# HMC CS 158, Fall 2017
# Problem Set 2 Exercises: Decision Trees, k-Nearest Neighbor

*Goals*:

- To investigate the theory behind decision trees.
- To practice evaluating classifier performance using k-nearest neighbors.

## Submission

You should submit any answers to the exercises in a single file `writeup.pdf`. This writeup should include your name and the assignment number at the top of the first page, and it should clearly label all problems. Additionally, cite any collaborators and sources of help you received (excluding course staff), and if you are using slip days, please also indicate this at the top of your document.

## 1 Splitting Heuristic for Decision Trees [6 pts]

Recall that the ID3 algorithm iteratively grows a decision tree from the root downwards. On each iteration, the algorithm replaces one leaf node with an internal node that splits the data based on one decision attribute (or feature). In particular, the ID3 algorithm chooses the split that reduces the entropy the most, but there are other choices. For example, since our goal in the end is to have the lowest error, why not instead choose the split that reduces error the most? In this problem, we will explore one reason why reducing entropy is a better criterion.

Consider the following simple setting. Let us suppose each example is described by $n$ boolean features: $X = \langle X_1, \dots, X_n \rangle$, where $X_i \in \{0, 1\}$, and where $n \geq 4$. Furthermore, the target function to be learned is $f : X \to Y$, where $Y = X_1 \vee X_2 \vee X_3$. That is, $Y = 1$ if $X_1 = 1$ or $X_2 = 1$ or $X_3 = 1$, and $Y = 0$ otherwise. Suppose that your training data contains all of the $2^n$ possible examples, each labeled by $f$. For example, when $n = 4$, the data set would be

| $X_1$ | $X_2$ | $X_3$ | $X_4$ | $Y$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 |

| $X_1$ | $X_2$ | $X_3$ | $X_4$ | $Y$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |

(a) **(2 pts)** How many mistakes does the best 1-leaf decision tree make over the $2^n$ training examples? (The 1-leaf decision tree does not split the data even once. Make sure you answer for the general case when $n \geq 4$.)

Parts of this assignment are adapted from course material by Andrea Danyluk (Williams), Tom Mitchell and Maria-Florina Balcan (CMU), and Stuart Russell (UC Berkeley).

(b) **(1 pts)** Is there a split that reduces the number of mistakes by at least one? (That is, is there a decision tree with 1 internal node with fewer mistakes than your answer to part (a)?) Why or why not?

(c) **(1 pts)** What is the entropy of the output label $Y$ for the 1-leaf decision tree (no splits at all)?

(d) **(2 pts)** There exists a split that reduces the entropy of the output $Y$ by a non-zero amount. What is it, and what is the resulting conditional entropy of $Y$ given this split?

## 2 Entropy and Information [2 pts]

The entropy of a Bernoulli (Boolean 0/1) random variable $X$ with $p(X=1)=q$ is given by

$$B(q) = -q\log q - (1-q)\log(1-q).$$

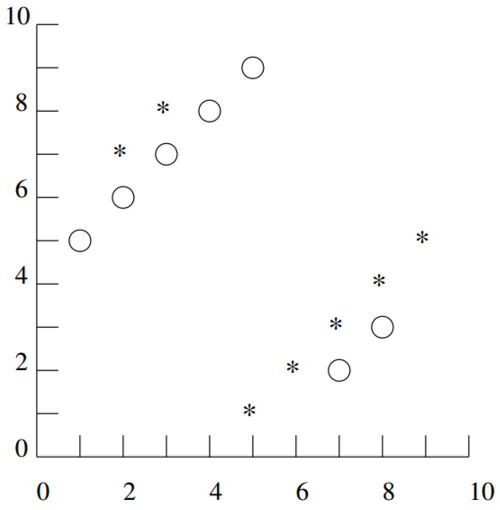
Suppose that a set $S$ of examples contains $p$ positive examples and $n$ negative examples. The entropy of $S$ is defined as $H(S) = B\left(\frac{p}{p+n}\right)$.

Based on an attribute $X_j$, we split our examples into disjoint subsets $S_k$, with $p_k$ positive and $n_k$ negative examples in each. If the ratio $\frac{p_k}{p_k+n_k}$ is the same for all $k$, show that the information gain of this attribute is 0.

*Optional*: Given that information gain is non-negative[1], what does this imply about the information content of data and about the process of constructing a decision tree?

## 3 k-Nearest Neighbor [6 pts]

One of the problems with $k$-nearest neighbor learning is selecting a value for $k$. Say you are given the following data set. This is a binary classification task in which the instances are described by two real-valued attributes.



[1] For the mathematically inclined, try proving this using Jensen's inequality.

(a) **(2 pts)** What value of $k$ minimizes training set error for this data set, and what is the resulting training set error? Why is training set error not a reasonable estimate of test set error, especially given this value of $k$?

(b) **(2 pts)** What value of $k$ minimizes the leave-one-out cross-validation error for this data set, and what is the resulting error? Why is cross-validation a better measure of test set performance?

(c) **(2 pts)** What are the LOOCV errors for the lowest and highest $k$ for this data set? Why might using too large or too small a value of $k$ be bad?