Chapter 4. Over-Fitting and Model Tuning

- The aim of this chapter is to explain and illustrate key principles of laying a foundation onto which trustworthy models can be built and subsequently used for prediction.

- We will describe strategies that enable us to have confidence that the model we build will predict new samples with a similar degree of accuracy on the set of data for which the model was evaluated.

  Without this confidence, the model's predictions are *useless*.

- Almost all predictive modeling techniques have tuning parameters that enable the model to flex to find the structure in the data.

  We must use the existing data to identify settings for the model's parameters that yield the best and most realistic predictive performance (known as model tuning).

4.1 The Problem of Over-Fitting

Over-fitting models usually have poor accuracy when predicting a new sample.

Estimating the utility of a model by re-predicting the training set is referred.

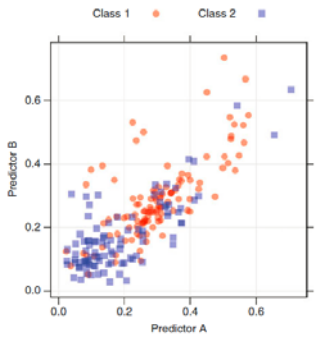

Fig. 4.1: An example of classification data that is used throughout the chapter

Consider a classification example in Fig. 4.1.

- There are two independent variables.
- 208 samples designated as "Class 1" or "Class 2."
- There are 111 samples in Class 1 and 97 in Class 2.
- There is a significant overlap between the classes.

One objective is to develop a model to classify new samples.

Figure 4.2 shows two distinct classification models.

- The left-hand panel ("Model #1") shows a boundary that is complex and attempts to encircle every possible data point.

  The pattern in this panel is not likely to generalize to new data.

- The right-hand panel shows an alternative model fit where the boundary is fairly smooth and does not overextend itself to correctly classify every data point in the training set.

- The estimated error rate for model #1 would be overly optimistic.

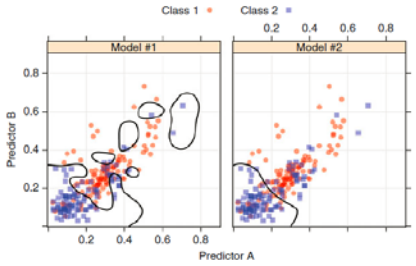

Fig. 4.2: An example of a training set with two classes and two predictors. The panels show two different classification models and their associated class boundaries

4.2 Model Tuning

Many models have important parameters which cannot be directly estimated from the data.

- For example, in the *K*-nearest neighbor classification model, K is the parameter.
- Number of PCs retained in the model.

An illustration of a 5-nearest neighbor model is shown in Fig. 4.3.

The question remains as to how many neighbors should be used.

- A choice of too few neighbors may over-fit the individual points of the training set
- Too many neighbors may not be sensitive enough to yield reasonable performance.

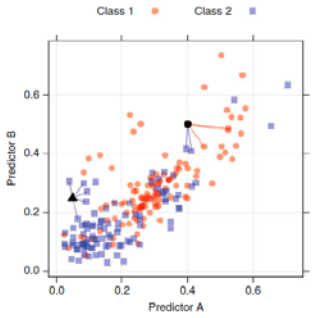

Fig. 4.3: The K-nearest neighbor classification model. Two new points, symbolized by *filled triangle* and *solid dot*, are predicted using the training set

This type of model parameter is referred to as a *tuning parameter* because there is no analytical formula available to calculate an appropriate value.

Many of tuning parameters control the complexity of the model, poor choices for the values can result in over-fitting.

There are different approaches to searching for the best parameters.

- A **general approach** is to define a set of candidate values, generate reliable estimates of model utility across the candidates' values, then choose the optimal settings.

A flowchart of this process is shown in Fig. 4.4.

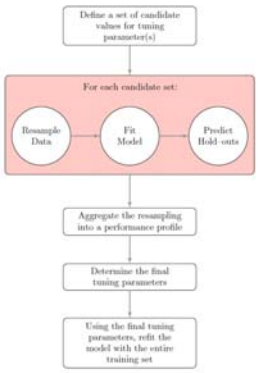

Fig. 4.4: A schematic of the parameter tuning process. An example of a candidate set of tuning parameter values for K-nearest neighbors might be odd numbers between 1 and 9. For each of these values, the data would be resampled multiple times to assess model performance for each value

4.3 Data Splitting

- We will discuss the heart of the process to find optimal tuning parameters: data splitting.
- Given a fixed amount of data, the modeler must decide how to "spend" their data.
- The "training" data set is the general term for the samples used to create the model.
- The "test" or "validation" data set is used to qualify performance.
- When the number of samples is not large, a strong case can be made that a test set should be avoided because every sample may be needed for model building.
  - ➢ Resampling methods, such as cross-validation, can be used to produce appropriate estimates of model performance using the training set.
- If a test set is deemed necessary, there are several methods for splitting the samples.
  - ➢ Nonrandom approaches
    - ➢ Random sampling methods
    - ➢ Stratified random sampling applies random sampling within subgroups (such as the classes).
      - ✓ The data can be split on the basis of the predictor values.
      - ✓ Willett (1999) and Clark (1997) propose data splitting based on *maximum dissimilarity sampling*.

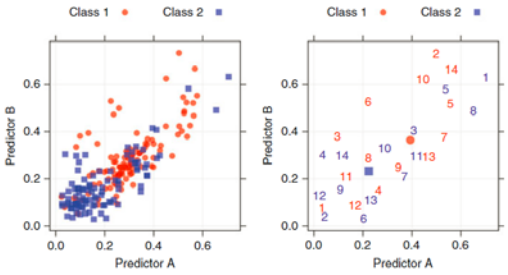

Fig. 4.5: An example of maximum dissimilarity sampling to create a test set. After choosing an initial sample within a class, 14 more samples were added

4.4 Resampling Techniques

Resampling techniques are used to estimate model performance:

- ➢ A subset of samples are used to fit a model
- ➢ The remaining samples are used to estimate the efficacy of the model.
- ➢ Repeat this process multiple times and the results are aggregated and summarized.

k-Fold Cross-Validation

- ➢ The samples are randomly partitioned into *k* sets of roughly equal size.
- ➢ A model is fit using all samples except the first subset (called the first *fold*).
- ➢ The held-out samples are predicted by this model and used to estimate performance measures.
- ➢ The first subset is returned to the training set and procedure repeats with the second subset held out, and so on.
- ➢ The *k* resampled estimates of performance are summarized (usually with the mean and standard error) and used to understand the relationship between the tuning parameter(s) and model utility.

The choice of *k* is usually 5 or 10, but there is no formal rule.

Example: a cross-validation process with *k* = 3.

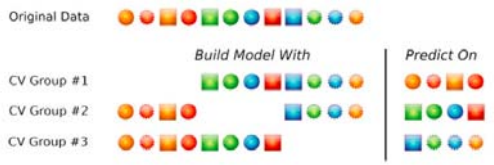

Fig. 4.6: A schematic of threefold cross-validation. Twelve training set samples are represented as symbols and are allocated to three groups. These groups are left out in turn as models are fit. Performance estimates, such as the error rate or R^2 are calculated from each set of held-out samples. The average of the three performance estimates would be the cross-validation estimate of model performance. In practice, the number of samples in the held-out subsets can vary but are roughly equal size

Leave-one-out cross-validation (LOOCV)

- Is the special case of the k-Fold Cross-Validation, where *k* is the number of samples.
- From a practical viewpoint, larger values of *k* are more computationally burdensome.

Molinaro (2005) found that leave-one-out and *k* =10-fold cross-validation yielded similar results, indicating that *k* = 10 is more attractive from the perspective of computational efficiency.

Also, small values of *k*, say 2 or 3, have high bias but are very computationally efficient.

Generalized Cross-Validation

- For linear regression models, there is a formula for approximating the leave-one-out error rate.
- The generalized cross-validation (GCV) statistic (Golub et al. 1979) for the *ith* training set outcome is

$$\text{GCV} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - df/n}\right)^2,$$

The Bootstrap

A bootstrap sample is a random sample of the data taken *with replacement*.

The samples not selected are usually referred to as the "out-of-bag" samples.

For a given iteration of bootstrap resampling, a model is built on the selected samples and is used to predict the out-of-bag samples.

4.5 Case Study: Credit Scoring

- Goals: create a model to predict the probability that applicants have good credit.
- This information can be used to quantify the risk to the lender.

Data:

- German credit data set contains 1,000 samples labeled with good and bad credit.
- 70% were rated as having good credit.
- Data were collected related to credit history, employment, account status, and so on.
- There were 41 predictors used to model the credit status of an individual.

4.6 Choosing Final Tuning Parameters

- Once model performance has been quantified across sets of tuning parameters, there are several philosophies on how to choose the final settings.
- The simplest approach is to pick the settings associated with the numerically best performance estimates.

For the credit scoring example, a nonlinear support vector machine model was evaluated over cost values ranging from 2^{-2} to 2^{7}.

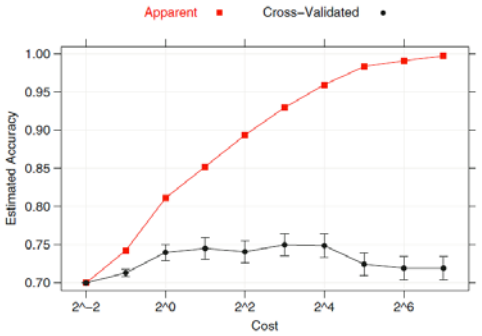

Fig. 4.9: The performance profile of a radial basis function support vector machine for the credit scoring example over different values of the cost parameter. The *vertical lines* indicate ± two-standard errors of the accuracy

Each model was evaluated using five repeats of 10-fold cross-validation.

Figure 4.9 and Table 4.1 show the accuracy profile across the candidate values of the cost parameter.

The shows an increase in accuracy until the cost value is one. Models with cost values between 1 and 16 are relatively constant; after which, the accuracy decreases (likely due to over-fitting).

Table 4.1: Repeated cross-validation accuracy results for the support vector machine model

	Resampled accuracy (%)		
Cost	Mean	Std. error	% Tolerance
0.25	70.0	0.0	−6.67
0.50	71.3	0.2	−4.90
1.00	74.0	0.5	−1.33
2.00	74.5	0.7	−0.63
4.00	74.1	0.7	−1.20
8.00	75.0	0.7	0.00
16.00	74.9	0.8	−0.13
32.00	72.5	0.7	−3.40
64.00	72.0	0.8	−4.07
128.00	72.0	0.8	−4.07

The one-standard error rule would select the simplest model with accuracy no less than 74.3 % (75 %–0.7 %). This corresponds to a cost value of 2. The "pick-the-best" solution is shown in bold

The numerically optimal value of the cost parameter is 8, with a corresponding accuracy rate of 75%.

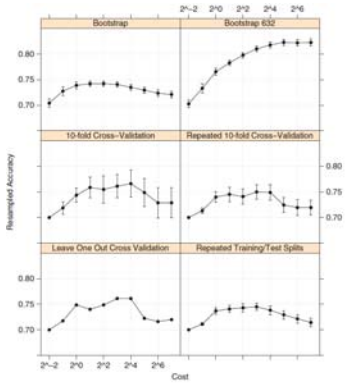

Fig. 4.10: The performance profile of nonlinear support vector machine over different values of the cost parameter for the credit scoring example using several different resampling procedures. The *vertical lines* indicate ± two-standard errors of the accuracy

4.7 Data Splitting Recommendations

There is a strong technical case to be made against a single, independent test set:

- A test set is a single evaluation of the model and has limited ability to characterize the uncertainty in the results.
- Proportionally large test sets divide the data in a way that increases bias in the performance estimates.
- With small sample sizes:
  - The model may need every possible data point to adequately determine model values.
  - The uncertainty of the test set can be considerably large to the point where different test sets may produce very different results.
- Resampling methods can produce reasonable predictions of how well the model will perform on future samples.

No resampling method is uniformly better than another:

- If the samples size is small, we recommend repeated 10-fold cross.
- If the goal is to choose between models, use one of the bootstrap procedures since these have very low variance.
- For large sample sizes, a simple 10-fold cross-validation should provide acceptable variance, low bias, and is relatively quick to compute.

4.8 Choosing Between Models

Once the settings for the tuning parameters have been determined for each model, the question remains: how do we choose between multiple models?

Again, this largely depends on the characteristics of the data and the type of questions being answered.

We suggest the following scheme for finalizing the type of model:

1. Start with several models that are the least interpretable and most flexible, such as boosted trees or support vector machines.
2. Investigate simpler models that are less opaque, such as multivariate adaptive regression splines (MARS), partial least squares, generalized additive models, or naive Bayes models.
3. Consider using the simplest model that reasonably approximates the performance of the more complex methods.

In many cases, a range of models will be equivalent in terms of performance so the practitioner can weight the benefits of different methodologies (e.g., computational complexity, easy of prediction, interpretability).

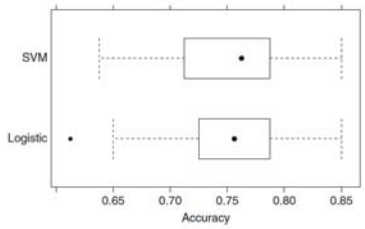

Fig. 4.11: A comparison of the cross-validated accuracy estimates from a support vector machine model and a logistic regression model for the credit scoring data described in Sect. 4.5