x_1=1 | y=0) = 2/3$ $\quad p(x_1=1 | y=1) = \emptyset$.

$p(x_2=1 | y=0) = 1/3$ $\quad p(x_2=1 | y=1) = 3/5$.

(b) Using your naïve Bayes model, what value of y is predicted given observation $(x_1, x_2) = (00)$?

Compare: $p(y=0)\, p(x_1=0|y=0)\, p(x_2=0|y=0)$ vs $p(y=1)\, p(x_1=0|y=1)\, p(x_2=0|y=1)$

$3/8 \cdot 1/3 \cdot 2/3$ vs $5/8 \cdot 1 \cdot 2/5$

$= \frac{6}{72}$ vs $1/4$.

$\rightarrow$ predict $\hat{y} = 1$.

(c) Using your naïve Bayes model, what is the probability $p(y = 1 | x_1 = 0, x_2 = 1)$?

Compare: $p(y=0)\, p(x_1=0|y=0)\, p(x_2=1|y=0)$ $\quad p(y=1)\, p(x_1=0|y=1)\, p(x_2=1|y=1)$

$3/8 \cdot 1/3 \cdot 1/3$ $\quad 5/8 \quad 1 \quad 3/5$

$= \frac{3}{8 \cdot 9}$ $\quad \frac{3}{8}$

$\rightarrow p(y=1 | x=01) = \frac{3/8}{3/8 + 3/72} = \frac{27}{27+3} = \frac{27}{30} = .9$.

Problem 4: (10 points) Gradient Descent

Suppose that we have training data $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(m)}, y^{(m)})\}$ and we wish to predict y using a nonlinear regression model with two parameters:

$$\hat{y} = a \exp(x_1 + b)$$

We decide to train our model using gradient descent on the mean squared error (MSE).

(a) Write down the expression for the MSE on our training set.

$$J(\theta) = \frac{1}{m}\sum_i (y^i - \hat{y}^i)^2 = \frac{1}{m}\sum_i \left[y^i - (a \exp(x^i + b))\right]^2$$

(b) Write down the gradient of the MSE.

$$\nabla J = \left[\frac{\partial J}{\partial a} \quad \frac{\partial J}{\partial b}\right]$$

$$\frac{\partial J}{\partial a} = \frac{1}{m}\sum_i (2)\left[y^i - \hat{y}^i\right](-1)\left[\exp(x^i + b)\right]$$

$$\frac{\partial J}{\partial b} = \frac{1}{m}\sum_i (2)\left[y^i - \hat{y}^i\right](-1)\left[a \exp(x^i + b)\right]$$

(c) Give pseudocode for a (batch) gradient descent function `theta = train(X,Y)`, including all necessary elements for it to work.

Initialize $\theta = [a \; b] = [\emptyset \; \emptyset]$; $\theta^{old} = \theta$.

Init step size α $(=1)$, stopping tolerance ϵ $(= 1e{-3})$

While ($\neg$ done) {

$\theta \leftarrow \theta - \alpha \nabla J$ (from (b))

if $(\|\theta - \theta^{old}\|_2^2 < \epsilon)$ done = true

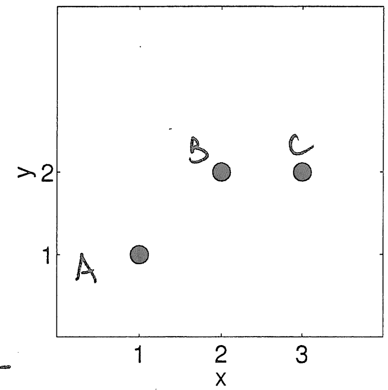
$\theta^{old} = \theta$.

}.

Problem 5: (8 points) Cross-validation and Linear Regression

Consider the following data points, copied in each part. We wish to perform linear regression to minimize mean squared error.

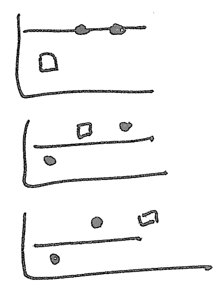
(a) Compute the leave-one-out cross-validation error of a zero-order (constant) predictor.



A : predict 2 $\Rightarrow$ $(1-2)^2 = 1^2$

B : predict $1\frac{1}{2}$ $\Rightarrow$ $(2-1\frac{1}{2})^2 \Rightarrow + (\frac{1}{2})^2$

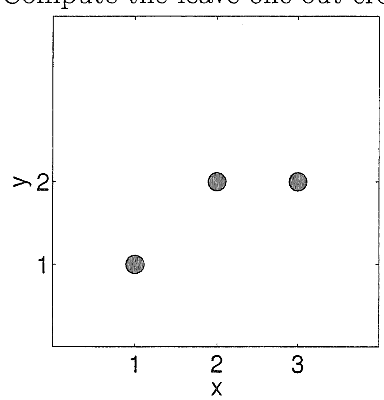
C : predict $1\frac{1}{2}$ $\Rightarrow$ " $+ (\frac{1}{2})^2$



$\Rightarrow$ MSE $= \frac{1}{3}(1 + \frac{1}{4} + \frac{1}{4}) = \frac{1}{3}(1\frac{1}{2}).$ $= \cancel{3/2}$.

$= \frac{1}{2}$.

(b) Compute the leave-one-out cross-validation error of a first-order (linear) predictor.

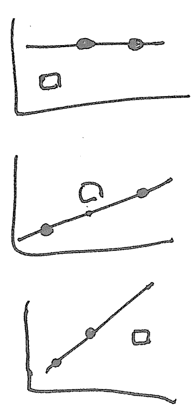


A : predict 2 $\Rightarrow$ 1^2

B : predict $1\frac{1}{2}$ $\Rightarrow$ $(\frac{1}{2})^2$

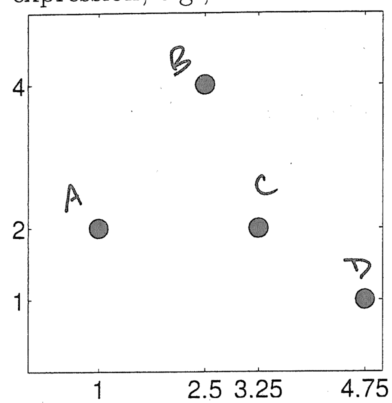
C : predict 3 $\Rightarrow$ 1^2

$\Rightarrow$ MSE $= \frac{1}{3}[1 + \frac{1}{4} + 1] = \frac{1}{3} \cdot \frac{9}{4} = 3/4$.



Problem 6: (12 points) K-Nearest Neighbor Regression

Consider a regression problem for predicting the following data points, using the k-nearest neighbor regression algorithm to minimize mean squared error (MSE). In the case of ties, we will prefer to use the neighbor to the left (smaller x value). Note: if you prefer, you may leave an arithmetic expression, e.g., leave values as "$(.6)^2$".



(a) For $k = 1$, compute the training error on the provided data.

ϕ

(b) For $k = 1$, compute the leave-one-out cross-validation error on the data.

A's NN is B $\Rightarrow 2^2$
B's NN is C $\Rightarrow 2^2$
C's NN is B $\Rightarrow 2^2$
D's NN is C $\Rightarrow 1^2$

$\Rightarrow \quad 13/4.$

(c) For $k = 3$, compute the training error on the provided data.

A's 3NN : ABC
B's 3NN : ABC
C's 3NN : BCD
D's 3NN : BCD

ABC $\rightarrow$ predict $8/3$
BCD $\Rightarrow$ predict $7/3$

$$\Rightarrow (2-8/3)^2 + (4-8/3)^2 + (2-7/3)^2 + (1-7/3)^2$$
$$= (2/3)^2 + (4/3)^2 + (1/3)^2 + (4/3)^2 = 37/9.$$

(d) For $k = 3$, compute the leave-one-out cross-validation error on the data.

A's 3NN : BCD
B's 3NN : ACD $\Rightarrow$
$\vdots$

predict $7/3$
$5/3$
$7/3$
$8/3$

$$\Rightarrow (2-7/3)^2 + (4-5/3)^2 + (2-7/3)^2 + (1-8/3)^2$$
$$= (1/3)^2 + (7/3)^2 + (1/3)^2 + (2/3)^2$$
$$= 55/9$$

Problem 7: (4 points) Multiple Choice

For the following questions, assume that we have m data points $y^{(i)}$, $x^{(i)}$, $i = 1 \ldots m$, each with n features, $x^{(i)} = [x_1^{(i)} \ \ldots \ x_n^{(i)}]$.

Circle one answer for each:

True or **false**: Linear regression can be solved using either matrix algebra or gradient descent.

True or **false**: The predictions of a k-nearest neighbor classifier will not be affected if we pre-process the data to normalize the magnitude of each feature. Changing feature scale ⇒ changes distances.

True or **false**: With enough hidden nodes, a Neural Network can separate any data set.

True or **false**: Increasing the regularization of a linear regression model will decrease the bias.

Problem 8: (4 points) Short Answer

Give one advantage of stochastic gradient descent over batch gradient descent, **and** one advantage of batch gradient descent over stochastic.

SGD: often faster, esp. initially & for very large data sets

Batch: less random:
easier to guage convergence, keep track of current loss value,
small step size ensures monotonic decrease in the loss.

Problem 9: (12 points) Perceptrons and VC Dimension

In this problem, consider the following perceptron model on two features:

$$\hat{y}(x) = \text{sign}(\, b + w_1 x_1 + w_2 x_2 \,)$$

and answer the following questions about the decision boundary and the VC dimension.

(a) Describe (in words, with diagrams if desired) the possible decision boundaries that can be realized by this classifier

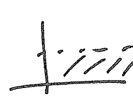
A standard perceptron – so – lines in 2D space; either decision on either side.

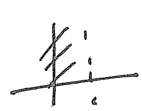
(b) What is its VC dimension?

3 (standard result from class)

Now suppose that I also enforce an additional condition on the parameters of the model: that **only one** of the two weights w_1, w_2 is non-zero (i.e., one of them must be zero, a "feature selection" criterion). Note that the training algorithm can choose which parameter is zero, depending on the data.

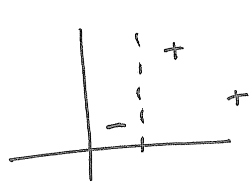
(c) Describe (in words, with diagrams if desired) the decision boundaries that can be realized by this classifier (there should be two "cases").

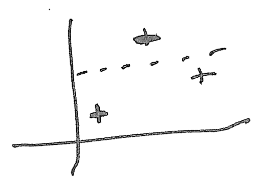
if $w_1 = 0 \Rightarrow$  horizontal boundaries (± 1 either side)

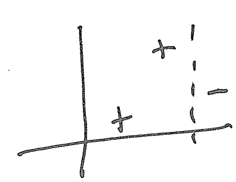
if $w_2 = 0 \Rightarrow$  vertical boundaries (± 1 either side)

(d) What is its VC dimension?

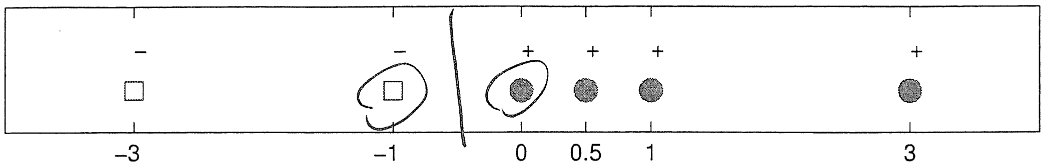
Still 3:







Problem 10: (9 points) Support Vector Machines



Using the above data with one feature x (whose values are given below each data point) and a class variable $y \in \{-1, +1\}$, with filled circles indicating $y = +1$ and squares $y = -1$ (the sign is also shown above each data point for redundancy), answer the following:

(a) Sketch the solution (decision boundary) of a linear SVM on the data, and identify the support vectors. Boundary at $x = -1/2$ SVs are $x = -1$, $x = 0$.

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

$w(-1) + b = -1$

$w(-1/2) + b = \emptyset.$ $\Rightarrow$ $b = +1$

$w(\emptyset) + b = +1$ $w = 2$.

(c) Calculate the training error:

$\emptyset$ - separates the data.

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

If any points except ~~...~~ $x=-1$, $x=0$ are left out, the boundary stays the same $\Rightarrow$ correct.

If $x=-1$ left out, boundary moves to -1.5 $\Rightarrow$ X

$x=\emptyset$ left out, " " to $-.25$ $\Rightarrow$ ✓

$\Rightarrow 1/6$