The Wrong and Right Way to Do Cross-validation

From ESL, Chapter 7.10.2

Consider a classification problem with a large number of predictors, as may arise, for example, in genomic or proteomic applications. A typical strategy for analysis might be as follows:

1. Screen the predictors: find a subset of "good" predictors that show fairly strong (univariate) correlation with the class labels
2. Using just this subset of predictors, build a multivariate classifier.
3. Use cross-validation to estimate the unknown tuning parameters and to estimate the prediction error of the final model.

Is this a correct application of cross-validation?

Simulation Consider a scenario with $n = 50$ samples in two equal-sized classes, and $p = 5000$ quantitative predictors (standard Gaussian) that are independent of the class labels. The true (test) error rate of any classifier is 50%.

1. choose the 100 predictors having highest correlation with the class labels
2. use a 1-nearest neighbor classifier, based on just these 100 *selected* predictors.
3. Use K-fold CV to estimate the test error of the final model

Over 50 simulations from this setting, the average CV error rate was 3%. This is far lower than the true error rate of 50%. What has happened?

Exercize Perform the simulation with $K = 5$

```
## 5-fold CV: 0.02
```

The problem is that the predictors have an unfair advantage, as they were chosen in step 1. on the basis of all of the observations. Leaving observations out after the variables have been selected does not correctly mimic the application of the classifier to a completely independent test set, since these predictors *have already seen* the left out observations.

Here is the correct way to carry out cross-validation in this example:

1. Divide the observations into K cross-validation folds at random.
2. For each fold $k = 1, \ldots, K$
    a. Find the best 100 predictors that have the largest (in absolute value) correlation with the class labels, using all of the observations except those in fold k.
    b. Using just this subset of predictors,fit a 1-nearest neighbor classifier, using all of the observations except those in fold k
    c. Use the classifier to predict the class labels for the observations in fold k

The error estimates from step 2.c are then accumulated over all K folds, to produce the cross-validation estimate of prediction error.

Exercize Try with $K = 5$

```
## 5-fold CV: 0.58
```

In general, with a multistep modeling procedure, cross-validation must be applied to the entire sequence of modeling steps. In particular, observations must be "left out" before any selection or filtering steps are applied.