4.3 Preventing Overfitting in Supervised Machine Learning

We have seen that overfitting impairs generalization, but overfitting potential is endemic to the supervised machine learning process due to the presence of noise. So, how do data scientists combat this risk? Two common methods are used to reduce overfitting: (1) preventing the algorithm from getting too complex during selection and training, which requires estimating an overfitting penalty, and (2) proper data sampling achieved by using **cross-validation**, a technique for estimating out-of-sample error directly by determining the error in validation samples.

The first strategy comes from Occam's razor, the problem-solving principle that the simplest solution tends to be the correct one. In supervised machine learning, it means limiting the number of features and penalizing algorithms that are too complex or too flexible by constraining them to include only parameters that reduce out-of-sample error.

The second strategy comes from the principle of avoiding sampling bias. But sampling bias can creep into machine learning in many ways. The challenge is having a large enough dataset to make both training and testing possible on representative samples. An unrepresentative sample or reducing the training sample size too much could obscure its true patterns, thereby increasing bias. In supervised machine learning, the technique for reducing sampling bias is through careful partitioning of the dataset into three groups: (1) training sample, the set of labeled training data where the target variable (Y) is known;(2) validation sample, the set of data used for making structural choices on the degree of model complexity, comparing various solutions, and tuning the selected model, thereby validating the model; and (3) test sample, the set of data held aside for testing to confirm the model's predictive or classifying power. The goal, of course, is to deploy the tested model on fresh data from the same domain.

To mitigate the problem of such **holdout samples** (i.e., data samples not used to train the model) reducing the training set size too much, modelers use special cross-validation techniques. One such technique is ***k*-fold cross-validation**, in which the data (excluding test sample and fresh data) are shuffled randomly and then are divided into k equal sub-samples, with $k - 1$ samples used as training samples and one sample, the kth, used as a validation sample. Note that k is typically set at 5 or 10. This process is then repeated k times, which helps minimize both bias and variance by insuring that each data point is used in the training set $k - 1$ times and in the validation set once. The average of the k validation errors (mean E_{val}) is then taken as a reasonable estimate of the model's out-of-sample error (E_{out}). A limitation of k-fold cross-validation is that it cannot be used with time-series data, where only the most recent data can reasonably be used for model validation.

In sum, mitigating overfitting risk by avoiding excessive out-of-sample error is critical to creating a supervised machine learning model that generalizes well to fresh datasets drawn from the same distribution. The main techniques used to mitigate overfitting risk in model construction are complexity reduction (or regularization) and cross-validation.