Ch. 5. Resampling Methods

Sampling methods involve repeatedly drawing samples from a training set and refitting a model of interest on each sample in order to obtain additional information about the fitted model.
For example, in order to estimate the variability of a linear regression fit, we can repeatedly draw different samples from the training data, fit a linear regression to each new sample and then examine the extent to which the resulting fits differ. Such an approach may allow us to obtain information that would not be available from fitting the model only once using the original training sample.

Resampling methods can be computationally intensive, because they involve fitting the same statistical method multiple times using different subsets of the training data. However, due to recent advances in computing power, the computational requirements of resampling methods generally are not prohibitive.
Main methods: cross-validation and bootstrap.

Cross-validation can be used to estimate the test error associated with a given statistical learning method in order to evaluate its performance, or to select the appropriate level of flexibility.

The process of evaluating a model's performance is known as **model assessment**, whereas the process of selecting the proper level of flexibility for a model is knowns as **model selection**.

The **bootstrap** is used in several contexts, most commonly to provide a measure of accuracy of a parameter estimate or of a given statistical method.

Cross-Validation

Usually test error ≠ training error.

Test error → average error that results from using a statistical learning method to predict the response on a new observation, a measurement that was not used in training the method. **Easy to compute if a designated test set is available. Not always the case → estimation.**

Training error → average error that results from using a statistical learning method to predict the response on the observations used in its training. **Straightforward to obtain.**

We are interested in test error obviously

Validation Set Approach

Randomly split data set in half into two sets: training set and validation set.

Train model using training set and evaluate the prediction on the validation set.

Problems
Result is dependent on the random split → depending on the observations that randomly are selected for training or validation we will have even very different assessments of our model.

- Validation estimate of the test error rate can be highly variable, depending on precisely which observations are included in the training set and which obs are included in the validation set.
- Only a subset of the observations - those included in the training set - are used to fit the model. Since statistical methods tend to perform worse when trained on fewer obs, this suggests that the validation set error rate may tend to overestimate the test error rate for the model fit on the entire data set.

Leave-One-Out-Cross-Validation

LOOCV is closely related to the validation set approach, but it attempts to address that method's drawbacks.

LOOCV splits the set of obs into two parts. Insteads of creating two subsets of comparable size, a single observation (x_1, y_1) is used for the validation set, and the remaining observations make up the training set. The statistical learning method is fit on the n - 1 training observations, and a prediction $\hat{y}_1$ is made for the excluded observation using its predictor value x_1.
Since (x_1, y_1) was not used in the fitting process, $MSE_1 = (y_1 - \hat{y}_1)^2$ provides an approximately unbiased estimate for the test error.
But even though MSE_1 is unbiased for the test error, it is a poor estimate because it is highly variable, since it is based upon a single observation (x_1, y_1).
We then repeat the procedure n times by selecting a different observation for the validation data and training the statistical learning procedure on the n - 1 remaining observations , and computing a new value for MSE_i each time.
The LOOCV estimate for the test MSE is the average of these n test error estimates.

Advantages of LOOCV over validation set approach:

- Far less bias, we repeatedly fit the model on almost the whole dataset vs fitting the model on half of the observations in the validation set appr. → LOOCV tends not to overestimate the test error rate as much as the valid. set appr. does

- In contrast with the valid. set appr. that will yield different approach for different splits, LOOCV will always yield the same results because there is no randomness in the training/validation set splits.

LOOCV has the potential to be expensive to implement since the model has to be fit n times. Shortcut for least squares linear or polynomial regression:

$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - h_i}\right)^2$$

Where h_i is the leverage of the observation (it lies between 1/n and 1, and reflects the amount that an observation influences its own fit). The residuals for high-leverage points are inflated in this formula by exactly the right amount for this equality to hold.

k-Fold Cross-Validation

An alternative to LOOCV is k-fold CV. This approach involves randomly dividing the set of observations into k groups, or folds, of approximately equal size. The first fold is treated as a validation set, and the method is fit on the remaining k - 1 folds. The mean squared error, MSE_1 is then computed on the observations in the held-out fold. This procedure is repeated k times; each time , a different group of observations is treated as a validation set. This process results in k estimates of the test error. The k-fold CV estimate is computed by averaging these values.

LOOCV is a special case of k-fold CV in which k is set to equal n.
In practice, one typically performs k-fold CV using k = 5 or k = 10. Advantage: mostly computational. There's also a non-computational advantage to performing 5-fold or 10-fold CV, which involves the bias-variance trade-off.

When we perform CV, our goal might be to determine how well a given statistical learning procedure can be expected to perform on independent data; in this case the actual estimate of the test MSE is of interest.

Other times we are interested only in the location of the minimum point in the estimated test MSE curve. This is because we might be performing cross-validation on a number of statistical learning methods, or on a single method using different levels of flexibility, in order to identify the method that results in the lowest estimated test MSE error. For this purpose, the location of the minimum point (method or hyperparameter combination) in the estimated test MSE curve is important, but the actual value of the estimated test MSE is not.

Bias-Variance Trade-Off for k-Fold Cross-Validation

An important advantage of k-fold CV is that it often gives more accurate estimates of the test error rate than does LOOCV. This had to do with a bias-variance trade-off.

The validation set approach can lead to overestimates of the test error rate, since in this approach the training set used to fit the statistical learning method contains only half the observations of the entire data set. Using this logic, it is not hard to see that LOOCV will give approximately unbiased estimates of the test error, since each training set contains n - 1 observations, which is almost as many as the number of observations in the full data set. And performing k-fold CV for, say k = 5 or k = 10 will lead to an intermediate level of bias, since each training set contains (k - 1)n/k observations -- fewer than in the LOOCV approach, but substantially more than in the validation set approach. Therefore, from the perspective of bias reduction, it is clear that LOOCV is to be preferred to k-fold CV.

However, we know that bias is not the only source for concern in an estimating procedure; we must also consider the procedure's variance. It turns out that LOOCV has higher variance than does k-fold CV with k < n. When we perform LOOCV, we are in effect averaging the outputs of n fitted models, each of which is trained on an almost identical set of observations; therefore, these outputs are highly (positively) correlated with each other. In contrast, when we perform k-fold CV with k < n, we are averaging the outputs of k fitted models that are somewhat less correlated with each other, since the overlap between the training sets in each model is smaller. Since the mean of many highly correlated quantities has higher variance than the mean of many quantities that are not as highly correlated, the test error estimate resulting from LOOCV tends to have higher variance than does the test error estimate resulting from k-fold CV.

To summarize, there is a bias-variance trade-off associated with the choice of k in k-fold cross-validation. Typically, given these considerations, one performs k-fold cross-validation using k = 5 or k = 10, as these values have been shown empirically to yield test error rate estimates that suffer neither from excessively high bias nor from very high variance.

Cross-Validation on Classification Problems

pg(184-186)
It works just as described for regression problems but instead of using MSE to quantify test error, we instead use the number of misclassified observations.

The Bootstrap

The bootstrap is a widely applicable and extremely powerful statistical tool that can be used to quantify the uncertainty associated with a given estimator or statistical learning method.

As a simple example, the bootstrap can be used to estimate the standard errors of the coefficients from a linear regression fit. In this specific case, this is not very useful since standard statistical softwares output such standard errors automatically. However, the power of the bootstrap lies in the fact that it can be easily applied to a wide range of statistical

learning methods, including some for which a measure of variability is otherwise difficult to obtain and is not automatically output by statistical software.

For simulated data, an approach to estimate the variance of an estimator would be to simulate the sample sets a large number of times, such as 100 or 1000. We would obtain 100 or 1000 values for the estimator (one estimate for each simulated sample set) and it would be straightforward to compute the standard deviation of the estimates.
With real data we don't have the possibility to simulate sample sets. Instead, with the bootstrap approach, we use a computer to emulate the process of obtaining new sample sets, so that we can estimate the variability of the estimator without generating additional samples. Rather than repeatedly obtaining independent data sets from the population, we instead obtain distinct data sets by repeatedly sampling observations from the original data set.
We randomly select n(total # of observations) observations from the data set in order to produce a bootstrap data set. The sampling is performed with replacement, which means that the same observation can occur more than once in the bootstrap data set.
We repeat this procedure many times, usually 100 or 1000 times and get our estimate of the standard deviation of our estimator.