In-Class Computing Project 3

Math 253: Statistical Computing & Machine Learning

Working with indices: Training vs testing data

In today's programming project, you are going to build some models of the attractiveness of colleges to accepted applicants. You'll use some data from the `ISLR` package; the package that goes along with the book. You may need to install this package.[1]

Task 1

Load the `ISLR` package into R. Then, to gain access to the `College` data in `ISLR`, use this command:

```
data(College, package = "ISLR")
```

Task 2

Create a variable called `Yield` within the `College` data table. The yield is defined by college admissions officers to be the number of students enrolled divided by the number of students accepted. (Applications are another matter altogether.)

Task 3

Divide `College` into two data frames, one for training and testing.

- Create an object `all_indices` with the integers 1, 2, 3, . . . , n, where n is the number of rows in `College`
- Create an object `train_indices` with 200 random indices between 1 and the number of cases in `College`. Hint: `sample()`
- Create another object `test_indices` with all the remaining cases from `College`. Hint: `setdiff()`.
- Create a data frame object `Train_data` with the rows from `College` corresponding to `train_indices`. Hint: `College[ , ]`
- Create a data frame object `Test_data` with the rows from `College` corresponding to `test_indices`.

Task 4

Using `Train_data`, construct a model of `Yield` as a function of `Top10perc`, `Outstate` (tuition), and `Expend`. Arrange things so that the name of the object holding the model is `Yield_mod1`.

```
Yield_mod1 <- lm(Yield ~ Top10perc + Outstate +
    Expend, data = Train_data)
```

[1] If so, you should do it with the RStudio "Packages" tab. **Do not** put the installation command in your `.Rmd` file.

Task 5

- Create an object `Y_train` which holds just the `Yield` from the training data. Hint: `Train_data$Yield`
- Create an object `fhat_train` which is the output of the model for the inputs in the training data. Hint: `predict(Yield_mod1, newdata = Train_data )`
- Create an object `MSE_train` that holds the mean square error for the training data. The value contained in this object will be a single number.

Task 6

Repeat Task 5, but for the testing data. Everywhere `Train` or `train` appears in step 5, use `Test` or `test` in this step. You'll end up with an object called `MSE_test`.

You might be interested to look at the ratio of `MSE_train` to `MSE_test`. This will be random, but should be close to 1.