

CS273 Final Exam  
Introduction to Machine Learning: Winter 2015  
**Tuesday March 17th, 2015**

**Your name:**

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

**Your seat (row and number):**

- Total time is 1 hour 50 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.

(This page intentionally left blank)

### Problem 1: (6 points) Multiple Choice

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

**Circle one answer for each:**

**True** or **false**: Early stopping can be used to reduce overfitting in neural networks.

**True** or **false**: The SVM learning algorithm will find the *globally optimal* model with respect to its objective function.

**True** or **false**: Increasing  $k$  in a k-nearest-neighbor classifier will decrease the bias.

**True** or **false**: Increasing the depth of a decision tree classifier will decrease the bias.

**True** or **false**: The VC dimension of a perceptron classifier is smaller than the VC dimension of a linear SVM.

**True** or **false**: If there exists a set of  $h$  instances that cannot be shattered by  $f(x)$ , then the VC dimension of  $f$  is less than  $h$ .

### Problem 2: (4 points) Short Answer

Give one advantage of the dual (kernel) form of support vector machines over the primal (linear) form, **and** one advantage of the primal form over the dual.

### Problem 3: (12 points) Decision Trees

Consider the table of measured data given at right. We will use a decision tree to predict the outcome  $y$  using the three features,  $x_1, \dots, x_3$ . In the case of ties, we prefer to use the feature with the smaller index ( $x_1$  over  $x_2$ , etc.) and prefer to predict class 1 over class 0. You may find the following values useful (although you may also leave logs unexpanded):

$$\begin{aligned} \log_2(1) &= 0 & \log_2(2) &= 1 & \log_2(3) &= 1.59 & \log_2(4) &= 2 \\ \log_2(5) &= 2.32 & \log_2(6) &= 2.59 & \log_2(7) &= 2.81 & \log_2(8) &= 3 \end{aligned}$$

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

- (a) What is the entropy of  $y$ ?
- (b) Which variable would you split first? Justify your answer.
- (c) What is the information gain of the variable you selected in part (b)?
- (d) Draw the rest of the decision tree learned on these data.

### Problem 4: (9 points) Gradient Descent & Latent Space Models

Suppose that, as in lecture, we wish to model a collection of text documents using a latent space model such as Latent Semantic Indexing. However, for interpretability, we would like our latent representation to be non-negative, so that each “direction” can only make a set of words more likely to appear, not less likely. As one solution, we use an exponential transform to ensure positive values, giving the model

$$x_j^{(i)} \approx \sum_k \exp(u_{ik}) \exp(v_{jk})$$

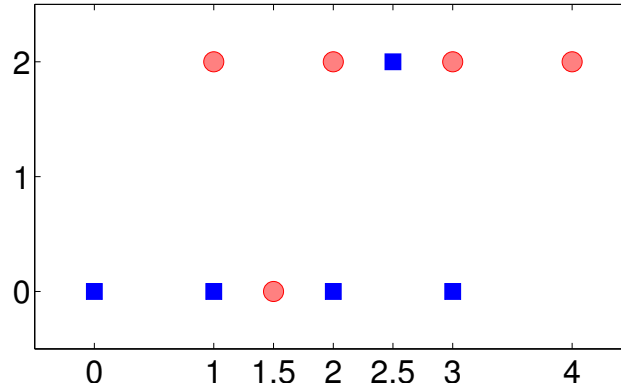
(so, data point  $i$  is a nonnegative linear combination  $u_i$  of nonnegative directions  $v_j$ ). We wish to train our model  $(U, V)$  to minimize squared error from the observed data, and will train it using gradient descent.

- [illegible]

### Problem 5: (8 points) Cross-validation

Suppose that we learn a classifier on the following binary classification data. There are two real-valued features,  $x_1$  and  $x_2$ , and a binary class  $y \in \{0, 1\}$ .

$x_1$	$x_2$	$y$
0	0	0
1	0	0
1.5	0	1
2	0	0
3	0	0
1	2	1
2	2	1
2.5	2	0
3	2	1
4	2	1



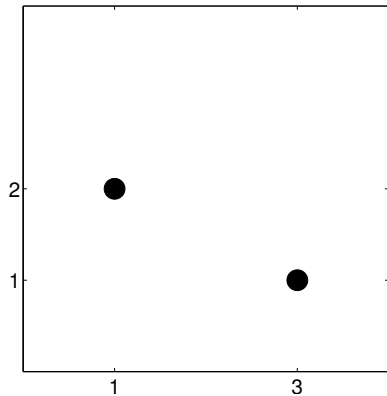
We decide to learn a decision tree as described in class. As in class, when the decision tree splits on the real-valued features, it puts the split threshold halfway between the data points on either side of the highest-scoring split. For example, if we first split on  $x_2$ , the algorithm would choose to split at  $x_2 = 1$ , which is halfway between the data at  $x_2 = 0$  and  $x_2 = 2$ .

- (a) What is the training error rate of a decision *stump* (decision tree with max depth 1) trained on these data?
- (b) What is the training error rate of a full decision tree (no maximum depth) trained on these data?
- (c) What is the leave-one-out cross-validation error rate of a decision *stump* (decision tree with max depth 1) trained on these data?
- (d) What is the leave-one-out cross-validation error rate of a full decision tree (no maximum depth) trained on these data?

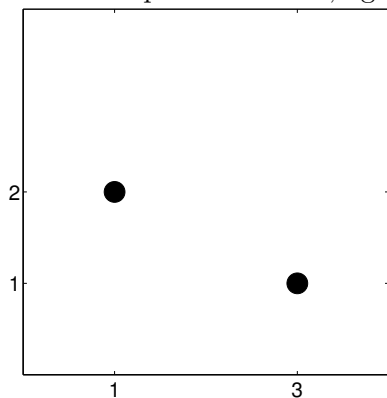
### Problem 6: (8 points) Bagging

Consider a data set, consisting of two data points, plotted in each part.

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



- (b) Suppose that we create a very large ensemble of *bagged* nearest-neighbor regressors, using data set draws of size  $m = 2$  during the bootstrap sampling. Compute the regression function of the complete ensemble, again labeling any necessary values.

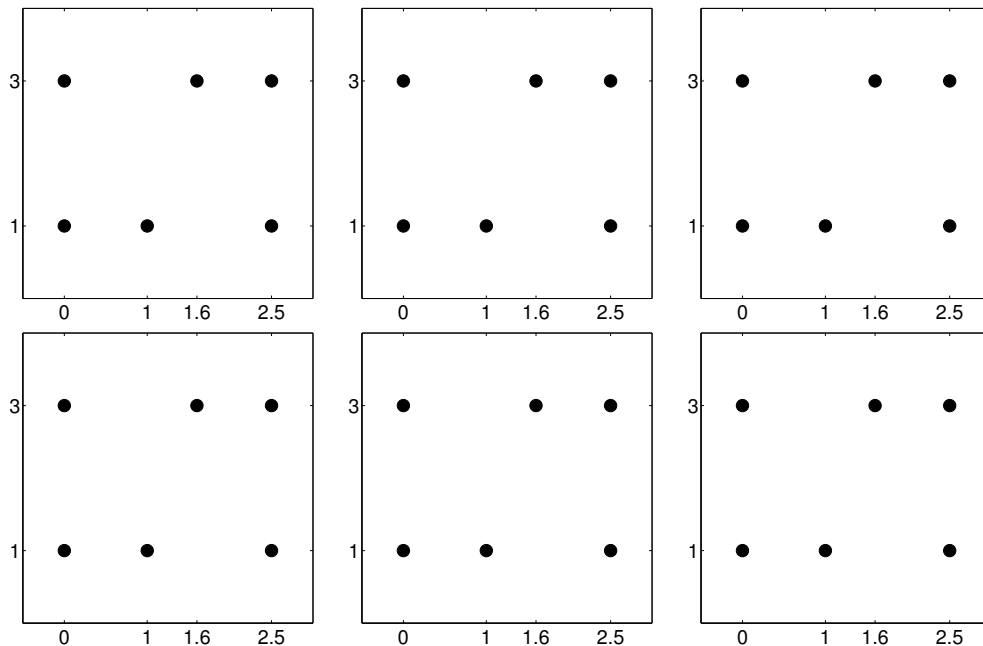


- (c) How does the training error in model (b) compare to the training error in (a)? (Note: you don't need to have answered (b) to answer this part.)

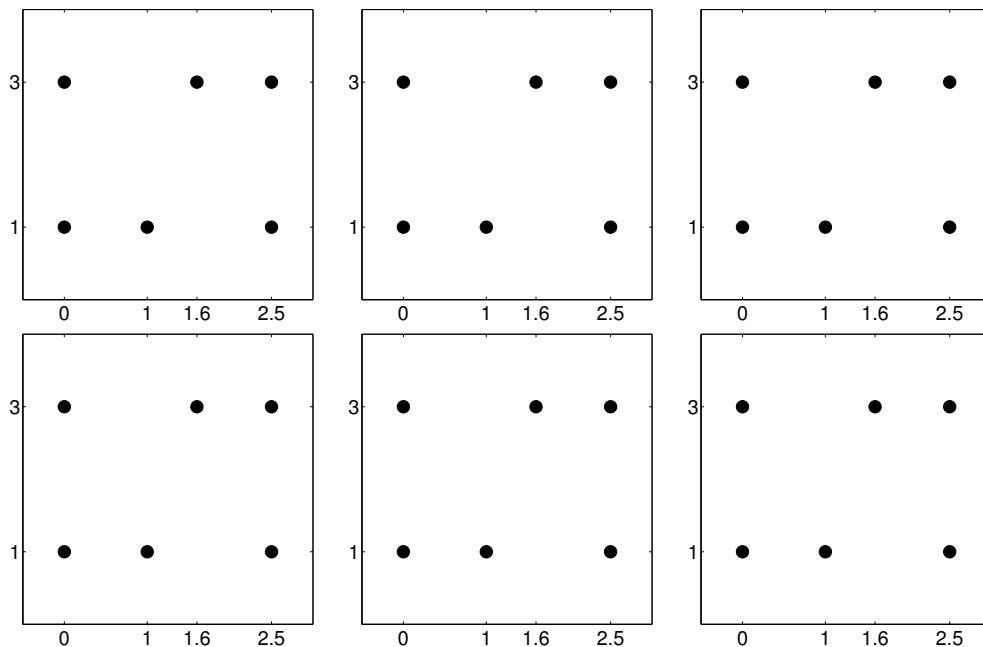
### Problem 7: (8 points) Clustering

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.

(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.



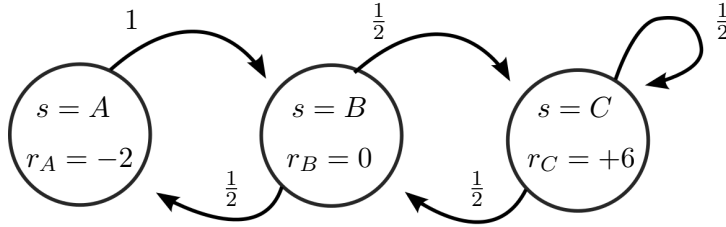
(b) Now execute hierarchical agglomerative clustering on the data points, but use “complete linkage” for the cluster scores. Stop when converged, or after 6 steps, whichever is first. Show each step separately in a panel.





### Problem 8: (8 points) Markov models

Consider the Markov model shown here:



---

$$\Pr[A \rightarrow B] = 1.0$$

$$\Pr[B \rightarrow A] = \frac{1}{2}$$

$$\Pr[B \rightarrow C] = \frac{1}{2}$$

$$\Pr[C \rightarrow B] = \frac{1}{2}$$

$$\Pr[C \rightarrow C] = \frac{1}{2}$$

---

where the transition probabilities are shown next to each arc and at right, and the rewards  $r_s$  associated with each state  $s$  are shown inside the circles. Assume a future discounting factor of  $\gamma = \frac{1}{2}$ .

- (a) Compute  $J^1(s)$ , the expected discounted reward for state sequences of length 1 starting in each state  $s$ .
  
  
  
  
  
  
  
  
  
  
- (b) Compute  $J^2(s)$ , the expected discounted reward for state sequences of length 2 starting in each state  $s$ .
  
  
  
  
  
  
  
  
  
  
- (c) Compute  $J^3(s)$ , the expected discounted reward for state sequences of length 3 starting in each state  $s$ .