CM146, Fall 2017
Problem Set 1: Decision Trees
Due Oct 23,2017

Author: Shivraj Gill

1 Problem 1

(a) Problem 1a

Solution: Solution to problem 1a

We define $P(X_i;\theta) = \theta^{X_i}(1-\theta)^{1-X_i}$
Then,

$$L(\theta) = P(X_1, ..., X_n;\theta) \text{ and because } X_1, ...X_n \text{ are i.i.d,}$$
$$= \prod_{i=1}^{n} P(X_i;\theta) = \prod_{i=1}^{n} \theta^{X_i}(1-\theta)^{1-X_i}$$

The likelihood does not depend on the order in which the random variables were observed because they are independent and identically distributed. In other words, because each random variable is independent, the ordering does not matter since the variables are not dependent on each other.

(b) Problem 1b **Solution:** Solution to problem 1b

$\ell(\theta) = log(L(\theta)) = \sum_{i=1}^{n}[X_i log(\theta) + (1-X_i)log(1-\theta)]$

Next we take the first and second derivatives,

$$\frac{\partial \ell(\theta)}{\partial \theta} = \sum_{i=1}^{n} \frac{X_i}{\theta} - \sum_{i=1}^{n} \frac{1-X_i}{1-\theta}$$
$$\frac{\partial^2 \ell(\theta)}{\partial \theta^2} = -\sum_{i=1}^{n} \frac{X_i}{\theta^2} - \sum_{i=1}^{n} \frac{1-X_i}{(1-\theta)^2} = -\sum_{i=1}^{n} \frac{X_i}{\theta^2} - \frac{n-\sum_{i=1}^{n} 1-X_i}{(1-\theta)^2}$$

To maximize θ, we need to set $\frac{\partial \ell(\theta)}{\partial \theta} = 0$ and solve for θ. Further, we need to check the sign of $\frac{\partial^2 \ell(\theta)}{\partial \theta^2}$. If $\frac{\partial^2 \ell(\theta)}{\partial \theta^2}$ is negative, then we have found the maximum. Let's first solve for θ.

$$0 = \frac{\partial \ell(\theta)}{\partial \theta} = \sum_{i=1}^{n} \frac{X_i}{\theta} - \sum_{i=1}^{n} \frac{1-X_i}{1-\theta}$$
$$0 = \frac{n}{\theta} \sum_{i=1}^{n} X_i - \frac{n - \sum_{i=1}^{n}(1-X_i)}{1-\theta}$$

After cross multiplying and rearranging terms we get,

$$\sum_{i=1}^{n} X_i - \theta \sum_{i=1}^{n} X_i = n\theta - \theta \sum_{i=1}^{n} X_i$$

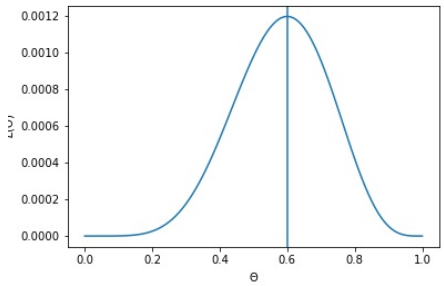
The $\theta \sum_{i=1}^{n} X_i$ terms cancels out, so we get

$$\hat{\theta}_{MLE} = \frac{\sum_{i=1}^{n} X_i}{n}$$

$\hat{\theta}_{MLE}$ is equal to the sample mean.

We know that $\frac{\partial^2 \ell(\theta)}{\partial \theta^2} = -\sum_{i=1}^{n} \frac{X_i}{\theta^2} - \frac{n - \sum_{i=1}^{n} 1 - X_i}{(1-\theta)^2}$ is negative for any value of θ because for both terms, the numerator and denominator are positive. For the first term, we know that $\sum_{i=1}^{n} X_i$ will be positive as will θ^2. For the second term, we know that $n - \sum_{i=1}^{n} 1 - X_i$ will always be positive since $n = \sum_{i=1}^{n} 1 - X_i$ if and only if $X_i = 1$ for all n, otherwise $n > \sum_{i=1}^{n} 1 - X_i$. Further, we know that $(1-\theta)^2$ will be positive. This proves that $\frac{\partial^2 \ell(\theta)}{\partial \theta^2}$ will be positive for any value of θ. Thus, $\hat{\theta}_{MLE} = \frac{\sum_{i=1}^{n} X_i}{n}$ maximizes $\ell(\theta)$ and hence $L(\theta)$.

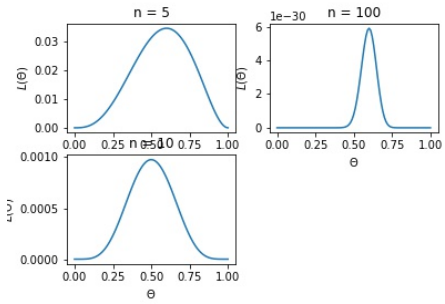
(c) Problem 1c **Solution:** Solution to problem 1c



The answer does agree with our closed form since the sample mean is 0.6.

(d) Problem 1d **Solution:** Solution to problem 1d

We see that the maximum likelihood stays the same for each data set. However, the shape of the likelihood function changes when the number of points changes as seen in the plots. For larger values of n (number of data points), the values of the likelihood function are closer to that of the maximum likelihood value, hence we got a narrow shaped curve.



2 Problem 2

(a) Problem 2a

Solution: Solution to problem 2a

The best 1-leaf decision tree would guess that all the labels are 1 since that will be the most common label. Thus, since the tree can only make a mistake when $X_1, X_2, and X_3$ are zero, then it can make a total of 2^{n-3} mistakes. This represents the total number of combinations that you can make with 2^n samples if $X_1, X_2, and X_3$ are zero. This will be true for n $n \geq 4$.

(b) Problem 2b

Solution: Solution to problem 2b

No, there is no split that reduces the number of mistakes by one since the outcome of the labels are determined by the value of three features $(X_1 \vee X_2 \vee X_3)$. For instance, if we split at $X_1, X_2, or X_3$, the data will be split onto one leaf that is all one's and another leaf that will have more ones than zeros. The other leaf will have more ones because we know that we have 2^{n-1} ones and 2^{n-3} zeros on this leaf. Thus, we have more ones than zeros on both leaves, and so the algorithm will choose 1 for all of them, so the number of mistakes will be the same as the one-leaf decision tree. Now say we split at $X_i, for i \geq 4$, then both leaves will still have majority ones since, it just essentially divides the dataset in half while keeping the proportion of zeros and ones the same. This is because $X_i for i \geq 4$, have no influence on the label Y. Because this split would have majority ones on both leaves, then the algorithm would choose ones for all the labels and the number of mistakes would be unchanged.

(c) Problem 2c

Solution: Solution to problem 2c

$H[Y] = -[P(Y=1)\log P(Y=1) + P(Y=0)\log P(Y=0)]$
We know that P(Y = 0) = $\frac{2^{n-3}}{2^n} = \frac{1}{8}$ since the numerator is all the different ways you can get Y = 0 and the denominator is the total number of samples.
P(Y = 1) = 1 - P(Y=0) = $\frac{7}{8}$.
Thus, $H[Y] = -[\frac{1}{8}\log\frac{1}{8} + \frac{7}{8}\log\frac{7}{8}] = 0.5435$

(d) Problem 2d

Solution: Solution to problem 2d

Yes, if we split at $X_1, X_2, or X_3$ we would reduce the entropy since these features hold information about our labels.
For i = 1, 2, or 3, the conditional entropy from splitting on X_i is
$H[Y|X_i] = P(X_i=1)H[Y|X_i=1] + P(X_i=0)H[Y|X_i=0]$
$H[Y|X_i] = \frac{1}{2}*0 + \frac{1}{2}[-P(Y=1|X_i=0)\log(P(Y=1|X_i=0)) - P(Y=1|X_i=1)\log(P(Y=0|X_i=1))] = 0 + \frac{1}{2}[-\frac{3}{4}\log(\frac{3}{4}) - \frac{1}{4}\log(\frac{1}{4})] = 0.406$

3 Problem 3

Solution: Solution to problem 3

We know that Gain = H[S] - $\sum_{k=1}^{s} P(X_j = a_k)H[S|X_j = a_k]$ where a_k represents all of the different values of X_j
We can say that $P(X_j = a_k) = \frac{|S_k|}{|S|}$ where $|S_k|$ represents the size of the set S_k since we know X_j splits the dataset into k disjoint subsets S_k and we know the number of positive and negative examples, p_k, n_k, in each set. Thus, $P(X_j = a_k)$ will equal the number of examples in each set over the total number of examples. In other words, $P(X_j = a_k) = \frac{p_k+n_k}{n+p}$.
Further we can say that $H[S|X_j = a_k] = H[S_k]$ since we know that when $X_j = a_k$ we are in the subset S_k. Because, $H[S_k] = B(\frac{p_k}{p_k+n_k})$ and $\frac{p_k}{p_k+n_k} = \frac{p}{p+n}$ since the ratio is the same for all S_k. Thus, $H[S_k] = H[S] = B(\frac{p}{p+n})$.
Combining this all together, we get
Gain = $B(\frac{p}{p+n}) - B(\frac{p}{p+n})\sum_{k=1}^{s}\frac{|S_k|}{|S|} = B(\frac{p}{p+n}) - B(\frac{p}{p+n})\frac{1}{p+n}\sum_{k=1}^{s} p_k + n_k$

$\sum_{k=1}^{s} p_k + n_k = p + n$ since we are covering all of S.
Then we get,

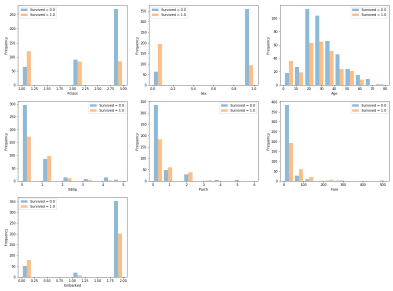
$$Gain = B(\frac{p}{p+n}) - B(\frac{p}{p+n})\frac{1}{p+n}(p+n) = 0$$

Thus, we get the intended result that Gain = 0.

4 Problem 4

(a) Problem 4a

Solution: Solution to problem 4a



Here I assume that Survive = 1 means surviving and Survive = 0 means not surviving.
For the Pclass feature, we see that people with Pclass = 3 had a significantly less chance of surviving and those with Pclass = 1, had a greater chance of surviving. For the sex feature, we see that females (Sex = 0) had a greater chance of surviving and men (Sex = 1) had a smaller chance of surviving. For the Age feature, we see that children (age ¡ 10) had a greater chance of surviving, whereas every other age group had a smaller chance of surviving especially for people in the 20-30s range. For the SibSp feature, we see that people with no siblings had a smaller chance of surviving. For the Parch feature, we see that when Parch = 1, people had a smaller chance of surviving. For the fare feature, we see that people with a fare less than 50, had a smaller chance of surviving. Lastly, for the Embarked feature, we see that people with Embarked = 2 had a smaller chance of surviving.

(b) Problem 4b

Solution: Solution to problem 4b

I got a training error of .485 as expected.

(c) Problem 4c

Solution: Solution to problem 4c

I get a training error of 0.014, which is a lot smaller than the other two training errors since a Decision Tree memorizes patterns in the training data and hence is able to predict the training labels almost perfectly.

(d) Problem 4d **Solution:** Solution to problem 4d

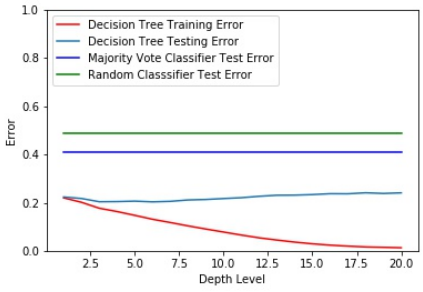
Majority Vote Classifier Average Training Error: 0.403778558875
Majority Vote Classifier Average Test Error: 0.407342657343

Random Classifier Average Training Error: 0.489015817223
Random Classifier Average Test Error: 0.486573426573

Decision Tree Classifier Average Training Error: 0.0115289982425
Decision Tree Classifier Average Test Error: 0.239090909091

We see that the average training error and average test error are the same for the Majority Vote Classifier and Random Classifier since they are very simple models that do not capture any patterns in the data, so they do not overfit. We see that the Decision Tree Classifier has a very small training error since our model is fit according to the training data, and so it is able to predict the training labels almost perfectly. Note that even though our Decision Tree average test error is a lot larger than the average training error, it is still smaller than the average test error of the other two classifiers .

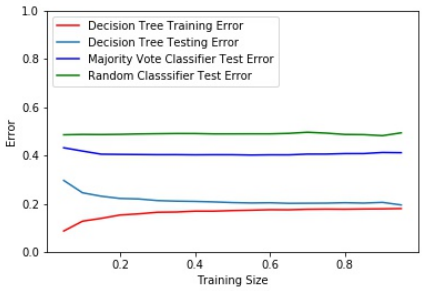
(e) Problem 4e **Solution:** Solution to problem 4e



According to our plot, the optimal depth would be at 3 since that gives us the smallest test error, and the test error increases after 3. Because

we see an steady increase in the test error after depth = 3, we can say that there is slight overfitting after 3 since we consistently get worse at predicting labels for the test set.

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



We see that as the size of the training set increases, the training error increases and the test error decreases. As a result, these two curves get increasingly closer, but the training error remains smaller than the test error. This shows that with more training data, the decision tree gets better at capturing the patterns in the data resulting in better predictions of the test labels. However, because the test error and training error begin to plateau when the training size ¿ 0.7 , there's a strong chance that using more training data, would not necessarily result in a significantly better test error. Alternatively, we could use feature engineering techniques such as reducing the number of redundant features and finding more relevant features in order to reduce the test error even further.