Quiz #3 – Solutions

1. *Occam's razor* prefers a simpler model to a more complicated model if they are equally powerful statistically. How does this idea relate to the use of validation samples in machine learning?

a) Validation samples can be used to check the assumptions of data mining models.

b) Complex models may overfit training data, so validation data allow an unbiased estimate of true predictive accuracy in the population.

c) Machine learning models can be estimated from validation models to compare simpler models to more complicated ones.

d) It is more complicated to use the entire sample dataset than to split the dataset into training and validation samples.

e) Two models are equally powerful statistically if the error rate for one is within one standard deviation of the error rate for the other.

Solution: B

Answer b is correct. Answer a is incorrect because model complexity has nothing to do with checking assumptions. Answer c is incorrect because we don't estimate models using validation data. Answer d is incorrect because model complexity is unrelated to the complexity of using the entire dataset versus partitioning the data. Answer e may be one way to define "statistical equivalence" but it doesn't relate directly to the use of validation samples.

2. Consider a machine learning analysis of a large dataset using the "validation set" approach that randomly divides the data set into a training set and a validation set. Put the following steps of the analysis into the most appropriate order.

a) Measure practical outcomes from applying the final model.
b) Randomly partition dataset into training and validation sets.
c) Select final model using validation data.
d) Transform variables, fit models using training data, assess models using validation data (repeat as necessary).
e) Collect data

Solution: E:1, B:2, D:3, C:4, A:5

The correct order is: Collect data; Randomly partition dataset into training and validation sets; Transform variables, fit models using training data, assess models using validation data (repeat as necessary); Select final model using validation data; Measure practical outcomes from applying the final model. This question brings up the question of when do we transform variables in machine learning? Ideally, transformations are suggested either by background knowledge before any data analysis takes place or during exploratory data analysis before we start fitting

any models. However, model building is often an iterative process whereby we might want to try out some transformations to see if they result in better models. Then, transforming variables might well take place after we've fit some models using training data and perhaps after we've assessed some models using validation data. This is why the phrase "repeat as necessary" appears in the step, "Transform variables, fit models using training data, assess models using validation data."

3. Match the following terms to the descriptions below:

a) Validation set approach
b) Leave-one-out cross-validation
c) 5 or 10-fold cross-validation

- Computationally expensive method that provides estimates of test error with low bias and high variance.
- Computationally intermediate method that provides estimates of test error with intermediate bias and low variance.
- Computationally cheap method that may overestimate test error and estimates can be highly variable.

Solution:

The definitions matched up as follows: Leave-one-out cross-validation (computationally expensive method that provides estimates of test error with low bias and high variance); 5 or 10-fold cross-validation (computationally intermediate method that provides estimates of test error with intermediate bias and low variance); Validation set approach (computationally cheap method that may overestimate test error and estimates can be highly variable).

4. What are reasons why test error could be LESS than training error? (Check all that apply.)

a) By chance, the test set has easier cases than the training set.
b) The model is highly complex, so training error systematically overestimates test error.
c) The model is not very complex, so training error systematically overestimates test error.

Solution: A

Training error usually underestimates test error when the model is very complex (compared to the training set size), and is a pretty good estimate when the model is not very complex. However, it's always possible we just get too few hard-to-predict points in the test set, or too many in the training set.

5. Suppose we want to use cross-validation to estimate the error of the following procedure:

- Step 1: Find the *k* variables most correlated with y.

- Step 2: Fit a linear regression using those variables as predictors.

We will estimate the error for each k from 1 to p, and then choose the best k.

True or false: a correct cross-validation procedure will possibly choose a different set of k variables for every fold?

Solution: True

True: we need to replicate our entire procedure for each training/validation split. That means the decision about which k variables are the best must be made on the basis of the training set alone. In general, different training sets will disagree on which are the best k variables.

6. One way of carrying out the bootstrap is to average equally over all possible bootstrap samples from the original data set (where two bootstrap data sets are different if they have the same data points but in different order). Unlike the usual implementation of the bootstrap, this method has the advantage of not introducing extra noise due to resampling randomly.

To carry out this implementation on a data set with $n = 10$ data points, how many bootstrap data sets would we need to average over?

Solution: n^n = 10^10 = 10000000000

Completely removing the bootstrap resampling noise is usually not worth incurring the extreme computational cost. If B is large, but still less than n^n, random resampling gives a good Monte Carlo estimate of the idealized bootstrap estimate for all n^n data sets.

7. If we have $n = 10$ data points, what is the probability that a given data point does not appear in a bootstrap sample (round to three decimal places)?

Solution: $(1 - \frac{1}{n})^n = 0.349$

To construct a bootstrap sample, we repeatedly draw a single data point from a sample of size n, n times. Any given data point has a 1-1/n chance of not being selected in each draw. Hence, the chance of not being selected in any of the n draws is (1-1/n)^n

8. When we fit a model to data, which is typically larger: test error or training error?

Solution: Test Error

Training error almost always underestimates test error, sometimes dramatically.