FINAL

CS273a ~~Midterm~~ Exam

Introduction to Machine Learning: Fall 2012

Tuesday December 11th. 2012

Your name: SOLUTIONS

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

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

- Total time is 1:~~15~~ 50. 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: VC Dimension (10p)

Argue by example / counterexample what is the VC dimension of each of the following classifiers.

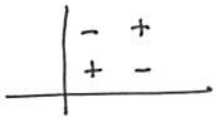
(5p) (a) A perceptron on two *binary* features

Binary features $\Rightarrow x_i \in \{0, 1\}$

VCDim = 3

Can shatter 3 points

Cannot shatter 4 points

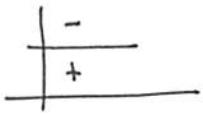


(5p) (b) A decision stump on two *binary* features

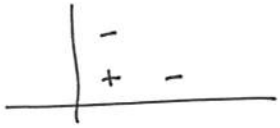
Binary features $\Rightarrow x_i \in \{0, 1\}$.

VC Dim = 2

Can shatter 2 points:



cannot shatter 3



(No axis-aligned split can separate)

Problem 2: Decision Trees (12p)

We plan to use a decision tree to predict an outcome y using four features, $x_1, \ldots, x_3$. We observe six training patterns, each of which we represent as $[x_1, x_2, x_3]$ (so, "010" means $x_1 = 0$, $x_2 = 1$, $x_3 = 0$). We observe the training data.

$y = 0$: [100], [111], [001]

$y = 1$: [110], [110], [011]

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

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

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

(3p) (a) What is the entropy of y?

$3/6 \log 6/3 + 3/6 \log 6/3 = 1$ bit

(3p) (b) Which variable would you split first? Justify your answer.

	X_1		X_2		X_3	
	=0	=1	=0	=1	=0	=1
y=0	001	100 111	100 001	111	100	111 001
y=1	011	110 110	—	110 110 011	110 110	011

By inspection, X_2 has the lowest expected entropy ($\Rightarrow$ higher info gain)

$H(y|x_1) \geq H(y|x_3) \geq H(y|x_2)$

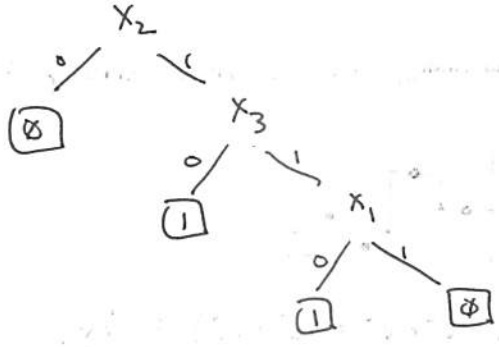
(3p) (c) What is the information gain of the variable you selected in part (b)?

$H(y) - [2/6\ H(\emptyset) + 4/6\ H(1/4)]$

$= 1 - [\emptyset + 2/3\ (1/4 \log 4 + 3/4 \log 4/3]$

$= 1 - 1/6 \log 4 - 1/2 \log 4/3 = 2/3 - 1/2 \lg(.41)$

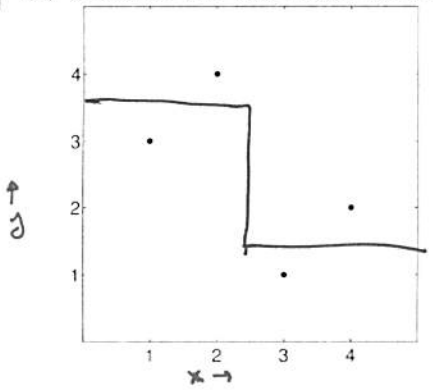
(3p) (d) Draw the rest of the decision tree learned on these data.



Problem 3: Gradient Boosting (9p)

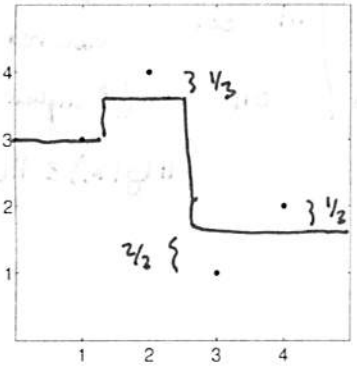
Consider the following data set consisting of four points; for convenience, the data are repeated in each part.

(3p) (a) Compute the best single decision stump regressor function, to minimize mean squared error.



(this predictor has MSE = $\frac{1}{4} \cdot (4 \cdot \frac{1}{2}^2) = \frac{1}{4}$.)

(3p) (b) Now, we wish to create a gradient boosted ensemble of decision stumps to minimize MSE. Starting from the decision stump learned in 9a), and using a learning rate of 1, what is the next predictor? Show your work.



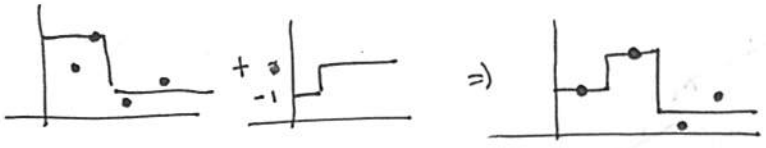
After (a), we have error residual

If we fit a decision stump, we get  ⇒ sum at left.

$MSE = \frac{1}{4} \cdot (0^2 + \frac{1}{3}^2 + \frac{2}{3}^2 + \frac{1}{3}^2) = \frac{3}{18}$.

(3p) (c) Is the resulting ensemble the best possible ensemble of two decision stumps? If yes, why? If not, give a better ensemble.

No - if you do not train them sequentially, you can get a better predictor, eg



⇒ MSE is $\frac{1}{4}\left(0^2 + 0^2 + \frac{1}{2}^2 + \frac{1}{2}^2\right) = \frac{1}{8}$.

Problem 4: Naïve Bayes (10p)

We plan to use a naïve Bayes classifier to predict an outcome y using four features, $x_1, \ldots, x_3$. We observe five training patterns, each of which we represent as $[x_1, x_2, x_3]$ (so, "010" means $x_1 = 0$, $x_2 = 1$, $x_3 = 0$). We observe the training data,

$y = 0$: [000], [111]
$y = 1$: [100], [010], [001]

(4p) (a) Compute (& show) all the necessary probabilities for a naïve Bayes model.

$p(y) = 3/5$

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

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

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

(3p) (b) Suppose you observe $x = [110]$. What class (value of y) would you predict? Show your work.

$p(y=0)\, p(\underline{x} | y=0) = 2/5 \cdot 1/2 \cdot 1/2 \cdot 1/2 = 1/20$

$p(y=1)\, p(\underline{x} | y=1) = 3/5 \cdot 1/3 \cdot 1/3 \cdot 2/3 = 2/45$ $\Rightarrow$ predict $y = 0$.

(3p) (c) Suppose you observe $x = [100]$. Compute the posterior probability, $p(y|x = 100)$.

$p(y=0)\, p(\underline{x} | y=0) = 2/5 \cdot 1/2 \cdot 1/2 \cdot 1/2 = 1/20 = \frac{9}{5 \cdot 2 \cdot 2 \cdot 3 \cdot 3}$

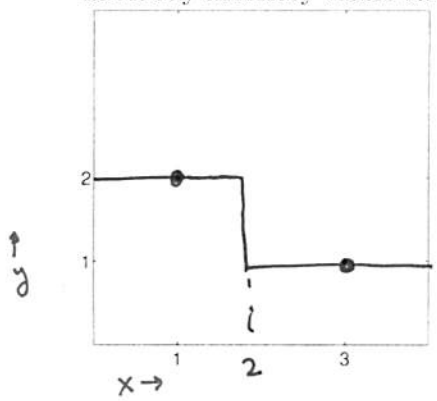
$p(y=1)\, p(\underline{x} | y=1) = 3/5 \cdot 1/3 \cdot 2/3 \cdot 2/3 = 4/45 = \frac{4 \cdot 4}{5 \cdot 3 \cdot 3 \cdot 2 \cdot 2}$

$\Rightarrow p(y=1 | \underline{x}) = \frac{16}{16+9} = \frac{16}{25}$.

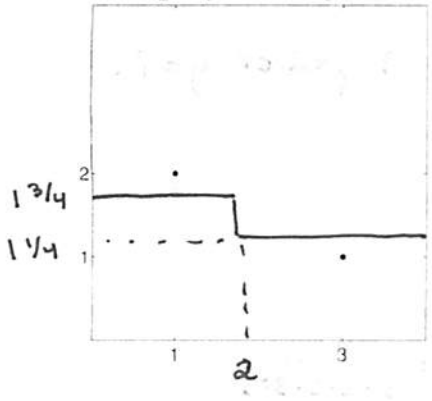
Problem 5: Bagging (9p)

Consider the data set, consisting of two data points, given in each part.

(3p) (a) Draw the regression function (predicted values for all x) using a nearest-neighbor regressor. Label any necessary values on your graph.



(4p) (b) Suppose that we create a very large ensemble of *bagged* nearest-neighbor regressors, using data set draws of size two. Compute the regression function of the complete ensemble, again labeling any necessary values.



There are four possible draws ⇒ 3 unique data sets:

{a,a}: occurs 1/4

{b,b}: 1/4

{a,b} = {b,a}: 1/2 of the time.

⇒ ensemble average predicts nearest point 3/4 of members; other point 1/4 of member ⇒ graph at left

(2p) (c) Is the model in (b) simpler or more complex than the model in (a)? Why?

Simpler: (1) Bagging tends to reduce complexity / overfitting

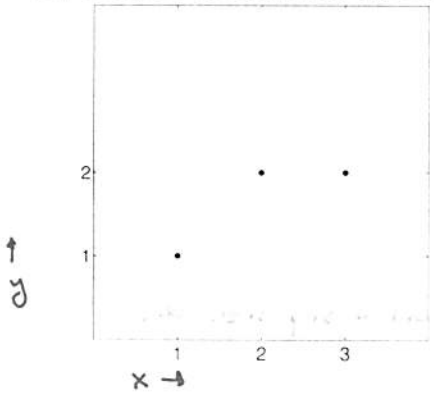
- you can see training error has gone up (& the data are no longer memorized)

(2) The function is "simpler" - it is closer to a constant predictor.

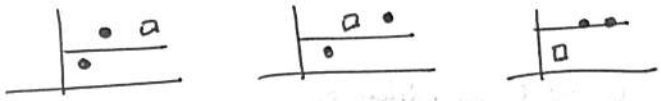
Problem 6: Cross-validation (8p)

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

(4p) (a) Compute the leave-one-out cross-validation error of a zero-order (constant) predictor.



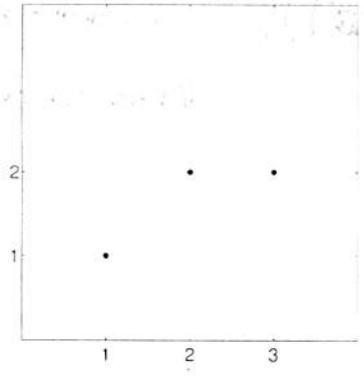
Leave out each data point ⇒



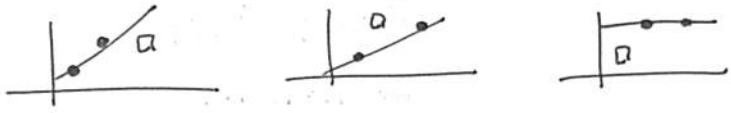
$(1/2)^2 + (1/2)^2 + (1)^2$

$\Rightarrow 1/3 \left[(1/2)^2 + (1/2)^2 + 1^2 \right) = 1/2.$

(4p) (b) Compute the leave-one-out cross-validation error of a first-order (linear) predictor.



Leave out each ⇒



$1^2 + (1/2)^2 + 1^2$

$\Rightarrow 1/3 \left[1^2 + 1/2^2 + 1^2 \right) = 3/4.$

Problem 7: Latent space models (8p)

Suppose that, as in HW5, we wish to model a collection of text documents using a latent space model. For interpretability, we would like our latent representation to be non-negative. As one solution, we use an exponential transform to ensure positive values, giving the model

$$x_j^{(i)} \approx \sum_k \exp(U_{ik}) \exp(V_{kj})$$

Give a stochastic gradient descent algorithm to learn this model, minimizing the mean squared error in the predicted values. Include all necessary details for the implementation.

A simple SGD algorithm is

1. Initialize U, V to something at random
2. Choose a stopping criterion (eg. # of steps) and a step size α.
3. while (!stop) {

```
for i = 1... m
  for j = 1..d
    E = (x_j^(i) - Σ_k exp(U_ik) exp(V_kj));   // compute signed error.
    Ũ = U;  Ṽ = V;                             // save old values
    for k = 1.. K.
      U_ik ← U_ik − α ∇_{U_ik} J
      V_ik ← V_ik − α ∇_{V_kj} J
    end
  end
end
```

where $J(\cdot) = \left(x_j^{(i)} - \sum_k \exp(u)\exp(v)\right)^2$ is the squared error loss

and

$$\nabla_{U_{ik}} J = -E \cdot \exp(\tilde{U}_{ik}) \exp(\tilde{V}_{kj})$$

$$\nabla_{V_{kj}} J = -E \cdot \exp(\tilde{U}_{ik}) \exp(\tilde{V}_{kj})$$

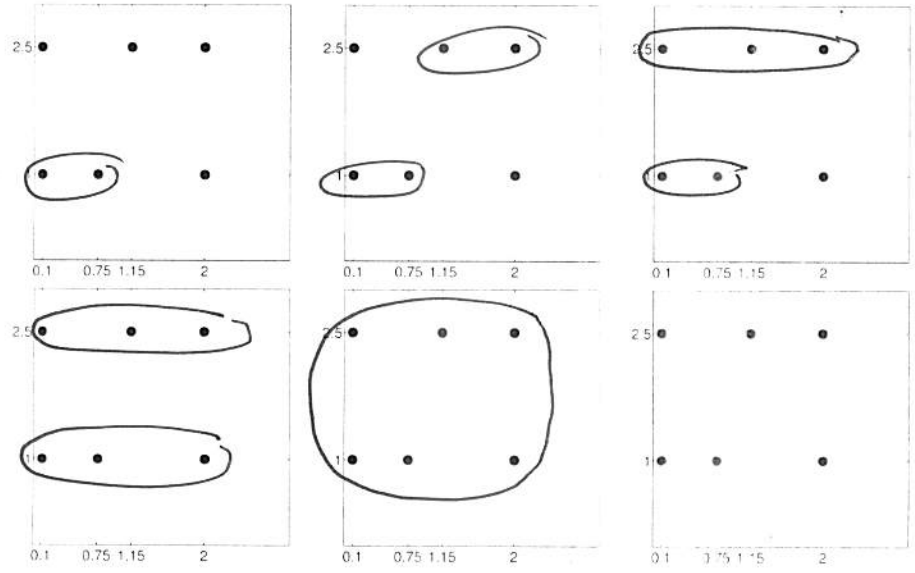
are the derivatives with respect to each element of u, v.

notation should really be $\frac{\partial}{\partial U_{ik}}$ and $\frac{\partial}{\partial V_{kj}}$ instead

Problem 8: Clustering (10p)

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.

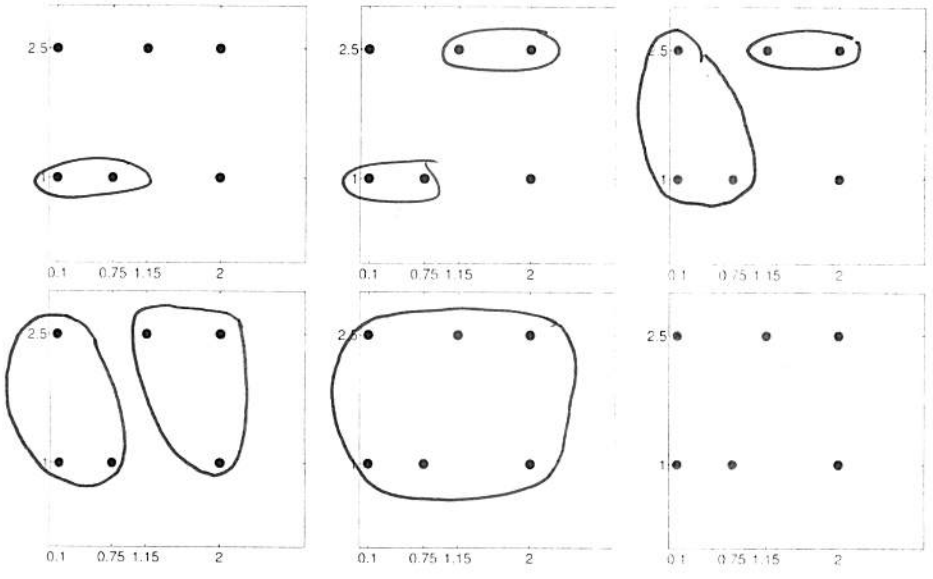
(5p) (a) Execute the hierarchical agglomerative clustering (linkage) algorithm on these data points, using "single linkage" for the cluster scores. Stop when converged, or after 6 steps, whichever is first. Show each step separately in a panel.



Join nearest clusters.

Cluster distance = nearest points in the two clusters.

(5p) (b) Now execute hierarchical agglomerative clustering on the data points, but use "complete linkage" for the cluster scores. Stop when converged, or after 6 steps, whichever is first. Show each step separately in a panel.



Join nearest two clusters.

Cluster distance = furthest distance between two points in the clusters.

From board:

$\sqrt{(.85)^2 + (1.5)^2} = 1.724$

$\sqrt{(.65)^2 + (1.5)^2} = 1.635$