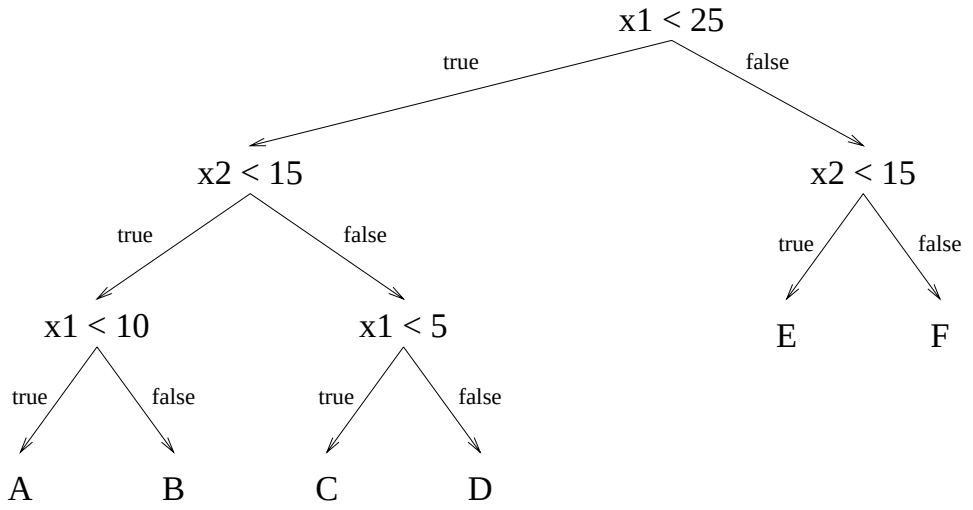
CS534 — Homework Assignment 4 — Due Monday, April 25

1. Consider the following decision tree:



   (a) [6] Draw the decision boundaries defined by this tree. Each leaf of the tree is labeled with a letter. Write this letter in the corresponding region of instance space.

   (b) [4] Give another decision tree that is syntactically different but defines the same decision boundaries ([2]). This demonstrates that the space of decision trees is syntactically redundant. Is this redundancy a statistical problem (i.e., does it affect the accuracy of the learned trees) ([1])? Is it a computational problem (i.e., does it increase the computational complexity of finding an accurate tree) ([1])?

2. In the basic decision tree algorithm, we choose the feature/value pair with the maximum mutual information as the test to use at each internal node of the decision tree. Suppose we modified the algorithm to choose at random from among those feature/value combinations that had non-zero mutual information, but that we kept all other parts of the algorithm unchanged.

   (a) [4] Prove that if a splitting feature/value combination has non-zero mutual information at an internal node, then at least one training example must be sent to each of the child nodes.

   (b) [2] What is the maximum number of leaf nodes that such a decision tree could contain if it were trained on m training examples?

   (c) [2] What is the maximum number of leaf nodes that a decision tree could contain if it were trained on m training examples using the original maximum mutual information version of the algorithm? Is it bigger, smaller, or the same as your answer to (b)?

   (d) [2] How do you think this change would affect the accuracy of the decision trees produced on average? Why?

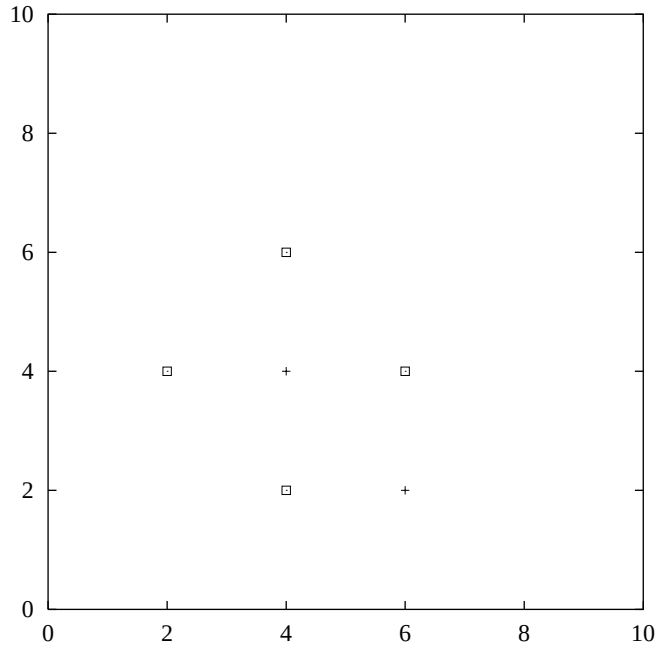
3. [8] Consider the following training examples:

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

What feature would be chosen for the split at the root of a decision tree using the Mutual Information criterion? Show your work.

4. Consider the set of training examples shown in the diagram below.

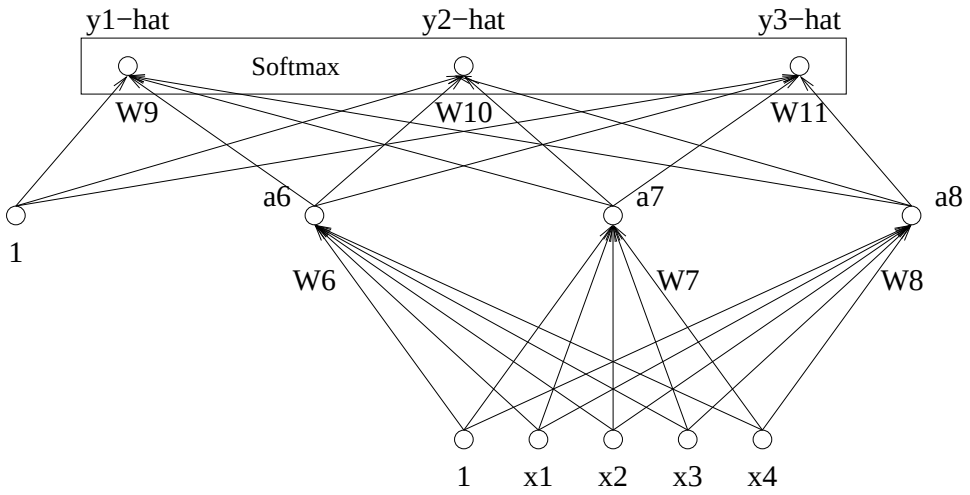
   a. [4] Draw the decision boundaries for the nearest neighbor algorithm assuming that we are using standard Euclidean distance to compute nearest neighbors. A plus indicates a positive example and a small square indicates a negative example.



   b. [1] How will the point (8, 1) be classified by the nearest-neighbor classifier?

   c. [1] How will the point (8, 8) be classified?

5. Consider the following neural network diagram from the lectures that has a softmax as the output layer. In this problem, we will compute the derivatives needed for the backpropagation algorithm for this kind of network.



a. [2] Write down the log likelihood objective function $J(\mathbf{w})$ for this network, where $\mathbf{w}$ is the concatenation of $W6, W7, W8, W9, W10$, and $W11$. You may assume that each training example has the form $(\mathbf{x}, y)$, where $\mathbf{x} = (1, x_1, x_2, x_3, x_4)$ and $y = (y_1, y_2, y_3)$. There are only three possible y values: $y = (1, 0, 0)$, $y = (0, 1, 0)$, and $y = (0, 0, 1)$.

b. [10] Compute the partial derivative

$$\frac{\partial J(\mathbf{w})}{\partial w_{9,6}}$$

c. [6] Compute the partial derivative

$$\frac{\partial J(\mathbf{w})}{\partial w_{6,3}}$$

d. [5] Generalize your answers to (b) and (c) and write the pseudo-code for the backpropagation algorithm using them.