DS 100: Principles and Techniques of Data Science **Date: April 19, 2019**

Discussion #10

Name:

Classification and Regression Trees and Random Forests

1. Answer the following multiple choice questions:

   (a) When predicting a qualitative outcome, the predicted class at each terminal node is:

   $\surd$ **the most common class in the node** ☐ the average class for the node ☐ neither

   (b) When predicting a quantitative outcome, the predicted value at each terminal node is:

   ☐ the most common outcome in the node $\surd$ **the average outcome for the node**
   ☐ neither

   (c) The main idea behind ensemble prediction methods such as Random Forests is that aggregating predictions from predictors applied to multiple bootstrap samples of the learning set:

   ☐ reduces bias $\surd$ **reduces variability** ☐ reduces both bias and variability
   ☐ none of these

   (d) The following question is a short answer question. What can you do if your decision tree is overfitting?

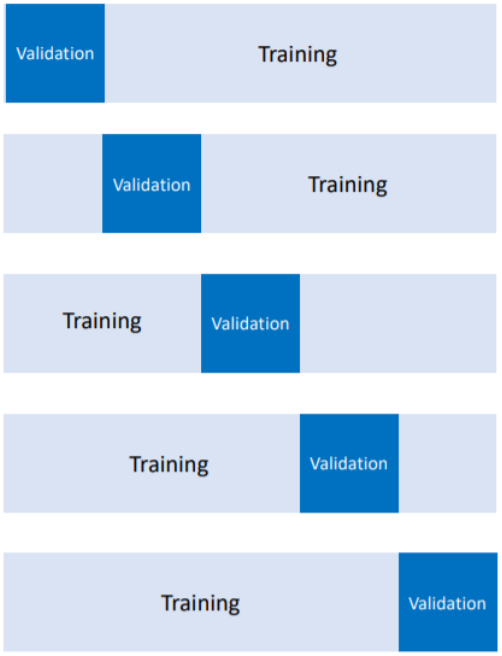
   Solution: Use a random forest, restrict the maximum tree depth, or prune tree to a particular depth after it is created (do cross-validation to check how much pruning is good).

Tree-Based Methods and Cross-Validation

2. Now that we're more familiar with tree based methods, we can discuss how to properly implement them to solve regression and classification problems. One of the most difficult aspects of using these methods is selecting optimal hyperparameters (e.g. the depth of the tree, or the number of trees in a Random Forest). Luckily, cross-validation methods can be used for this task.

(a) Recall that cross-validation (CV) can be used to assess the risk (i.e. the performance) of a model using only the learning set. Describe in your own words how this is accomplished using K-fold CV. Can you illustrate the procedure of 5-fold CV?

Solution: In the case of K-fold CV, the learning set is randomly divided into K disjoint sets (i.e. folds) of almost equal size. Then, for each of the K iterations, a model is trained on K-1 of the folds (which is called the training set), and the risk of the model is computed on the fold that was held out (the validation set). Each fold is held out once. The cross-validated risk is computed by taking the average of the risks from each iteration.



(b) Now suppose we wish to fit a tree in a binary classification context. We can use CV methods to select the optimal value for the maximum tree depth parameter. Write the steps to accomplish this using K-fold cross-validation and a vector d of possible maximum tree depths, $d = (1, 2, 3, ..., N)$.

Solution:

1. Split the data into a learning set and a test set
2. Split the learning data into K-folds
3. Initialize a vector r to contain the CV risk of each tree model
4. For each depth in d, compute the CV-risk of the tree model with depth d and store it in r
5. Identify the maximum tree depth value associated with the minimum CV-risk in r. This is the optimal model.
6. Train the tree model on the entire leaning set using the optimal tree depth found in step 5.

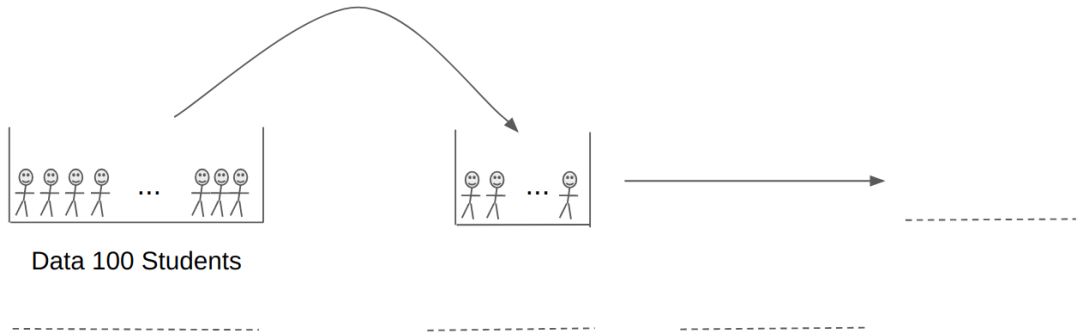
(c) Suppose now you wish to fit a Random Forest model instead of a classification tree. How would you modify your pseudo code to identify the appropriate maximum depth and the number of trees to include in your model?

Solution: In addition to creating a vector of possible maximum tree depths, create a vector containing the possible number of trees. A grid can then be created, where each entry is a unique combination of the elements in each vector. The CV-risk can then be calculated for each possible combination of these hyperparameters.

The Bootstrap

3. Suppose we wish to estimate the proportion of cigarette smokers among the students enrolled in Data 100.

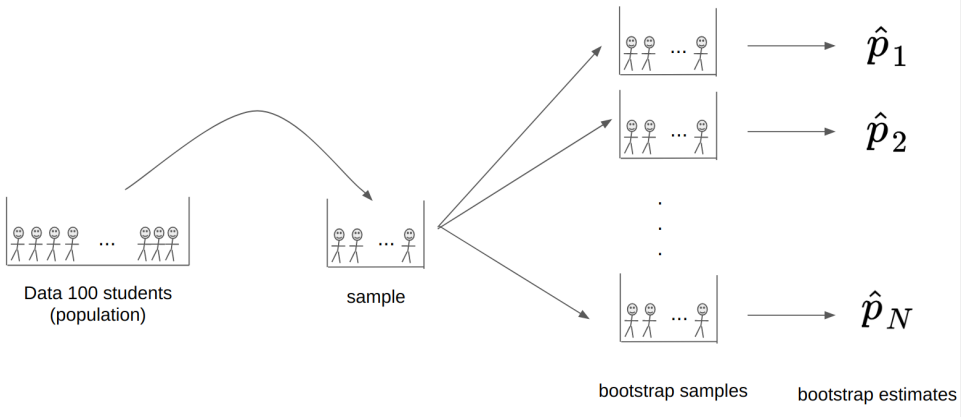
(a) Using a box model, we wish to represent the process of estimating the proportion of cigarette smokers. Fill the dotted lines in the image below using the following words: population, sample, compute estimate, and $\hat{p}$. Given that there are about 900 students enrolled in Data 100, should the random sample be drawn with or without replacement?



Solution: The box to the left represents all students enrolled in Data 100. This is the population. A simple random sample of students from this population is then selected and placed in the box to the right. This is the sample. We then compute the estimate of the proportion of smokers, $\hat{p}$.

For a fixed population size, the choice of sampling with or without replacement depends on the size of the sample. If the sample is much smaller than the size of the population, then sampling without replacement is acceptable since observations are approximately independent. However, if the size of the sample is large, sampling should be perfomed with replacement to ensure that each observation in the sample is drawn independently.

(b) Now that we have our estimate, we would like to carry out some statistical inference by creating a confidence interval. Assuming that our sample is a good representation of the population, we can use the bootstrap method to do this. Using a box model, illustrate the process of generating bootstrap estimates for the proportion of smokers in Data 100.



Solution:

The bootstrap samples are generated by sample with replacement from the original sample. Each bootstrap sample has the same length as the original sample.

(c) Describe how you would compute the 95% confidence interval of proportion of smokers

using bootstrap estimates.

> **Solution:** We can use the bootstrap estimates to estimate the distribution of the sample proportion of smokers. Therefore, the lower-endpoint of the 95% confidence interval is approximated by the 2.5^{th} percentile of the bootstrap estimates distribution, and the upper-endpoint is approximated by the 97.5^{th} percentile.

(d) Suppose estimate that the proportion of smokers in Data 100 is 3.5%, and the 95% confidence interval found using the bootstrap is $(2.3\%, 4.7\%)$. How do we interpret this interval?

> **Solution:** If this bootstrap procedure were to be repeated multiple times for different samples of students from the Data 100 student body, we would expect each sample's 95% bootstrap confidence interval (which will be different for each sample) to contain the true proportion of smokers 95% of the time.

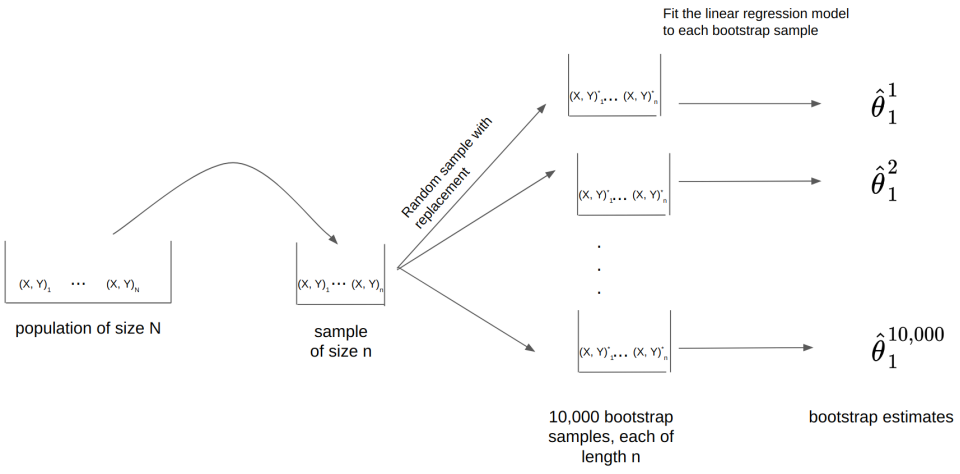
4. We can use the bootstrap to carry out inference on the slope of a simple linear regression. Recall that a simple linear regression model is defined as follows

$$E[Y|X] = \theta_0 + \theta_1 X$$

where (X, Y) are continuous random variables. Y is the response, X is the feature. Using the data to estimate the intercept and the slope, we arrive at the following equation:

$$f_{\hat{\theta}}(x) = \hat{\theta}_0 + \hat{\theta}_1 x_1$$

(a) Using a box model, describe the process of computing the 95% confidence interval of $\hat{\theta}_1$.

> **Solution:**
>
> 
>

Given the bootstrap estimates, we can estimate the sampling distribution of θ_1. The 95% confidence interval is approximated by the 2.5^{th} percentile of the bootstrap estimates distribution, and the upper-endpoint is approximated by the 97.5^{th} percentile.
Depending on the size of N and n, the original sample can be a simple random sample with or without replacement. See the solution of 2a for more details.

(b) Now that we have some intuition for how the bootstrap works in a simple linear regression, let's think about how we might implement this for a multivariate linear regression. Suppose we wish to fit a model of the following form:

$$f_{\hat{\theta}}(x) = \hat{\theta}_0 + \hat{\theta}_1 x_1 + \cdots + \hat{\theta}_p x_p$$

and we would like to generate confidence intervals around our estimates of θ. Outline in pseudo-code a non-parametric bootstrap based approach to estimate the 95% confidence interval around each θ_i. Assume there are n data points.

Solution:
$theta\_hats$ = []

For $i = 1$ to $num\_replicates$:
$bootstrap\_sample = SampleWithReplacement(data, n)$
$theta\_hat = LinearModel.fit(bootstrap\_sample).coefficients$
$theta\_hats.append(theta\_hat)$

$.025\_CI = percentile(theta\_hats, .025)$
$.975\_CI = percentile(theta\_hats, .975)$