CS 189 Introduction to Machine Learning
Spring 2019 Jonathan Shewchuk

DIS9

1 Surprise and Entropy

In this section, we will clarify the concepts of surprise and entropy. Recall that entropy is one of the standards for us to split the nodes in decision trees until we reach a certain level of homogeneity.

(a) Suppose you have a bag of balls, all of which are black. What's the surprise of taking out a ball whose color is black?

(b) With the same bag of balls, what's the surprise of taking out a white ball?

(c) Now we have 10 balls in the bag, each of which is black or white. Under what color distribution(s) is the entropy of the bag minimized? And under what color distribution(s) is the entropy maximized? Calculate the entropy in each case.
Recall: The entropy of an index set S is the expected surprise of choosing an element from S. For a set S, the entropy

$$H(S) = -\sum p_c \log_2(p_c), \text{ where } p_c = \frac{|i \in S : y_i = c|}{|S|}$$

(d) Draw the graph of entropy $H(p_c)$ when there are only two classes. Is the entropy function strictly concave, concave, strictly convex, or convex? Why? What is the significance?
Hint: For the significance, recall the information gain.

2 Ensemble Learning

Ensemble learning is a general technique to combat overfitting, by combining the predictions of many varied models into a single prediction based on their average or majority vote.

(a) **The motivation of averaging.** Consider a set of uncorrelated random variables $\{Y_i\}_{i=1}^n$ with mean μ and variance σ^2. Calculate the expectation and variance of their average. (In the context of ensemble methods, these Y_i are analogous to the prediction made by classifier i.)

(b) **Ensemble Learning – Bagging.** In lecture, we covered bagging (Bootstrap AGGregatING). Bagging is a randomized method for creating many different learners from the same data set.

Given a training set of size n, generate B random subsamples of size n' by sampling with replacement. Some points may be chosen multiple times, while some may not be chosen at all. If $n' = n$, around 63% are chosen, and the remaining 37% are called out-of-bag (OOB) samples.

   (a) Why 63%?

   (b) If we use bagging to train our model, How should we choose the hyperparameter B? Recall, B is the number of subsamples, and typically, a few hundred to several thousand trees are used, depending on the size and nature of the training set.

(c) In part (a), we see that averaging reduces variance for uncorrelated classifiers. Real world prediction will of course not be completely uncorrelated, but reducing correlation will generally reduce the final variance. Reconsider a set of correlate random variables $\{Z_i\}_{i=1}^n$. Suppose $\forall i \neq j$, $\text{Corr}(Z_i, Z_j) = \rho$. Calculate the variance of their average.

3 Decision Trees and Random Forests

Random forests are a specific ensemble method where the individual models are decision trees trained in a randomized way so as to reduce correlation among them. Because the basic decision tree building algorithm is deterministic, it will produce the same tree every time if we give it the same dataset and use the same hyperparameters (stopping conditions, etc.).

Consider constructing a decision tree on data with d features and n training points where each feature is real-valued and each label takes one of m possible values. The splits are two-way, and are chosen to maximize the information gain. We only consider splits that form a linear boundary parallel to one of the axes. In parts (a), (b) and (c) we will consider a standalone decision tree and not a random forest, so no randomization.

(a) **(From Discussion 8)** Prove or give a counter-example: In any path from the root to a leaf, the same feature will never be split on twice. If false, can you modify the conditions of the problem so that this statement is true?

(b) **(From Discussion 8)** Prove or give a counter-example: The information gain at the root is at least as much as the information gain at any other node.
Hint: Think about the XOR function.

(c) Prove or give a counter-example: For every value of $a > 3$, there exists some probability distribution on a objects such that its entropy is less than -1.

(d) One may be concerned that the randomness introduced in random forests may cause trouble. For example, some features or samples may not be considered at all. We will investigate this phenomenon in the next two parts. Consider n training points in a feature space of d dimensions. Consider building a random forest with T binary trees, each having exactly h internal nodes. Let m be the number of features randomly selected at each node. In order to simplify our calculations, we will let $m = 1$. For this setting, compute the probability that a certain feature (say, the first feature) is never considered for splitting.

(e) Now let us investigate the concern regarding the random selection of the samples. Suppose each tree employs n bootstrapped training samples. Compute the probability that a particular sample (say, the first sample) is never considered in any of the trees.

(f) Compute the values of the probabilities you obtained in the previous two parts for the case when there are $n = 2$ training points, $d = 2$ dimensions, $t = 10$ trees of depth $h = 4$ (you may leave your answer in a fraction and exponentiated form, e.g., as $\left(\frac{51}{100}\right)^2$. What conclusions can you draw from your answer with regard to the concern mentioned in the beginning of the problem?