Homework 4: Decision Trees and Cross-validation

Due: Electronic Submission by Tuesday, March 26, 2019 at 11:59 pm

1. Build a decision tree to predict whether or not a credit card user will default on his/her credit card payment next month using the dataset found here `https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients#`.

   You need to complete the following steps:

   (a) Import the data into R. Convert all categorical variables to "factor" type variables.

   (b) Start by setting a random seed so that your results will be reproducible by me when I grade your work.

   (c) Randomly separate the data into training and test datasets of equal size.

   (d) Build a single, unpruned tree on the training data and use it to predict for the test data. Be sure to remove the ID variable when specifying the model in the tree() function.

      i. Plot the resulting tree with labels on each node.

      ii. Give the overall misclassification rate for both training and test data.

      iii. Give the misclassification rate for the subgroups of people who will default and people who won't default next month.

      iv. How many terminal nodes are in the unpruned tree?

   (e) Use the built-in cross-validation function for trees to determine the optimal pruning of your first tree. What is the optimal number of terminal nodes?

   (f) Prune the tree. Use the pruned tree to make predictions for your test data and determine the misclassification rate.

   (g) Run a random forest using your training data (you may use all the data, but it will take a while) using the default value for `mtry`. What is your estimate of the misclassification rate? (It is one of the outputs of the random forest function). Compare the misclassification rate of your pruned tree and the random forest.

   (h) Rerun the random forest where each split is chosen from a random sample of 4 of the predictors. Is this better than the default?

2. Using the built-in dataset called `iris`, a decision tree was constructed to predict `Species` based on the 4 remaining variables using the code below:

```
> head(iris)

  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa

> library(tree)
> iris.tree <- tree(Species~.,data=iris)
> summary(iris.tree)

Classification tree:
tree(formula = Species ~ ., data = iris)
Variables actually used in tree construction:
[1] "Petal.Length" "Petal.Width"  "Sepal.Length"
Number of terminal nodes:  6
Residual mean deviance:  0.1253 = 18.05 / 144
Misclassification error rate: 0.02667 = 4 / 150
```

```
> iris.tree

node), split, n, deviance, yval, (yprob)
      * denotes terminal node

 1) root 150 329.600 setosa ( 0.33333 0.33333 0.33333 )
   2) Petal.Length < 2.45 50   0.000 setosa ( 1.00000 0.00000 0.00000 ) *
   3) Petal.Length > 2.45 100 138.600 versicolor ( 0.00000 0.50000 0.50000 )
     6) Petal.Width < 1.75 54  33.320 versicolor ( 0.00000 0.90741 0.09259 )
      12) Petal.Length < 4.95 48   9.721 versicolor ( 0.00000 0.97917 0.02083 )
        24) Sepal.Length < 5.15 5   5.004 versicolor ( 0.00000 0.80000 0.20000 ) *
        25) Sepal.Length > 5.15 43   0.000 versicolor ( 0.00000 1.00000 0.00000 ) *
      13) Petal.Length > 4.95 6   7.638 virginica ( 0.00000 0.33333 0.66667 ) *
     7) Petal.Width > 1.75 46   9.635 virginica ( 0.00000 0.02174 0.97826 )
      14) Petal.Length < 4.95 6   5.407 virginica ( 0.00000 0.16667 0.83333 ) *
      15) Petal.Length > 4.95 40   0.000 virginica ( 0.00000 0.00000 1.00000 ) *

> levels(iris$Species)

[1] "setosa"     "versicolor" "virginica"
```

Use only the output above to answer the following questions:

(a) How many observations fell into the 6th node?

(b) Of the observations in node 6, what percent are in each species categories?

(c) What is the predicted category for observations in this node?

(d) According to what rule will all observations in node 6 be split in the next level of the tree?

(e) The deviance of node 6 is defined as $-2(n_1 \log p_1 + n_2 \log p_2)$ where n_i is the number of node observations in category i and p_i is the proportion of node observations in category i. Here, category 1 is versicolor and category 2 is virginica. Recall that larger deviance indicates more impurity and less homogeneity in a node. Use arithmetic in R to verify the calculated deviance of node 6 is 33.32.

(f) Think of a set of predictor variables which would cause an observation to be sent to the right at each possible split in the tree.