12

Cross-Validation and C_p Estimates of Prediction Error

Prediction has become a major branch of twenty-first-century commerce. Questions of prediction arise naturally: how credit-worthy is a loan applicant? Is a new email message `spam`? How healthy is the kidney of a potential donor? Two problems present themselves: how to construct an effective prediction rule, and how to estimate the accuracy of its predictions. In the language of Chapter 1, the first problem is more algorithmic, the second more inferential. Chapters 16–19, on *machine learning*, concern prediction rule construction. Here we will focus on the second question: having chosen a particular rule, how do we estimate its predictive accuracy?

Two quite distinct approaches to prediction error assessment developed in the 1970s. The first, depending on the classical technique of cross-validation, was fully general and nonparametric. A narrower (but more efficient) model-based approach was the second, emerging in the form of Mallows' C_p estimate and the Akaike information criterion (AIC). Both theories will be discussed here, beginning with cross-validation, after a brief overview of prediction rules.

12.1 Prediction Rules

Prediction problems typically begin with a *training set* $\boldsymbol{d}$ consisting of N pairs (x_i, y_i),

$$\boldsymbol{d} = \{(x_i, y_i),\ i = 1, 2, \ldots, N\}, \tag{12.1}$$

where x_i is a vector of p *predictors* and y_i a real-valued *response*. On the basis of the training set, a *prediction rule* $r_{\boldsymbol{d}}(x)$ is constructed such that a prediction $\hat{y}$ is produced for any point x in the predictor's sample space $\mathcal{X}$,

$$\hat{y} = r_{\boldsymbol{d}}(x) \qquad \text{for } x \in \mathcal{X}. \tag{12.2}$$

The inferential task is to assess the accuracy of the rule's predictions. (In practice there are usually several competing rules under consideration and the main question is determining which is best.)

In the **spam** data of Section 8.1, x_i comprised $p = 57$ keyword counts, while y_i (8.18) indicated whether or not message i was **spam**. The rule $r_{\boldsymbol{d}}(x)$ in Table 8.3 was an MLE logistic regression fit. Given a new message's count vector, say x_0, $r_{\boldsymbol{d}}(x_0)$ provided an estimated probability $\hat{\pi}_0$ of it being **spam**, which could be converted into a prediction $\hat{y}_0$ according to

$$\hat{y}_0 = \begin{cases} 1 & \text{if } \hat{\pi}_0 \geq 0.5 \\ 0 & \text{if } \hat{\pi}_0 < 0.5. \end{cases} \tag{12.3}$$

The diabetes data of Table 7.2, Section 7.3, involved the $p = 10$ predictors $x = (\mathbf{age}, \mathbf{sex}, \ldots, \mathbf{glu})$, obtained at baseline, and a response y measuring disease progression one year later. Given a new patient's baseline measurements x_0, we would like to predict his or her progression y_0. Table 7.3 suggests two possible prediction rules, ordinary least squares and ridge regression using ridge parameter $\lambda = 0.1$, either of which will produce a prediction $\hat{y}_0$. In this case we might assess prediction error in terms of squared error, $(y_0 - \hat{y}_0)^2$.

In both of these examples, $r_{\boldsymbol{d}}(x)$ was a regression estimator suggested by a probability model. One of the charms of prediction is that the rule $r_{\boldsymbol{d}}(x)$ need not be based on an explicit model. Regression trees, as pictured in Figure 8.7, are widely used[1] prediction algorithms that do not require model specifications. Prediction, perhaps because of its model-free nature, is an area where algorithmic developments have run far ahead of their inferential justification.

Quantifying the prediction error of a rule $r_{\boldsymbol{d}}(x)$ requires specification of the discrepancy $D(y, \hat{y})$ between a prediction $\hat{y}$ and the actual response y. The two most common choices are *squared error*

$$D(y, \hat{y}) = (y - \hat{y})^2, \tag{12.4}$$

and *classification error*

$$D(y, \hat{y}) = \begin{cases} 1 & \text{if } y \neq \hat{y} \\ 0 & \text{if } y = \hat{y}, \end{cases} \tag{12.5}$$

when, as with the **spam** data, the response y is dichotomous. (Prediction of a dichotomous response is often called "classification.")

[1] *Random forests*, one of the most popular machine learning prediction algorithms, is an elaboration of regression trees. See Chapter 17.

For the purpose of error estimation, we suppose that the pairs (x_i, y_i) in the training set $\boldsymbol{d}$ of (12.1) have been obtained by random sampling from some probability distribution F on $(p+1)$-dimensional space $\mathcal{R}^{p+1}$,

$$(x_i, y_i) \overset{\text{iid}}{\sim} F \qquad \text{for } i = 1, 2, \ldots, N. \tag{12.6}$$

The *true error rate* $\text{Err}_{\boldsymbol{d}}$ of rule $r_{\boldsymbol{d}}(x)$ is the expected discrepancy of $\hat{y}_0 = r_{\boldsymbol{d}}(x_0)$ from y_0 given a new pair (x_0, y_0) drawn from F independently of $\boldsymbol{d}$,

$$\text{Err}_{\boldsymbol{d}} = E_F\{D(y_0, \hat{y}_0)\}; \tag{12.7}$$

$\boldsymbol{d}$ (and $r_{\boldsymbol{d}}(\cdot)$) is held fixed in expectation (12.7), only (x_0, y_0) varying.

Figure 12.1 concerns the *supernova data*, an example we will return to in the next section.[†] Absolute magnitudes y_i have been measured for $N = 39$ relatively nearby Type Ia supernovas, with the data scaled such that

$$y_i \overset{\text{ind}}{\sim} \mathcal{N}(\mu_i, 1), \qquad i = 1, 2, \ldots, 39, \tag{12.8}$$

is a reasonable model. For each supernova, a vector x_i of $p = 10$ spectral energies has been observed,

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{i10}), \qquad i = 1, 2, \ldots, 39. \tag{12.9}$$

Table 12.1 shows (x_i, y_i) for $i = 1, 2, \ldots, 5$. (The frequency measurements have been standardized to have mean 0 and variance 1, while y has been adjusted to have mean 0.)

On the basis of the training set $\boldsymbol{d} = \{(x_i, y_i),\ i = 1, 2, \ldots, 39\}$, we wish to construct a rule $r_{\boldsymbol{d}}(x)$ that, given the frequency vector x_0 for a newly observed Type Ia supernova, accurately predicts[2] its absolute magnitude y_0. To this end, a *lasso* estimate $\tilde{\beta}(\lambda)$ was fit, with y in (7.42) the vector $(y_1, y_2, \ldots, y_{39})$ and x the 39×10 matrix having ith row x_i; λ was selected to minimize a C_p estimate of prediction error, Section 12.3, yielding prediction rule

$$\hat{y}_0 = x_0' \tilde{\beta}(\lambda). \tag{12.10}$$

(So in this case constructing $r_{\boldsymbol{d}}(x)$ itself involves error rate estimation.)

[2] Type Ia supernovas were used as "standard candles" in the discovery of dark energy and the cosmological expansion of the Universe, on the grounds that they have constant absolute magnitude. This isn't exactly true. Our training set is unusual in that the 39 supernovas are close enough to Earth to have y ascertained directly. This allows the construction of a prediction rule based on the frequency vector x, which *is* observable for distant supernovas, leading to improved calibration of the cosmological expansion.

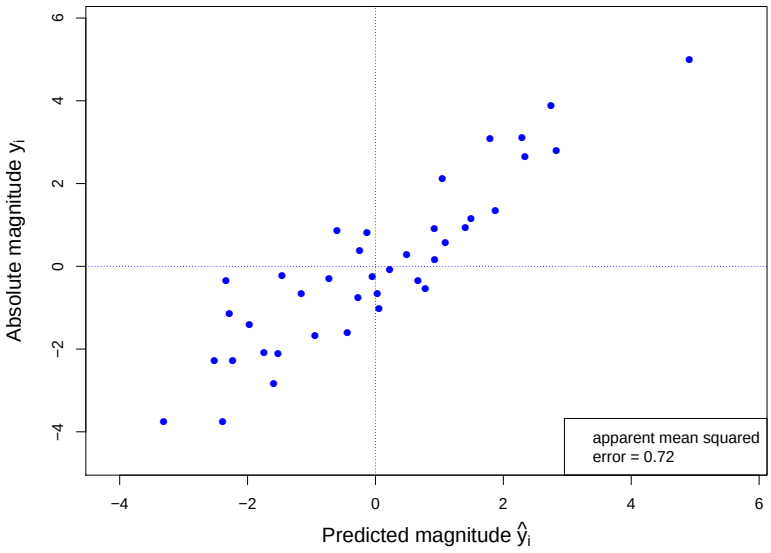

Figure 12.1 The supernova data; observed absolute magnitudes y_i (on log scale) plotted versus predictions $\hat{y}_i$ obtained from lasso rule (12.10), for $N = 39$ nearby Type Ia supernovas. Predictions based on 10 spectral power measurements, 7 of which had nonzero coefficients in $\tilde{\beta}(\lambda)$.

The plotted points in Figure 12.1 are $(\hat{y}_i, y_i)$ for $i = 1, 2, \ldots, N = 39$. These gave *apparent error*

$$\text{err} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 = 0.720. \tag{12.11}$$

Comparing this with $\sum(y_i - \bar{y})^2/N = 3.91$ yields an impressive-looking "R squared" value

$$R^2 = 1 - 0.720/3.91 = 0.816. \tag{12.12}$$

Things aren't really that good (see (12.23)). Cross-validation and C_p methods allow us to correct apparent errors for the fact that $r_{\boldsymbol{d}}(x)$ was chosen to make the predictions $\hat{y}_i$ fit the data y_i.

Prediction and estimation are close cousins but they are not twins. As discussed earlier, prediction is less model-dependent, which partly accounts for the distinctions made in Section 8.4. The prediction criterion Err (12.7)

Table 12.1 *Supernova data; 10 frequency measurements and response variable "absolute magnitude" for the first 5 of $N = 39$ Type Ia supernovas. In terms of notation* (12.1)*, frequency measurements are x and magnitude y.*

	SN1	SN2	SN3	SN4	SN5
x_1	−.84	−1.89	.26	−.08	.41
x_2	−.93	−.46	−.80	1.02	−.81
x_3	.32	2.41	1.14	−.21	−.13
x_4	.18	.77	−.86	−1.12	1.31
x_5	−.68	−.94	.68	−.86	−.65
x_6	−1.27	−1.53	−.35	.72	.30
x_7	.34	.09	−1.04	.62	−.82
x_8	−.43	.26	−1.10	.56	−1.53
x_9	−.02	.18	−1.32	.62	−1.49
x_{10}	−.3	−.54	−1.70	−.49	−1.09
mag	**−.54**	**2.12**	**−.22**	**.95**	**−3.75**

is an expectation over the (x, y) space. This emphasizes good overall performance, without much concern for behavior at individual points x in $\mathcal{X}$.

Shrinkage usually improves prediction. Consider a Bayesian model like that of Section 7.1,

$$\mu_i \sim \mathcal{N}(0, A) \quad \text{and} \quad x_i|\mu_i \sim \mathcal{N}(\mu_i, 1) \qquad \text{for } i = 1, 2, \ldots, N. \tag{12.13}$$

The Bayes shrinkage estimator, which is ideal for estimation,

$$\hat{\mu}_i = Bx_i, \qquad B = A/(A+1), \tag{12.14}$$

is also ideal for prediction. Suppose that in addition to the observations x_i there are independent unobserved replicates, one for each of the N x_i values,

$$y_i \sim \mathcal{N}(\mu_i, 1) \qquad \text{for } i = 1, 2, \ldots, N, \tag{12.15}$$

that we wish to predict. The Bayes predictor

$$\hat{y}_i = Bx_i \tag{12.16}$$

has overall Bayes prediction error

$$E\left\{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2\right\} = B + 1, \tag{12.17}$$

which cannot be improved upon. The MLE rule $\hat{y}_i = x_i$ has Bayes prediction error 2, which is always worse than (12.17).

As far as prediction is concerned it pays to overshrink, as illustrated in Figure 7.1 for the James–Stein version of situation (12.13). This is fine for prediction, but less fine for estimation if we are concerned about extreme cases; see Table 7.4. Prediction rules sacrifice the extremes for the sake of the middle, a particularly effective tactic in dichotomous situations (12.5), where the cost of individual errors is bounded. The most successful machine learning prediction algorithms, discussed in Chapters 16–19, carry out a version of local Bayesian shrinkage in selected regions of $\mathcal{X}$.

12.2 Cross-Validation

Having constructed a prediction rule $r_{\boldsymbol{d}}(x)$ on the basis of training set $\boldsymbol{d}$, we wish to know its prediction error $\text{Err} = E_F\{D(y_0, \hat{y}_0)\}$ (12.7) for a new case obtained independently of $\boldsymbol{d}$. A first guess is the apparent error

$$\text{err} = \frac{1}{N}\sum_{i=1}^{N} D(y_i, \hat{y}_i), \tag{12.18}$$

the average discrepancy in the training set between y_i and its prediction $\hat{y}_i = r_{\boldsymbol{d}}(x_i)$; err usually underestimates Err since $r_{\boldsymbol{d}}(x)$ has been adjusted[3] to fit the observed responses y_i.

The ideal remedy, discussed in Section 12.4, would be to have an independent *validation set* (or *test* set) $\boldsymbol{d}_{\text{val}}$ of N_{val} additional cases,

$$\boldsymbol{d}_{\text{val}} = \left\{(x_{0j}, y_{0j}),\ j = 1, 2, \ldots, N_{\text{val}}\right\}. \tag{12.19}$$

This would provide as unbiased estimate of Err,

$$\widehat{\text{Err}}_{\text{val}} = \frac{1}{N_{\text{val}}}\sum_{j=1}^{N_{\text{val}}} D(y_{0j}, \hat{y}_{0j}), \qquad \hat{y}_{0j} = r_{\boldsymbol{d}}(x_{0j}). \tag{12.20}$$

Cross-validation attempts to mimic $\widehat{\text{Err}}_{\text{val}}$ without the need for a validation set. Define $\boldsymbol{d}(i)$ to be the reduced training set in which pair (x_i, y_i) has been omitted, and let $r_{\boldsymbol{d}(i)}(\cdot)$ indicate the rule constructed on the basis

[3] Linear regression using ordinary least squares fitting provides a classical illustration: $\text{err} = \sum_i (y_i - \hat{y}_i)^2/N$ must be increased to $\sum_i (y_i - \hat{y}_i)^2/(N - p)$, where p is the degrees of freedom, to obtain an unbiased estimate of the noise variance σ^2.

of $\boldsymbol{d}(i)$. The *cross-validation estimate* of prediction error is

$$\widehat{\text{Err}}_{\text{cv}} = \frac{1}{N}\sum_{i=1}^{N} D(y_i, \hat{y}_{(i)}), \qquad \hat{y}_{(i)} = r_{\boldsymbol{d}(i)}(x_i). \tag{12.21}$$

Now (x_i, y_i) is *not* involved in the construction of the prediction rule for y_i.

$\widehat{\text{Err}}_{\text{cv}}$ (12.21) is the "leave one out" version of cross-validation. A more common tactic is to leave out several pairs at a time: $\boldsymbol{d}$ is randomly partitioned into J groups of size about N/J each; $\boldsymbol{d}(j)$, the training set with group j omitted, provides rule $r_{\boldsymbol{d}(j)}(x)$, which is used to provide predictions for the y_i in group j. Then $\widehat{\text{Err}}_{\text{cv}}$ is evaluated as in (12.21). Besides reducing the number of rule constructions necessary, from N to J, grouping induces larger changes among the J training sets, improving the predictive performance on rules $r_{\boldsymbol{d}}(x)$ that include discontinuities. (The argument here is similar to that for the jackknife, Section 10.1.)

Cross-validation was applied to the supernova data pictured in Figure 12.1. The 39 cases were split, randomly, into $J = 13$ groups of three cases each. This gave

$$\widehat{\text{Err}}_{\text{cv}} = 1.17, \tag{12.22}$$

(12.21), 62% larger than err $= 0.72$ (12.9). The R^2 calculation (12.12) now yields the smaller value

$$R^2 = 1 - 1.17/3.91 = 0.701. \tag{12.23}$$

We can apply cross-validation to the **spam** data of Section 8.1, having $N = 4061$ cases, $p = 57$ predictors, and dichotomous response y. For this example, each of the 57 predictors was itself dichotomized to be either 0 or 1 depending on whether the original value x_{ij} equaled zero or not. A logistic regression, Section 8.1, regressing y_i on the 57 dichotomized predictors, gave apparent classification error (12.5)

$$\text{err} = 0.064, \tag{12.24}$$

i.e., 295 wrong predictions among the 4061 cases. Cross-validation, with $J = 10$ groups of size 460 or 461 each, increased this to

$$\widehat{\text{Err}}_{\text{cv}} = 0.069, \tag{12.25}$$

an increase of 8%.

Glmnet is an automatic model building program that, among other things, constructs a lasso sequence of logistic regression models, adding

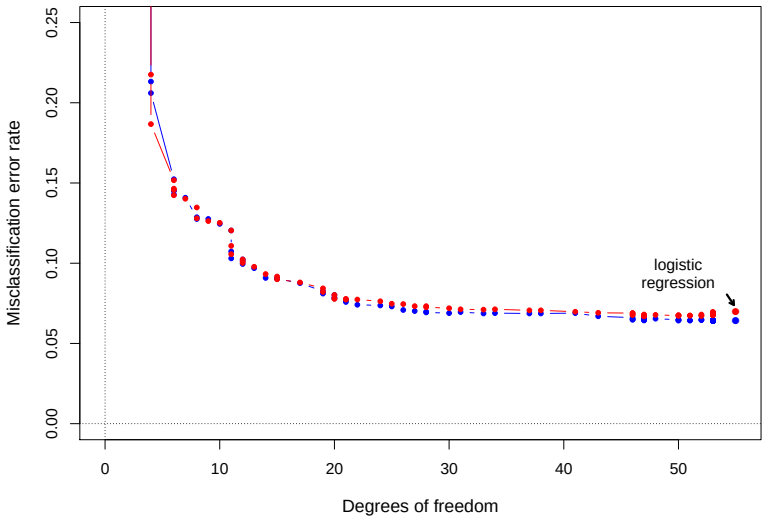

Figure 12.2 Spam data. Apparent error rate err (blue) and cross-validated estimate (red) for a sequence of prediction rules generated by `glmnet`. The degrees of freedom are the number of nonzero regression coefficients: df = 57 corresponds to ordinary logistic regression, which gave apparent err 0.064, cross-validated rate 0.069. The minimum cross-validated error rate is 0.067.

variables one at a time in their order of apparent predictive power; see Chapter 16. The solid curve in Figure 12.2 tracks the apparent error err (12.18) as a function of the number of predictors employed. Aside from numerical artifacts, err is monotonically decreasing, declining to err = 0.064 for the full model that employs all 57 predictors, i.e., for the usual logistic regression model, as in (12.24).

`Glmnet` produced prediction error estimates $\widehat{\text{Err}}_{\text{cv}}$ for each of the successive models, shown by the dashed curve. These are a little noisy themselves, but settle down between 4% and 8% above the corresponding err estimates. The minimum value

$$\widehat{\text{Err}}_{\text{cv}} = 0.067 \tag{12.26}$$

occurred for the model using 47 predictors.

The difference between (12.26) and (12.25) is too small to take seriously given the noise in the $\widehat{\text{Err}}_{\text{cv}}$ estimates. There is a more subtle objection: the choice of "best" prediction rule based on comparative $\widehat{\text{Err}}_{\text{cv}}$ estimates is not itself cross-validated. Each case (x_i, y_i) is involved in choosing its

own best prediction, so $\widehat{\mathrm{Err}}_{\mathrm{cv}}$ at the apparently optimum choice cannot be taken entirely at face value.

Nevertheless, perhaps the principal use of cross-validation lies in choosing among competing prediction rules. Whether or not this is fully justified, it is often the only game in town. That being said, minimum predictive error, no matter how effectuated, is a notably weaker selection principle than minimum variance of estimation.

As an example, consider an iid normal sample

$$x_i \overset{\text{iid}}{\sim} \mathcal{N}(\mu, 1), \qquad i = 1, 2, \ldots, 25, \tag{12.27}$$

having mean $\bar{x}$ and median $\check{x}$. Both are unbiased for estimating μ, but $\bar{x}$ is much more efficient,

$$\mathrm{var}(\check{x}) / \mathrm{var}(\bar{x}) \doteq 1.57. \tag{12.28}$$

Suppose we wish to predict a future observation x_0 independently selected from the same $\mathcal{N}(\mu, 1)$ distribution. In this case there is very little advantage to $\bar{x}$,

$$E\left\{(x_0 - \check{x})^2\right\} \Big/ E\left\{(x_0 - \bar{x})^2\right\} = 1.02. \tag{12.29}$$

The noise in $x_0 \sim \mathcal{N}(\mu, 1)$ dominates its prediction error. Perhaps the proliferation of prediction algorithms to be seen in Part III reflects how weakly changes in strategy affect prediction error.

Table 12.2 *Ratio of predictive errors $E\{(\bar{x}_0 - \check{x})^2\}/E\{(\bar{x}_0 - \bar{x})^2\}$ for $\bar{x}_0$ the mean of an independent sample of size N_0 from $\mathcal{N}(\mu, 1)$; $\bar{x}$ and $\check{x}$ are the mean and median from $x_i \sim \mathcal{N}(\mu, 1)$ for $i = 1, 2, \ldots, 25$.*

N_0	1	10	100	1000	∞
Ratio	1.02	1.16	1.46	1.56	1.57

In this last example, suppose that our task was to predict the *average* $\bar{x}_0$ of N_0 further draws from the $\mathcal{N}(\mu, 1)$ distribution. Table 12.2 shows the ratio of predictive errors as a function of N_0. The superiority of the mean compared to the median reveals itself as N_0 gets larger. In this supersimplified example, the difference between prediction and estimation lies in predicting the average of *one* versus an *infinite number* of future observations.

Does $\widehat{\mathrm{Err}}_{\mathrm{cv}}$ actually estimate $\mathrm{Err}_{\boldsymbol{d}}$ as defined in (12.7)? It seems like the answer must be yes, but there is some doubt expressed in the literature, for

reasons demonstrated in the following simulation: we take the true distribution F in (12.6) to be the discrete distribution $\hat{F}$ that puts weight $1/39$ on each of the 39 (x_i, y_i) pairs of the supernova data.[4] A random sample with replacement of size 39 from $\hat{F}$ gives simulated data set $\boldsymbol{d}^*$ and prediction rule $r_{\boldsymbol{d}^*}(\cdot)$ based on the lasso/C_p recipe used originally. The same cross-validation procedure as before, applied to $\boldsymbol{d}^*$, gives $\widehat{\text{Err}}^*_{\text{cv}}$. Because this is a simulation, we can also compute the actual mean-squared error rate of rule $r_{\boldsymbol{d}^*}(\cdot)$ applied to the true distribution $\hat{F}$,

$$\text{Err}^* = \frac{1}{39}\sum_{i=1}^{39} D\left(y_i, r_{\boldsymbol{d}^*}(x_i)\right). \tag{12.30}$$

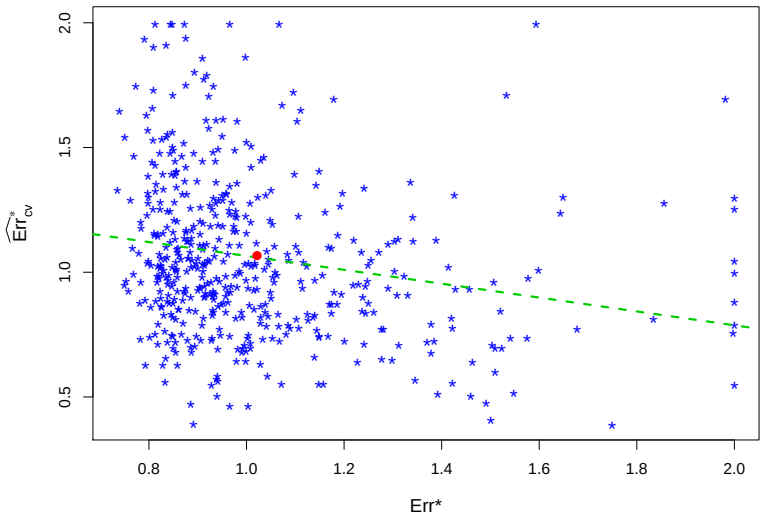

Figure 12.3 Simulation experiment comparing true error Err with cross-validation estimate $\widehat{\text{Err}}^*_{\text{cv}}$; 500 simulations based on the supernova data. $\widehat{\text{Err}}^*_{\text{cv}}$ and Err are negatively correlated.

Figure 12.3 plots $(\text{Err}^*, \widehat{\text{Err}}^*_{\text{cv}})$ for 500 simulations, using squared error discrepancy $D(y, \hat{y}) = (y - \hat{y})^2$. Summary statistics are given in Table 12.3. $\widehat{\text{Err}}^*_{\text{cv}}$ has performed well overall, averaging 1.07, quite near the true Err 1.02, both estimates being 80% greater than the average apparent error 0.57. However, the figure shows something unsettling: there is a

[4] Simulation based on $\hat{F}$ is the same as nonparametric bootstrap analysis, Chapter 10.

Table 12.3 *True error* Err^**, cross-validated error* $\widehat{\text{Err}}^*_{\text{cv}}$*, and apparent error* err^**; 500 simulations based on supernova data. Correlation* -0.175 *between* Err^* *and* $\widehat{\text{Err}}^*_{\text{cv}}$.

	Err^*	$\widehat{\text{Err}}^*_{\text{cv}}$	err^*
Mean	1.02	1.07	.57
St dev	.27	.34	.16

negative correlation between $\widehat{\text{Err}}^*_{\text{cv}}$ and Err^*. Large values of $\widehat{\text{Err}}^*_{\text{cv}}$ go with smaller values of the true prediction error, and vice versa.

Our original definition of Err,

$$\text{Err}_{\boldsymbol{d}} = E_F\{D(y_0, r_{\boldsymbol{d}}(x_0))\}, \tag{12.31}$$

took $r_{\boldsymbol{d}}(\cdot)$ fixed as constructed from $\boldsymbol{d}$, only $(x_0, y_0) \sim F$ random. In other words, $\text{Err}_{\boldsymbol{d}}$ was the expected prediction error for the specific rule $r_{\boldsymbol{d}}(\cdot)$, as is Err^* for $r_{\boldsymbol{d}^*}(\cdot)$. If $\widehat{\text{Err}}^*_{\text{cv}}$ is tracking Err^* we would expect to see a positive correlation in Figure 12.3.

As it is, all we can say is that $\widehat{\text{Err}}^*_{\text{cv}}$ is estimating the *expected* predictive error, where $\boldsymbol{d}$ as well as (x_0, y_0) is random in definition (12.31). This makes cross-validation a *strongly* frequentist device: $\widehat{\text{Err}}_{\text{cv}}$ is estimating the average prediction error of the algorithm producing $r_{\boldsymbol{d}}(\cdot)$, not of $r_{\boldsymbol{d}}(\cdot)$ itself.

12.3 Covariance Penalties

Cross-validation does its work nonparametrically and without the need for probabilistic modeling. Covariance penalty procedures require probability models, but within their ambit they provide less noisy estimates of prediction error. Some of the most prominent covariance penalty techniques will be examined here, including *Mallows' C_p*, *Akaike's information criterion* (AIC), and *Stein's unbiased risk estimate* (SURE).

The covariance penalty approach treats prediction error estimation in a regression framework: the predictor vectors x_i in the training set $\boldsymbol{d} = \{(x_i, y_i),\ i = 1, 2, \ldots, N\}$ (12.1) are considered *fixed* at their observed values, not random as in (12.6). An unknown vector $\boldsymbol{\mu}$ of expectations $\mu_i = E\{y_i\}$ has yielded the observed vector of responses $\boldsymbol{y}$ according to some given probability model, which to begin with we assume to have the

simple form

$$\boldsymbol{y} \sim (\boldsymbol{\mu}, \sigma^2 \boldsymbol{I}); \tag{12.32}$$

that is, the y_i are uncorrelated, with y_i having unknown mean μ_i and variance σ^2. We take σ^2 as known, though in practice it must usually be estimated.

A regression rule $r(\cdot)$ has been used to produce an estimate of vector $\boldsymbol{\mu}$,

$$\hat{\boldsymbol{\mu}} = r(\boldsymbol{y}). \tag{12.33}$$

(Only $\boldsymbol{y}$ is included in the notation since the predictors x_i are considered fixed and known.) For instance we might take

$$\hat{\boldsymbol{\mu}} = r(\boldsymbol{y}) = \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y}, \tag{12.34}$$

where $\boldsymbol{X}$ is the $N \times p$ matrix having x_i as the ith row, as suggested by the linear regression model $\boldsymbol{\mu} = \boldsymbol{X}\beta$.

In covariance penalty calculations, the estimator $\hat{\boldsymbol{\mu}}$ also functions as a predictor. We wonder how accurate $\hat{\boldsymbol{\mu}} = r(\boldsymbol{y})$ will be in predicting a new vector of observations $\boldsymbol{y}_0$ from model (12.32),

$$\boldsymbol{y}_0 \sim (\boldsymbol{\mu}, \sigma^2 \boldsymbol{I}), \qquad \text{independent of } \boldsymbol{y}. \tag{12.35}$$

To begin with, prediction error will be assessed in terms of squared discrepancy,

$$\mathrm{Err}_i = E_0\left\{(y_{0i} - \hat{\mu}_i)^2\right\} \tag{12.36}$$

for component i, where E_0 indicates expectation with y_{0i} random but $\hat{\mu}_i$ held fixed. Overall prediction error is the average[5]

$$\mathrm{Err.} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{Err}_i\ . \tag{12.37}$$

The *apparent error* for component i is

$$\mathrm{err}_i = (y_i - \hat{\mu}_i)^2. \tag{12.38}$$

A simple but powerful lemma underlies the theory of covariance penalties.

Lemma *Let E indicate expectation over both $\boldsymbol{y}$ in* (12.32) *and $\boldsymbol{y}_0$ in* (12.35)*. Then*

$$E\{\mathrm{Err}_i\} = E\{\mathrm{err}_i\} + 2\,\mathrm{cov}(\hat{\mu}_i, y_i), \tag{12.39}$$

[5] Err. is sometimes called "insample error," as opposed to "outsample error" Err (12.7), though in practice the two tend to behave similarly.

where the last term is the covariance between the i th components of $\hat{\boldsymbol{\mu}}$ *and* $\boldsymbol{y}$,

$$\operatorname{cov}(\hat{\mu}_i, y_i) = E\{(\hat{\mu}_i - \mu_i)(y_i - \mu_i)\}. \tag{12.40}$$

(Note: (12.40) does not require $E\{\hat{\mu}_i\} = \mu_i$.)

Proof Letting $\epsilon_i = y_i - \mu_i$ and $\delta_i = (\hat{\mu}_i - \mu_i)$, the elementary equality $(\epsilon_i - \delta_i)^2 = \epsilon_i^2 - 2\epsilon_i\delta_i + \delta_i^2$ becomes

$$(y_i - \hat{\mu}_i)^2 = (y_i - \mu_i)^2 - 2(\hat{\mu}_i - \mu_i)(y_i - \mu_i) + (\hat{\mu}_i - \mu_i)^2, \tag{12.41}$$

and likewise

$$(y_{0i} - \hat{\mu}_i)^2 = (y_{0i} - \mu_i)^2 - 2(\hat{\mu}_i - \mu_i)(y_{0i} - \mu_i) + (\hat{\mu}_i - \mu_i)^2. \tag{12.42}$$

Taking expectations, (12.41) gives

$$E\{\text{err}_i\} = \sigma^2 - 2\operatorname{cov}(\hat{\mu}_i, y_i) + E(\hat{\mu}_i - \mu_i)^2, \tag{12.43}$$

while (12.42) gives

$$E\{\text{Err}_i\} = \sigma^2 + E(\hat{\mu}_i - \mu_i)^2, \tag{12.44}$$

the middle term on the right side of (12.42) equaling zero because of the independence of y_{0i} and $\hat{\mu}_i$. Taking the difference between (12.44) and (12.43) verifies the lemma. ■

Note: The lemma remains valid if σ^2 varies with i.

The lemma says that, on average, the apparent error err_i understimates the true prediction error Err_i by the *covariance penalty* $2\operatorname{cov}(\hat{\mu}_i, y_i)$. (This makes intuitive sense since $\operatorname{cov}(\mu_i, y_i)$ measures the amount by which y_i influences its own prediction $\hat{\mu}_i$.) Covariance penalty estimates of prediction error take the form

$$\widehat{\text{Err}}_i = \text{err}_i + 2\widehat{\operatorname{cov}}(\hat{\mu}_i, y_i), \tag{12.45}$$

where $\widehat{\operatorname{cov}}(\hat{\mu}_i, y_i)$ approximates $\operatorname{cov}(\mu_i, y_i)$; overall prediction error (12.37) is estimated by

$$\widehat{\text{Err}}. = \text{err} + \frac{2}{N}\sum_{i=1}^{N}\widehat{\operatorname{cov}}(\hat{\mu}_i, y_i), \tag{12.46}$$

where $\text{err} = \sum \text{err}_i / N$ as before.

The form of $\widehat{\operatorname{cov}}(\hat{\mu}_i, y_i)$ in (12.45) depends on the context assumed for the prediction problem.

(1) Suppose that $\hat{\boldsymbol{\mu}} = r(\boldsymbol{y})$ in (12.32)–(12.33) is *linear*,

$$\hat{\boldsymbol{\mu}} = \boldsymbol{c} + \boldsymbol{M}\boldsymbol{y}, \tag{12.47}$$

where $\boldsymbol{c}$ is a known N-vector and $\boldsymbol{M}$ a known $N \times N$ matrix. Then the covariance matrix between $\hat{\boldsymbol{\mu}}$ and $\boldsymbol{y}$ is

$$\operatorname{cov}(\hat{\boldsymbol{\mu}}, \boldsymbol{y}) = \sigma^2 \boldsymbol{M}, \tag{12.48}$$

giving $\operatorname{cov}(\hat{\mu}_i, y_i) = \sigma^2 M_{ii}$, M_{ii} the ith diagonal element of $\boldsymbol{M}$,

$$\widehat{\mathrm{Err}}_i = \mathrm{err}_i + 2\sigma^2 M_{ii}, \tag{12.49}$$

and, since $\mathrm{err} = \sum_i (y_i - \hat{\mu}_i)^2/N$,

$$\widehat{\mathrm{Err}}. = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{\mu}_i)^2 + \frac{2\sigma^2}{N}\operatorname{tr}(\boldsymbol{M}). \tag{12.50}$$

Formula (12.50) is *Mallows' C_p* estimate of prediction error. For OLS estimation (12.34), $\boldsymbol{M} = \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'$ has $\operatorname{tr}(\boldsymbol{M}) = p$, the number of predictors, so

$$\widehat{\mathrm{Err}}. = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{\mu}_i)^2 + \frac{2}{N}\sigma^2 p. \tag{12.51}$$

For the supernova data (12.8)–(12.9), the OLS predictor $\hat{\boldsymbol{\mu}} = \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y}$ yielded $\mathrm{err} = \sum(y_i - \hat{\mu}_i)^2/39 = 0.719$. The covariance penalty, with $N = 39$, $\sigma^2 = 1$, and[6] $p = 10$, was 0.513, giving C_p estimate of prediction error

$$\widehat{\mathrm{Err}}. = 0.719 + 0.513 = 1.23. \tag{12.52}$$

For OLS regression, the *degrees of freedom* p, the rank of matrix $\boldsymbol{X}$ in (12.34), determines the covariance penalty $(2/N)\sigma^2 p$ in (12.51). Comparing this with (12.46) leads to a general definition of degrees of freedom df for a regression rule $\hat{\boldsymbol{\mu}} = r(\boldsymbol{y})$,

$$\mathrm{df} = (1/\sigma^2)\sum_{i=1}^{N}\widehat{\operatorname{cov}}(\hat{\mu}_i, y_i). \tag{12.53}$$

This definition provides common ground for comparing different types of regression rules. Rules with larger df are more flexible and tend toward better apparent fits to the data, but require bigger covariance penalties for fair comparison.

[6] We are not counting the intercept as an 11th predictor since $\boldsymbol{y}$ and all the x_i were standardized to have mean 0, all our models assuming zero intercept.

(2) For lasso estimation (7.42) and (12.10), it can be shown that formula (12.51), with p equaling the number of nonzero regression coefficients, holds to a good approximation.† The lasso rule used in Figure 12.1 for the supernova data had $p = 7$; err was 0.720 for this rule, almost the same as for the OLS rule above, but the C_p penalty is less, $2{\cdot}7/39 = 0.359$, giving

$$\widehat{\text{Err}}. = 0.720 + 0.359 = 1.08, \tag{12.54}$$

compared with 1.23 for OLS. This estimate does not account for the data-based selection of the choice $p = 7$, see item **(4)** below.

(3) If we are willing to add multivariate normality to model (12.32),

$$\boldsymbol{y} \sim \mathcal{N}_p(\boldsymbol{\mu}, \sigma^2 \boldsymbol{I}), \tag{12.55}$$

we can drop the assumption of linearity (12.47). In this case it can be shown that, for any differentiable estimator $\hat{\boldsymbol{\mu}} = r(\boldsymbol{y})$, the covariance in formula (12.51) is given by†

$$\text{cov}(\hat{\mu}_i, y_i) = \sigma^2 E\{\partial \hat{\mu}_i / \partial y_i\}, \tag{12.56}$$

σ^2 times the partial derivative of $\hat{\mu}_i$ with respect to y_i. (Another measure of y_i's influence on its own prediction.) The SURE formula (Stein's unbiased risk estimator) is

$$\widehat{\text{Err}}_i = \text{err}_i + 2\sigma^2 \frac{\partial \hat{\mu}_i}{\partial y_i}, \tag{12.57}$$

with corresponding estimate for overall prediction error

$$\widehat{\text{Err}}. = \text{err} + \frac{2\sigma^2}{N} \sum_{i=1}^{N} \frac{\partial \hat{\mu}_i}{\partial y_i}. \tag{12.58}$$

SURE was applied to the rule $\hat{\boldsymbol{\mu}} =$ `lowess(x,y,1/3)` for the kidney fitness data of Figure 1.2. The open circles in Figure 12.4 plot the component-wise degrees of freedom estimates[7]

$$\frac{\partial \hat{\mu}_i}{\partial y_i}, \qquad i = 1, 2, \ldots, N = 157, \tag{12.59}$$

(obtained by numerical differentiation) versus age_i. Their sum

$$\sum_{i=1}^{N} \frac{\partial \hat{\mu}_i}{\partial y_i} = 6.67 \tag{12.60}$$

[7] Notice that the factor σ^2 in (12.56) cancels out in (12.53).

estimates the total degrees of freedom, as in (12.53), implying that `lowess(x,y,1/3)` is about as flexible as a sixth-degree polynomial fit, with df = 7.

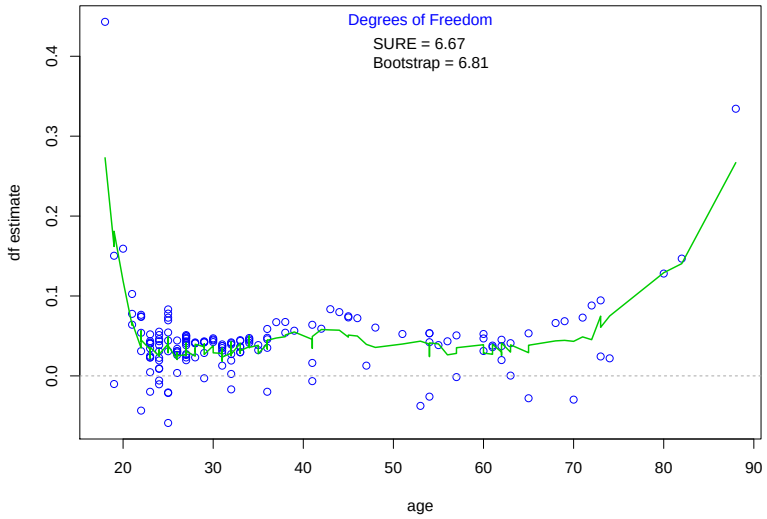

Figure 12.4 Analysis of the `lowess(x,y,1/3)` fit to kidney data of Figure 1.2. Open circles are SURE coordinate-wise df estimates $\partial\hat{\mu}_i/\partial y_i$, plotted versus age_i, giving total degrees of freedom 6.67. The solid curve tracks bootstrap coordinate-wise estimates (12.65), with their sum giving total df $= 6.81$.

(4) The *parametric bootstrap*[8] of Section 10.4 can be used to estimate the covariances $\text{cov}(\hat{\mu}_i, y_i)$ in the lemma (12.39). The data vector $\boldsymbol{y}$ is assumed to be generated from a member $f_{\boldsymbol{\mu}}(\boldsymbol{y})$ of a given parametric family

$$\mathcal{F} = \{f_{\boldsymbol{\mu}}(\boldsymbol{y}),\ \boldsymbol{\mu} \in \Omega\}, \tag{12.61}$$

yielding $\hat{\boldsymbol{\mu}} = r(\boldsymbol{y})$,

$$f_{\boldsymbol{\mu}} \to \boldsymbol{y} \to \hat{\boldsymbol{\mu}} = r(\boldsymbol{y}). \tag{12.62}$$

Parametric bootstrap replications of $\boldsymbol{y}$ and $\hat{\boldsymbol{\mu}}$ are obtained by analogy with

[8] There is also a *nonparametric* bootstrap competitor to cross-validation, the ".632 estimate;" see the chapter endnote †4.

(12.62),[9]

$$f_{\hat{\boldsymbol{\mu}}} \to \boldsymbol{y}^* \to \hat{\boldsymbol{\mu}}^* = r(\boldsymbol{y}^*). \tag{12.63}$$

A large number B of replications then yield bootstrap estimates

$$\widehat{\mathrm{cov}}(\hat{\mu}_i, y_i) = \frac{1}{B}\sum_{b=1}^{B}(\hat{\mu}_i^{*b} - \hat{\mu}_i^{*\cdot})(y_i^{*b} - y_i^{*\cdot}), \tag{12.64}$$

the dot notation indicating averages over the B replications.

$B = 1000$ parametric bootstrap replications $(\hat{\boldsymbol{\mu}}^*, \boldsymbol{y}^*)$ were obtained from the normal model (12.55), taking $\hat{\boldsymbol{\mu}}$ in (12.63) to be the estimate from `lowess(x,y,1/3)` as in Figure 1.2. A standard linear regression, of y as a 12th-degree polynomial function of age, gave $\hat{\sigma}^2 = 3.28$. Covariances were computed as in (12.64), yielding coordinate-wise degrees of freedom estimates (12.53),

$$\mathrm{df}_i = \widehat{\mathrm{cov}}(\hat{\mu}_i, y_i)/\hat{\sigma}^2. \tag{12.65}$$

The solid curve in Figure 12.4 plots df_i as a function of age_i. These are seen to be similar to but less noisy than the SURE estimates. They totaled 6.81, nearly the same as (12.60). The overall covariance penalty term in (12.46) equaled 0.284, increasing $\widehat{\mathrm{Err}}$. by about 9% over err $= 3.15$.

The advantage of parametric bootstrap estimates (12.64) of covariance penalties is their applicability to *any* prediction rule $\hat{\boldsymbol{\mu}} = r(\boldsymbol{y})$ no matter how exotic. Applied to the lasso estimates for the supernova data, $B = 1000$ replications yielded total df $= 6.85$ for the rule that always used $p = 7$ predictors, compared with the theoretical approximation df $= 7$. Another 1000 replications, now letting $\hat{\boldsymbol{\mu}}^* = r(\boldsymbol{y}^*)$ choose the apparently best p^* each time, increased the df estimate to 7.48, so the adaptive choice of p cost about 0.6 extra degrees of freedom. These calculations exemplify modern computer-intensive inference, carrying through error estimation for complicated adaptive prediction rules on a totally automatic basis.

(5) Covariance penalties can apply to measures of prediction error other than squared error $D(y_i, \hat{\mu}_i) = (y_i - \hat{\mu}_i)^2$. We will discuss two examples of a general theory. First consider *classification*, where y_i equals 0 or 1 and

[9] It isn't necessary for the $\hat{\boldsymbol{\mu}}$ in (12.63) to equal $\hat{\boldsymbol{\mu}} = \boldsymbol{r}(\boldsymbol{y})$. The calculation (12.64) was rerun taking $\hat{\boldsymbol{\mu}}$ in (12.63) from `lowess(x,y,1/6)` (but with $\boldsymbol{r}(\boldsymbol{y})$ still from `lowess(x,y,1/3)`) with almost identical results. In general, one might take $\hat{\boldsymbol{\mu}}$ in (12.63) to be from a more flexible, less biased, estimator than $\boldsymbol{r}(\boldsymbol{y})$.

similarly the predictor $\hat{\mu}_i$, with dichotomous error

$$D(y_i, \hat{\mu}_i) = \begin{cases} 1 & \text{if } y_i \neq \hat{\mu}_i \\ 0 & \text{if } y_i = \hat{\mu}_i, \end{cases} \tag{12.66}$$

as in (12.5).[10] In this situation, the apparent error is the observed proportion of prediction mistakes in the training set (12.1),

$$\text{err} = \#\{y_i \neq \hat{\mu}_i\}/N. \tag{12.67}$$

Now the true prediction error for case i is

$$\text{Err}_i = \Pr_0\{y_{0i} \neq \hat{\mu}_i\}, \tag{12.68}$$

the conditional probability given $\hat{\mu}_i$ that an independent replicate y_{0i} of y_i will be incorrectly predicted. The lemma holds as stated in (12.39), leading to the prediction error estimate

$$\widehat{\text{Err}} = \frac{\#\{y_i \neq \hat{\mu}_i\}}{N} + \frac{2}{N}\sum_{i=1}^{N} \text{cov}(\hat{\mu}_i, y_i). \tag{12.69}$$

Some algebra yields

$$\text{cov}(\hat{\mu}_i, y_i) = \mu_i(1-\mu_i)\left(\Pr\{\hat{\mu}_i = 1 | y_i = 1\} - \Pr\{\hat{\mu}_1 = 1 | y_i = 0\}\right), \tag{12.70}$$

with $\mu_i = \Pr\{y_i = 1\}$, showing again the covariance penalty measuring the self-influence of y_i on its own prediction.

As a second example, suppose that the observations y_i are obtained from different members of a one-parameter exponential family $f_\mu(y) = \exp\{\lambda y - \gamma(\lambda)\} f_0(y)$ (8.32),

$$y_i \sim f_{\mu_i}(y_i) \qquad \text{for } i = 1, 2, \ldots, N, \tag{12.71}$$

and that error is measured by the deviance (8.31),

$$D(y, \hat{\mu}) = 2\int_{\mathcal{Y}} f_y(Y) \log\left(\frac{f_y(Y)}{f_{\hat{\mu}}(Y)}\right) dY. \tag{12.72}$$

According to (8.33), the apparent error $\sum D(y_i, \hat{\mu}_i)$ is then

$$\text{err} = \frac{2}{N}\sum_{i=1}^{N} \log\left(\frac{f_{y_i}(y_i)}{f_{\hat{\mu}_i}(y_i)}\right) = \frac{2}{N}\left\{\log\left(f_{\boldsymbol{y}}(\boldsymbol{y})\right) - \log\left(f_{\hat{\boldsymbol{\mu}}}(\boldsymbol{y})\right)\right\}. \tag{12.73}$$

[10] More generally, $\hat{\pi}_i$ is some predictor of $\Pr\{y_i = 1\}$, and $\hat{\mu}_i$ is the indicator function $I(\hat{\pi}_i \geq 0.5)$.

In this case the general theory gives overall covariance penalty

$$\text{penalty} = \frac{2}{N}\sum_{i=1}^{N}\text{cov}\left(\hat{\lambda}_i, \hat{\mu}_i\right), \tag{12.74}$$

where $\hat{\lambda}_i$ is the natural parameter in family (8.32) corresponding to $\hat{\mu}_\lambda$ (e.g., $\hat{\lambda}_i = \log\hat{\mu}_i$ for Poisson observations). Moreover, if $\hat{\boldsymbol{\mu}}$ is obtained as the MLE of $\boldsymbol{\mu}$ in a generalized linear model with p degrees of freedom (8.22),

$$\text{penalty} \doteq \frac{2p}{N} \tag{12.75}$$

to a good approximation. The corresponding version of $\widehat{\text{Err}}$. (12.46) can then be written as

$$\widehat{\text{Err}}. \doteq -\frac{2}{N}\left\{\log\left(f_{\hat{\boldsymbol{\mu}}}(\boldsymbol{y})\right) - p\right\} + \text{ constant}, \tag{12.76}$$

the constant $(2/N)\log(f_{\boldsymbol{y}}(\boldsymbol{y}))$ not depending on $\hat{\boldsymbol{\mu}}$.

The term in brackets is the *Akaike information criterion* (AIC): if the statistician is comparing possible prediction rules $r^{(j)}(\boldsymbol{y})$ for a given data set $\boldsymbol{y}$, the AIC says to select the rule maximizing the *penalized* maximum likelihood

$$\log\left(f_{\hat{\boldsymbol{\mu}}^{(j)}}(\boldsymbol{y})\right) - p^{(j)}, \tag{12.77}$$

where $\hat{\boldsymbol{\mu}}^{(j)}$ is rule j's MLE and $p^{(j)}$ its degrees of freedom. Comparison with (12.76) shows that for GLMs, the AIC amounts to selecting the rule with the smallest value of $\widehat{\text{Err}}_{\cdot}^{(j)}$.

Cross-validation does not require a probability model, but if such a model is available then the error estimate $\widehat{\text{Err}}_{\text{cv}}$ can be improved by *bootstrap smoothing*.[11] With the predictor vectors x_i considered fixed as observed, a parametric model generates the data set $\boldsymbol{d} = \{(x_i, y_i),\ i = 1, \ldots, N\}$ as in (12.62), from which we calculate the prediction rule $r_{\boldsymbol{d}}(\cdot)$ and the error estimate $\widehat{\text{Err}}_{\text{cv}}$ (12.21),

$$f_{\boldsymbol{\mu}} \to \boldsymbol{d} \to r_{\boldsymbol{d}}(\cdot) \to \widehat{\text{Err}}_{\text{cv}}. \tag{12.78}$$

Substituting the estimated density $f_{\hat{\boldsymbol{\mu}}}$ for f_μ, as in (12.63), provides

[11] Perhaps better known as "bagging;" see Chapter 17.

parametric bootstrap replicates of $\widehat{\text{Err}}_{\text{cv}}$,

$$f_{\hat{\boldsymbol{\mu}}} \to \boldsymbol{d}^* \to r_{\boldsymbol{d}^*}(\cdot) \to \widehat{\text{Err}}^*_{\text{cv}}. \tag{12.79}$$

Some large number B of replications can then be averaged to give the smoothed estimate

$$\overline{\text{Err}} = \frac{1}{B}\sum_{b=1}^{B} \widehat{\text{Err}}^{*b}_{\text{cv}}. \tag{12.80}$$

$\overline{\text{Err}}$ averages out the considerable noise in $\widehat{\text{Err}}_{\text{cv}}$, often significantly reducing its variability.[12]

A surprising result, referenced in the endnotes, shows that $\overline{\text{Err}}$ approximates the covariance penalty estimate Err.. Speaking broadly, Err. is what's left after excess randomness is squeezed out of $\widehat{\text{Err}}_{\text{cv}}$ (an example of "Rao–Blackwellization," to use classical terminology). Improvements can be quite substantial. † Covariance penalty estimates, when believable parametric †4
models are available, should be preferred to cross-validation.

12.4 Training, Validation, and Ephemeral Predictors

Good Practice suggests splitting the full set of observed predictor–response pairs (x, y) into a training set $\boldsymbol{d}$ of size N (12.1), and a validation set $\boldsymbol{d}_{\text{val}}$, of size N_{val} (12.16). The validation set is put into a vault while the training set is used to develop an effective prediction rule $r_{\boldsymbol{d}}(x)$. Finally, $\boldsymbol{d}_{\text{val}}$ is removed from the vault and used to calculate $\widehat{\text{Err}}_{\text{val}}$ (12.20), an honest estimate of the predictive error rate of $r_{\boldsymbol{d}}$.

This *is* a good idea, and seems foolproof, at least if one has enough data to afford setting aside a substantial portion for a validation set during the training process. Nevertheless, there remains some peril of underestimating the true error rate, arising from *ephemeral* predictors, those whose predictive powers fade away over time. A contrived, but not completely fanciful, example illustrates the danger.

The example takes the form of an imaginary microarray study involving 360 subjects, 180 patients and 180 healthy controls, coded

$$y_i = \begin{cases} 1 & \text{patient} \\ 0 & \text{control,} \end{cases} \qquad i = 1, 2, \ldots, 360. \tag{12.81}$$

[12] A related tactic pertaining to grouped cross-validation is to repeat calculation (12.21) for several different randomly selected splits into J groups, and then average the resulting $\widehat{\text{Err}}_{\text{cv}}$ estimates.

Each subject is assessed on a microarray measuring the genetic activity of $p = 100$ genes, these being the predictors

$$x_i = (x_{i1}, x_{i2}, x_{i3}, \ldots, x_{i100})'. \tag{12.82}$$

One subject per day is assessed, alternating patients and controls.

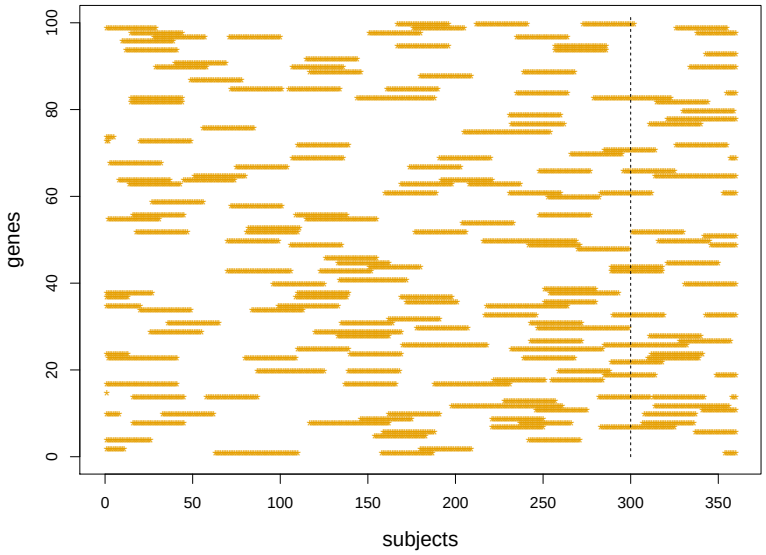

Figure 12.5 Orange bars indicate transient episodes, (12.84) and the reverse, for imaginary medical study (12.81)–(12.82).

The measurements x_{ij} are independent of each other and of the y_i's,

$$x_{ij} \overset{\text{ind}}{\sim} \mathcal{N}(\mu_{ij}, 1) \qquad \text{for } i = 1, 2, \ldots, 360 \quad \text{and} \quad j = 1, 2, \ldots, 100. \tag{12.83}$$

Most of the μ_{ij} equal zero, but each gene's measurements can experience "transient episodes" of two possible types: in type 1,

$$\mu_{ij} = \begin{cases} 2 & \text{if } y_i = 1 \\ -2 & \text{if } y_i = 0, \end{cases} \tag{12.84}$$

while type 2 reverses signs. The episodes are about 30 days long, randomly and independently located between days 1 and 360, with an average of two episodes per gene. The orange bars in Figure 12.5 indicate the episodes.

For the purpose of future diagnoses we wish to construct a prediction rule $\hat{y} = r_{\boldsymbol{d}}(x)$. To this end we *randomly* divide the 360 subjects into a

training set $\boldsymbol{d}$ of size $N = 300$ and a validation set $\boldsymbol{d}_{\text{val}}$ of size $N_{\text{val}} = 60$. The popular "machine learning" prediction program *Random Forests*, Chapter 17, is applied. Random Forests forms $r_{\boldsymbol{d}}(x)$ by averaging the predictions of a large number of randomly subsampled regression trees (Section 8.4).

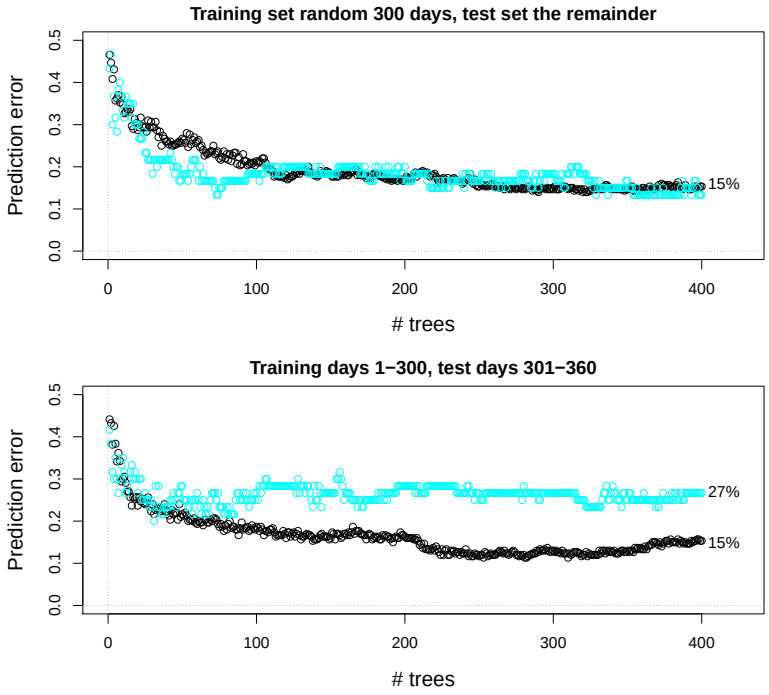

Figure 12.6 Test error (blue) and cross-validated training error (black), for Random Forest prediction rules using the imaginary medical study (12.81)–(12.82). *Top panel*: training set randomly selected 300 days, test set the remaining 60 days. *Bottom panel*: training set the first 300 days, test set the last 60 days.

The top panel of Figure 12.6 shows the results, with blue points indicating test-set error and black the (cross-validated) training-set error. Both converge to 15% as the number of Random Forest trees grows large. This seems to confirm an 85% success rate for prediction rule $r_{\boldsymbol{d}}(x)$.

One change has been made for the bottom panel: now the training set is the data for days 1 through 300, and the test set days 301 through 360.

The cross-validated training-set prediction error still converges to 15%, but $\widehat{\text{Err}}_{\text{val}}$ is now 27%, nearly double.

The reason isn't hard to see. Any predictive power must come from the transient episodes, which lose efficacy outside of their limited span. In the first example the test days are located among the training days, and inherit their predictive accuracy from them. This mostly fails in the second setup, where the test days are farther removed from the training days. (Only the orange bars crossing the 300-day line can help lower $\widehat{\text{Err}}_{\text{val}}$ in this situation.)

An obvious, but often ignored, dictum is that $\widehat{\text{Err}}_{\text{val}}$ is more believable if the test set is further separated from the training set. "Further" has a clear meaning in studies with a time or location factor, but not necessarily in general. For J-fold cross-validation, separation is improved by removing contiguous blocks of N/J cases for each group, rather than by random selection, but the amount of separation is still limited, making $\widehat{\text{Err}}_{\text{cv}}$ less believable than a suitably constructed $\widehat{\text{Err}}_{\text{val}}$.

The distinction between transient, ephemeral predictors and dependable ones is sometimes phrased as the difference between correlation and causation. For prediction purposes, if not for scientific exegesis, we may be happy to settle for correlations as long as they are persistent enough for our purposes. We return to this question in Chapter 15 in the discussion of large-scale hypothesis testing.

A notorious cautionary tale of fading correlations concerns *Google Flu*
†5 *Trends*,† a machine-learning algorithm for predicting influenza outbreaks. Introduced in 2008, the algorithm, based on counts of internet search terms, outperformed traditional medical surveys in terms of speed and predictive accuracy. Four years later, however, the algorithm failed, badly overestimating what turned out to be a nonexistent flu epidemic. Perhaps one lesson here is that the Google algorithmists needed a validation set years—not weeks or months—removed from the training data.

Error rate estimation is mainly frequentist in nature, but the very large data sets available from the internet have encouraged a disregard for inferential justification of any type. This can be dangerous. The heterogeneous nature of "found" data makes statistical principles of analysis more, not less, relevant.

12.5 Notes and Details

The evolution of prediction algorithms and their error estimates nicely illustrates the influence of electronic computation on statistical theory and

practice. The classical recipe for cross-validation recommended splitting the full data set in two, doing variable selection, model choice, and data fitting on the first half, and then testing the resulting procedure on the second half. Interest revived in 1974 with the independent publication of papers by Geisser and by Stone, featuring leave-one-out cross-validation of predictive error rates.

A question of bias versus variance arises here. A rule based on only $N/2$ cases is less accurate than the actual rule based on all N. Leave-one-out cross-validation minimizes this type of bias, at the expense of increased variability of error rate estimates for "jumpy" rules of a discontinuous nature. Current best practice is described in Section 7.10 of Hastie *et al.* (2009), where J-fold cross-validation with J perhaps 10 is recommended, possibly averaged over several random data splits.

Nineteen seventy-three was another good year for error estimation, featuring Mallows' C_p estimator and Akaike's information criterion. Efron (1986) extended C_p methods to a general class of situations (see below), established the connection with AIC, and suggested bootstrapping methods for covariance penalties. The connection between cross-validation and covariance penalties was examined in Efron (2004), where the Rao–Blackwell-type relationship mentioned at the end of Section 12.3 was demonstrated. The SURE criterion appeared in Charles Stein's 1981 paper. Ye (1998) suggested the general degrees of freedom definition (12.53).

†1 [p. 210] *Standard candles and dark energy.* Adam Riess, Saul Perlmutter, and Brian Schmidt won the 2011 Nobel Prize in physics for discovering increasing rates of expansion of the Universe, attributed to an Einsteinian concept of *dark energy*. They measured cosmic distances using Type Ia supernovas as "standard candles." The type of analysis suggested by Figure 12.1 is intended to improve the cosmological distance scale.

†2 [p. 222] *Data-based choice of a lasso estimate.* The regularization parameter λ for a lasso estimator (7.42) controls the number of nonzero coefficients of $\tilde{\beta}(\lambda)$, with larger λ yielding fewer nonzeros. Efron *et al.* (2004) and Zou *et al.* (2007) showed that a good approximation for the degrees of freedom df (12.53) of a lasso estimate is the number of its nonzero coefficients. Substituting this for p in (12.51) provides a quick version of $\widehat{\mathrm{Err}}$.. This was minimized at df $= 7$ for the supernova example in Figure 12.1 (12.54).

†3 [p. 222] *Stein's unbiased risk estimate.* The covariance formula (12.56) is obtained directly from integration by parts. The computation is clear from

the one-dimensional version of (12.55), $N = 1$:

$$\begin{aligned}\operatorname{cov}(\hat{\mu}, y) &= \int_{-\infty}^{\infty} \left[\frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{1}{2}\frac{(y-\mu)^2}{\sigma^2}} (y-\mu)\right] \hat{\mu}(y)\, dy \\ &= \sigma^2 \int_{-\infty}^{\infty} \left[\frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{1}{2}\frac{(y-\mu)^2}{\sigma^2}}\right] \frac{\partial \hat{\mu}(y)}{\partial y}\, dy \\ &= \sigma^2 E\left\{\frac{\partial \hat{\mu}(y)}{\partial y}\right\}.\end{aligned} \tag{12.85}$$

Broad regularity conditions for SURE are given in Stein (1981).

†4 [p. 227] *The .632 rule.* Bootstrap competitors to cross-validation are discussed in Efron (1983) and Efron and Tibshirani (1997). The most successful of these, the ".632 rule" is generally less variable than leave-one-out cross-validation. We suppose that nonparametric bootstrap data sets $\boldsymbol{d}^{*b}$, $b = 1, 2, \ldots, B$, have been formed, each by sampling with replacement N times from the original N members of $\boldsymbol{d}$ (12.1). Data set $\boldsymbol{d}^{*b}$ produces rule

$$r^{*b}(x) = r_{\boldsymbol{d}^{*b}}(x), \tag{12.86}$$

giving predictions

$$y_i^{*b} = r^{*b}(x_i). \tag{12.87}$$

Let $I_i^b = 1$ if pair (x_i, y_i) is *not* in $\boldsymbol{d}^{*b}$, and 0 if it is. (About $e^{-1} = 0.368$ of the $N \cdot B$ I_i^b will equal 1, the remaining 0.632 equaling 0.) The "out of bootstrap" estimate of prediction error is

$$\widehat{\mathrm{Err}}_{\mathrm{out}} = \sum_{i=1}^{N}\sum_{j=1}^{B} I_i^b\, D\left(y_i, \hat{y}_i^{*b}\right) \Big/ \sum_{i=1}^{N}\sum_{j=1}^{B} I_i^b, \tag{12.88}$$

the average discrepancy in the omitted cases.

$\widehat{\mathrm{Err}}_{\mathrm{out}}$ is similar to a grouped cross-validation estimate that omits about 37% of the cases each time. The .632 rule compensates for the upward bias in $\widehat{\mathrm{Err}}_{\mathrm{out}}$ by incorporating the downwardly biased apparent error (12.18),

$$\widehat{\mathrm{Err}}_{.632} = 0.632\, \widehat{\mathrm{Err}}_{\mathrm{out}} + 0.368\, \mathrm{err}\,. \tag{12.89}$$

$\widehat{\mathrm{Err}}_{\mathrm{out}}$ has resurfaced in the popular Random Forests prediction algorithm, Chapter 17, where a closely related procedure gives the "out of bag" estimate of Err.

†5 [p. 230] *Google Flu Trends.* Harford's 2014 article, "Big data: A big mistake?," concerns the enormous "found" data sets available in the internet age, and the dangers of forgetting the principles of statistical inference in their analysis. Google Flu Trends is his primary cautionary example.