Machine Learning, 48, 51–84, 2002

Metric-Based Methods for Adaptive Model Selection and Regularization

DALE SCHUURMANS dale@cs.uwaterloo.ca
FINNEGAN SOUTHEY fdjsouth@cs.uwaterloo.ca
Department of Computer Science, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada

Editor: Yoshua Bengio

Abstract. We present a general approach to model selection and regularization that exploits *unlabeled* data to adaptively control hypothesis complexity in supervised learning tasks. The idea is to impose a metric structure on hypotheses by determining the discrepancy between their predictions across the distribution of unlabeled data. We show how this metric can be used to detect untrustworthy training error estimates, and devise novel model selection strategies that exhibit theoretical guarantees against over-fitting (while still avoiding under-fitting). We then extend the approach to derive a general training criterion for supervised learning—yielding an adaptive regularization method that uses unlabeled data to automatically set regularization parameters. This new criterion adjusts its regularization level to the specific set of training data received, and performs well on a variety of regression and conditional density estimation tasks. The only proviso for these methods is that sufficient unlabeled training data be available.

Keywords: model selection, regularization, unlabeled examples

1. Introduction

In supervised learning, one takes a sequence of training pairs $\langle x_1, y_1\rangle, \ldots, \langle x_t, y_t\rangle$ and attempts to infer a hypothesis function $h : X \to Y$ that achieves small prediction error $err(h(x), y)$ on future test examples. This basic paradigm covers many of the tasks studied in machine learning research, including: *regression*, where Y is typically $I\!R$ and we measure prediction error by squared difference $err(\hat{y}, y) = (\hat{y} - y)^2$ or some similar loss; *classification*, where Y is typically a small discrete set and we measure prediction error with the misclassification loss $err(\hat{y}, y) = 1_{(\hat{y} \neq y)}$; and *conditional density estimation*, where we assume, for example, that Y is a classification label from $\{0, 1\}$ and $\hat{Y}$ is a probabilistic prediction in $[0, 1]$, and we measure prediction error using the log loss $err(\hat{y}, y) = -y \log \hat{y} - (1 - y) \log(1 - \hat{y})$ (also known as the cross-entropy error (Bishop, 1995)).

Regardless of the specifics of these scenarios, one always faces the classical over-fitting versus under-fitting dilemma in supervised learning: If the hypothesis is chosen from a class that is too complex for the data, there is a good chance it will exhibit large test error even though its training error is small. This occurs because complex classes generally contain several hypotheses that behave similarly on the training data and yet behave quite differently in other parts of the domain—thus diminishing the ability to distinguish good hypotheses from bad. (Note that significantly different hypotheses cannot be simultaneously accurate.)

Therefore, one must restrict the set of hypotheses to be able to reliably differentiate between accurate and inaccurate predictors. On the other hand, selecting hypotheses from an overly restricted class can prevent one from being able to express a good approximation to the ideal predictor, thereby causing important structure in the training data to be ignored. Since both under-fitting and over-fitting result in large test error, they must be avoided simultaneously.

This tradeoff between over-fitting and under-fitting is a fundamental dilemma in machine learning and statistics. In this paper, we are primarily interested in investigating *automated* methods for calibrating hypothesis complexity to given training data. Most of the techniques that have been developed for this process fall into one of three basic categories: model selection, regularization, and model combination.

In *model selection* one first takes a base hypothesis class, H, decomposes it into a discrete collection of subclasses $H_0 \subset H_1 \subset \cdots = H$ (say, organized in a nested chain, or lattice) and then, given training data, attempts to identify the optimal subclass from which to choose the final hypothesis. There have been a variety of methods proposed for choosing the optimal subclass, but most techniques fall into one of two basic categories: *complexity penalization* (e.g., the minimum description length principle (Rissanen, 1986) and various statistical selection criteria (Foster & George, 1994)); and *hold-out testing* (e.g., cross-validation and bootstrapping (Efron, 1979).

Regularization is similar to model selection except that one does not impose a discrete decomposition on the base hypothesis class. Instead a penalty criterion is imposed on the individual hypotheses, which either penalizes their parametric form (e.g., as in ridge regression or weight decay in neural network training (Cherkassky & Mulier, 1998; Ripley, 1996; Bishop, 1995) or penalizes their global smoothness properties (e.g., minimizing curvature (Poggio & Girosi, 1990)).

Model combination methods do not select a single hypothesis but rather take a weighted combination of base hypotheses to form a composite predictor. Composing base functions in this way can have the effect of smoothing out erratic hypotheses (e.g., as in Bayesian model averaging (MacKay, 1992) and bagging (Breiman, 1996)), or increasing the representation power of the base hypothesis class through linear combinations (e.g., as in boosting (Freund & Schapire, 1997) and neural network ensemble methods (Krogh & Vedelsby, 1995)).

All of these methods have shown impressive improvements over naive learning algorithms in every area of supervised learning research. However, one difficulty with these techniques is that they usually require expertise to apply properly, and often involve free parameters that must be set by an informed practitioner.

In this paper we introduce alternative methods for model selection and regularization that attempt to improve on the robustness of standard approaches. Our idea is to use unlabeled data to automatically penalize hypotheses that behave erratically off the labeled training set. In Section 3 we first investigate how unlabeled data can be used to perform *model selection* in nested sequences of hypothesis spaces. The strategies we develop are shown to experimentally outperform standard model selection methods, and are proved to be robust in theory. Then in Section 4 we consider *regularization* and show how our proposed model selection strategies can be extended to a generalized training objective for supervised learning. Here the idea is to use unlabeled data to automatically tune the degree of regularization for a given task without having to set free parameters by hand. We show

that the resulting regularization technique adapts its behavior to a given training set and can outperform standard fixed regularizers for a given problem. Note, however, that we do not address model combination methods in this paper (Krogh & Vedelsby, 1995), instead leaving this to future work.

The work reported here extends the earlier conference papers (Schuurmans, 1997; Schuurmans & Southey, 2000).

2. Metric structure of supervised learning

In this paper we will consider the metric structure on a space of hypothesis functions that arises from a simple statistical model of the supervised learning problem: Assume that the examples $\langle x, y \rangle$ are generated by a stationary joint distribution P_{XY} on $X \times Y$. In learning a hypothesis function $h : X \to Y$ we are primarily interested in modeling the conditional distribution $\mathrm{P}_{Y|X}$. However, here we will investigate the utility of using extra information about the marginal domain distribution P_X to choose a good hypothesis. Note that information about P_X can be obtained from a collection of *un*labeled training examples $x_1, \ldots, x_r$ (these are often in abundant supply in many applications—for example, text processing and computer perception). The significance of having information about the domain distribution P_X is that it defines a natural *(pseudo) metric* on the space of hypotheses. That is, for any two hypothesis functions f and g we can obtain a measure of the distance between them by computing the expected disagreement in their predictions

$$d(f, g) \stackrel{\triangle}{=} \varphi\left(\int err(f(x), g(x))\, d\mathrm{P}_X\right) \tag{1}$$

where $err(\hat{y}, y)$ is the natural measure of prediction error for the problem at hand (e.g., regression or classification) and φ is an associated normalization function that recovers the standard metric axioms. Specifically, we will be interested in obtaining the metric properties: nonnegativity $d(f, g) \geq 0$, symmetry $d(f, g) = d(g, f)$, and the triangle inequality $d(f, g) \leq d(f, h) + d(h, g)$. It turns out that most typical prediction error functions admit a metric of this type.

For example, in regression we measure the distance between two prediction functions by

$$d(f, g) = \left(\int (f(x) - g(x))^2\, d\mathrm{P}_X\right)^{1/2}$$

where the normalization function $\varphi(z) = z^{1/2}$ establishes the metric properties. In classification, we measure the distance between two classifiers by

$$\begin{aligned} d(f, g) &= \int 1_{(f(x) \neq g(x))}\, d\mathrm{P}_X \\ &= \mathrm{P}_X(f(x) \neq g(x)) \end{aligned}$$

where no normalization is required to achieve a metric. (In conditional density estimation, one can measure the "distance" between two conditional probability models by their

Kullback-Leibler divergence, which technically is not a metric but nevertheless supplies a useful measure (Cover & Thomas, 1991)).

In each of these cases, the resulting distances can be efficiently calculated by making a single pass down a list of unlabeled examples. Importantly, these definitions can be generalized to include the target *conditional distribution* in an analogous manner:

$$d(h, \mathrm{P}_{Y|X}) \stackrel{\triangle}{=} \varphi\left(\int\int err(h(x), y)\, d\mathrm{P}_{Y|x}\, d\mathrm{P}_X\right) \tag{2}$$

That is, we can interpret the true error of a hypothesis function h with respect to a target conditional $\mathrm{P}_{Y|X}$ as a *distance* between h and $\mathrm{P}_{Y|X}$. The significance of this definition is that it is consistent with the previous definition (1) and we can therefore embed the entire supervised learning problem in a common metric space structure.

To illustrate, in regression the definition (2) yields the root mean squared error of a hypothesis

$$d(h, \mathrm{P}_{Y|X}) = \left(\int\int (h(x) - y)^2\, d\mathrm{P}_{Y|x}\, d\mathrm{P}_X\right)^{1/2}$$

and in classification it gives the true misclassification probability

$$\begin{aligned} d(h, \mathrm{P}_{Y|X}) &= \int\int 1_{(h(x)\neq y)}\, d\mathrm{P}_{Y|x}\, d\mathrm{P}_X \\ &= \mathrm{P}_{XY}(h(x) \neq y) \end{aligned}$$

(In conditional probability modeling it gives the expected log loss—or KL-divergence to $\mathrm{P}_{Y|X}$—which again, yields a useful measure, although it is not a metric).

Together, definitions (1) and (2) show how we can impose a global metric space view of the supervised learning problem (Figure 1): Given labeled training examples $\langle x_1, y_1\rangle, \ldots, \langle x_t, y_t\rangle$, the goal is to find the hypothesis h in a space H that is closest to a target conditional

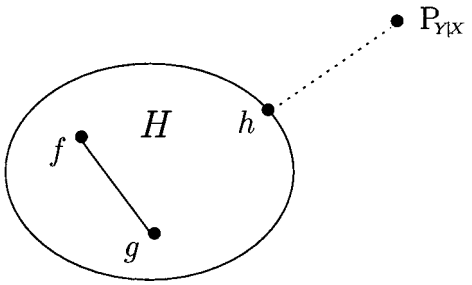


Figure 1. Metric space view of supervised learning: Unlabeled data can accurately estimate distances between functions f and g within H, however only limited labeled data is available to estimate the closest function h to $\mathrm{P}_{Y|X}$.

$P_{Y|X}$ under the distance measure (2). If we are also given a large set of auxiliary unlabeled examples $x'_1, \ldots, x'_r$, then we can also accurately estimate the distances between alternative hypotheses f and g within H; effectively giving us (1)

$$d(f, g) \stackrel{\triangle}{=} \varphi\left(\frac{1}{r}\sum_{j=1}^{r} err(f(x'_j), g(x'_j))\right) \tag{3}$$

That is, for sufficiently large r, the distances defined in (3) will be very close to the distances defined in (1). However, the distances between hypotheses and the target conditional $P_{Y|X}$ (2) can only be weakly estimated using the (presumably much smaller) set of labeled training data

$$d(h, \widehat{P_{Y|X}}) \stackrel{\triangle}{=} \varphi\left(\frac{1}{t}\sum_{i=1}^{t} err(h(x_i), y_i)\right) \tag{4}$$

which need not be close to (2). The challenge then is to approximate the closest hypothesis to the target conditional as accurately as possible using the available information (3) and (4) in place of the true distances (1) and (2).

Below we will use this metric space perspective to devise novel model selection and regularization strategies that exploit inter-hypothesis distances measured on an auxiliary set of unlabeled examples. Our approach is applicable to any supervised learning problem that admits a reasonable metric structure. In particular, all of our strategies will be expressed in terms of a generic distance measure which does not depend on other aspects of the problem. (However, for the sake of concreteness, we will focus on *regression* as a source of demonstration problems initially, and return to classification and conditional density estimation examples near the end of the paper.)

3. Model selection

We first consider the process of using *model selection* to choose the appropriate level of hypothesis complexity to fit to data. This, conceptually, is the simplest approach to automatic complexity control for supervised learning: the idea is to stratify the hypothesis class H into a sequence (or lattice) of nested subclasses $H_0 \subset H_1 \subset \cdots = H$, and then, given training data, somehow choose a class that has the proper complexity for the given data. To understand how one might make this choice, note that for a given training sample $\langle x_1, y_1\rangle, \ldots, \langle x_t, y_t\rangle$ we can, in principle, obtain the corresponding sequence of empirically optimal functions $h_0 \in H_0, h_1 \in H_1, \ldots$

$$h_k = \arg\min_{h \in H_k} \varphi\left(\frac{1}{t}\sum_{i=1}^{t} err(h(x_i), y_i)\right) = \arg\min_{h \in H_k} d(h, \widehat{P_{Y|X}})$$

The problem is to select one of these functions based on the observed training errors $d(h_0, \widehat{P_{Y|X}}), d(h_1, \widehat{P_{Y|X}}), \ldots$ (figure 2). Note, however, that these errors are monotonically

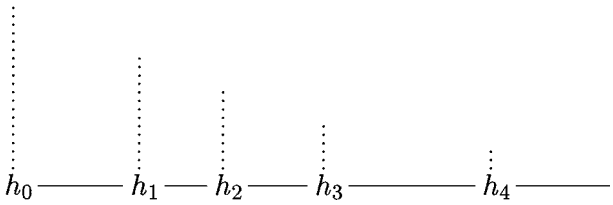


Figure 2. Sequence of empirically optimal functions induced by a chain $H_0 \subset H_1 \subset \cdots$ on a given training set: Dotted lines indicate decreasing optimal training distances $d(h_0, \widehat{\mathrm{P}_{Y|X}})$, $d(h_1, \widehat{\mathrm{P}_{Y|X}})$, ... and solid lines indicated distances *between* hypotheses. The final hypothesis must be selected on the basis of these estimates.

decreasing (assuming we can fully optimize in each class) and therefore choosing the function with smallest training error inevitably leads to over-fitting. So the trick is to invoke some other criterion beyond mere empirical error minimization to make the final selection.

As mentioned, two basic model selection strategies currently predominate: *complexity penalization* and *hold-out testing*. However, neither of these approaches attends to the metric distances between hypotheses, nor do they offer an obvious way to exploit auxiliary unlabeled data. But by adopting the metric space view of Section 2 we obtain an useful new perspective on model selection: In our setting, the chain $H_0 \subset H_1 \subset \cdots \subset H$ can be interpreted as a sequence of hypothesis *spaces* wherein we can measure the distance between candidate hypotheses (using unlabeled data). Unfortunately, we still cannot directly measure the distances from hypotheses to the target conditional $\mathrm{P}_{Y|X}$ (just as before) and therefore must estimate them based on a small labeled training sample. However, we can now exploit the fact that we have the distances *between* functions in the sequence, and hence attempt to use this additional information to make a better choice (figure 2).

3.1. Strategy 1: Triangle inequality

The first intuition we explore is that inter-hypothesis distances can help us detect over-fitting in a very simple manner: Consider two hypotheses h_k and h_{k+1} that both have a small estimated distance to $\mathrm{P}_{Y|X}$ and yet have a large true distance between them. In this situation, it should be clear that we should be concerned about selecting the second hypothesis, because if the true distance between h_k and h_{k+1} is indeed large then both functions cannot be simultaneously close to $\mathrm{P}_{Y|X}$, by simple geometry. This implies that at least one of the distance estimates to $\mathrm{P}_{Y|X}$ must be inaccurate, and we know intuitively to trust the earlier estimate more than the latter (since h_{k+1} is chosen from a larger class). In fact, if both $d(h_k, \widehat{\mathrm{P}_{Y|X}})$ and $d(h_{k+1}, \widehat{\mathrm{P}_{Y|X}})$ really were accurate estimates they would have to satisfy the *triangle inequality* with the known distance $d(h_k, h_{k+1})$; that is

$$d(h_k, \widehat{\mathrm{P}_{Y|X}}) + d(h_{k+1}, \widehat{\mathrm{P}_{Y|X}}) \geq d(h_k, h_{k+1}) \tag{5}$$

Since these empirical distances eventually become significant underestimates in general (because the h_i are explicitly chosen to minimize the empirical distance on the labeled training set) the triangle inequality provides a useful test to detect when these estimates

Procedure TRI
- Given hypothesis sequence $h_0, h_1, \ldots$
- Choose the last hypothesis h_ℓ in the sequence that satisfies the triangle inequality $d(h_k, h_\ell) \leq d(h_k, \widehat{\mathrm{P}_{\gamma|X}}) + d(h_\ell, \widehat{\mathrm{P}_{\gamma|X}})$ with every preceding hypothesis h_k, $0 \leq k < \ell$. (Note that the inter-hypothesis distances $d(h_k, h_\ell)$ are measured on the *un*labeled training data.)

Figure 3. Triangle inequality model selection procedure.

become inaccurate. In fact, this basic test forms the basis of a simple model selection strategy, TRI, that works surprisingly well in many situations (figure 3).

3.2. Example: Polynomial regression

To demonstrate this method (and all subsequent methods we develop in this paper) we first consider the problem of polynomial curve fitting. This is a supervised learning problem where $X = I\!R$, $Y = I\!R$, and the goal is to minimize the squared prediction error, $err(\hat{y}, y) = (\hat{y} - y)^2$. Specifically, we consider polynomial hypotheses $h : I\!R \rightarrow I\!R$ under the natural stratification $H_0 \subset H_1 \subset \cdots$ into polynomials of degree $0, 1, \ldots$, etc. The motivation for studying this task is that it is a classical well-studied problem, that still attracts a lot of interest (Cherkassky, Mulier, & Vapnik, 1997; Galarza, Rietman, & Vapnik, 1996; Vapnik, 1996). Moreover, polynomials create a difficult model selection problem which has a strong tendency to produce catastrophic over-fitting effects (figure 4). Another benefit is that polynomials are an interesting and nontrivial class for which there are efficient techniques for computing best fit hypotheses.

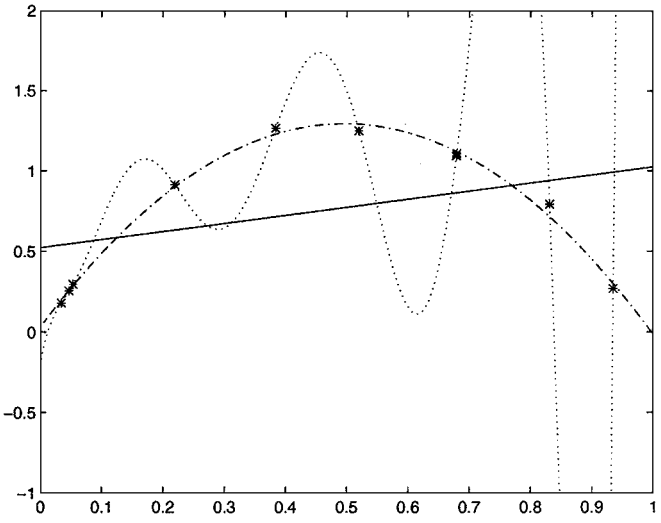


Figure 4. An example of minimum squared error polynomials of degrees 1, 2, and 9 for a set of 10 training points. The large degree polynomial demonstrates erratic behavior off the training set.

To apply the metric based approach to this task, we define the metric d in terms of the squared prediction error $err(\hat{y}, y) = (\hat{y} - y)^2$ with a square root normalization $\varphi(z) = z^{1/2}$, as discussed in Section 2. To evaluate the efficacy of TRI in this problem we compared its performance to a number of standard model selection strategies, including: structural risk minimization, SRM (Cherkassky, Mulier, & Vapnik, 1997; Vapnik, 1996), RIC (Foster & George, 1994), SMS (Shibata, 1981), GCV (Craven & Wahba, 1979), BIC (Schwarz, 1978), AIC (Akaike, 1974), CP (Mallows, 1973), and FPE (Akaike, 1970). We also compared it to 10-fold cross validation, CVT (a standard hold-out method (Efron, 1979; Weiss & Kulikowski, 1991; Kohavi, 1995)).

We conducted a simple series of experiments by fixing a domain distribution P_X on $X = I\!R$ and then fixing various target functions $f : I\!R \to I\!R$. (The specific target functions we used in our experiments are shown in figure 5). To generate training samples we first drew a sequence of values, $x_1, \ldots, x_t$, computed the target function values $f(x_1), \ldots, f(x_t)$, and added independent Gaussian noise to each, to obtain the labeled training sequence $\langle x_1, y_1 \rangle, \ldots, \langle x_t, y_t \rangle$. For a given training sample we then computed the series of best fit polynomials $h_0, h_1, \ldots$ of degree 0, 1, ..., etc. Given this sequence, each model selection strategy will choose some hypothesis h_k on the basis of the observed empirical errors. To implement TRI we gave it access to auxiliary unlabeled examples $x'_1, \ldots, x'_r$ in order to compute the true distances between polynomials in the sequence.

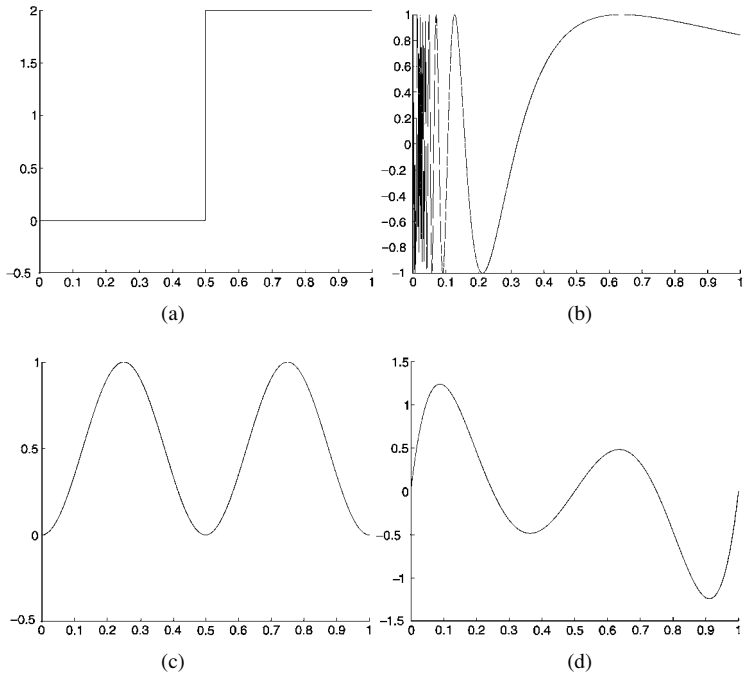


Figure 5. Target functions used in the polynomial curve fitting experiments (in order): (a) step($x \geq 0.5$), (b) $\sin(1/x)$, (c) $\sin^2(2\pi x)$, and (d) a fifth degree polynomial.

Our main emphasis in these experiments was to minimize the true distance between the final hypothesis and the target conditional $P_{Y|X}$. That is, we are primarily concerned with choosing a hypothesis that obtains a small prediction error on future test examples, independent of its complexity level.[1] To determine the effectiveness of the various selection strategies, we therefore measured the *ratio* of the true error (distance) of the polynomial they selected to the best true error among polynomials in the sequence $h_0, h_1, \ldots$, etc. (This means that the optimum achievable ratio is 1). The rationale for doing this is that we wish to measure the model selection strategy's ability to approximate the best hypothesis in the given sequence—not find a better function from outside the sequence.[2]

Table 1 shows the results obtained for approximating a step function $f(x) = \text{step}(x \geq 0.5)$ corrupted by Gaussian noise. (The strategy ADJ in the tables is explained in Section 3.3 below.) We obtained these results by repeatedly generating training samples of a fixed size and recording the approximation ratio achieved by each strategy. These tables record the distribution of ratios produced by each strategy for training sample sizes of $t = 20$ and 30 respectively, using $r = 200$ unlabeled examples to measure inter-hypothesis distances—repeated over 1000 trials. The initial results appear to be quite positive. TRI achieves median approximation ratios of 1.06 and 1.08 for training sample sizes 20 and 30 respectively. This compares favorably to the median approximation ratios 1.39 and 1.54 achieved by SRM, and 1.17 achieved by CVT in both cases. (The remaining complexity penalization strategies, GCV, FPE, etc., all performed significantly worse on these trials.) However, the most notable difference was TRI's robustness against over-fitting. In fact, although the penalization strategy SRM performed reasonably well much of the time, it was prone to making periodic but catastrophic over-fitting errors. Even the normally well-behaved cross-validation strategy CVT made significant over-fitting errors from time to time. This is

Table 1. Fitting $f(x) = \text{step}(x \geq 0.5)$ with $P_X = U(0, 1)$ and $\sigma = 0.05$. Table gives distribution of approximation ratios achieved at training sample size $t = 20$ and $t = 30$, showing percentiles of approximation ratios achieved in 1000 repeated trials.

		TRI	CVT	SRM	RIC	GCV	BIC	AIC	FPE	ADJ
	$t = 20$									
	25	1.00	1.06	1.14	7.54	5.47	15.2	22.2	25.8	1.02
	50	1.06	1.17	1.39	224	118	394	585	590	1.12
Percentiles	75	1.17	1.42	3.62	5.8e3	3.9e3	9.8e3	1.2e4	1.2e4	1.24
	95	1.44	6.75	56.1	6.1e5	3.7e5	7.8e5	9.2e5	8.2e5	1.54
	100	2.41	1.1e4	2.2e4	1.5e8	6.5e7	1.5e8	1.5e8	8.2e7	3.02
	$t = 30$									
	25	1.00	1.08	1.17	4.69	1.51	5.41	5.45	2.72	1.06
	50	1.08	1.17	1.54	34.8	9.19	39.6	40.8	19.1	1.14
Percentiles	75	1.19	1.37	9.68	258	91.3	266	266	159	1.25
	95	1.45	6.11	419	4.7e3	2.7e3	4.8e3	5.1e3	4.0e3	1.51
	100	2.18	643	1.6e7	1.6e7	1.6e7	1.6e7	1.6e7	1.6e7	2.10

evidenced by the fact that in 1000 trials with a training sample of size 30 (Table 1) TRI produced a *maximum* approximation ratio of 2.18, whereas CVT produced a worst case approximation ratio of 643, and the penalization strategies SRM and GCV both produced worst case ratios of 1.6×10^7. (The 95th percentiles were TRI 1.45, CVT 6.11, SRM 419, GCV 2.7×10^3).[3]

In fact, TRI's robustness against over-fitting is not a surprise: One can prove that TRI cannot produce an approximation ratio greater than 3 if we make two simple assumptions: (i) that TRI makes it to the best hypothesis h_m in the sequence, and (ii) that the empirical error of h_m is an underestimate; that is, $d(h_m, \widehat{\mathrm{P}_{Y|X}}) \leq d(h_m, \mathrm{P}_{Y|X})$. (Note that this second assumption is likely to hold because we are choosing hypotheses by explicitly minimizing $d(h_m, \widehat{\mathrm{P}_{Y|X}})$ rather than $d(h_m, \mathrm{P}_{Y|X})$; see Table 6.)

Proposition 1. *Let h_m be the optimal hypothesis in the sequence $h_0, h_1, \ldots$ (that is, $h_m = \arg\min_{h_k} d(h_k, \mathrm{P}_{Y|X})$) and let h_ℓ be the hypothesis selected by TRI. If* (i) $m \leq \ell$ *and* (ii) $d(h_m, \widehat{\mathrm{P}_{Y|X}}) \leq d(h_m, \mathrm{P}_{Y|X})$ *then*

$$d(h_\ell, \mathrm{P}_{Y|X}) \leq 3d(h_m, \mathrm{P}_{Y|X}) \tag{6}$$

Proof: Consider a hypothesis h_n which follows h_m in the sequence, and assume $d(h_n, \mathrm{P}_{Y|X}) > 3d(h_m, \mathrm{P}_{Y|X})$. We show that h_n must fail the triangle test (5) with h_m and therefore TRI will not select h_n. First, notice that the initial assumption about h_n's error along with the triangle inequality imply that $3d(h_m, \mathrm{P}_{Y|X}) < d(h_n, \mathrm{P}_{Y|X}) \leq d(h_m, h_n) + d(h_m, \mathrm{P}_{Y|X})$, and hence $d(h_m, h_n) > 2d(h_m, \mathrm{P}_{Y|X})$. But now recall that $d(h_n, \widehat{\mathrm{P}_{Y|X}}) \leq d(h_m, \widehat{\mathrm{P}_{Y|X}})$ for $n \geq m$ (since the training errors are monotonically decreasing), and also, by assumption, $d(h_m, \widehat{\mathrm{P}_{Y|X}}) < d(h_m, \mathrm{P}_{Y|X})$. Therefore we have $d(h_m, h_n) > 2d(h_m, \mathrm{P}_{Y|X}) > d(h_m, \widehat{\mathrm{P}_{Y|X}}) + d(h_n, \widehat{\mathrm{P}_{Y|X}})$, which contradicts (5). Thus TRI will not consider h_n. Finally, since h_ℓ cannot precede h_m (by assumption (i)), h_ℓ must satisfy $d(h_\ell, \mathrm{P}_{Y|X}) \leq 3d(h_m, \mathrm{P}_{Y|X})$. □

(Note that in Proposition 1, as well as Propositions 2 and 3 below, we implicitly assume that we have the true inter-hypothesis distances $d(h_m, h_\ell)$, which in principle must be measured on unlimited amounts of unlabeled data. We discuss relaxing this assumption in Section 3.4 below).

Continuing with the experimental investigation, we find that the basic flavor of the results remains unchanged at different noise levels and for different domain distributions P_X. In fact, much stronger results are obtained for wider tailed domain distributions like Gaussian (Table 2) and "difficult" target functions like $\sin(1/x)$ (Table 3). Here the complexity penalization methods (SRM, GCV, etc.) can be forced into a regime of constant catastrophe, CVT noticeably degrades, and yet TRI retains similar performance levels shown in Table 1.

Of course, these results might be due to considering a pathological target function from the perspective of polynomial curve fitting. It is therefore important to consider other more natural targets that might be better suited to polynomial approximation. In fact, by repeating the previous experiments with a more benign target function $f(x) = \sin^2(2\pi x)$ we obtain quite different results. Table 4 shows that procedure TRI does not fare as well in this case–obtaining median approximation ratios of 3.11 and 3.51 for training sample sizes 20

Table 2. Fitting $f(x) = \text{step}(x \geq 0.5)$ using $\sigma = 0.05$ (as in Table 1), but here using $P_X = N(0.5, 1)$ instead of $P_X = U(0, 1)$. Table gives distribution of approximation ratios achieved at training sample size $t = 20$ and $t = 30$, showing percentiles of approximation ratios achieved in 1000 repeated trials.

		TRI	CVT	SRM	RIC	GCV	BIC	AIC	FPE	ADJ
	$t = 20$									
	25	1.00	1.01	1.23	1.9e7	3.0e5	7.0e7	1.4e8	3.5e7	1.00
	50	1.09	1.36	10.2	9.8e8	1.5e8	2.2e9	3.1e9	1.4e9	1.00
Percentiles	75	1.27	3.75	982	2e10	5.9e9	4e10	5e10	3e10	1.04
	95	2.32	47.5	5.6e4	1e12	5e11	1e12	1e12	1e12	1.21
	100	33.2	4.9e5	7.3e6	2e14	1e14	4e14	4e14	1e14	2.24
	$t = 30$									
	25	1.01	1.02	10.9	1.9e7	3.2e4	2.1e7	2.3e7	2.4e6	1.00
	50	1.13	1.36	606	9.9e7	7.3e6	1.1e8	1.1e8	3.7e7	1.00
Percentiles	75	1.51	4.82	8.4e5	6.1e8	1.1e8	6.2e8	6.5e8	2.8e8	1.08
	95	3.68	92.0	2.8e8	5.6e9	2.4e9	5.9e9	5.9e9	4.2e9	1.20
	100	44.4	5.2e5	2e10	2e11	1e11	2e11	2e11	2e11	2.05

Table 3. Fitting $f(x) = \sin(1/x)$ with $P_X = U(0, 1)$ and $\sigma = 0.05$. Table gives distribution of approximation ratios achieved at training sample size $t = 20$ and $t = 30$, showing percentiles of approximation ratios achieved in 1000 repeated trials.

		TRI	CVT	SRM	RIC	GCV	BIC	AIC	FPE	ADJ
	$t = 20$									
	25	1.00	1.08	1.20	3.11	4.34	3.86	4.81	9.89	1.07
	50	1.11	1.21	1.64	11.9	22.8	15.8	24.9	72.8	1.18
Percentiles	75	1.30	1.76	6.58	77.4	193	104	196	1.3e3	1.38
	95	1.77	18.5	39.0	1.4e4	2.6e4	2.4e4	3.7e4	1.2e5	3.79
	100	3.80	5.8e3	9.2e3	1.0e9	1.0e9	1.0e9	1.0e9	2.7e9	22.9
	$t = 30$									
	25	1.02	1.08	1.34	2.80	1.89	3.16	3.67	2.80	1.08
	50	1.14	1.20	4.74	12.1	9.67	14.1	15.8	13.8	1.17
Percentiles	75	1.30	1.63	33.2	61.5	55.2	70.1	81.6	72.4	1.30
	95	1.72	23.5	306	1.2e3	479	1.3e3	1.3e3	1.3e3	1.81
	100	2.68	325	1.4e5	5.2e5	1.4e5	5.2e5	5.2e5	3.9e5	9.75

and 30 respectively (compared to 1.33 and 1.03 for SRM, and 1.37 and 1.16 for CVT). A closer inspection of TRI's behavior reveals that the reason for this performance drop is that TRI systematically gets stuck at low even-degree polynomials (cf. Table 6). In fact, there is a simple geometric explanation for this: the even-degree polynomials (after

Table 4. Fitting $f(x) = \sin^2(2\pi x)$ with $P_X = U(0, 1)$ and $\sigma = 0.05$. Table gives distribution of approximation ratios achieved at training sample size $t = 20$ and $t = 30$, showing percentiles of approximation ratios achieved in 1000 repeated trials.

		TRI	CVT	SRM	RIC	GCV	BIC	AIC	FPE	ADJ
	$t = 20$									
	25	2.04	1.03	1.00	1.00	1.06	1.00	1.01	1.58	1.02
	50	3.11	1.37	1.33	1.34	1.94	1.35	1.61	18.2	1.32
Percentiles	75	3.87	2.23	2.30	2.13	10.0	2.75	4.14	1.2e3	1.83
	95	5.11	9.45	8.84	8.26	5.0e3	11.8	82.9	1.8e5	3.94
	100	8.92	105	526	105	2.0e7	2.1e3	2.7e5	2.4e7	6.30
	$t = 30$									
	25	1.50	1.00	1.00	1.00	1.00	1.00	1.00	1.02	1.01
	50	3.51	1.16	1.03	1.05	1.11	1.02	1.08	1.45	1.27
Percentiles	75	4.15	1.64	1.45	1.48	2.02	1.39	1.88	6.44	1.60
	95	5.51	5.21	5.06	4.21	26.4	5.01	19.9	295	3.02
	100	9.75	124	1.4e3	20.0	9.1e3	28.4	9.4e3	1.0e4	8.35

degree 4) all give reasonable fits to $\sin^2(2\pi x)$ whereas the odd-degree fits have a tail in the wrong direction. This creates a significant distance between successive polynomials and causes the triangle inequality test to fail between the even and odd degree fits, even though the larger even-degree polynomials give a good approximation. Therefore, although the metric-based TRI strategy is robust against over-fitting, it can be prone to systematic under-fitting in seemingly benign cases. Similar results were obtained for fitting a fifth degree target polynomial corrupted by the same level of Gaussian noise (Table 5). This problem demonstrates that the first assumption used in Proposition 1 above can be violated in natural situations (see Table 6). Consideration of this difficulty leads us to develop a reformulated procedure.

3.3. Strategy 2: Adjusted distance estimates

The final idea we explore for model selection is to observe that we are actually 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 $\langle x_1, y_1\rangle, \ldots, \langle x_t, y_t\rangle$. Note that the previous model selection strategy TRI ignored the fact that we could measure the empirical distance between hypotheses $\hat{d}(h_k, h_\ell)$ on the *labeled* training data, as well as measure their "true" distance $d(h_k, h_\ell)$ on the unlabeled data. However, the fact that we can measure both inter-hypothesis distances actually gives us an *observable* relationship between $\hat{d}$ and d in the local vicinity. We now exploit this observation to attempt to derive an improved model selection procedure.

Given the two metrics d and $\hat{d}$, consider the triangle formed by two hypotheses h_k and h_ℓ and the target conditional $P_{Y|X}$ (figure 6). Notice that there are six distances involved—three

Table 5. Fitting a fifth degree polynomial $f(x)$ with $P_X = U(0, 1)$ and $\sigma = 0.05$. Table gives distribution of approximation ratios achieved at training sample size $t = 20$ and $t = 30$, showing percentiles of approximation ratios achieved in 1000 repeated trials.

		TRI	CVT	SRM	RIC	GCV	BIC	AIC	FPE	ADJ
	$t = 20$									
	25	7.52	1.00	1.00	1.00	1.00	1.00	1.00	1.10	1.00
	50	8.62	1.00	1.00	1.00	1.05	1.00	1.00	10.2	1.00
Percentiles	75	9.75	1.20	1.03	1.01	2.68	1.04	1.35	850	1.06
	95	12.1	3.89	2.17	1.35	2.2e3	2.68	28.5	2.3e5	2.32
	100	17.6	582	233	15.2	2.6e8	3.5e3	1.0e6	3.3e8	16.9
	$t = 30$									
	25	7.80	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	50	8.58	1.01	1.00	1.00	1.01	1.00	1.00	1.08	1.00
Percentiles	75	9.36	1.11	1.01	1.00	1.20	1.01	1.14	2.40	1.02
	95	11.0	2.59	1.42	1.13	8.92	1.35	5.46	131	1.18
	100	14.2	45.3	24.1	8.00	3.1e4	11.8	9.9e3	1.4e5	13.6

Table 6. Strengths of the assumptions used in Propositions 1 and 2. Table shows frequency (in percent) that the assumptions hold over 1000 repetitions of the experiments conducted in Tables 1, 3, 4 and 5 (at sample size $t = 20$).

	step($x \geq 0.5$) (Table 1)	sin($1/x$) (Table 3)	$\sin^2(2\pi x)$ (Table 4)	$\text{poly}^5(x)$ (Table 5)
Proposition 1(i) holds	73	80	10	4
Proposition 1(ii) holds	87	86	99	98
Proposition 1 holds	61	66	9	4
Proposition 2(i) holds	27	32	28	67
Proposition 2(ii) holds	22	26	14	24
Proposition 2 holds	15	17	12	21

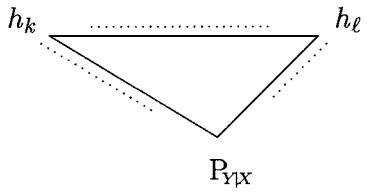


Figure 6. The real and estimated distances between successive hypotheses h_k and h_ℓ and the target $P_{Y|X}$. Solid lines indicate real distances, dotted lines indicate empirical distance estimates.

Procedure ADJ
- Given hypothesis sequence $h_0, h_1, \ldots$
- For each hypothesis h_ℓ in the sequence
  multiply its estimated distance to the target $d(h_\ell, \mathrm{P}_{YX})$ by the worst ratio of unlabeled and labeled distance to some predecessor h_k to obtain an adjusted distance estimate $d(\widehat{\widehat{h_\ell, \mathrm{P}_{YX}}}) = d(\widehat{h_\ell, \mathrm{P}_{YX}}) \frac{d(h_k, h_\ell)}{\widehat{d(h_k, h_\ell)}}$.
- Choose the hypothesis h_n with the smallest adjusted distance $d(\widehat{\widehat{h_n, \mathrm{P}_{YX}}})$.

Figure 7. Adjusted-distance-estimate model selection procedure.

real and three estimated, of which the true distances to $\mathrm{P}_{Y|X}$ are the only two we care about, and yet these are the only two that we do not have. However, we can now exploit the observed relationship between d and $\hat{d}$ to adjust the empirical training error estimate $d(\widehat{h_\ell, \mathrm{P}_{Y|X}})$. In fact, one could first consider the simplest possible adjustment based on the naive assumption that the observed relationship of the metrics $\hat{d}$ and d between h_k and h_ℓ also holds between h_ℓ and $\mathrm{P}_{Y|X}$. Note that if this were actually the case, we would obtain a better estimate of $d(h_\ell, \mathrm{P}_{Y|X})$ simply by re-scaling the training distance $d(\widehat{h_\ell, \mathrm{P}_{Y|X}})$ according to the observed ratio $d(h_k, h_\ell)/d(\widehat{h_k, h_\ell})$. (Since we expect $\hat{d}$ to be an underestimate in general, we expect this ratio to be larger than 1). In fact, by adopting this as a simple heuristic we obtain another model selection procedure, ADJ, which is also surprisingly effective (figure 7). This simple procedure overcomes some of the under-fitting problems associated with TRI and yet retains much of TRI's robustness against over-fitting.

Although at first glance this procedure might seem to be ad hoc, it turns out that one can prove an over-fitting bound for ADJ that is analogous to that established for TRI. In particular, if we assume that (i) ADJ makes it to the best hypothesis h_m in the sequence, and (ii) the adjusted error estimate $d(\widehat{\widehat{h_m, \mathrm{P}_{Y|X}}})$ is an underestimate, then ADJ cannot over-fit by a factor much greater than 3.

Proposition 2. *Let h_m be the optimal hypothesis in the sequence $h_0, h_1, \ldots$ and let h_ℓ be the hypothesis selected by ADJ. If* (i) $m \leq \ell$ *and* (ii) $d(\widehat{\widehat{h_m, \mathrm{P}_{Y|X}}}) \leq d(h_m, \mathrm{P}_{Y|X})$ *then*

$$d(h_\ell, \mathrm{P}_{Y|X}) \leq \left(2 + \frac{d(\widehat{h_m, \mathrm{P}_{Y|X}})}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}})}\right) d(h_m, \mathrm{P}_{Y|X}) \tag{7}$$

Proof: By the definition of ADJ we have that

$$d(\widehat{\widehat{h_\ell, \mathrm{P}_{Y|X}}}) \leq d(\widehat{\widehat{h_m, \mathrm{P}_{Y|X}}}) \tag{8}$$

since ADJ selects h_ℓ in favor of h_m. We show that this implies a bound on h_ℓ's true test error $d(h_\ell, \mathrm{P}_{Y|X})$ in terms of the optimum available test error $d(h_m, \mathrm{P}_{Y|X})$. First, by the triangle inequality we have $d(h_\ell, \mathrm{P}_{Y|X}) \leq d(h_\ell, h_m) + d(h_m, \mathrm{P}_{Y|X})$ as well as $d(\widehat{h_\ell, h_m}) \leq d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) + d(\widehat{h_m, \mathrm{P}_{Y|X}})$, and hence

$$\frac{d(h_\ell, h_m)}{d(\widehat{h_\ell, h_m})} \geq \frac{d(h_\ell, \mathrm{P}_{Y|X}) - d(h_m, \mathrm{P}_{Y|X})}{d(\widehat{h_m, \mathrm{P}_{Y|X}}) + d(\widehat{h_\ell, \mathrm{P}_{Y|X}})}$$

Note that by the definition of ADJ (and since $m \leq \ell$) this yields

$$\begin{aligned} d(\widehat{\widehat{h_\ell, \mathrm{P}_{Y|X}}}) &\geq \frac{d(h_\ell, h_m)}{d(\widehat{h_\ell, h_m})} d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) \\ &\geq d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) \frac{d(h_\ell, \mathrm{P}_{Y|X}) - d(h_m, \mathrm{P}_{Y|X})}{d(\widehat{h_m, \mathrm{P}_{Y|X}}) + d(\widehat{h_\ell, \mathrm{P}_{Y|X}})} \end{aligned} \tag{9}$$

So from (9) and (8) and the assumption that $d(\widehat{\widehat{h_m, \mathrm{P}_{Y|X}}}) \leq d(h_m, \mathrm{P}_{Y|X})$, we obtain

$$\begin{aligned} d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) \frac{d(h_\ell, \mathrm{P}_{Y|X}) - d(h_m, \mathrm{P}_{Y|X})}{d(\widehat{h_m, \mathrm{P}_{Y|X}}) + d(\widehat{h_\ell, \mathrm{P}_{Y|X}})} &\leq d(\widehat{\widehat{h_\ell, \mathrm{P}_{Y|X}}}) \\ &\leq d(\widehat{\widehat{h_m, \mathrm{P}_{Y|X}}}) \\ &\leq d(h_m, \mathrm{P}_{Y|X}) \end{aligned}$$

Simple algebraic manipulation then shows that

$$\begin{aligned} d(h_\ell, \mathrm{P}_{Y|X}) &\leq \frac{d(h_m, \mathrm{P}_{Y|X})}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}})} (d(\widehat{h_m, \mathrm{P}_{Y|X}}) + d(\widehat{h_\ell, \mathrm{P}_{Y|X}})) + d(h_m, \mathrm{P}_{Y|X}) \\ &= \frac{d(h_m, \mathrm{P}_{Y|X})}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}})} (d(\widehat{h_m, \mathrm{P}_{Y|X}}) + d(\widehat{h_\ell, \mathrm{P}_{Y|X}})) + \frac{d(h_m, \mathrm{P}_{Y|X})}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}})} d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) \\ &= \frac{d(h_m, \mathrm{P}_{Y|X})}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}})} (d(\widehat{h_m, \mathrm{P}_{Y|X}}) + 2d(\widehat{h_\ell, \mathrm{P}_{Y|X}})) \\ &= d(h_m, \mathrm{P}_{Y|X}) \left(\frac{d(\widehat{h_m, \mathrm{P}_{Y|X}})}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}})} + 2 \right) \end{aligned}$$

□

In this respect, not only does ADJ exhibit robustness against over-fitting, it also has a (weak) theoretical guarantee against under-fitting. That is, if we make the assumptions that: (i) the empirical distance estimates are underestimates, and (ii) the adjusted distance estimates strictly increase the empirical distance estimates; then if the true error of a successor hypothesis h_m improves the true error of all of its predecessors h_ℓ by a significant factor, h_m will be selected in lieu of its predecessors.

Proposition 3. *Consider a hypotheses h_m, and assume that* (i) $d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) \leq d(h_\ell, \mathrm{P}_{Y|X})$ *for all* $0 \leq \ell \leq m$, *and* (ii) $d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) \leq d(\widehat{\widehat{h_\ell, \mathrm{P}_{Y|X}}})$ *for all* $0 \leq \ell < m$. *Then if*

$$d(h_m, \mathrm{P}_{Y|X}) < \frac{1}{3} \frac{d(\widehat{h_\ell, \mathrm{P}_{Y|X}})^2}{d(h_\ell, \mathrm{P}_{Y|X})} \tag{10}$$

for all $0 \leq \ell < m$ (*that is,* $d(h_m, \mathrm{P}_{Y|X})$ *is sufficiently small*) *it follows that* $d(\widehat{\widehat{h_m, \mathrm{P}_{Y|X}}}) < d(\widehat{\widehat{h_\ell, \mathrm{P}_{Y|X}}})$ *for all* $0 \leq \ell < m$, *and therefore ADJ will not choose any predecessor in lieu of* h_m.

Proof: By the triangle inequality we have $d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) \leq d(\widehat{h_\ell, h_m}) + d(\widehat{h_m, \mathrm{P}_{Y|X}})$ and $d(h_\ell, h_m) \leq d(h_\ell, \mathrm{P}_{Y|X}) + d(h_m, \mathrm{P}_{Y|X})$, yielding

$$\frac{d(h_\ell, h_m)}{d(\widehat{h_\ell, h_m})} \leq \frac{d(h_\ell, \mathrm{P}_{Y|X}) + d(h_m, \mathrm{P}_{Y|X})}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) - d(\widehat{h_m, \mathrm{P}_{Y|X}})} \tag{11}$$

Recall that by the definition of $\hat{\hat{d}}$ we have

$$d(\widehat{\widehat{h_m, \mathrm{P}_{Y|X}}}) = \frac{d(h_\ell, h_m)}{d(\widehat{h_\ell, h_m})} d(\widehat{h_\ell, \mathrm{P}_{Y|X}})$$

for some $0 \leq \ell < m$ (specifically, the ℓ leading to the largest $d(\widehat{\widehat{h_m, \mathrm{P}_{Y|X}}})$). Therefore by applying (11) to this particular ℓ we obtain

$$\begin{aligned} d(\widehat{\widehat{h_m, \mathrm{P}_{Y|X}}}) &\leq d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) \frac{d(h_\ell, \mathrm{P}_{Y|X}) + d(h_m, \mathrm{P}_{Y|X})}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) - d(\widehat{h_m, \mathrm{P}_{Y|X}})} \\ &< d(h_m, \mathrm{P}_{Y|X}) \frac{2d(h_\ell, \mathrm{P}_{Y|X})}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) - d(h_m, \mathrm{P}_{Y|X})} \end{aligned}$$

The second step above follows from the assumption (i) that $d(\widehat{h_m, \mathrm{P}_{Y|X}}) \leq d(h_m, \mathrm{P}_{Y|X})$ and the fact that $d(h_m, \mathrm{P}_{Y|X}) < d(h_\ell, \mathrm{P}_{Y|X})$ (by both (i) and (10)). Now, by applying (10) to both occurrences of $d(h_m, \mathrm{P}_{Y|X})$ we obtain

$$\begin{aligned} d(\widehat{\widehat{h_m, \mathrm{P}_{Y|X}}}) &< \frac{d(\widehat{h_\ell, \mathrm{P}_{Y|X}})^2}{3d(h_\ell, \mathrm{P}_{Y|X})} \left(\frac{2d(h_\ell, \mathrm{P}_{Y|X})}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) - d(\widehat{h_\ell, \mathrm{P}_{Y|X}})^2 / 3d(h_\ell, \mathrm{P}_{Y|X})} \right) \\ &= \frac{2d(h_\ell, \mathrm{P}_{Y|X}) d(\widehat{h_\ell, \mathrm{P}_{Y|X}})^2}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) 3d(h_\ell, \mathrm{P}_{Y|X}) - d(\widehat{h_\ell, \mathrm{P}_{Y|X}})^2} \\ &< \frac{2d(h_\ell, \mathrm{P}_{Y|X}) d(\widehat{h_\ell, \mathrm{P}_{Y|X}})^2}{d(\widehat{h_\ell, \mathrm{P}_{Y|X}})(2d(h_\ell, \mathrm{P}_{Y|X}) + d(\widehat{h_\ell, \mathrm{P}_{Y|X}})) - d(\widehat{h_\ell, \mathrm{P}_{Y|X}})^2} \\ &\qquad \text{since } d(h_\ell, \mathrm{P}_{Y|X}) > d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) \text{ by assumption (i)} \\ &= \frac{2d(h_\ell, \mathrm{P}_{Y|X}) d(\widehat{h_\ell, \mathrm{P}_{Y|X}})^2}{2d(h_\ell, \mathrm{P}_{Y|X}) d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) + d(\widehat{h_\ell, \mathrm{P}_{Y|X}})^2 - d(\widehat{h_\ell, \mathrm{P}_{Y|X}})^2} \\ &= d(\widehat{h_\ell, \mathrm{P}_{Y|X}}) \\ &< d(\widehat{\widehat{h_\ell, \mathrm{P}_{Y|X}}}) \quad \text{by assumption (ii)} \end{aligned}$$

□

Therefore, although ADJ might not have originally appeared to be well motivated, it possesses worst case bounds against over-fitting and under-fitting that cannot be established for conventional methods. However, these bounds remain somewhat weak: Table 6 shows that both ADJ and TRI systematically under-fit in our experiments. That is, even though assumption (ii) of Proposition 1 is almost always satisfied (as expected), assumption (ii) of

Proposition 2 is only true one quarter of the time. Therefore, Propositions 1 and 2 can only provide a loose characterization of the quality of these methods. However, both metric-based procedures remain robust against over-fitting.

To demonstrate that ADJ is indeed effective, we repeated the previous experiments with ADJ as a new competitor. Our results show that ADJ robustly outperformed the standard complexity penalization and hold-out methods in all cases considered—spanning a wide variety of target functions, noise levels, and domain distributions P_X. Tables 1–5 show the previous data along with the performance characteristics of ADJ. In particular, Tables 4–6 show that ADJ avoids the extreme under-fitting problems that hamper TRI; it appears to responsively select high order approximations when this is supported by the data. Moreover, Tables 1–3 show that ADJ is still extremely robust against over-fitting, even in situations where the standard approaches make catastrophic errors. Overall, this is the best model selection strategy we have observed for these polynomial regression tasks, even though it possesses a weaker guarantee against over-fitting than TRI.

Note that both model selection procedures we propose add little computational overhead to traditional methods, since computing inter-hypothesis distances involves making only a single pass down the reference list of unlabeled examples. This is an advantage over standard hold-out techniques like CVT which repeatedly call the hypothesis generating mechanism to generate pseudo-hypotheses—an extremely expensive operation in many applications.

Finally, we note that ADJ possesses a subtle limitation: the multiplicative re-scaling it employs cannot penalize hypotheses that have zero training error. (Therefore, we had to limit the degree of the polynomials to $t - 2$ in the above experiments to avoid null training errors). However, despite this shortcoming, the ADJ procedure turns out to perform very well in practice and most often outperforms the more straightforward TRI strategy.

3.4. Robustness to unlabeled data

Before moving on to regularization, we briefly investigate the robustness of these model selection techniques to limited amounts of auxiliary unlabeled data. In principle, one can always argue that the preceding empirical results are not useful because the metric-based strategies TRI and ADJ might require significant amounts of unlabeled data to perform well in practice. (However, the 200 unlabeled examples used in the previous experiments does not seem that onerous.) In fact, the previous theoretical results (Propositions 1–3) assumed infinite unlabeled data. To explore the issue of robustness to limited amounts of unlabeled data, we repeated our previous experiments but gave TRI and ADJ only a small auxiliary sample of unlabeled data to estimate inter-hypothesis distances. In this experiment we found that these strategies were actually quite robust to using approximate distances. Table 7 shows that small numbers of unlabeled examples were still sufficient for TRI and ADJ to perform nearly as well as before. Moreover, Table 7 shows that these techniques only seem to significantly degrade once we consider fewer unlabeled than labeled training examples. This robustness was observed across the range of problems considered.

Table 7. Fitting $f(x) = \text{step}(x \geq 0.5)$ with $P_X = U(0, 1)$ and $\sigma = 0.05$ (as in Table 1). This table gives distribution of approximation ratios achieved with $t = 30$ labeled training examples and $r = 500$, $r = 200$, $r = 100$, $r = 50$, $r = 25$ unlabeled examples, showing percentiles of approximation ratios achieved after 1000 repeated trials. The experimental set up of Table 1 is repeated, except that a smaller number of unlabeled examples are used.

	Percentiles of approximation ratios				
$t = 30$	25	50	75	95	100
TRI ($r = 500$)	1.00	1.07	1.19	1.48	2.21
TRI ($r = 200$)	1.00	1.08	1.19	1.45	2.18
TRI ($r = 100$)	1.00	1.08	1.19	1.45	2.49
TRI ($r = 50$)	1.01	1.08	1.19	1.65	7.26
TRI ($r = 25$)	1.01	1.10	1.27	2.74	64.6
ADJ ($r = 500$)	1.06	1.14	1.26	1.51	1.99
ADJ ($r = 200$)	1.06	1.14	1.25	1.51	2.10
ADJ ($r = 100$)	1.07	1.16	1.31	1.67	2.21
ADJ ($r = 50$)	1.07	1.17	1.29	1.58	3.19
ADJ ($r = 25$)	1.09	1.22	1.40	1.85	8.68

In fact, it is a straightforward exercise to theoretically analyze the robustness of these procedures TRI and ADJ to approximation errors in the estimated inter-hypothesis distances. In a model selection sequence $h_0, h_1, \ldots, h_{K-1}$, there are only $K(K-1)/2$ pairwise distances that need to be estimated from unlabeled data. This means that a straightforward "union bound" can be combined with standard uniform convergence results (Anthony & Bartlett, 1999) to obtain an $O(\frac{1}{\sqrt{r}} \ln \frac{K}{\delta})$ error bar on these estimates (at the $1-\delta$ confidence level). These error bars could easily be used to suitably adjust Propositions 1–3 to account for the estimation errors. However, we do not pursue this analysis here since it is straightforward but unrevealing.

Although the empirical results in this section are anecdotal, the paper (Schuurmans, Ungar, & Foster, 1997) pursues a more systematic investigation of the robustness of these procedures and reaches similar conclusions (also based on artificial data). Rather than present a detailed investigation of these model selection strategies in more serious case studies, we first consider a further improvement to the basic method.

4. Regularization

One of the difficulties with model selection is that its generalization behavior depends on the specific decomposition of the base hypothesis class one considers. That is, different decompositions of H can lead to different outcomes. To avoid this issue, we extend the previous ideas to a more general training criterion that uses unlabeled data to decide how to penalize *individual* hypotheses in the global space H. The main contribution of this section is a simple, generic training objective that can be applied to a wide range of supervised learning problems.

Continuing from above, we assume that we have access to a sizable collection of unlabeled data which we now use to globally penalize complex hypotheses. Specifically, we formulate an alternative training criterion that measures the behavior of individual hypotheses on both the labeled and unlabeled data. The intuition behind our criterion is simple—instead of minimizing empirical training error alone, we in addition seek hypotheses that behave *similarly* both on and off the labeled training data. This objective arises from the observation that a hypothesis which fits the training data well but behaves erratically off the labeled training set is not likely to generalize to unseen examples. To detect erratic behavior we measure the distance a hypothesis exhibits to a fixed "origin" function ϕ (chosen arbitrarily) on both data sets. If a hypothesis is behaving erratically off the labeled training set then it is likely that these distances will disagree. This effect is demonstrated in figure 8 for two large degree polynomials that fit the labeled training data well, but differ dramatically in their true error and their differences between on and off training set distance to a simple origin function. (Note that we will use trivial origin functions throughout this section, such as the zero function $\phi = 0$ or the constant function $\phi = \bar{y}$ at the mean of the y labels).

To formulate a concrete training objective we first propose the following tentative measures: empirical training error plus an additive penalty

$$d(\widehat{h, \mathrm{P}_{Y|X}}) + d(h, \phi) - d(\widehat{h, \phi}) \tag{12}$$

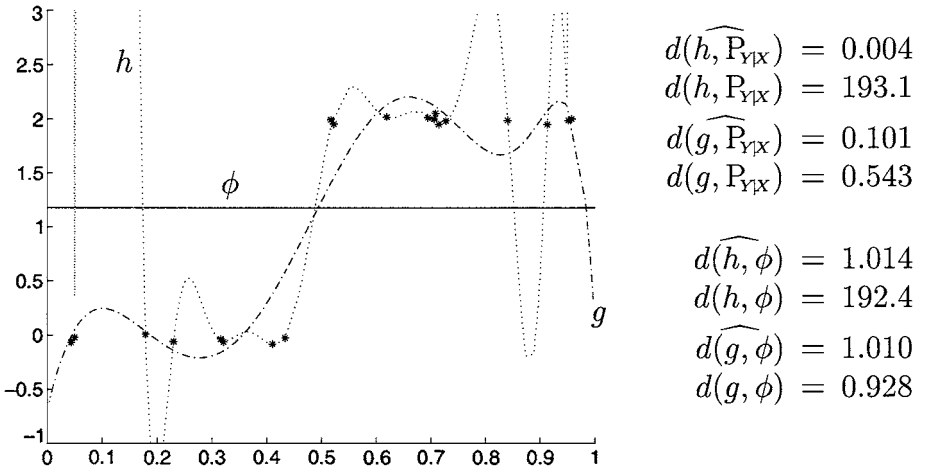


Figure 8. Two nineteenth degree polynomials h and g that fit 20 given training points. Here h approximately minimizes $d(\widehat{h, \mathrm{P}_{Y|X}})$, whereas g optimizes an alternative training criterion defined in (13). This plot demonstrates how the labeled training data estimate $d(\widehat{g, \mathrm{P}_{Y|X}})$ for the smoother polynomial g is much closer to its true distance $d(g, \mathrm{P}_{Y|X})$. However, for both functions the proximity of the estimated errors $d(\cdot, \widehat{\mathrm{P}_{Y|X}})$ to the true errors $d(\cdot, \mathrm{P}_{Y|X})$ appear to be reflected on the relative proximity of the estimated distances $d(\widehat{\cdot, \phi})$ to the true distances $d(\cdot, \phi)$ to the simple constant origin function ϕ.

and empirical error times a multiplicative penalty

$$d(\widehat{h, \mathrm{P}_{Y|X}}) \times \frac{d(h, \phi)}{d(\widehat{h, \phi})} \tag{13}$$

In each case we compare the behavior of a candidate hypothesis h to the fixed origin ϕ. Thus, in both cases we seek to minimize empirical training error $d(h, \widehat{\mathrm{P}_{Y|X}})$ plus (or times) a penalty that measures the discrepancy between the distance to the origin on the labeled training data and the distance to the origin on unlabeled data. The regularization effect of these criteria is illustrated in figure 8. Somewhat surprisingly, we have found that the *multiplicative* objective (13) generally performs much better than (12), as it more harshly penalizes discrepancies between on and off training set behavior. Therefore, this is the form we adopt below.

Although these training criteria might appear to be ad hoc, they are not entirely unprincipled. One useful property they have is that if the origin function ϕ happens to be equal to the target conditional $\mathrm{P}_{Y|X}$, then minimizing (12) or (13) becomes equivalent to minimizing the true prediction error $d(h, \mathrm{P}_{Y|X})$. However, despite the utility of this technique, it turns out that these initial training objectives have the inherent drawback that they subtly bias the final hypotheses towards the origin function ϕ. That is, both (12) and (13) allow minima that have "artificially" large origin distances on the labeled data $d(\widehat{h, \phi})$ and simultaneously small distances on unlabeled data $d(h, \phi)$. For example, this is illustrated in figure 8 for a hypothesis function g that minimizes (13) but is clearly attracted to the origin ϕ at the right end of the domain (off of the labeled training data). Of course, such a bias towards ϕ can be desirable if ϕ happens to be near the target conditional $\mathrm{P}_{Y|X}$. In this sense, ϕ could serve as a useful prior on hypotheses. However, there is no reason to expect ϕ to be anywhere near $\mathrm{P}_{Y|X}$ in practice, especially when considering the trivial constant functions used in this paper.

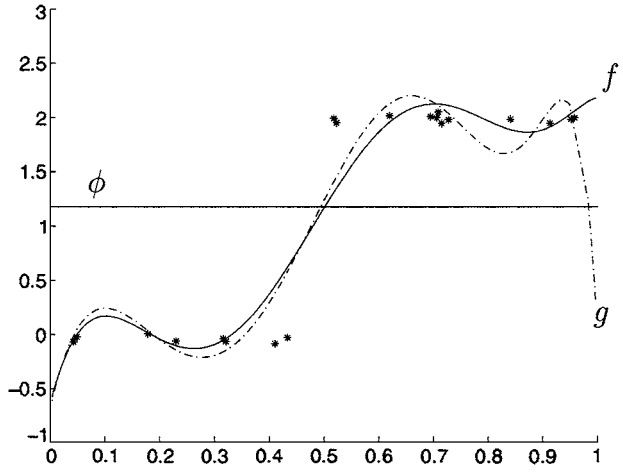
Nevertheless, there is an intuitive way to counter this difficulty: to avoid the bias towards ϕ, we introduce *symmetric* forms of the previous criteria that also penalize hypotheses which are unnaturally *close* to the origin off the labeled data. That is, one could consider a symmetrized form of the additive penalty (12)

$$d(\widehat{h, \mathrm{P}_{Y|X}}) + \left| d(h, \phi) - d(\widehat{h, \phi}) \right| \tag{14}$$

as well as a symmetrized form of the multiplicative penalty (13)

$$d(\widehat{h, \mathrm{P}_{Y|X}}) \times \max\left(\frac{d(h, \phi)}{d(\widehat{h, \phi})}, \frac{d(\widehat{h, \phi})}{d(h, \phi)} \right) \tag{15}$$

These penalties work in both directions: hypotheses that are much further from the origin on the training data than off are penalized, but so are hypotheses that are significantly *closer* to the origin on the training data than off. The rationale behind this symmetric criterion is that both types of erratic behavior indicate that the observed training error is likely to be an unrepresentative reflection of the hypothesis's true error. The value of this intuition is demonstrated in figure 9, where the hypothesis f that minimizes the symmetric criterion



$$\widehat{d(g, \mathrm{P}_{Y|X})} = 0.101$$
$$d(g, \mathrm{P}_{Y|X}) = 0.543$$
$$\widehat{d(f, \mathrm{P}_{Y|X})} = 0.098$$
$$d(f, \mathrm{P}_{Y|X}) = 0.488$$

$$\widehat{d(g, \phi)} = 1.010$$
$$d(g, \phi) = 0.928$$
$$\widehat{d(f, \phi)} = 1.011$$
$$d(f, \phi) = 1.023$$

Figure 9. A comparison of the asymmetric and symmetrized training objectives. Here g is the nineteenth degree polynomial which minimizes the original asymmetric criterion (13) on 20 data points, whereas f minimizes the symmetrized criterion (15). This plot shows how g is inappropriately drawn towards the origin ϕ near the right end of the interval, whereas f behaves neutrally with respect to ϕ.

(15) is not drawn towards the origin inappropriately, and thereby achieves a smaller true prediction error than the hypothesis g that minimizes (13).

These symmetric training criteria can also be given a technical justification: First, if the origin function ϕ happens to be equal to the target conditional $\mathrm{P}_{Y|X}$, then minimizing either (14) or (15) comes very close to minimizing the true prediction error $d(h, \mathrm{P}_{Y|X})$. To see this for the multiplicative criterion (15), let h be the hypothesis that achieves the minimum and note that if $\widehat{d(h, \mathrm{P}_{Y|X})} \leq d(h, \mathrm{P}_{Y|X})$ the criterion becomes equivalent to $\widehat{d(h, \mathrm{P}_{Y|X})} d(h, \mathrm{P}_{Y|X}) / \widehat{d(h, \mathrm{P}_{Y|X})} = d(h, \mathrm{P}_{Y|X})$, and otherwise if $\widehat{d(h, \mathrm{P}_{Y|X})} > d(h, \mathrm{P}_{Y|X})$ the criterion becomes equivalent to $d(h, \mathrm{P}_{Y|X}) r^2$ for $r = \widehat{d(h, \mathrm{P}_{Y|X})} / d(h, \mathrm{P}_{Y|X})$. In the latter case, since h minimizes (15) we must have $d(h, \mathrm{P}_{Y|X}) < d(h, \mathrm{P}_{Y|X}) r^2 \leq d(h^*, \mathrm{P}_{Y|X}) r^{*2}$ for the Bayes optimal hypothesis h^*. But since h^* is not directly optimized on the training set (it remains fixed), we will usually have $d(h^*, \mathrm{P}_{Y|X}) \approx \widehat{d(h^*, \mathrm{P}_{Y|X})}$ and hence $r^* \approx 1$, which means that $d(h, \mathrm{P}_{Y|X})$ will tend to be close to $d(h^*, \mathrm{P}_{Y|X})$. Thus, minimizing (15) will result in near optimal generalization performance in this scenario. (Note that this property would not hold for naively smoothed versions of this objective.)

In the more general case where the origin does not match the target, the symmetric criteria will also still provably penalize hypotheses that have small training error and large test error. To see this for (15), note that for any hypothesis h

$$\frac{d(h, \phi)}{\widehat{d(h, \phi)}} \geq \frac{d(h, \mathrm{P}_{Y|X}) - d(\phi, \mathrm{P}_{Y|X})}{\widehat{d(h, \mathrm{P}_{Y|X})} + \widehat{d(\phi, \mathrm{P}_{Y|X})}} \tag{16}$$

by the triangle inequality. Since ϕ and $\mathrm{P}_{Y|X}$ are not optimized on the training set we can expect $\widehat{d(\phi, \mathrm{P}_{Y|X})} \approx d(\phi, \mathrm{P}_{Y|X})$ for moderate sample sizes. Thus, (16) shows that if $\widehat{d(h, \mathrm{P}_{Y|X})}$

is small (say, less than $d(\phi, \mathrm{P}_{Y|X})$) and $d(h, \mathrm{P}_{Y|X})$ is large (greater than $k \times d(\phi, \mathrm{P}_{Y|X})$, $k \geq 3$), then h's training error must be penalized by a significant ratio (at least $\frac{k-1}{2}$). By contrast, an alternative hypothesis g that achieves comparable training error and yet exhibits balanced behavior on and off the labeled training set (that is, such that $d(g, \widehat{\mathrm{P}_{Y|X}}) \approx d(g, \mathrm{P}_{Y|X})$) will be strongly preferred; in fact, such a g cannot over-fit by the same amount as h without violating (16). Importantly, the Bayes optimal hypothesis h^* will also tend to have $d(h^*, \widehat{\mathrm{P}_{Y|X}}) \approx d(h^*, \mathrm{P}_{Y|X})$ and $d(\widehat{h^*}, \phi) \approx d(h^*, \phi)$ since it too does not depend on the training set. Thus, h^* will typically achieve a small value of the objective, which will force any hypothesis that has a large over-fitting error (relative to $d(\phi, \mathrm{P}_{Y|X})$) to exhibit an objective value greater than the minimum.

Note that the sensitivity of the lower bound (16) clearly depends on the distance between the origin and the target. If the origin is too far from the target then the lower bound is weakened and the criterion (15) becomes less sensitive to over-fitting. However, our experiments show that the objective is not unduly sensitive to the choice of ϕ, so long as is not too far from the data. In fact, even simple constant functions generally suffice.[4]

The outcome is a new regularization procedure that uses the training objective (15) to penalize hypotheses based on the given training data and on the unlabeled data. The resulting procedure, in effect, uses the unlabeled data to automatically set the level of regularization for a given problem. Our goal is to apply the new training objective to various hypothesis classes and see if it regularizes effectively across different data sets. We demonstrate this for several classes below. However, the regularization behavior is even subtler: Since the penalization factor in (15) also depends on the specific labeled training set under consideration, the resulting procedure regularizes in a data dependent way. That is, the procedure adapts the penalization to the particular set of observed data. This raises the possibility of outperforming any regularization scheme that keeps a fixed penalization level across different training samples drawn from the same problem. In fact, we demonstrate below that such an improvement can be achieved in realistic hypothesis classes on real data sets.

4.1. Example: Polynomial regression

The first supervised learning task we consider is the polynomial regression problem considered in Section 3.2. The regularizer introduced above (15) turns out to perform very well in such problems. In this case, our training objective can be expressed as choosing a hypothesis to minimize

$$\sum_{i=1}^{t} (h(x_i) - y_i)^2/t \times \max\left(\frac{\sum_{j=1}^{r} (h(x_j) - \phi(x_j))^2/r}{\sum_{i=1}^{t} (h(x_i) - \phi(x_i))^2/t}, \frac{\sum_{i=1}^{t} (h(x_i) - \phi(x_i))^2/t}{\sum_{j=1}^{r} (h(x_j) - \phi(x_j))^2/r} \right)$$

where $\{\langle x_i, y_i \rangle\}_{i=1}^{t}$ is the set of labeled training data, $\{\langle x_j \rangle\}_{j=1}^{r}$ is a set of unlabeled examples, and ϕ is a fixed origin (which we usually just set to be the constant function at the mean of the y labels). Note again that this training objective seeks hypotheses that fit the labeled training data well while simultaneously behaving similarly on the labeled and unlabeled data.

To test the basic effectiveness of our approach, we repeated the experiments of Section 3.2. The first class of methods we compared against were the same *model selection* methods considered before: 10-fold cross validation CVT, structural risk minimization SRM (Cherkassky, Mulier, & Vapnik, 1997), RIC (Foster & George, 1994), SMS (Shibata, 1981), GCV (Craven & Wahba, 1979), BIC (Schwarz, 1978), AIC (Akaike, 1974), CP (Mallows, 1973), FPE (Akaike, 1970), and the metric based model selection strategy, ADJ, introduced in Section 3.3. However, since none of the statistical methods, RIC, SMS, GCV, BIC, AIC, CP, FPE, performed competitively in our experiments, we report results only for GCV which performed the best among them. For comparison, we also report results for the optimal model selector OPT* which makes an oracle choice of the best available hypothesis in any given model selection sequence. In these experiments, the model selection methods considered polynomials of degree 0 to $t - 2$.[5]

The second class of methods we compared against were *regularization* methods, which consider polynomials of maximum degree $(t-2)$ but penalize individual polynomials based on the size of their coefficients or their smoothness properties. The specific methods we considered were: a standard form of "ridge" penalization (or weight decay) which places a penalty $\lambda \sum_k a_k^2$ on polynomial coefficients a_k (Cherkassky & Mulier, 1998), and Bayesian *maximum a posteriori* inference with zero-mean Gaussian priors on polynomial coefficients a_k with diagonal covariance matrix λI (MacKay, 1992).[6] Both of these methods require a regularization parameter λ to be set by hand. We refer to these methods as REG and MAP respectively.

To test the ability of our technique to automatically set the regularization level we tried a range of (fourteen) regularization parameters λ for the fixed regularization methods REG and MAP. For comparison purposes, we also report the results of the oracle regularizers, REG* and MAP*, which select the best λ value for each training set. Our experiments were conducted by repeating the experimental conditions of Section 3.2. Specifically, Table 8 repeats Table 1 (fitting a step function), Table 9 repeats Table 3 (fitting $\sin(1/x)$), Table 10 repeats Table 4 (fitting $\sin^2(2\pi x)$), and Table 11 repeats Table 5 (fitting a fifth degree polynomial). The regularization criterion based on minimizing (15) is listed as ADA in our figures (for "adaptive" regularization).[7] We also tested ADA using different origin functions $\phi = \text{mean}\, y, \max y, 2 \max y, 4 \max y, 8 \max y$ to examine its robustness to ϕ, and also tested the one-sided version of ADA (13) to verify the benefits of the symmetrized criterion (15) over (13).

The results once again are quite positive. The first observation is that the model selection methods generally did not fare as well as the regularization techniques on these problems. Model selection seems prone to making catastrophic over-fitting errors in these polynomial regression problems, whereas regularization appears to retain robust control. As noted, even the frequently trusted 10-fold cross validation procedure CVT did not fare well in our experiments. The only model selection strategy to perform reasonably well (besides the oracle model selector OPT*) was the metric-based method ADJ, which also exploits unlabeled data.

The new adaptive regularization scheme ADA performed the best among all procedures in these experiments. Tables 8–11 show that it outperforms the fixed regularization strategies (REG and MAP) for all fixed choices of regularization parameter λ, even though the optimal

Table 8. Fitting $f(x) = \text{step}(x \geq 0.5)$ with $P_X = U(0, 1)$ and $\sigma = 0.05$. Test errors (true distances) achieved at training sample size $t = 20$, using $r = 200$ auxiliary unlabeled examples for the metric procedures ADA and ADJ, results of 1000 repeated trials. This repeats the conditions of Table 1.

		Absolute test errors (distance)		
		Mean	Median	Stdev
ADA (15)	$\phi = \text{mean } y$	0.391	0.366	0.113
	$\phi = 2 \max y$	0.460	0.355	0.319
	$\phi = 4 \max y$	0.556	0.367	0.643
	$\phi = 8 \max y$	0.596	0.369	1.004
Asymmetric (13)		0.403	0.378	0.111
REG	$\lambda = 10^{-9}$	7.940	0.664	38.50
	$\lambda = 10^{-7}$	3.930	0.469	13.10
	$\lambda = 10^{-5}$	2.570	0.457	8.360
	$\lambda = 10^{-4}$	1.750	0.441	5.640
	$\lambda = 10^{-3}$	1.050	0.388	2.620
	$\lambda = 10^{-2}$	0.697	0.397	0.825
	$\lambda = 10^{-1}$	0.529	0.407	0.480
	$\lambda = 0.5$	0.495	0.416	0.243
	$\lambda = 1.0$	**0.483**	0.468	0.048
	$\lambda = 5.0$	0.512	0.498	0.050
	$\lambda = 50$	0.554	0.541	0.042
REG*		0.371	0.355	0.049
MAP*		0.496	0.400	0.385
Model sel	OPT*	0.387	0.374	0.076
	ADJ	0.458	0.466	0.112
	CVT	14.90	0.420	340.0
	SRM	29.00	0.510	311.0
	GCV	3.2e5	51.9	3.1e6

choice varies across problems (MAP was inferior to REG in these experiments, and therefore we do not report detailed results). This demonstrates that ADA is able to effectively tune its penalization behavior to the problem at hand. Moreover, since it outperforms even the best choice of λ for each data set, ADA also demonstrates the ability to adapt its penalization behavior to the specific training set, not just the given problem. In fact, ADA is competitive with the oracle regularizers REG* and MAP* in these experiments, and even outperformed the oracle model selection strategy OPT* on two problems. It is clear that ADA is fairly robust to the choice of ϕ, since moving ϕ to a distant constant origin (even up to eight times the max y value) did not completely damage its performance. The results also show that the one-sided version of ADA based on (13) is inferior to the symmetrized version in these experiments, confirming our prior expectations.

Table 9. Fitting $f(x) = \sin(1/x)$ with $P_X = U(0, 1)$ and $\sigma = 0.05$. Test errors (true distances) achieved at training sample size $t = 20$, using $r = 200$ auxiliary unlabeled examples for the metric procedures ADA and ADJ, results of 1000 repeated trials. This repeats the conditions of Table 3.

		Absolute test errors (distance)		
		Mean	Median	Stdev
ADA (15)	$\phi = \text{mean } y$	0.444	0.425	0.085
	$\phi = 2 \max y$	0.495	0.436	0.171
	$\phi = 4 \max y$	0.533	0.427	0.326
	$\phi = 8 \max y$	0.591	0.426	0.639
Asymmetric (13)		0.466	0.439	0.102
REG	$\lambda = 10^{-9}$	4.250	0.758	28.00
	$\lambda = 10^{-7}$	3.250	0.588	28.50
	$\lambda = 10^{-5}$	1.830	0.588	12.80
	$\lambda = 10^{-4}$	1.060	0.486	2.640
	$\lambda = 10^{-3}$	0.774	0.489	1.560
	$\lambda = 10^{-2}$	0.558	0.452	0.550
	$\lambda = 10^{-1}$	0.514	0.464	0.156
	$\lambda = 0.5$	0.488	0.459	0.104
	$\lambda = 1.0$	**0.484**	0.473	0.040
	$\lambda = 5.0$	0.494	0.485	0.032
	$\lambda = 50$	0.509	0.502	0.029
REG*		0.429	0.424	0.041
MAP*		0.651	0.476	0.989
Model sel	OPT*	0.433	0.427	0.049
	ADJ	0.712	0.504	0.752
	CVT	2.410	0.516	14.20
	SRM	29.40	0.781	469.0
	GCV	1.4e5	11.3	2.6e6

4.2. Example: Radial basis function regression

To test our approach on a more realistic task, we considered the problem of regularizing radial basis function (RBF) networks for regression. RBF networks are a natural generalization of interpolation and spline fitting techniques. Given a set of prototype centers $c_1, \ldots, c_k$, an RBF representation of a prediction function h is given by

$$h(x) = \sum_{i=1}^{k} w_i g\left(\frac{\|x - c_i\|}{\sigma}\right) \tag{17}$$

where $\|x - c_i\|$ is the Euclidean distance between x and center c_i, and g is a response function with width parameter σ. In this experiment we use a standard local (Gaussian) basis function $g(z) = e^{-z^2/\sigma^2}$.

Table 10. Fitting $f(x) = \sin^2(2\pi x)$ with $P_X = U(0, 1)$ and $\sigma = 0.05$. Test errors (true distances) achieved at training sample size $t = 20$, using $r = 200$ auxiliary unlabeled examples for the metric procedures ADA and ADJ, results of 1000 repeated trials. This repeats the conditions of Table 4.

		Absolute test errors (distance)		
		Mean	Median	Stdev
ADA (15)	$\phi = \text{mean } y$	0.107	0.081	0.066
	$\phi = 2 \max y$	0.137	0.083	0.168
	$\phi = 4 \max y$	0.157	0.084	0.273
	$\phi = 8 \max y$	0.230	0.084	0.844
Asymmetric (13)		0.111	0.087	0.060
REG	$\lambda = 10^{-9}$	0.964	0.115	3.850
	$\lambda = 10^{-7}$	0.797	0.124	3.120
	$\lambda = 10^{-5}$	0.660	0.159	2.370
	$\lambda = 10^{-4}$	0.714	0.181	1.570
	$\lambda = 10^{-3}$	0.582	0.237	1.150
	$\lambda = 10^{-2}$	0.446	0.212	0.940
	$\lambda = 10^{-1}$	0.509	0.291	0.500
	$\lambda = 0.5$	0.405	0.355	0.145
	$\lambda = 1.0$	0.358	0.342	0.066
	$\lambda = 5.0$	**0.353**	0.341	0.040
	$\lambda = 50$	**0.353**	0.342	0.033
REG*		0.140	0.092	0.099
MAP*		0.496	0.232	0.983
Model sel	OPT*	0.122	0.085	0.086
	ADJ	0.188	0.114	0.150
	CVT	0.559	0.132	1.980
	SRM	0.576	0.128	2.430
	GCV	4.8e3	0.227	5.6e4

Fitting with RBF networks is straightforward. The simplest approach is to place a prototype center on each training example and then determine the weight vector **w** that allows the network to fit the training y labels. The best fit weight vector can be obtained by solving for **w** in

$$\begin{bmatrix} g\left(\frac{\|x_1 - x_1\|}{\sigma}\right) & \cdots & g\left(\frac{\|x_1 - x_t\|}{\sigma}\right) \\ \vdots & & \vdots \\ g\left(\frac{\|x_t - x_1\|}{\sigma}\right) & \cdots & g\left(\frac{\|x_t - x_t\|}{\sigma}\right) \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_t \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_t \end{bmatrix}$$

Table 11. Fitting a fifth degree polynomial $f(x)$ with $P_X = U(0, 1)$ and $\sigma = 0.05$. Test errors (true distances) achieved at training sample size $t = 20$, using $r = 200$ auxiliary unlabeled examples for the metric procedures ADA and ADJ, results of 1000 repeated trials. This repeats the conditions of Table 5.

		Absolute test errors (distance)		
		Mean	Median	Stdev
ADA (15)	$\phi = \text{mean } y$	0.077	0.060	0.090
	$\phi = 2 \max y$	0.073	0.059	0.054
	$\phi = 4 \max y$	0.072	0.059	0.056
	$\phi = 8 \max y$	0.075	0.059	0.115
Asymmetric (13)		0.110	0.074	0.088
REG	$\lambda = 10^{-9}$	0.753	0.099	2.850
	$\lambda = 10^{-7}$	0.514	0.094	1.780
	$\lambda = 10^{-5}$	0.440	0.118	1.330
	$\lambda = 10^{-4}$	0.462	0.195	1.030
	$\lambda = 10^{-3}$	0.558	0.225	1.190
	$\lambda = 10^{-2}$	0.524	0.360	0.539
	$\lambda = 10^{-1}$	**0.454**	0.337	0.508
	$\lambda = 0.5$	0.523	0.396	0.337
	$\lambda = 1.0$	0.532	0.499	0.086
	$\lambda = 5.0$	0.520	0.511	0.038
	$\lambda = 50$	0.519	0.513	0.030
REG*		0.147	0.082	0.121
MAP*		0.460	0.352	0.511
Model sel	OPT*	0.071	0.060	0.071
	ADJ	0.116	0.062	0.188
	CVT	0.321	0.065	3.160
	SRM	0.163	0.062	1.230
	GCV	2421	0.072	4.2e4

(the solution is guaranteed to exist and be unique for distinct training points and most natural basis functions g, including the Gaussian basis used here (Bishop, 1995)).

Although exactly fitting data with RBF networks is natural, it has the problem that it generally over-fits the training data in the process of replicating the y labels. Many approaches therefore exist for regularizing RBF networks. However, these techniques are often hard to apply because they involve setting various free parameters or controlling complex methods for choosing prototype centers, etc. (Cherkassky & Mulier, 1998; Bishop, 1995). The simplest regularization approaches are to add a ridge penalty to the weight vector, and minimize

$$\sum_{i=1}^{t} (h(x_i) - y_i)^2 + \lambda \sum_{i=1}^{t} w_i^2 \tag{18}$$

where h is given as in (17) (Cherkassky & Mulier, 1998). An alternative approach is to add a non-parametric penalty on curvature (Poggio & Girosi, 1990), but the resulting procedure is similar. To apply these methods in practice one has to make an intelligent choice of the width parameter σ and the regularization parameter λ. Unfortunately, these choices interact and it is often hard to set them by hand without extensive visualization and experimentation with the data set.

In this section we investigate how effectively the ADA regularizer is able to automatically select the width parameter σ and regularization parameter λ in an RBF network on real regression problems. Here the basic idea is to use unlabeled data to make these choices automatically and adaptively. We compare ADA (15) to a large number of ridge regularization procedures, each corresponding to the penalty (18) with different

Table 12. RBF results showing mean test errors (distances) on the AAUP data set (1074 instances on 12 independent attributes). Results are averaged over 100 splits of the dataset (1/10 train, 7/10 unlabeled, 2/10 test), with standard deviations given for ADA and REG*.

AAUP data set					
ADA (15) 0.0197 ± 0.004			REG* 0.0329 ± 0.009		
REG	$\lambda = 0.0$	0.1	0.25	0.5	1.0
$\sigma = 0.0001$	0.0400	0.0479	0.0508	0.0535	0.0566
0.0005	0.0363	0.0447	0.0482	0.0515	0.0554
0.001	0.0353	0.0435	0.0475	0.0512	0.0554
0.0025	**0.0350**	0.0425	0.0473	0.0514	0.0555
0.005	0.0359	0.0423	0.0475	0.0516	0.0554
0.0075	0.0368	0.0424	0.0478	0.0517	0.0553
0.01	0.0376	0.0426	0.0480	0.0518	0.0551

Table 13. RBF results showing mean test errors (distances) on the ABALONE data set (1000 instances on 8 independent attributes). Results are averaged over 100 splits of the dataset (1/10 train, 7/10 unlabeled, 2/10 test), with standard deviations given for ADA and REG*.

ABALONE data set					
ADA (15) 0.034 ± 0.0046			REG* 0.049 ± 0.0063		
REG	$\lambda = 0.0$	0.1	0.25	0.5	1.0
$\sigma = 4$	0.4402	0.04954	0.04982	0.05008	0.05061
6	0.3765	0.04952	0.04979	0.05007	0.05063
8	0.3671	**0.04951**	0.04979	0.05007	0.05069
10	0.3474	0.04952	0.04979	0.05007	0.05073
12	0.3253	0.04953	0.04979	0.05008	0.05079
14	0.5702	0.04954	0.04979	0.05009	0.05084
16	1.0549	0.04954	0.04980	0.05010	0.05089

fixed choices of σ and λ (thirty five in total). To apply ADA in this case we simply ran a standard optimizer over the parameter space (σ, λ) while explicitly solving for the $\mathbf{w}$ vector that minimizes (18) for each choice of σ and λ (which involves solving a linear system (Cherkassky & Mulier, 1998; Bishop, 1995)). Thus, given σ, λ and $\mathbf{w}$ we could calculate (15) and supply the resulting value to the optimizer as the objective to be minimized (cf. Footnote 7).

To conduct an experiment we investigated a number of regression problems from the StatLib and UCI machine learning repositories.[8] In our experiments, a data set was randomly split into a training (1/10), unlabeled (7/10), and test set (2/10), and then each of the methods was run on this split. We repeated the random splits 100 times to obtain our results. Tables 12–15 show that ADA regularization is able to choose width and regularization

Table 14. RBF results showing mean test errors (distances) on the BODYFAT data set (252 instances on 14 independent attributes). Results are averaged over 100 splits of the dataset (1/10 train, 7/10 unlabeled, 2/10 test), with standard deviations given for ADA and REG*.

BODYFAT data set					
ADA (15) 0.131 ± 0.0171			REG* 0.125 ± 0.0151		
REG	$\lambda = 0.0$	0.1	0.25	0.5	1.0
$\sigma = 0.05$	0.1623	0.1303	0.1328	0.1344	0.1357
0.1	0.1658	0.1299	0.1325	0.1341	0.1354
0.5	0.1749	0.1294	0.1321	0.1337	0.1352
1	0.1792	**0.1294**	0.1321	0.1336	0.1353
2	0.1837	0.1296	0.1322	0.1337	0.1356
4	0.1883	0.1299	0.1323	0.1339	0.1362
6	0.1910	0.1301	0.1325	0.1340	0.1366

Table 15. RBF results showing mean test errors (distances) on the BOSTON-C data set (506 instances on 12 independent attributes). Results are averaged over 100 splits of the dataset (1/10 train, 7/10 unlabeled, 2/10 test), with standard deviations given for ADA and REG*.

BOSTON-C data set					
ADA (15) 0.150 ± 0.0212			REG* 0.151 ± 0.0197		
REG	$\lambda = 0.0$	0.1	0.25	0.5	1.0
$\sigma = 0.01$	0.1611	0.15908	0.1622	0.1650	0.1684
0.05	0.1614	0.15798	0.1615	0.1645	0.1679
0.075	0.1619	0.15785	0.1614	0.1645	0.1679
0.1	0.1624	0.15779	0.1614	0.1645	0.1679
0.15	0.1633	**0.15776**	0.1615	0.1646	0.1680
0.2	0.1642	0.15777	0.1615	0.1647	0.1682
0.25	0.1649	0.15780	0.1616	0.1648	0.1683

parameters that achieve effective generalization performance across a range of data sets. Here ADA performs better than any fixed regularizer on every problem (except BODYFAT), and even beats the oracle regularizer REG* on all but one problem. This shows that the adaptive criterion is not only effective at choosing good regularization parameters for a given problem, it can choose them adaptively based on the given training data to yield improvements over fixed regularizers.

5. Classification

Finally, we note that the regularization approach developed in this paper can also be easily applied to classification and conditional density estimation problems. In conditional density estimation, one can use KL divergence as a proxy distance measure and still achieve interesting results (however we do not report these experiments here).

In classification, the label set Y is usually a small discrete set and we measure prediction error by the misclassification loss, $err(\hat{y}, y) = 1_{(\hat{y} \neq y)}$. Here, distances are measured by the disagreement probability $d(f, g) = \mathrm{P}_X(f(x) \neq g(x))$. Using this metric, our generic regularization objective (15) can be directly applied to classification problems. In fact, we have applied (15) to the problem of decision tree pruning in classification, obtaining the results shown in Table 16. Unfortunately, the results achieved in this experiment are not strong, and it appears that the techniques proposed in this paper may not work as decisively for classification problems as they do for regression and conditional density estimation problems.

We believe that the weakness of the proposed methods for classification might have an intuitive explanation however: Since classification functions are essentially histogram-like (i.e., piecewise constant), they limit the ability of unlabeled data to detect erratic behavior off the labeled training sample. This is because histograms, being flat across large regions, tend to behave similarly in large neighborhoods around training points—to the extent that

Table 16. Some decision tree pruning results on UCI repository data sets, showing size and test error over 100 splits.

	Un-pruned		C4.5 Pruned		ADA-pruned	
	Size	Test	Size	Test	Size	Test
random	120	50.5	105	50.5	51	50.2
optdigit	269	15.3	250	15.2	234	15.2
iris	7	8.9	6	8.8	6	9.3
glass	11	10.8	11	10.8	10	12.8
ecoli	32	24.1	22	22.4	22	23.6
vote	21	6.7	8	5.2	14	6.9
crx	56	19.8	28	18.0	23	17.3
soybean	146	19.7	75	17.5	124	19.7
hypo	25	0.94	19	0.83	23	0.87

distances on labeled and unlabeled data points are often very similar, even for complex histograms. Coping with this apparent limitation in our approach remains grounds for future research.

6. Conclusion

We have introduced a new approach to the classical complexity-control problem that is based on exploiting the intrinsic geometry of the function learning task. These new techniques seem to outperform standard approaches in a wide range of regression problems. The primary source of this advantage is that the proposed metric-based strategies are able to detect dangerous situations and avoid making catastrophic over-fitting errors, while still being responsive enough to adopt reasonably complex models when this is supported by the data. They accomplish this by attending to the real distances between hypotheses. (Standard complexity-penalization strategies completely ignore this information. Hold-out methods implicitly take some of this information into account, but do so indirectly and less effectively than the metric-based strategies introduced here.) Although there is no "free lunch" in general (Schaffer, 1994) and we cannot claim to obtain a universal improvement for every complexity-control problem (Schaffer, 1993), we claim 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 for regression support this view.

A substantial body of literature has investigated unlabeled data in the context of supervised learning, although not in the same way we have considered in this paper. Most work in this area adopts the perspective of parametric probability modeling and uses unlabeled data as part of a maximum likelihood (EM) or discriminative training procedure (Miller & Uyar, 1997; Castelli & Cover, 1996; Ratsaby & Venkatesh, 1995; Gutfinger & Sklansky, 1991; O'Neill, 1978). Another common idea is to supply artificial labels to unlabeled examples and use this data directly in a supervised training procedure (Blum & Mitchell, 1998; Towell, 1996). Unlabeled examples can also be used to construct a "cover" of the hypothesis space and improve some worst case bounds on generalization error (Lugosi & Pinter, 1996). However, none of this previous research explicitly uses unlabeled data for automated complexity control. Perhaps the closest work in spirit to ours is Krogh and Vedelsby (1995) which uses unlabeled examples to calculate optimal combination weights in an ensemble of regressors. The emphasis in Krogh and Vedelsby (1995) is on model combination rather than model selection and regularization, but nevertheless there appears to be a close relationship between their ideas and ours.

An important direction for future research is to develop theoretical support for our strategies—in particular, a stronger theoretical justification of the regularization methods proposed in Section 4 and an improved analysis of the model selection methods proposed in Section 3. It remains open as to whether the proposed methods TRI, ADJ, and ADA are in fact the best possible ways to exploit the hypothesis distances provided by P_X. We plan to continue investigating alternative strategies which could potentially be more effective in this regard. For example, it remains future work to extend the multiplicative ADJ and ADA methods to cope with zero training errors. Finally, it would be interesting to adapt

the approach to model combination methods, extending the ideas of Krogh and Vedelsby (1995) to other combination strategies, including boosting (Freund & Schapire, 1997) and bagging (Breiman, 1996).

Acknowledgements

Research supported by NSERC, MITACS, CITO and BUL. Thanks to Yoshua Bengio, Adam Grove, Rob Holte, John Lafferty, Joel Martin, John Platt, Lyle Ungar, Jason Weston and anonymous referees for very helpful comments at various stages of this research.

Notes

1. Prediction error is not the only criteria one could imagine optimizing in model selection. For example, one could be interested in finding a simple model of the underlying phenomenon that gives some insight into its fundamental nature, rather than simply producing a function that predicts well on future test examples (Heckerman & Chickering, 1996). However, we will focus on the traditional machine learning goal of minimizing prediction error.
2. One could consider more elaborate strategies that choose hypotheses from outside the sequence; e.g., by averaging several hypotheses together (Krogh & Vedelsby, 1995; Opitz & Shavlik, 1996; Breiman, 1996). However, as mentioned, we will not pursue this idea in this paper.
3. Although one might suspect that the large failures could be due to measuring relative instead of absolute error, it turns out that all of these large relative errors also correspond to large absolute errors—which we verify in Section 4.1 below.
4. One could easily imagine trying more complex origin functions such as low dimensional polynomials or smooth interpolant functions. We did not explore these ideas in this paper, primarily because we wished to emphasize the robustness of the method to even very simple choices of origin. However, one extension that we did investigate was to use a *set* of origin functions $\phi_1, \ldots, \phi_n$ and penalize according to the maximum ratio—but this did not yield any significant improvements.
5. Note that we restricted the degree to be less than $t-1$ to prevent the maximum degree polynomials from achieving zero training error, which as discussed in Section 3, destroys the regularization effect of the multiplicative penalty.
6. We did not test the more elaborate approach to Bayesian learning of polynomials described in Young (1977).
7. We used a standard optimization routine (Matlab 5.3 "fminunc") to determine coefficients that minimize (14) and (15). Although the nondifferentiability of (15) creates difficulty for the optimizer, it does not prevent reasonable results from being achieved. Another potential problem could arise if h gets close to the origin ϕ. However, since we chose simple origins that were never near $P_{Y|X}$, h was not drawn near ϕ in our experiments and thus the resultant numerical instability did not arise.
8. The URLs are lib.stat.cmu.edu and www.ics.uci.edu/~mlearn/MLRepository.html.

References

Akaike, H. (1970). Statistical predictor information. *Annals of the Institute of Statistical Mathematics, 22*, 203–271.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control, 19*, 716–723.

Anthony, M., & Bartlett, P. (1999). *Neural network learning: Theoretical foundations*. Cambridge: Cambridge University Press.

Bishop, C. (1995). *Neural networks for pattern recognition*. Oxford: Clarendon Press.

Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings Annual Conference on Computational Learning Theory, COLT-98* (pp. 92–100).

Breiman, L. (1996). Bagging predictors. *Machine Learning, 24*, 123–140.

Castelli, V., & Cover, T. (1996). The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter. *IEEE Transactions on Information Theory, 42:6*, 2102–2117.

Cherkassky, V., & Mulier, F. (1998). *Learning from data: Concepts, theory, and methods*. New York: Wiley.

Cherkassky, V., Mulier, F., & Vapnik, V. (1997). Comparison of VC-method with classical methods for model selection. In *Proceedings World Congress on Neural Networks* (pp. 957–962).

Cover, T., & Thomas, J. (1991). *Elements of information theory*. New York: Wiley.

Craven, P., & Wahba, G. (1979). Smoothing noisy data with spline functions. *Numerische Mathematik, 31*, 377–403.

Efron, B. (1979). Computers and the theory of statistics: Thinking the unthinkable. *SIAM Review, 21*, 460–480.

Foster, D., & George, E. (1994). The risk inflation criterion for multiple regression. *Annals of Statistics, 22*, 1947–1975.

Freund, Y., & Schapire, R. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences, 55:1*, 119–139.

Galarza, C., Rietman, E., & Vapnik, V. (1996). Applications of model selection techniques to polynomial approximation. Preprint.

Gutfinger, D., & Sklansky, J. (1991). Robust classifiers by mixed adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 13:6*, 552–567.

Heckerman, D., & Chickering, D. (1996). A comparison of scientific and engineering criteria for Bayesian model selection. Technical Report MSR-TR-96-12, Microsoft Research.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of International Joint Conference on Artificial Intelligence, IJCAI-95*.

Krogh, A., & Vedelsby, J. (1995). Neural network ensembles, cross validation, and active learning. In *Advances in neural information processing systems 7* (pp. 231–238).

Lugosi, G., & Pinter, M. (1996). A data-dependent skeleton estimate for learning. In *Proceedings Annual Conference on Computational Learning Theory, COLT-96* (pp. 51–56).

MacKay, D. (1992). Bayesian interpolation. *Neural Computation, 4*, 415–447.

Mallows, C. (1973). Some comments on C_p. *Technometrics, 15*, 661–676.

Miller, D., & Uyar, H. (1997). A mixture of experts classifier with learning based on both labelled and unlabelled data. In *Advances in neural information processing systems 9* (pp. 571–577).

O'Neill, T. (1978). Normal discrimination with unclassified observations. *Journal of the American Statistical Association, 73:364*, 821–826.

Opitz, D., & Shavlik, J. (1996). Generating accurate and diverse members of a neural-network ensemble. In *Advances in neural information processing systems 8*.

Poggio, T., & Girosi, F. (1990). Regularization algorithms for learning that are equivalent to multilayer networks. *Science, 247*, 978–982.

Ratsaby, J., & Venkatesh, S. (1995). Learning from a mixture of labeled and unlabeled examples with parametric side information. In *Proceedings of Annual Conference on Computational Learning Theory, COLT-95* (pp. 412–417).

Ripley, B. (1996). *Pattern recognition and neural networks*. Cambridge: Cambridge University Press.

Rissanen, J. (1986). Stochastic complexity and modeling. *Annals of Statistics, 14*, 1080–1100.

Schaffer, C. (1993). Overfitting avoidance as bias. *Machine Learning, 10:2*, 153–178.

Schaffer, C. (1994). A conservation law for generalization performance. In *Proceedings of International Conference on Machine Learning, ICML-94* (pp. 683–690).

Schuurmans, D. (1997). A new metric-based approach to model selection. In *Proceedings of National Conference on Artificial Intelligence, AAAI-97* (pp. 552–558).

Schuurmans, D., & Southey, F. (2000). An adaptive regularization criterion for supervised learning. In *Proceedings of International Conference on Machine Learning, ICML-2000* (pp. 847–854).

Schuurmans, D., Ungar, L., & Foster, D. (1997). Characterizing the generalization performance of model selection strategies. In *Proceedings of International Conference on Machine Learning, ICML-97* (pp. 340–348).

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics, 6*, 461–464.
Shibata, R. (1981). An optimal selection of regression variables. *Biometrika, 68*, 45–54.
Towell, G. (1996). Using unlabeled data for supervised learning. In *Advances in neural information processing systems 8* (pp. 647–653).
Vapnik, V. (1996). *The nature of statistical learning theory*. New York: Springer-Verlag.
Weiss, S. M., & Kulikowski, C. A. (1991). *Computer systems that learn*. San Mateo: Morgan Kaufmann.
Young, A. (1977). A Bayesian approach to prediction using polynomials. *Biometrika, 64:2*, 309–317.

Received September 5, 2000
Revised January 23, 2001
Accepted January 24, 2001
Final manuscript February 20, 2001