CS188, Winter 2017

Problem Set 1: Decision Trees

Due Feb. 2,2017

1 Problem 1

(a) Problem 1a

Solution:

$$L(\theta) = P(X_1, X_2, ...X_n; \theta)$$

Due to independence,

$$L(\theta) = P(X_1; \theta) * P(X_2; \theta) * ...P(X_n; \theta) = \prod_{i=1}^{n} P(X_i; \theta)$$

Since

$$X_i Ber(p) :$$

$$P(X_i; \theta) = \theta^{X_i}(1-\theta)^{1-X_i}, X_i \in [0, 1]$$

Thus,

$$L(\theta) = \prod_{i=1}^{n} \theta^{X_i}(1-\theta)^{1-X_i}$$

The likelihood does not depend on the order of the random variables, since they are independent.

(b) Problem 1b

Solution:

$$l(\theta) = log(L(\theta)) = log \prod_{i=1}^{N} \theta^{X_i}(1-\theta)^{1-X_i} = \sum_{i=1}^{N} log(\theta^{X_i}(1-\theta)^{1-X_i})$$

$$l(\theta) = \sum_{i=1}^{N} (X_i log(\theta) + (1-X_i) log(1-\theta))$$

$$\frac{\delta l}{\delta \theta} = \sum_{i=1}^{N} (\frac{X_i}{\theta} - \frac{1 - X_i}{1 - \theta})$$

Setting

$$\sum_{i=1}^{N} X_i = \bar{X}$$

$$\frac{\delta l}{\delta \theta} = n(\frac{\bar{X}}{\theta} - \frac{1 - \bar{X}}{1 - \theta})$$

$$\frac{\delta^2 l}{\delta \theta^2} = n(-\frac{\bar{X}}{\theta^2} - \frac{1 - \bar{X}}{(1 - \theta)^2})$$

$$\frac{\delta l}{\delta \theta} = 0 \longrightarrow \theta = \bar{X} = \sum_{i=1}^{N} X_i$$

Since

$$\frac{\delta^2 l}{\delta \theta^2} < 0$$

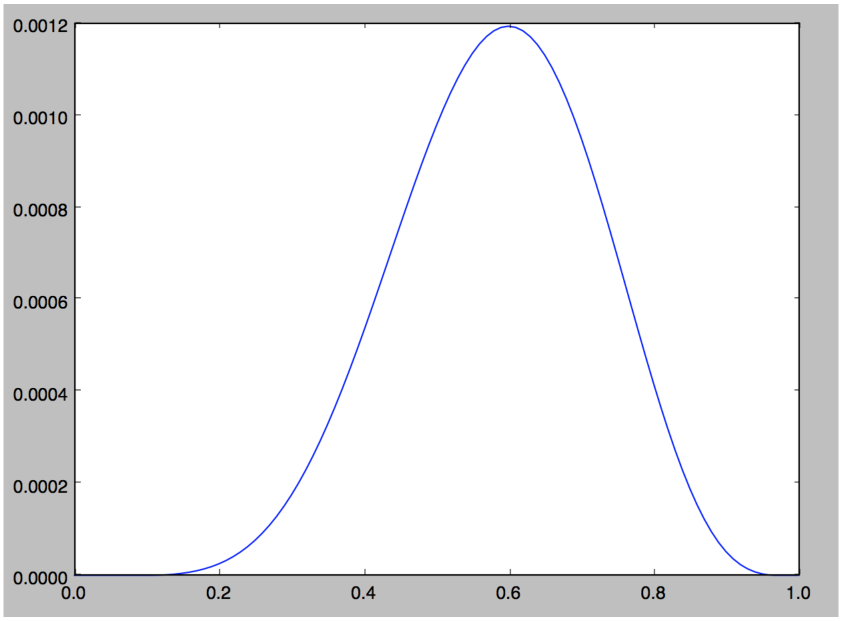
at

$$\theta = \bar{X}$$

$$\theta = \bar{X} = \sum_{i=1}^{N} X_i$$

is the MLE.

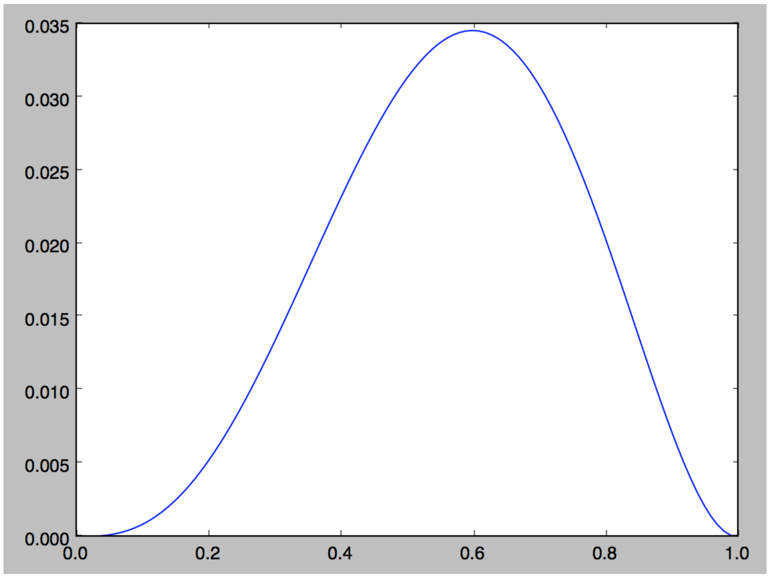
(c) Problem 1c **Solution:**



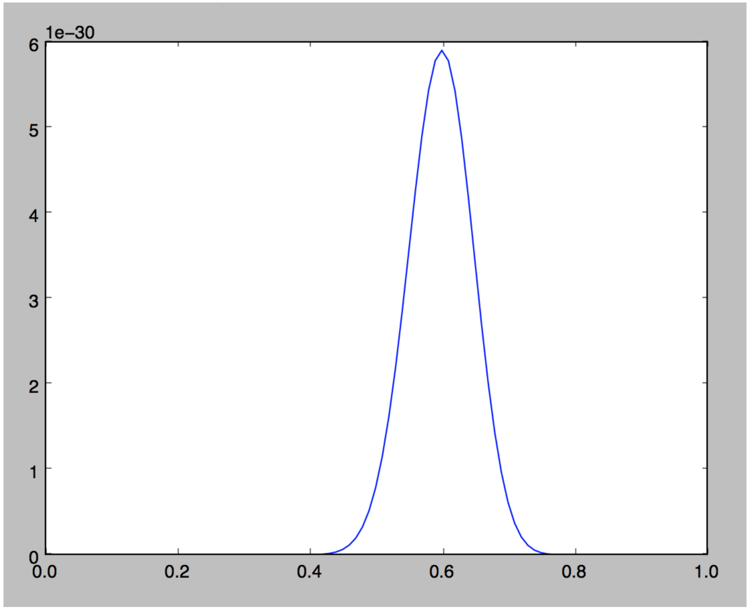
The maximum can clearly be seen to occur at theta = 0.6 (the computer finds the max to occur at 0.599999...). This agrees with the closed form answer since from the data, we calculate that

$$\bar{X} = 0.6$$

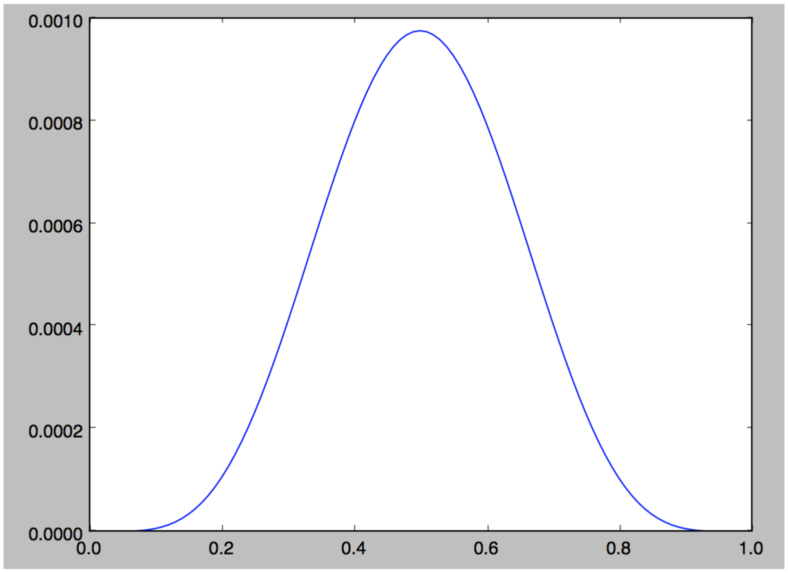
(d) Problem 1d **Solution:** Plot for n = 5, three 1's:



Plot for n = 100, 60 ones:



Plot for n = 10, 5 ones:



For both the n = 5 and n = 100 datasets, the theta that maximized the likelihoods were both 0.6 (computer gave 0.5999999...). However, the maximum likelihoods themselves were 0.03456 and $5.9 * 10^{-30}$. This makes sense since it is much less likely to see a specific dataset from the bernoulli distribution if the dataset is very large, but since both datasets have the same expected value for X, the theta should be the same. For the n = 10 and 5 ones dataset, theta = 0.5 and ML = $9.7 * 10^{-4}$ was obtained. The parameter given by the plot matches the closed form answer, since for this dataset, $\theta = \bar{X} = 0.5$

In general, the MLE is equivalent to the proportion of observed ones, and as we take $n \longrightarrow \infty$ the function's spread narrows and the peak becomes more apparent, but the value of the likelihood decreases.

2 Problem 2

Solution: Solution to problem 2

(a) Problem 2a

Solution: The best one-leaf decision tree will always predict a 1. Since the label is 0 when all the first three features are 0, we must compute the number of different arrangements such that the first three features are 0. This is $1 * 1 * 1 * 2 * 2 * ...2 = 2^{n-3}$ possibilities. So the best decision tree will make 2^{n-3} mistakes, or an error rate of $\frac{1}{8}$.

(b) Problem 2b

Solution: There is no such split. If we split on the value of $X_i, i >= 4$, the proportion of ones in each split will be the same $(\frac{7}{8})$ so a 1 will be predicted in both leaves. If we split on one of $X_i, i < 4$, one leaf will have data that is only ones, and the proportion of data in the other leaf with label 1 will be $\frac{7}{8} - \frac{1}{8} = \frac{3}{4}$. Therefore, the tree will predict 1 in both leaves. So any case with one split will make the same number of mistakes as a 1-leaf tree.

(c) Problem 2c **Solution:**

$$H[Y] = -\sum_{i=1}^{K} P(X = a_i) log P(X = a_i) = \frac{1}{8} log(\frac{1}{8}) + \frac{7}{8} log(\frac{7}{8})$$

$= 0.543$

(d) Problem 2d **Solution:** Yes. Using any of $X_i, i <= 3$ gives an entropy of $\frac{1}{8} log(4) + \frac{3}{4} log(\frac{4}{3}) = 0.406$

3 Problem 3

Solution: Solution to Problem 3

(a) Problem 3a **Solution:** If $\frac{p_k}{p_k+n_k}$ is the same for all k, then we must have, for all k:

$$\frac{p_k}{p_k + n_k} = \frac{p}{p+n}$$

. Using the equation for gain, we obtain:

$$G = B(\frac{p}{p+n}) - B(\frac{p}{p+n})\frac{1}{p+n}\sum_{k=1}^{d} p_k + n_k$$

$$= B(\frac{p}{p+n}) - B(\frac{p}{p+n})\frac{1}{p+n}(n+p)$$

$$= B(\frac{p}{p+n}) - B(\frac{p}{p+n}) = 0$$

4 Problem 4

Solution: Solution to Problem 4

(a) Problem 4a **Solution:**
Pclass: Roughly the same number of people survived or didn't survive for a pclass value of 2. For first-class, about 60 more people survived than didn't. For pclass = 3, the didn't survive (survived = 0) was significantly higher.
Sex: If sex = 0, survived was more frequent. For a value of 1, didn't survive was much more frequent.
Age: The highest non-survival frequency age range was between 20 and 30 years. The only time the survived frequency is more than the didn't survive frequency is for ages between 0 and 10.
SibSp: The sibSp value of 0 was the most common, and this value had a higher non-survival rate. For a SibSp value of 1, the rates are roughly equal with a few more surviving than not, and SipSp values of 3, 4, and 5 are relatively uncommon.
Parch: The most frequent value was 0, for which a majority did not survive. For parch values of 1 and 2, the frequency of survival was slightly higher. A parch value of 3 through 7 was very uncommon.
Fare: Those who paid the lowest fare had the highest non-survival ratio. As fare increased, the frequency of survival exceeds the frequency of non-survival.
Embarked: The most frequent value was 2, which had a higher non-survival frequency. An embarked value of 0 had a higher survival rate, while the uncommon value of 1 had a slightly higher non-survival rate.

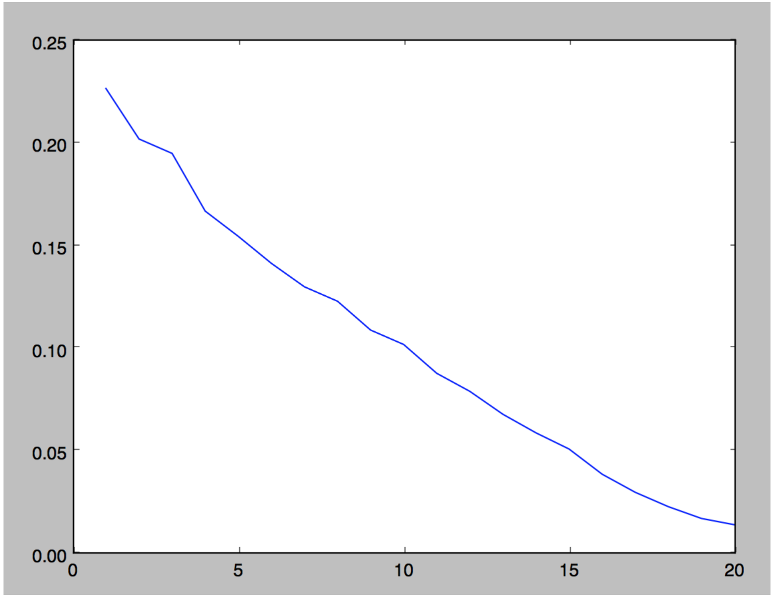
(b) Problem 4b **Solution:** I obtained a training error of 0.485.

(c) Problem 4c **Solution:** The training error for sci-kit learn's DecisionTreeClassifier was 0.014. The "entropy" criterion was used, and all other parameters were unmodified.

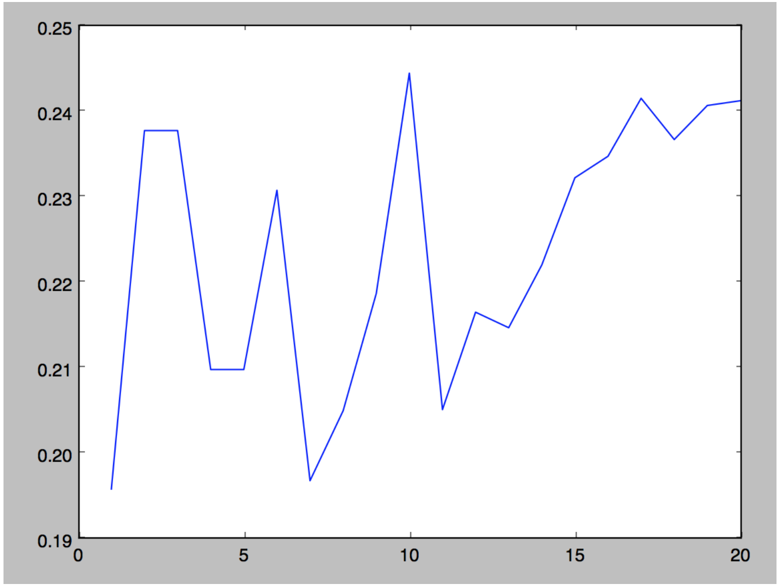
(d) Problem 4d **Solution:** The training and testing error were accumulated over n = 100 trials with an 80/20 random split of data. The errors were:
For DecisionTreeClassifier: training error: 0.01230228471
test error: 0.243706293706
For RandomClassifier: training error: 0.516695957821
test error: 0.482517482517

For MajorityClassifier: training error: 0.397188049209
test error: 0.433566433566

(e) Problem 4e **Solution:** Plot of depths (1-20) vs training error:



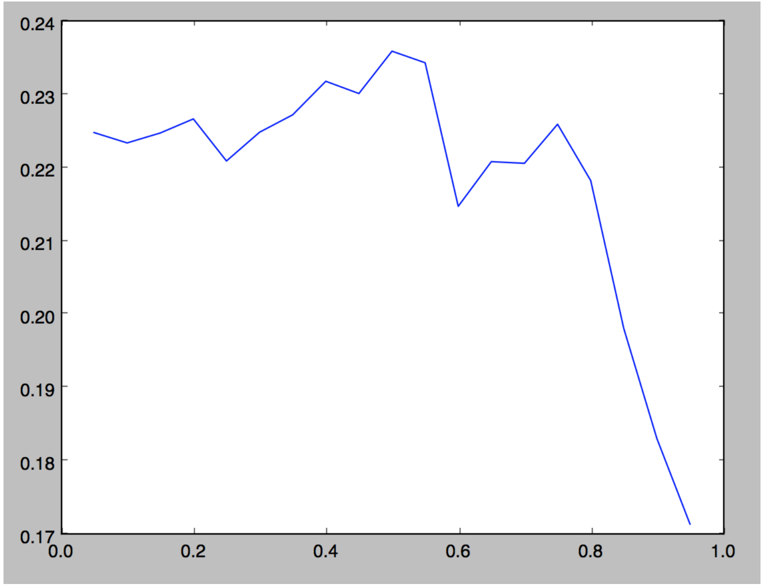
Plot of depths (1-20) vs testing error:



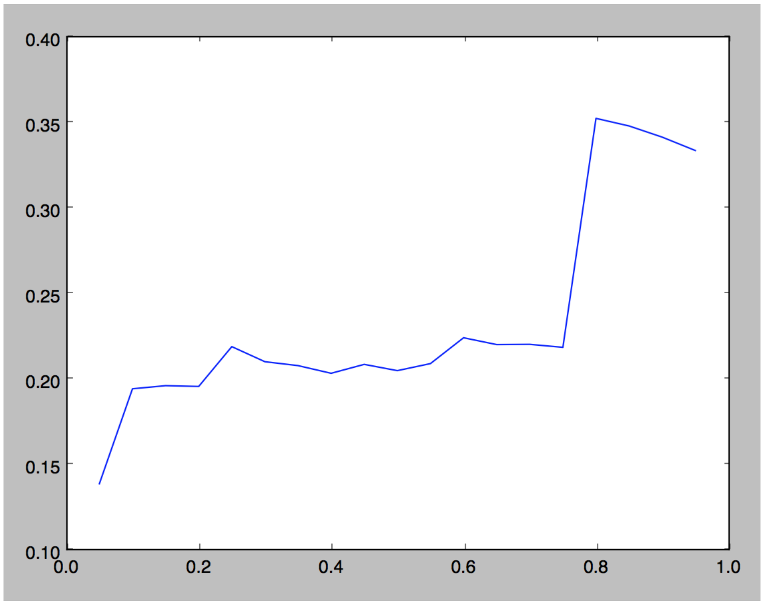
We see overfitting as the maximum depth approaches 20. This is indicated by the continued reduction in error on the training data, but increased error on the testing data as our max depth increases. This indicates that the model has low bias and high variance, and has begun to ”memorize” the training data. The best depth limit, if we consider the ability to generalize, should be determined by the error on the testing data. In this case, a max depth of 1 would minimize the training error the most, but a max depth of 7 is also a close second.

(f) Problem 4f **Solution:**

Plot of splits against training error:



Plot of splits against testing error:



Similar to above, we again see overfitting as we use more data for training. This is again because we have more data to train on, so our classifier mistakes this data as more and more resemblent of the entire population, leading to worse generalization.