B365 Homework 6

1. Consider the table below

node	errors/n	terminal
1	50/100	0
2	25/50	0
3	25/50	0
4	10/25	0
5	5/25	1
6	1/30	1
7	5/20	1
8	2/10	1
9	7/15	0
18	1/10	1
19	1/5	1

giving the number of errors and the number of examples at each node in a tree classifier. The nodes are numbered so that 1 is the root, while the children of node k are $2k$ and $2k+1$. The "terminal" column says if a node is terminal or not.

   (a) Construct the tree as a graph (the usual depiction of a tree) labeling the nodes with numbers and giving the errors/n fractions.

   (b) Compute $R(T)$ where T is the tree given in the example and $R(T)$ is our estimated probabability of error for the tree.

   (c) Explain why you do or do not believe this is an accurate representation of the tree's performance on new data.

   (d) Compute the optimal penalized risk, R^*_α for each node of T where $\alpha = .04$. Give the corresponding optimal tree T_α.

   (e) T_0 is the optimal tree when $\alpha = 0$ which is the initial tree T. As α increases from 0, which will be the first branch to be pruned?

   (f) As α continues to increase, other branches will be pruned. Construct the entire family of optimal rooted trees for all values of α by continuing to increase α and noting where prunings occur. Sketch the family of optimal trees, $\{T_\alpha\}$, from left to right, beginning with T_0 on the left, giving the smallest value of α for which each tree T_α would be the optimal choice.

2. The following code shows two things. First we show how to create a matrix from the file "tree_data.dat," available from Canvas, which stores the data above. In the resulting matrix, the columns contain the number of errors, the number of examples, and the boolean variable describing the node as terminal or not. The rows 10 through 17 are all zeros and are unused. This way $X[i,]$ gives the data associated with tree node i. The factorial function, shown below, gives an example of a simple recursive function in R, which you would call by, e.g. factorial(5).

```
X = matrix(scan("tree_data.dat"),byrow=T,ncol = 3)

factorial <- function(i) {
  if (i == 1) { return(1); }
  else return(i*factorial(i-1))
}
```

   (a) Write a recursive function in R that takes as input the number of a node and returns the optimal risk associated with that node, with a split penalty of $\alpha = .04$. When you run your function with input 1 (the root node) it should return the optimal risk for the entire tree.

(b) $T_{\alpha=.04}$ is the associated optimal tree. Explain precisely how you would construct this tree from the results of the recursive routine.

(c) Explain what problem has $T_{\alpha=.04}$ as its optimal solution.

3. Consider the following table of cross validation on tree induction for a two-class classification problem,as discussed in class:

```
Root node error: 1524/3100 = 0.49161

n= 3100
           CP nsplit rel error   xerror      xstd
1  0.54757282      0 1.0000000 1.025890 0.0180141
2  0.11909385      1 0.4524272 0.462136 0.0151731
3  0.06601942      2 0.3333333 0.368932 0.0139601
4  0.05372168      3 0.2673139 0.293204 0.0127297
5  0.05242718      4 0.2135922 0.238835 0.0116698
6  0.03430421      5 0.1611650 0.186408 0.0104615
7  0.01326861      6 0.1268608 0.150809 0.0095013
8  0.01165049      8 0.1003236 0.127508 0.0087912
9  0.01035599      9 0.0886731 0.119741 0.0085368
10 0.00776699     10 0.0783172 0.100324 0.0078541
11 0.00550162     12 0.0627832 0.083495 0.0071968
12 0.00517799     14 0.0517799 0.077023 0.0069238
13 0.00453074     15 0.0466019 0.073786 0.0067825
14 0.00291262     17 0.0375405 0.066019 0.0064285
15 0.00258900     19 0.0317152 0.062783 0.0062741
16 0.00226537     20 0.0291262 0.057605 0.0060178
17 0.00194175     22 0.0245955 0.055663 0.0059185
18 0.00161812     24 0.0207120 0.054369 0.0058512
19 0.00129450     26 0.0174757 0.054369 0.0058512
20 0.00097087     30 0.0122977 0.050485 0.0056440
21 0.00064725     34 0.0084142 0.045955 0.0053910
22 0.00032362     42 0.0032362 0.048544 0.0055371
23 0.00000000     52 0.0000000 0.046602 0.0054279
```

(a) In the "rel error" column we get a value of 0. for the 23rd row. Explain what this number means.

(b) In terms of error rate, how well do you think the resulting maximally-deep tree will perform on *different* data from the sample population.

(c) Consider the tree that makes *no* splits — i.e. the one that simply classifies according to the most likely class. How well will this tree classify new data from the same population.

(d) Juding from the table, what appears to be your best choice of complexity parameter α? In what sense is your α value best?

4. As a result of a recent exam, an instructor of a class believes that 80% of the students do yet not understand a topic sufficiently well. The instructor wishes to implement a Naive Bayes classifier to estimate each student's probability of understanding. Students are asked a sequence of 7 true or false questions. The instructor assumes that the responses to these questions are conditionally independent given the student's state of knowledge — understands or does not understand. Of course, understanding is not really a binary attribute in real life as there are degrees of understanding and various aspects to understanding, though we regard it as binary here.

   The following code fragment creates a 2x7 matrix, x, when $x[i, j]$ is the probability that a student will answer the jth question correctly where her state of knowledge is i. Here $i = 1$ corresponds to "not understanding" and $i = 2$ corresponds to understanding. For ease of computation we compute the 2x7x2 array, z, where $z[i, j, k]$ gives the probability that a student will give answer k (k=1 means wrong and k=2 means right) to question j, given her state of knowledge is i.

```
x = matrix(c(.7, .6, .5, .5, .5, .5, .7, .8, .7, .6, .7, .9, .8, .9 ),byrow=T,nrow=2);
z = array(0,c(2,7,2));
```

(a) Write R code to fill in the matrix z to be as described in the problem.

(b) Using your z matrix create an R function that receives a vector of 7 test answers which are either wrong or right. For instance, if the answers are c(0,0,0,0,1,1,1), that would mean the student answered only the last three questions correctly. The function should return the probability that the student has understood the subject, using a Naive Bayes classifier.