

# Characterizing the generalization performance of model selection strategies

**Dale E. Schuurmans**<sup>\*,†</sup>

Institute for Research in Cognitive Science

**Lyle H. Ungar**

Department of Computer and Information Science

**Dean P. Foster**

Department of Statistics

University of Pennsylvania

Philadelphia, PA 19104

**Abstract:** We investigate the structure of model selection problems via the bias/variance decomposition. In particular, we characterize the essential structure of a model selection task by the bias and variance profiles it generates over the sequence of hypothesis classes. This leads to a new understanding of complexity-penalization methods: First, the penalty terms in effect postulate a particular profile for the variances as a function of model complexity—if the postulated and true profiles do not match, then systematic under-fitting or over-fitting results, depending on whether the penalty terms are too large or too small. Second, it is usually best to penalize according to the true variances of the task, and therefore no fixed penalization strategy is optimal across all problems. We then use this bias/variance characterization to identify the notion of easy and hard model selection problems. In particular, we show that if the variance profile grows too rapidly in relation to the biases then standard model selection techniques become prone to significant errors. This can happen for example in regression when the independent variables are drawn from wide-tailed distributions. Finally, we discuss a new model selection strategy that dramatically outperforms standard complexity-penalization and hold-out methods on these hard tasks.

**Keywords:** inductive learning, model selection, overfitting, bias/variance decomposition

---

\***Also:** NEC Research Institute, Princeton, NJ

†**Mailing address:** Dale Schuurmans, Institute for Research in Cognitive Science, University of Pennsylvania, 3401 Walnut Street, Suite 400A, Philadelphia, PA 19104-6228. **Email:** daes@linc.cis.upenn.edu **Phone:** 1-215-573-6285 **Fax:** 1-215-573-9247

# Characterizing the generalization performance of model selection strategies

**Abstract:** We investigate the structure of model selection problems via the bias/variance decomposition. In particular, we characterize the essential structure of a model selection task by the bias and variance profiles it generates over the sequence of hypothesis classes. This leads to a new understanding of complexity-penalization methods: First, the penalty terms in effect postulate a particular profile for the variances as a function of model complexity—if the postulated and true profiles do not match, then systematic under-fitting or over-fitting results, depending on whether the penalty terms are too large or too small. Second, it is usually best to penalize according to the true variances of the task, and therefore no fixed penalization strategy is optimal across all problems. We then use this bias/variance characterization to identify the notion of easy and hard model selection problems. In particular, we show that if the variance profile grows too rapidly in relation to the biases then standard model selection techniques become prone to significant errors. This can happen for example in regression when the independent variables are drawn from wide-tailed distributions. Finally, we discuss a new model selection strategy that dramatically outperforms standard complexity-penalization and hold-out methods on these hard tasks.

## 1 Introduction

When learning a function  $h : X \rightarrow Y$  from random training examples  $\langle x_1, y_1 \rangle, \dots, \langle x_t, y_t \rangle$ , there is a well-known tradeoff between the size of the training sample and the complexity of the function class being considered: If the class is too complex for the sample size, there is a risk of "overfitting" the training data and guessing a function that performs poorly on future test examples. On the other hand, an overly restricted class can prevent us from considering any good candidate functions. The most common strategy for coping with this dilemma in practice is to use some form of automatic *model selection* such as complexity-penalization or repeated hold-out testing to balance the tradeoff between complexity and fit to the data.

Under the simplest formulation of model selection, the idea is to first stratify the hypothesis class  $H$  into a sequence of nested subclasses  $H_0 \subset H_1 \subset \dots = H$  and then (somehow) choose a class which has the appropriate complexity for the given training data. To understand how we might make this choice, note that for a given

training sample  $S = \langle x_1, y_1 \rangle, \dots, \langle x_t, y_t \rangle$  we obtain a corresponding sequence of empirically optimal functions  $h_0^*, h_1^*, \dots$ , one from each subclass, that achieve minimum error on  $S$ . The essence of the model selection problem is to choose one of these functions based on their observed empirical errors  $err(h_1^*, S), err(h_2^*, S), \dots$ . Note however that these errors are monotonically decreasing, and therefore choosing the function with minimum training error simply leads to choosing a function from the largest class. Therefore the trick is to invoke some other criteria beyond empirical error minimization to make this choice.

Currently, two basic model selection strategies predominate. The most common strategy is *complexity-penalization*. Here we assign increasing complexity values  $c_0, c_1, \dots$  to the successive function classes, and then choose the hypothesis from  $h_1^*, h_2^*, \dots$  that minimizes some prior combination of complexity and empirical error (e.g., the additive combination  $c_i + \lambda err(h_i^*, S)$ ). There are many variants of this basic approach, including generalized cross validation [2], minimum description length principle [10], structural risk minimization [12, 13], “Bayesian” maximum a posteriori selection, and regularization [8]. These strategies differ in the specific complexity values they assign and the particular tradeoff function they optimize, but the basic idea is still the same.

The other most common strategy is *hold-out testing*. Here one asks: for the given set of training data, which hypothesis class  $H_i$  generalizes best? This is answered by partitioning the training set,  $1, \dots, t$ , into a pseudo-training set,  $1, \dots, k$ , and a hold-out test set,  $k + 1, \dots, t$ , and then using the pseudo-training set to obtain a sequence of pseudo-hypotheses  $\hat{h}_0, \hat{h}_1, \dots$ , etc. The hold-out test set can then be used to obtain an *unbiased* estimate of the true errors of these pseudo-hypotheses. (Note that the training set errors tend to be gross underestimates in general.) From these unbiased estimates, one simply chooses the hypothesis class  $H_i$  that yields the pseudo-hypothesis  $\hat{h}_i$  with the smallest estimated error. Once  $H_i$  has been selected, we return the function  $h_i^* \in H_i$  that obtains minimum empirical error on the *entire* training sequence. Again, there are many variants to this basic strategy—having to do with repeating the pseudo-train pseudo-test split many times and averaging the results to choose the final hypothesis class; e.g., 10-fold cross validation, leave-one-out testing, bootstrapping, etc. [3, 14].

The abundance of model selection strategies and different approaches to the problem raises the question of which techniques are best and when. We attempt to answer this question by appealing to the standard bias/variance decomposition of generalization error [4]. In particular, we characterize model selection problems by the bias and variance profiles they generate over the sequence of hypothesis classes.

Given this characterization, we address a number of topics regarding the behavior of model selection strategies and the structure of model selection tasks: First, we investigate complexity-penalization methods, which attempt to directly adjust the empirical error estimates to account for the unseen variances. Here we observe that no single penalization strategy dominates in every task—all penalization methods have conditions where they perform well and conditions where they fail. Next, we investigate the structure of model selection *problems*, and identify the notion of easy versus hard model selection tasks. Here we show that some problems are inherently more difficult than others for standard complexity-penalization and hold-out methods. Given the inadequacy of standard techniques in these cases, we finally discuss a new model selection procedure that significantly outperforms standard approaches on these hard tasks. Throughout, we establish these points via controlled simulation studies, but also suggest directions towards establishing more rigorous theoretical results.

## 2 Bias/variance decomposition

This paper will focus on least squares regression problems. Here the goal is to learn a prediction function  $h : X \rightarrow \mathbb{R}$  that minimizes the squared difference between predicted  $\hat{y}$  and true  $y$ -values, as specified by the loss function  $err(\hat{y}, y) = (\hat{y} - y)^2$ . The prototypical approach to this problem is to first conjecture a suitable class of hypothesis functions  $H$  (e.g., by specifying a neural net architecture, or some other representation class), and then choose the hypothesis  $h^* \in H$  that minimizes the empirical error  $err(h^*, S) \triangleq \sum_{j=1}^t err(h^*(x_j), y_j)$  on the training set  $S$ . Of course, the key to making this work is to choose the right hypothesis class  $H$ .

One way to assess  $H$ 's suitability is to consider the bias/variance decomposition of the resulting prediction error. Consider a fixed example distribution  $P_{XY}$  and training sample size  $t$ . Following Geman *et al.* [4] we can decompose the expected error of the empirically optimal hypothesis  $h^*$  into bias and variance components as follows: First note that each training set  $S$  determines some hypothesis  $h^*$  with minimum error on  $S$ . Therefore, from the distribution over length  $t$  training sequences,  $P_{XY}^t$ , we obtain a corresponding distribution over hypotheses,  $P_H$ . Now notice that each hypothesis  $h^*$  has a true expected error with respect to the example distribution  $P_{XY}$ , given by  $err(h^*) \triangleq \int_X \int_Y (h^*(x) - y)^2 dP_{Y|x} dP_X$ . Thus, the distribution over hypotheses generates a corresponding distribution over true errors. The resulting

expected true error,  $E_{h^*} \text{err}(h^*) \triangleq \int_H \text{err}(h^*) dP_H$ , can be decomposed as

$$E_{h^*} \text{err}(h^*) = \overset{\text{“bias”}}{\text{err}(\bar{h}^*)} + \overset{\text{“variance”}}{E_{h^*} \text{err}(\bar{h}^*, h^*)}, \quad (1)$$

where  $\bar{h}^*$  is the mean hypothesis of the distribution  $P_H$ , and  $\text{err}(\bar{h}^*, h^*) \triangleq \int_X (\bar{h}^*(x) - h^*(x))^2 dP_X$  is the average discrepancy between the empirically optimal hypothesis  $h^*$  and  $\bar{h}^*$  (cf. [4]). Thus, we decompose the expected hypothesis error into two components: the true error of the mean hypothesis (bias), and the average discrepancy between a random data generated hypothesis and the mean hypothesis (variance).<sup>1</sup>

Now consider the model selection task. For a given instance of a model selection problem we are given a nested *sequence* of hypothesis classes  $H_0 \subset H_1 \subset \dots$ , and faced with a particular example distribution  $P_{XY}$  and training sample size  $t$ . Note that for fixed  $P_{XY}$  and  $t$  we obtain particular bias and variance values,  $b_i$  and  $v_i$ , for each hypothesis class  $H_i$ . Thus, each instance of a model selection problem yields a particular *profile* of biases and variances over the sequence of hypothesis classes  $H_1, H_2, \dots$ . Intuitively, we expect the variance terms to increase for larger hypothesis classes, as there are a wider variety of functions that give similar fits to the data. On the other hand, we expect the bias terms to decrease as we are better able to approximate the optimal regression for the given distribution. A model selection strategy needs to infer how the combination of bias + variance behaves, based on the structure of  $H_1 \subset H_2 \subset \dots$  and the training set errors  $\text{err}(h_1^*, S), \text{err}(h_2^*, S), \dots$

By adopting the perspective that these bias and variance profiles capture the essential aspects of the task, we are able to make several useful predictions about the behavior of model selection strategies, as well as characterize the difficulty of model selection problems—based solely on the shapes of these bias and variance profiles, and disregarding other aspects of the problem.

### 3 Performance of penalization strategies

We begin by investigating the behavior of complexity-penalization strategies. Recall that for a training sample  $S$  and corresponding hypothesis sequence  $h_1^*, h_2^*, \dots$ , a penalization strategy will choose the hypothesis  $h_i^*$  that minimizes some combination of class complexity  $c_i$  and empirical error  $\text{err}(h_i^*, S)$ . The point is that the empirical errors  $\text{err}(h_i^*, S)$  tend to be gross underestimates of  $\text{err}(h_i^*)$  in general

---

<sup>1</sup>Note that in many situations (e.g., classical linear regression) the mean hypothesis  $\bar{h}^*$  actually corresponds to the hypothesis in  $H$  with minimum true error.

(since the  $h^*$  are explicitly chosen to minimize the error on  $S$ ), and the degree of underestimation tends to become worse at higher complexity levels. Complexity-penalization, therefore, seeks to adjust the empirical error estimates to compensate for this fact. This results in a generic model selection strategy where one first penalizes the empirical errors to obtain better estimates

$$\widehat{err}_{pen}(h_i^*) = err(h_i^*, S) + penalty_i, \quad (2)$$

and then chooses the hypothesis  $h_i^*$  with the smallest adjusted estimate  $\widehat{err}_{pen}(h_i^*)$ .

As mentioned, there are many variants of this strategy, but to illustrate our main points it will suffice to consider two strategies that embody distinct penalization policies. To describe these strategies, let  $r = i/t$  be the number of complexity levels being considered per training example.<sup>2</sup> The first penalization strategy we consider is Generalized Cross Validation **GCV** [2]. Following [9] we can write the adjusted error estimate of this strategy as

$$\widehat{err}_{GCV}(h_i^*) = err(h_i^*, S) + \frac{2r - r^2}{(1 - r)^2} err(h_i^*, S). \quad (3)$$

The other penalization strategy we consider is Vapnik’s Structural Risk Minimization procedure **SRM** [13], which following [1] can be formulated

$$\widehat{err}_{SRM}(h_i^*) = err(h_i^*, S) + \frac{\sqrt{\tilde{r}}}{(1 - \sqrt{\tilde{r}})_+} err(h_i^*, S), \quad (4)$$

where  $\tilde{r} = r(1 + \ln 1/r) + (\ln t)/2t$ . For our purposes, the key difference between these two policies is that **SRM** uses a much steeper penalization profile than **GCV**. (This can be seen in Figures 2–3 below.)

Now reconsider the bias/variance characterization developed above. This offers an interesting interpretation of complexity-penalization methods—which can be seen by directly comparing equations (1) and (2) and noting that the first terms can be naturally aligned. Notice here that, although the empirical error  $err(h_i^*, S)$  is normally considered to be an estimate of  $h_i^*$ ’s error, we can alternatively view  $err(h_i^*, S)$  as an estimate of the error of the *mean* hypothesis for the class,  $\bar{h}_i^*$ . In

---

<sup>2</sup>For most natural orderings  $H_1 \subset H_2 \dots$ , the complexity level  $i$  corresponds to the number of free parameters used in the definition of function class  $H_i$ . Therefore, intuitively  $r$  gives the number of distinct parameters being estimated per training example [1, 13].

fact,  $err(h_i^*, S)$  is often a much *better* estimate of  $err(\bar{h}_i^*)$  than it is of  $err(h_i^*)$ !<sup>3</sup> Although not often explicitly made, this elementary observation leads to an interesting interpretation of complexity-penalization strategies: if the empirical error term  $err(h_i^*, S)$  accurately estimates the bias term for class  $H_i$ , then the *penalty* <sub>$i$</sub>  term must be accounting for the unobserved *variance* of  $H_i$ . Thus, we can interpret the sequence of penalty terms,  $penalty_1, penalty_2, \dots$ , as in effect postulating a particular profile of variance terms for the classes  $H_1, H_2, \dots$ . So for example, a steep penalization profile encodes the assumption that the variances grow rapidly as a function of the complexity level  $i$ , whereas a flat profile asserts that the variances grow more slowly. This observation leads to a series of specific predictions about the behavior of complexity-penalization strategies: (1) if the penalization profile is much steeper than the true variance profile, we expect systematic *underfitting* since the latter hypotheses will be over-penalized relative to the true variances; (2) on the other hand, if the penalization profile is much flatter than the true variance profile, we expect systematic *overfitting* since the latter hypotheses will be under-penalized; and finally (3) we expect good generalization performance if the penalty profile matches the true variance profile for the task.

**Experiment** To test these hypotheses we ran a series of experiments to investigate the behavior of GCV and SRM on model selection tasks with different bias and variance profiles. Recall that GCV and SRM propose very different penalization policies and therefore we expect them to behave quite differently as we vary the task structure. To conduct our experiments we considered a traditional linear regression problem where the goal is to learn a linear function  $h(x_1, \dots, x_n) = a_1x_1 + \dots + a_nx_n$  that minimizes the mean squared error on an unknown  $P_{XY}$ . In this context, a natural model selection task arises by considering the nested sequence of function classes  $H_1 \subset H_2 \subset \dots$  defined by the first 1, 2, ... variables respectively (which assumes in effect that the variables have been ordered by importance). To design test problems, we set  $n = 10$ ,  $t = 20$ , and considered a series of distributions  $P_{XY}$  that yield different bias and variance profiles for the task.<sup>4</sup> For these tasks, we evaluated model selection strategies by measuring the *ratio* of the true error of the hypotheses  $h_i^*$  they

<sup>3</sup>Note that for any class  $H$  such that  $err(h^*, S) \rightarrow err(h^*)$  for all  $P_{XY}$  and  $\bar{h}$  is well-defined, we also have  $err(h^*, S) \rightarrow err(\bar{h}^*)$ . This is in fact quite easy to prove using the uniform convergence results of Vapnik [12, 13]. The interesting part of this observation is that  $err(h^*, S)$  evidently converges much faster to  $err(\bar{h}^*)$  than to  $err(h^*)$ . Currently, this is largely an empirical observation, but we are pursuing a theoretical analysis that quantifies these differing *rates* of convergence.

<sup>4</sup>Specifically, we used distributions defined by a simple additive model  $Y = \alpha_1X_1 + \dots + \alpha_nX_n + \varepsilon$ , where the  $X_i$ 's and  $\varepsilon$  are independent and  $\varepsilon \sim N(0, \sigma^2)$ . We generated  $X_i$ 's by a Cauchy(0, 1) distribution, which was then truncated (and renormalized) at  $(-\beta_i, +\beta_i)$  for different

chose to the true error of the best hypothesis in the sample-dependent sequence  $h_1^*, h_2^*, \dots$ . (The rationale for doing this is that we wish to measure the selection strategy’s ability to approximate the best hypothesis in the given sequence—not find a better function from outside the sequence.) We ran our experiments by fixing a distribution  $P_{XY}$ , repeatedly generated training samples of size  $t = 20$ , and recording the ratio of chosen to best-in-sequence errors achieved by each strategy. This was repeated 500 times to estimate the performance of the model selection strategies, as well as to estimate the bias/variance characteristics of the given problem.

The first problem we considered, shown in Figure 2a, was designed to have a *flat* variance profile comparable in size to the bias profile.<sup>5</sup> Here we expect GCV to outperform SRM, since its penalization profile more closely matches the true variance profile of this task (Figure 2). In fact our results show exactly this. Table 1 shows that GCV significantly outperforms SRM at this task, obtaining a mean approximation ratio of 1.8 over 500 trials, compared to 2.8 obtained by SRM.<sup>6</sup> That is, GCV chose a function from the sample-determined sequence  $h_1^*, h_2^*, \dots$  that had a true error 1.8 times larger than the best true error of any function in the sequence, on average. Moreover, GCV chose functions at complexity levels that were close to the optimum complexity levels for the given training sets—the last column of Table 1 shows that GCV underestimated the best complexity level by only 0.7 on average. For this problem SRM significantly underfit the data, choosing function complexities that were 5.6 levels smaller than optimum complexity on average. These results support our predictions based on the variance and penalization profiles involved.

We next considered a problem that had a much steeper variance profile, more closely resembling the penalization profile of SRM; see Figure 2b.<sup>7</sup> In sharp contrast to the previous results, Table 2 shows that SRM significantly outperforms GCV in this case, achieving a mean approximation ratio of 2.1 versus GCV’s mean ratio

---

choices of  $\beta_i$ . We also set the linear model coefficients to be  $\alpha_i = 1/\beta_i$ , to normalize the  $X_i$  variances. Thus, our test distributions  $P_{XY}$  were determined by  $\sigma$  and the truncation constants  $\beta_1, \dots, \beta_{10}$ .

The reason for using these Cauchy-like distributions instead of more conventional Gaussians is that we wished to construct *difficult* model selection problems. That is, wide-tailed distributions like Cauchy create difficult variable selection problems, because small training samples in this case will not accurately capture the significant range of  $X_i$  values that will be observed in testing. Therefore small errors in  $\hat{\alpha}_i$  result in hypotheses with huge test set errors, since we evaluate these functions on large unobserved  $X_i$  values. In this way we achieve a large *variance* between hypotheses; as desired.

<sup>5</sup>This problem was defined by setting  $\sigma = 0.5$ ,  $\beta_i = 10$ , as described in Footnote 4.

<sup>6</sup>The other strategies mentioned in Table 1 are explained below.

<sup>7</sup>This problem was defined by setting  $\sigma = 0.5$ ,  $\beta_i = 10 \times 2^{i-1}$ , as described in Footnote 4.

of 24.3. The last column in Table 2 also shows that GCV now overshoots the best complexity level by an average of 0.4 on this problem, which leads to devastating consequences given the nature of the variance profile for this task.

These results show that there is no *best* penalization method in general. The performance one obtains depends on how closely the penalization profile of the strategy matches the true variances of the task. If the penalty terms are much larger than the variances, we obtain systematic underfitting; whereas if the penalties are much smaller, we systematically overfit the data.<sup>8</sup> Not surprisingly, directly penalizing by the true variances of the task (Procedure VAR) always seems to yield good performance for any slope of variance profile. *E.g.*, Tables 1 and 2 show that VAR performs well on Problems 1 and 2, even though the variance profiles behave quite differently in these two cases. Of course, VAR avoids systematic over/underfitting by using different penalization profiles for each problem, and is certainly not achievable in practice. However, these results suggest that one should try to directly penalize according to the true variances of the task as much as possible.<sup>9</sup>

Overall, these results suggest that one can interpret penalizers as asserting a particular structure for the problem: The postulated penalization profile makes a specific assumption about the behavior of the variances in the given task. The obvious conclusion is that one should set the penalty terms according to whatever prior knowledge one has about the variance profile for the task at hand. Accurate assumptions tend to yield excellent generalization performance, whereas inaccurate assumptions lead to poor performance. However, we will see that there are situations where one might not want to use penalty methods in any event.

## 4 Difficulty of model selection problems

The bias/variance decomposition can also be used to characterize the notion of *hard* versus *easy* model selection problems. Specifically, in terms of our previous definitions, we find that if the variance profile is flat (grows slowly and is not large in comparison to the bias profile) then almost any reasonable penalization strategy will do reasonably well. On the other hand, if the variance profile grows explosively relative to the bias profile, then disaster results for any penalization strategy that does not use the exact variance profile for the task.

---

<sup>8</sup>Note that this is similar to an observation made by Kearns *et al.* [5] in the context of learning classifications. However, they do not explicitly invoke a bias/variance characterization of model selection problems to explain their results.

<sup>9</sup>We note that VAR performed well across a much wider range of tasks than reported here.

**Experiment** To demonstrate the distinction between easy and hard problems, we conducted a series of experiments in the same setup as before. The first case we considered was a model selection problem which had a *low* variance profile in relation to the bias terms (Figure 3a).<sup>10</sup> The point here is that we expect such a problem to be easy for most reasonable selection strategies, since the variances play a minor role and there are no serious consequences to minor over or under-fitting. Table 3 demonstrates the relatively benign behavior of the penalization strategies on this task (although the variance profile distinctly favors GCV in this case and this is reflected in the results).<sup>11</sup> However, to contrast with this, we next considered a problem (Figure 3b) which had a variance profile that grows explosively in complexity of the hypothesis class<sup>12</sup> We expect this to give a *hard* model selection problem because of the drastic consequences that would befall even minor overfitting. Table 4 shows that both GCV and SRM fail badly at this task. Both strategies make *catastrophic* mistakes from time to time, and choose hypotheses that are many orders of magnitude worse than the best available. Interestingly, Procedure VAR, which penalizes according to the true variances of the task, still works reasonably well in this case. Table 4 shows that VAR avoids the catastrophes suffered by GCV and SRM. Of course, VAR is not a practically realizable strategy. Model selection problems of this form seem to be inherently difficult for penalization strategies in general. (We tried many other well known penalization strategies, and obtained uniformly poor results on this problem.)

These results lead us to conclude that complexity-penalization can be an inherently risky strategy. There seems to be a potential for disaster whenever the task happens to be hard; *i.e.*, whenever the variance profile grows explosively in an unpredictable manner. The only way to avoid catastrophe in these situations (without mindlessly under-fitting) seems to be the omnipotent solution of knowing the true variances of the task *a priori*, which is highly unlikely for most interesting applications.

---

<sup>10</sup>This problem was defined by setting  $\sigma = 0.1$ ,  $\beta_i = i$ , as described in Footnote 4.

<sup>11</sup>It might seem surprising at first that the penalty and variance terms can actually *decrease* on these problems, as shown in Figure 3a. However, this is a consequence of the fact that the penalty terms depend on the training set errors, which can decrease faster than the multiplicative adjustments used by GCV and SRM. For the variances, the easiest way to see how they can decrease is to imagine a case where  $Y$  is a deterministic linear function of the variables. Here, any linearly independent set of training examples determines the target function exactly, and thus we would observe *zero* variance if given a linearly independent set (which could happen with probability 1 for  $t \geq n$ ). The additive noise component  $\varepsilon$  has the effect of monotonically increasing these variances, in counterbalance.

<sup>12</sup>This problem was defined by setting  $\sigma = 1$ ,  $\beta_i = 10^i$ , as described in Footnote 4.

**Alternative hold-out methods** An obvious idea in these situations is to consider alternative hold-out–based methods, like 10-fold cross-validation (10CV) or some other resampling procedure [6, 14]. However, it turns out that these strategies are prone to the very same mistakes suffered by penalty-based methods, as Table 4 clearly demonstrates for 10CV. The strikingly bad performance obtained by all standard model selection methods on these difficult tasks raises the question of whether it is possible to do better on hard problems, or whether we have to live with the potential of making disastrous mistakes.

## 5 A new model selection technique

In a recent paper (submitted to AAAI’97) one of the authors introduces a new strategy for model selection that takes a fundamentally different approach to the problem than previous techniques. This new strategy seems to avoid many of the catastrophic overfitting errors that plague standard complexity-penalization and hold-out methods on difficult model selection tasks. This procedure implicitly attempts to estimate the variance of a function class  $H_j$  by examining how  $h_j^*$  compares to  $h_i^*$  for  $i < j$ .

The basic idea behind this new strategy is to exploit the intrinsic geometry of the function learning task which arises from a simple statistical model of the problem: Assume the training and test examples are independent random observations drawn from a joint distribution  $P_{XY}$  on  $X \times Y$ . Then we can decompose this distribution into the conditional distribution of  $Y$  given  $X$ ,  $P_{Y|X}$ , and the marginal distribution  $P_X$  on  $X$ . Note that when learning a function  $h : X \rightarrow Y$  we are really only interested in approximating the conditional  $P_{Y|X}$ . However our approach is to exploit knowledge about  $P_X$  to help us make better decisions about which hypothesis  $h$  to choose. In fact, for now, assume that we actually *know*  $P_X$  and see how far this gets us. (Note that any information we require about  $P_X$  can be obtained from *unlabeled* training examples.)

How can knowing  $P_X$  help? Well, the first thing it does is give us a natural measure of the “distance” between two hypotheses  $h$  and  $g$ . In fact, we can obtain a natural (pseudo) *metric* on the space of hypotheses via the definition  $d(h, g) = (\int_X (h(x) - g(x))^2 dP_X)^{1/2}$ ; that is, we measure the distance between two functions by their root mean squared difference. Moreover, we can extend this definition to include the target conditional  $P_{Y|X}$  via the definition  $d(h, P_{Y|X}) = (\int_X \int_Y (h(x) - y)^2 dP_{Y|x} dP_X)^{1/2}$ ; which means that we can interpret the true error of a function  $h$  as the *distance* between  $h$  and the target object  $P_{Y|X}$ . Importantly, these definitions are compatible in

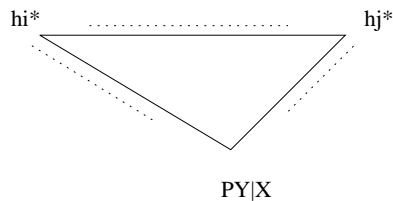


Figure 1: The real and estimated distances between successive hypotheses  $h_i$  and  $h_j$  and the target  $P_{Y|X}$ . Solid lines indicate real distances, dashed lines indicate empirical distance estimates.

the sense that the defined distance measure  $d$  satisfies the standard (pseudo) metric axioms over  $H \cup \{P_{Y|X}\}$ . This now gives us a nice geometric view of the problem: We have a nested sequence of spaces  $H_0 \subset H_1 \subset \dots$ , each with a closest function  $h_0, h_1, \dots$  to the target  $P_{Y|X}$ , where the distances are decreasing. However, we do not observe these real distances. Rather, we are given a training sample and have to choose from the sequence of *empirically* closest functions  $h_0^*, h_1^*, \dots$ , which have monotonically decreasing estimates  $d(\widehat{h}, P_{Y|X}) = \left(\sum_{j=1}^t (h(x_j) - y_j)^2 / t\right)^{1/2}$ . The key point is that we now have more information at our disposal: not only do we have estimated distances to  $P_{Y|X}$ , we now know the true distances *between* functions in the sequence!

Our idea is to use this additional information to choose a better hypothesis. Observe that we are dealing with two metrics here: the true metric  $d$  defined by the joint distribution  $P_{XY}$  and an empirical metric  $\hat{d}$  determined by the labeled training sequence. Given these two metrics, consider the triangle formed by two hypotheses  $h_i^*$  and  $h_j^*$  and the target conditional  $P_{Y|X}$  (Figure 1). Notice that there are six distances involved, three real and three estimated—of which the true distances to  $P_{Y|X}$  are the only two we care about, and yet these are the only two we don't have! The key observation though is that the real and estimated distances between hypotheses  $d(h_i^*, h_j^*)$  and  $d(\widehat{h_i^*}, \widehat{h_j^*})$  give us an *observable* relationship between  $d$  and  $\hat{d}$  in the local vicinity. In fact, we can adopt the naive assumption that observed relationship between  $h_i^*$  and  $h_j^*$  also holds between  $h_j^*$  and  $P_{Y|X}$ . Note that if this were the case, we would obtain a better estimate of  $d(h_j^*, P_{Y|X})$  simply by adjusting the training set distance  $d(\widehat{h_j^*}, P_{Y|X})$  according to the observed ratio  $d(h_i^*, h_j^*) / d(\widehat{h_i^*}, \widehat{h_j^*})$ .<sup>13</sup> In fact,

<sup>13</sup>Note that since we expect  $\hat{d}$  to be an underestimate in general, we expect this ratio to be typically larger than 1.

adopting this as a simple heuristic leads to a surprisingly effective model selection procedure (ADJ): given the hypothesis sequence  $h_1^*, h_2^*, \dots$ , first multiply each estimated distance  $d(h_j^*, \widehat{P_{Y|X}})$  by the largest observed ratio  $d(h_i^*, h_j^*)/d(\widehat{h_i^*}, \widehat{h_j^*})$ ,  $i < j$ , and then choose the function in the sequence with the smallest *adjusted* distance estimate to  $P_{Y|X}$ . (Note that this adjustment to  $h_j^*$ 's distance can be interpreted as an estimate for the variance of  $H_j$ , indirectly achieved by referring to  $H_i \subset H_j$ .)

**Experiment** Tables 1–4 show that this technique does indeed work effectively on the model selection problems considered here. In particular, Table 4 shows that ADJ completely avoids the catastrophic mistakes made by the standard model selection strategies, and even outperforms the ideal variance penalizer VAR as well. This is a somewhat surprising result, but it follows from the fact that VAR does not pay explicit attention to the inter-hypothesis distances, and can therefore be fooled from time to time. Of course, we do not expect a free lunch [11] and there are certainly model selection problems where ADJ does not dominate (Table 3). However, the claim is that one should be able to exploit additional information about the task (here knowledge of  $P_X$ ) to obtain significant improvements across a wide range of problem types and conditions. Our empirical results support this view. (Further support to this claim is provided in the AAI'97 submission which considers a different class of polynomial curve-fitting problems.)

To summarize, this new metric-based technique ADJ appears to effectively avoid both under and over-fitting, and provides a safe and responsive model selection strategy, at least for the regression problems considered here. Interestingly, the performance of ADJ does not seem to degrade too severely when we move to consider hard model selection problems, even when these hard problems cause tremendous difficulty for standard techniques.

Of course, one can always argue that these results are not terribly useful since the metric-based strategy ADJ requires knowledge of the true domain distribution  $P_X$ . This is clearly an unreasonable assumption in practice. However, one can obtain information about  $P_X$  from *unlabeled* training instances. In fact, many important function learning applications have large corpora of unlabeled training data available (*e.g.*, image, speech and text databases), so these metric-based techniques could still apply to a wide range of practical situations—provided they are robust to using only *estimated* distances. In fact, ADJ turns out to be extremely robust to using approximate distances. Tables 1–4 show that as few as 100 reference examples were sufficient for the approximate ADJ procedure to perform nearly as well as ADJ. Finally, note that this still yields a reasonably efficient model selection

procedure since computing inter-hypothesis distances involves making only a single pass down the reference list of unlabeled examples. This is a strong advantage over standard hold-out techniques like 10CV which repeatedly call the hypothesis generating mechanism to generate pseudo-hypotheses.

## 6 Conclusions

We considered a simple characterization of model selection problems based on the standard bias/variance decomposition of expected hypothesis error. This analysis allows us to make predictions about and distinguish the performance of different model selection strategies based on two simple but essential aspects of the task: the shapes of the bias and variance profiles generated across the sequence of hypothesis classes. With this characterization, we distinguished between easy and hard model selection problems (a distinction which has not been explicitly stated nor characterized as far as we know). This distinction is important because difficult model selection problems arise in fairly natural conditions; for example, linear regression on broadly-distributed variables (as we demonstrated), or polynomial curve-fitting [1, 13] to name just a few.

These observations lead to specific recommendations, the first being that one needs to incorporate as much prior knowledge as possible about the shape of the variance profile in order to choose a model selection policy that works effectively while avoiding disastrous mistakes. Also, the new metric-based model selection strategies seem to be much more robust against catastrophic overfitting errors than standard techniques, and apparently could be usefully applied in difficult cases.

Among the many avenues for future work, we are currently pursuing the same style of bias/variance analysis to *classification* (as opposed to regression) problems [5]. We do not expect the same issues to be important here, since there is no potential for catastrophe in this case (unless one is interested in small relative rather than absolute approximation). The real issue is one of over versus under-penalization, as pointed out by Kearns *et al.* [5]. Note however that, for classification, the decomposition of prediction error into additive bias and variance components is not so obvious [7].

We are also pursuing more rigorous theoretical analyses of the various questions raised in this paper.

## References

- [1] V. Cherkassky, F. Mulier, and V. Vapnik. Comparison of VC-method with classical methods for model selection. Preprint, 1996.
- [2] P. Craven and G. Wahba. Smoothing noisy data with spline functions. *Numer. Math.*, 31:377–403, 1979.
- [3] B. Efron. Computers and the theory of statistics. *SIAM Review*, 21:460–80, 1979.
- [4] S. Geman, E. Bienenstock, and R. Doursat. Neural networks and the bias/variance dilemma. *Neural Comp.*, 4:1–58, 1992.
- [5] M. Kearns, Y. Mansour, A. Ng, and D. Ron. An experimental and theoretical comparison of model selection methods. In *COLT-95*, 1995.
- [6] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI-95*, 1995.
- [7] R. Kohavi and D. Wolpert. Bias plus variance decomposition for zero-one loss functions. In *ML-96*, pages 275–283, 1996.
- [8] J. Moody. The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems. In *NIPS-4*, 1992.
- [9] J. Moody and J. Utans. Principled architecture selection for neural networks: Application to corporate bond rating prediction. In *NIPS-4*, 1992.
- [10] J. Rissanen. Stochastic complexity and modeling. *Ann. Statist.*, 14:1080–100, 1986.
- [11] C Schaffer. Overfitting avoidance as bias. *Mach. Learn.*, 10(2):153–78, 1993.
- [12] V. Vapnik. *Estimation of Dependences Based on Empirical Data*. Springer-Verlag, New York, 1982.
- [13] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1996.
- [14] S. M. Weiss and C. A. Kulikowski. *Computer Systems that Learn*. Morgan Kaufmann, San Mateo, 1991.

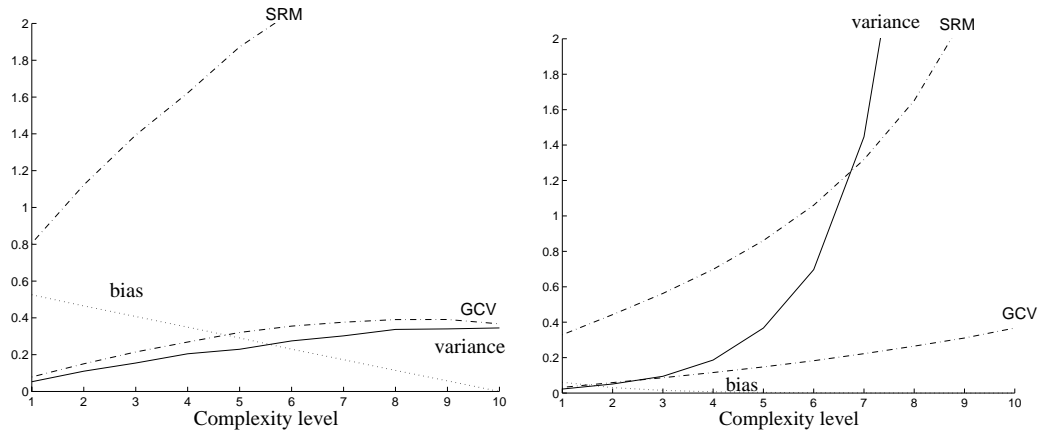


Figure 2: Bias/variance profiles for Problem 1 (*flat* variance) and Problem 2 (*steep* variance), showing the corresponding penalty profiles used by GCV and SRM.

	mean error ratio	percentiles of error ratios after 500 repetitions						mean complexity difference
		5	25	50	75	95	100	
GCV	1.848	1.0	1.000	1.424	2.123	4.094	10.35	-0.712
SRM	2.758	1.0	1.572	2.268	3.321	6.443	11.39	-5.644
VAR	1.579	1.0	1.000	1.120	1.734	3.580	10.35	0.042
10CV	2.093	1.0	1.138	1.625	2.486	4.954	11.39	-1.624
ADJ	1.974	1.0	1.084	1.559	2.361	4.717	11.98	-2.046
$\widehat{\text{ADJ}}$	1.973	1.0	1.071	1.530	2.328	4.779	11.98	-2.050

Table 1: Results for Problem 1—*flat* variance profile.

	mean error ratio	percentiles of error ratios after 500 repetitions						mean complexity difference
		5	25	50	75	95	100	
GCV	24.26	1.0	1.196	1.770	4.449	146.10	872.9	0.424
SRM	2.08	1.0	1.126	1.553	2.261	4.68	25.2	-1.174
VAR	2.01	1.0	1.091	1.437	2.041	3.99	25.3	-0.892
10CV	16.61	1.0	1.143	1.588	3.057	37.71	3270.5	-0.002
ADJ	1.54	1.0	1.000	1.233	1.742	3.10	8.8	-0.626
$\widehat{\text{ADJ}}$	1.84	1.0	1.000	1.253	1.826	3.78	34.4	-0.348

Table 2: Results for Problem 2—*steep* variance profile.

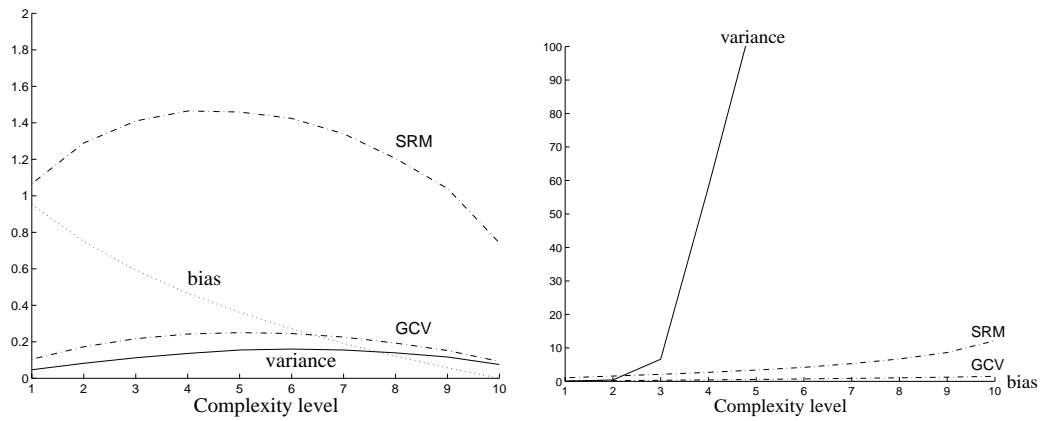


Figure 3: Bias/variance profiles for Problem 3 (*low* variance) and Problem 4 (*extreme* variance), showing the corresponding penalty profiles used by GCV and SRM.

	mean error ratio	percentiles of error ratios after 500 repetitions						mean complexity difference
		5	25	50	75	95	100	
GCV	1.227	1.0	1.0	1.0	1.000	2.263	9.41	-0.096
SRM	2.485	1.0	1.0	1.0	2.223	8.535	29.02	-0.840
VAR	1.031	1.0	1.0	1.0	1.000	1.101	2.87	0.096
10CV	1.603	1.0	1.0	1.0	1.101	4.421	27.90	-0.256
ADJ	2.230	1.0	1.0	1.0	2.223	7.547	27.89	-0.610
$\widehat{\text{ADJ}}$	2.287	1.0	1.0	1.0	2.321	7.844	27.89	-0.642

Table 3: Results for Problem 3—*low* variance profile; easy problem.

	mean error ratio	percentiles of error ratios after 100 repetitions						mean complexity difference
		5	25	50	75	95	100	
GCV	7497	1.0	1.0	1.097	253.5	41,790	253,700	1.120
SRM	1251	1.0	1.0	1.000	1.0	3.3	98,700	-0.110
VAR	2.1	1.0	1.0	1.000	1.0	1.7	103	-0.230
10CV	3177	1.0	1.0	1.006	38.8	6483	98,700	0.710
ADJ	1.3	1.0	1.0	1.000	1.0	2.6	8.3	0.040
$\widehat{\text{ADJ}}$	5.5	1.0	1.0	1.000	1.5	13	190	0.350

Table 4: Results for Problem 4—*catastrophic* variance profile; hard problem.