Performance measure	Formula
mean-squared error	$\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{n}$
root mean-squared error	$\sqrt{\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{n}}$
mean absolute error	$\frac{\lvert p_1-a_1\rvert+\ldots+\lvert p_n-a_n\rvert}{n}$
relative squared error	$\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{(a_1-\bar{a})^2+\ldots+(a_n-\bar{a})^2}$, where $\bar{a}=\frac{1}{n}\sum_i a_i$
root relative squared error	$\sqrt{\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{(a_1-\bar{a})^2+\ldots+(a_n-\bar{a})^2}}$
relative absolute error	$\frac{\lvert p_1-a_1\rvert+\ldots+\lvert p_n-a_n\rvert}{\lvert a_1-\bar{a}\rvert+\ldots+\lvert a_n-\bar{a}\rvert}$
correlation coefficient	$\frac{S_{PA}}{\sqrt{S_P S_A}}$, where $S_{PA}=\frac{\sum_i(p_i-\bar{p})(a_i-\bar{a})}{n-1}$, $S_p=\frac{\sum_i(p_i-\bar{p})^2}{n-1}$, and $S_A=\frac{\sum_i(a_i-\bar{a})^2}{n-1}$

* *p* are predicted values and *a* are actual values.

relation, to −1 when the results are perfectly correlated negatively. Of course, negative values should not occur for reasonable prediction methods. Correlation is slightly different from the other measures because it is scale independent in that, if you take a particular set of predictions, the error is unchanged if all the predictions are multiplied by a constant factor and the actual values are left unchanged. This factor appears in every term of S_{PA} in the numerator and in every term of S_P in the denominator, thus canceling out. (This is not true for the relative error figures, despite normalization: if you multiply all the predictions by a large constant, then the difference between the predicted and the actual values will change dramatically, as will the percentage errors.) It is also different in that good performance leads to a large value of the correlation coefficient, whereas because the other methods measure error, good performance is indicated by small values.

Which of these measures is appropriate in any given situation is a matter that can only be determined by studying the application itself. What are we trying to minimize? What is the cost of different kinds of error? Often it is not easy to decide. The squared error measures and root squared error measures weigh large discrepancies much more heavily than small ones, whereas the absolute error measures do not. Taking the square root (root mean-squared error) just reduces the figure to have the same dimensionality as the quantity being predicted. The relative error figures try to compensate for the basic predictability or unpredictability of the output variable: if it tends to lie fairly close to its average value, then you expect prediction to be good and the relative figure compensate for this. Otherwise, if the error figure in one situation is far greater than that in another situation, it may be because the quantity in the first situation is inherently more variable and therefore harder to predict, not because the predictor is any worse.

Table 5.9 **Performance measures for four numeric prediction models.**

	A	B	C	D
root mean-squared error	67.8	91.7	63.3	57.4
mean absolute error	41.3	38.5	33.4	29.2
root relative squared error	42.2%	57.2%	39.4%	35.8%
relative absolute error	43.1%	40.1%	34.8%	30.4%
correlation coefficient	0.88	0.88	0.89	0.91

Fortunately, it turns out that in most practical situations the best numeric prediction method is still the best no matter which error measure is used. For example, Table 5.9 shows the result of four different numeric prediction techniques on a given dataset, measured using cross-validation. Method D is the best according to all five metrics: it has the smallest value for each error measure and the largest correlation coefficient. Method C is the second best by all five metrics. The performance of methods A and B is open to dispute: they have the same correlation coefficient, method A is better than method B according to both mean-squared and relative squared errors, and the reverse is true for both absolute and relative absolute error. It is likely that the extra emphasis that the squaring operation gives to outliers accounts for the differences in this case.

When comparing two different learning schemes that involve numeric prediction, the methodology developed in Section 5.5 still applies. The only difference is that success rate is replaced by the appropriate performance measure (e.g., root mean-squared error) when performing the significance test.

5.9 The minimum description length principle

What is learned by a machine learning method is a kind of "theory" of the domain from which the examples are drawn, a theory that is predictive in that cost values at the left and right sides of the graph are *fp* and *fn*, just as they are for the error curve, so you can draw the cost curve for any classifier very easily.

Figure 5.4(b) also shows classifier B, whose expected cost remains the same across the range—that is, its false positive and false negative rates are equal. As you can see, it outperforms classifier A if the probability cost function exceeds about 0.45, and knowing the costs we could easily work out what this corresponds to in terms of class distribution. In situations that involve different class distributions, cost curves make it easy to tell when one classifier will outperform another.

In what circumstances might this be useful? To return to the example of predicting when cows will be in estrus, their 30-day cycle, or 1/30 prior probability, is unlikely to vary greatly (barring a genetic cataclysm!). But a particular herd may have different proportions of cows that are likely to reach estrus in any given week, perhaps synchronized with—who knows?—the phase of the moon. Then, different classifiers would be appropriate at different times. In the oil spill example, different batches of data may have different spill probabilities. In these situations cost curves can help to show which classifier to use when.

Each point on a lift chart, ROC curve, or recall–precision curve represents a classifier, typically obtained using different threshold values for a method such as Naïve Bayes. Cost curves represent each classifier using a straight line, and a suite of classifiers will sweep out a curved envelope whose lower limit shows how well that type of classifier can do if the parameter is well chosen. Figure 5.4(b) indicates this with a few gray lines. If the process were continued, it would sweep out the dotted parabolic curve.

The operating region of classifier B ranges from a probability cost value of about 0.25 to a value of about 0.75. Outside this region, classifier B is outperformed by the trivial classifiers represented by dashed lines. Suppose we decide to use classifier B within this range and the appropriate trivial classifier below and above it. All points on the parabola are certainly better than this scheme. But how much better? It is hard to answer such questions from an ROC curve, but the cost curve makes them easy. The performance difference is negligible if the probability cost value is around 0.5, and below a value of about 0.2 and above 0.8 it is barely perceptible. The greatest difference occurs at probability cost values of 0.25 and 0.75 and is about 0.04, or 4% of the maximum possible cost figure.

5.8 Evaluating numeric prediction

All the evaluation measures we have described pertain to classification situations rather than numeric prediction situations. The basic principles—using an independent test set rather than the training set for performance evaluation, the

holdout method, and cross-validation—apply equally well to numeric prediction. But the basic quality measure offered by the error rate is no longer appropriate: errors are not simply present or absent; they come in different sizes.

Several alternative measures, summarized in Table 5.8, can be used to evaluate the success of numeric prediction. The predicted values on the test instances are $p_1, p_2, \ldots, p_n$; the actual values are $a_1, a_2, \ldots, a_n$. Notice that p_i means something very different here from what it did in the last section: there it was the probability that a particular prediction was in the *i*th class; here it refers to the numeric value of the prediction for the *i*th test instance.

Mean-squared error is the principal and most commonly used measure; sometimes the square root is taken to give it the same dimensions as the predicted value itself. Many mathematical techniques (such as linear regression, explained in Chapter 4) use the mean-squared error because it tends to be the easiest measure to manipulate mathematically: it is, as mathematicians say, "well behaved." However, here we are considering it as a performance measure: all the performance measures are easy to calculate, so mean-squared error has no particular advantage. The question is, is it an appropriate measure for the task at hand?

Mean absolute error is an alternative: just average the magnitude of the individual errors without taking account of their sign. Mean-squared error tends to exaggerate the effect of outliers—instances whose prediction error is larger than the others—but absolute error does not have this effect: all sizes of error are treated evenly according to their magnitude.

Sometimes it is the *relative* rather than *absolute* error values that are of importance. For example, if a 10% error is equally important whether it is an error of 50 in a prediction of 500 or an error of 0.2 in a prediction of 2, then averages of absolute error will be meaningless: relative errors are appropriate. This effect would be taken into account by using the relative errors in the mean-squared error calculation or the mean absolute error calculation.

Relative squared error in Table 5.8 refers to something quite different. The error is made relative to what it would have been if a simple predictor had been used. The simple predictor in question is just the average of the actual values from the training data. Thus relative squared error takes the total squared error and normalizes it by dividing by the total squared error of the default predictor.

The next error measure goes by the glorious name of *relative absolute error* and is just the total absolute error, with the same kind of normalization. In these three relative error measures, the errors are normalized by the error of the simple predictor that predicts average values.

The final measure in Table 5.8 is the *correlation coefficient*, which measures the statistical correlation between the *a*'s and the *p*'s. The correlation coefficient ranges from 1 for perfectly correlated results, through 0 when there is no cor-