An Experimental and Theoretical Comparison of Model Selection Methods

Michael Kearns
AT&T Bell Laboratories
Murray Hill, New Jersey

Yishay Mansour*
Tel Aviv University
Tel Aviv, Israel

Andrew Y. Ng
Carnegie Mellon University
Pittsburgh, Pennsylvania

Dana Ron
Hebrew University
Jerusalem, Israel

1 Introduction

In the model selection problem, we must balance the complexity of a statistical model with its goodness of fit to the training data. This problem arises repeatedly in statistical estimation, machine learning, and scientific inquiry in general. Instances of the model selection problem include choosing the best number of hidden nodes in a neural network, determining the right amount of pruning to be performed on a decision tree, and choosing the degree of a polynomial fit to a set of points. In each of these cases, the goal is not to minimize the error on the training data, but to minimize the resulting generalization error.

Many model selection algorithms have been proposed in the literature of several different research communities, too many to productively survey here. (A more detailed history of the problem will be given in the full paper.) Perhaps surprisingly, despite the many proposed solutions for model selection and the diverse methods of analysis, direct comparisons between the different proposals (either experimental or theoretical) are rare.

The goal of this paper is to provide such a comparison, and more importantly, to describe the general conclusions to which it has led. Relying on evidence that is divided between controlled experimental results and related formal analysis, we compare three well-known model selection algorithms. We attempt to identify their relative and absolute strengths and weaknesses, and we provide some general methods for analyzing the behavior and performance of model selection algorithms. Our hope is that these results may aid the informed practitioner in making an educated choice of model selection algorithm (perhaps based in part on some known properties of the model selection problem being confronted).

The summary of the paper follows. In Section 2, we provide a formalization of the model selection problem. In this formalization, we isolate the problem of choosing the appropriate *complexity* for a hypothesis or model. We also introduce the specific model selection problem that will be the basis for our experimental results, and describe an initial experiment demonstrating that the problem is nontrivial. In Section 3, we introduce the three model selection algorithms we examine in the experiments: Vapnik's Guaranteed Risk Minimization (GRM) [11], an instantiation of Rissanen's Minimum Description Length Principle (MDL) [7], and Cross Validation (CV).

Section 4 describes our controlled experimental comparison of the three algorithms. Using artificially generated data from a known target function allows us to plot complete learning curves for the three algorithms over a wide range of sample sizes, and to directly compare the resulting generalization error to the hypothesis complexity selected by each algorithm. It also allows us to investigate the effects of varying other natural parameters of the problem, such as the amount of noise in the data. These experiments support the following assertions: the behavior of the algorithms examined can be complex and incomparable, even on simple problems, and there are fundamental difficulties in identifying a "best" algorithm; there is a strong connection between hypothesis complexity and generalization error; and it may be impossible to uniformly improve the performance of the algorithms by slight modifications (such as introducing constant multipliers on the complexity penalty terms).

In Sections 5, 6 and 7 we turn our efforts to formal results providing explanation and support for the experimental findings. We begin in Section 5 by upper bounding the error of any model selection algorithm falling into a wide class (called *penalty-based* algorithms) that includes both GRM and MDL (but not cross validation). The form of this bound highlights the competing desires for powerful hypotheses and controlled complexity. In Section 6, we upper bound the additional error suffered by cross validation compared to any other model selection algorithm. This quality of this bound depends on the extent to which the function classes have learning curves obeying a classical power law. Finally, in Section 7, we give an impossibility result demonstrating a fundamental handicap suffered by the entire class of penalty-based algorithms that does not afflict cross validation. In Section 8, we weigh the evidence and find that it provides concrete arguments favoring the use of cross validation (or at least cause for caution in using any penalty-based algorithm).

*Thanks to Joe Blow.

2 Definitions

Throughout the paper we assume that a fixed boolean *target function* f is used to label inputs drawn randomly according to a fixed distribution D. For any boolean function h, we define the *generalization error* $\epsilon(h) = \epsilon_{f,D}(h) \equiv \mathbf{Pr}_{x \in D}[h(x) \neq f(x)]$. We use S to denote the random variable $S = \langle x_1, b_1 \rangle, \ldots, \langle x_m, b_m \rangle$, where m is the *sample size*, each x_i is drawn randomly and independently according to D, and $b_i = f(x_i) \oplus c_i$, where the noise bit $c_i \in \{0, 1\}$ is 1 with probability η; we call $\eta \in [0, 1/2)$ the *noise rate*. In the case that $\eta \neq 0$, we will sometimes wish to discuss the generalization error of h with respect to the noisy examples, so we define $\epsilon^\eta(h) \equiv \mathbf{Pr}_{x \in D, c}[h(x) \neq f(x) \oplus c]$, where c is the noise bit. Note that $\epsilon(h)$ and $\epsilon^\eta(h)$ are related by the equality $\epsilon^\eta(h) = (1-\eta)\epsilon(h) + \eta(1-\epsilon(h)) = (1-2\eta)\epsilon(h) + \eta$. For simplicity, we will use the expression "with high probability" to mean with probability $1-\delta$ over the draw of S, at a cost of a factor of $\log(1/\delta)$ in the bounds — *thus, our bounds all contain "hidden" logarithmic factors*, but our handling of confidence is entirely standard and will be spelled out in the full paper.

We assume a nested sequence of *hypothesis classes* (or *models*) $F_1 \subseteq \cdots \subseteq F_d \subseteq \cdots$. The target function f may or may not be contained in any of these classes, so we define $h_d \equiv \text{argmin}_{h \in F_d}\{\epsilon(h)\}$ and $\epsilon_{opt}(d) \equiv \epsilon(h_d)$ (similarly, $\epsilon^\eta_{opt}(d) \equiv \epsilon^\eta(h_d)$). Thus, h_d is the best approximation to f (with respect to D) in the class F_d, and $\epsilon_{opt}(d)$ measures the quality of this approximation. Note that $\epsilon_{opt}(d)$ is a nonincreasing function of d since the hypothesis function classes are nested. Thus, larger values of d can only improve the *potential* approximative power of the hypothesis class. Of course, the difficulty is to realize this potential on the basis of a small sample.

With this notation, the model selection problem can be stated informally: on the basis of a random sample S of a fixed size m, the goal is to choose a hypothesis *complexity* $\tilde{d}$, and a *hypothesis* $\tilde{h} \in F_{\tilde{d}}$, such that the resulting generalization error $\epsilon(\tilde{h})$ is minimized. In many treatments of model selection, including ours, it is explicitly or implicitly assumed that the model selection algorithm has control only over the choice of the complexity $\tilde{d}$, but not over the choice of the final hypothesis $\tilde{h} \in F_{\tilde{d}}$. It is assumed that there is a fixed algorithm that chooses a set of *candidate* hypotheses, one from each hypothesis class. Given this set of candidate hypotheses, the model selection algorithm then chooses one of the candidates as the final hypothesis.

To make these ideas more precise, we define the *training error* $\hat{\epsilon}(h) = \hat{\epsilon}_S(h) \equiv |\{\langle x_i, b_i \rangle \in S : h(x_i) \neq b_i\}|/m$, and the *version space* $VS(d) = VS_S(d) \equiv \{h \in F_d : \hat{\epsilon}(h) = \min_{h' \in F_d}\{\hat{\epsilon}(h')\}\}$. Note that $VS(d) \subseteq F_d$ may contain more than one function in F_d — several functions may minimize the training error. If we are lucky, we have in our possession a (possibly randomized) *learning algorithm* L that takes as input any sample S and any complexity value d, and outputs a member $\tilde{h}_d$ of $VS(d)$ (using some unspecified criterion to break ties if $|VS(d)| > 1$). More generally, it may be the case that finding *any* function in $VS(d)$ is intractable, and that L is simply a heuristic (such as backpropogation or ID3) that does the best job it can at finding $\tilde{h}_d \in F_d$ with small training error on input S and d. In either case, we define $\tilde{h}_d = L(S, d)$ and $\hat{\epsilon}(d) = \hat{\epsilon}_{L,S}(d) \equiv \hat{\epsilon}(\tilde{h}_d)$. Note that we expect $\hat{\epsilon}(d)$, like $\epsilon_{opt}(d)$, to be a nonincreasing function of d — by going to a larger complexity, we can only reduce our training error.

We can now give a precise statement of the model selection problem. First of all, an *instance* of the model selection problem consists of a tuple $(\{F_d\}, f, D, L)$, where $\{F_d\}$ is the hypothesis function class sequence, f is the target function, D is the input distribution, and L is the underlying learning algorithm. The *model selection problem* is then: Given the sample S, and the sequence of functions $\tilde{h}_1 = L(S, 1), \ldots, \tilde{h}_d = L(S, d), \ldots$ determined by the learning algorithm L, select a complexity value $\tilde{d}$ such that $\tilde{h}_{\tilde{d}}$ minimizes the resulting generalization error. Thus, a model selection algorithm is given both the sample S and the sequence of (increasingly complex) hypotheses derived by L from S, and must choose one of these hypotheses.

The current formalization suffices to motivate a key definition and a discussion of the fundamental issues in model selection. We define $\epsilon(d) = \epsilon_{L,S}(d) \equiv \epsilon(\tilde{h}_d)$. Thus, $\epsilon(d)$ is a random variable (determined by the random variable S) that gives the *true generalization error* of the function $\tilde{h}_d$ chosen by L from the class F_d. Of course, $\epsilon(d)$ is not directly accessible to a model selection algorithm; it can only be estimated or guessed in various ways from the sample S. A simple but important observation is that no model selection algorithm can achieve generalization error less than $\min_d\{\epsilon(d)\}$. Thus the behavior of the function $\epsilon(d)$ — especially the location and value of its minimum — is in some sense the essential quantity of interest in model selection.

The prevailing folk wisdom in several research communities posits that $\epsilon(d)$ will typically have a global minimum that is nontrivial — that is, at an "intermediate" value of d away from the extremes $d = 0$ and $d \approx m$. As a demonstration of the validity of this view, and as an introduction to a particular model selection problem that we will examine in our experiments, we call the reader's attention to Figure 1. In this model selection problem (which we shall refer to as the *intervals model selection problem*), the input domain is simply the real line segment $[0, 1]$, and the hypothesis class F_d is simply the class of all boolean functions over $[0, 1]$ in which we allow at most d alternations of label; thus F_d is the class of all binary step functions with at most $d/2$ steps. For the experiments, the underlying learning algorithm L that we have implemented performs training error minimization. This is a rare case where efficient minimization is possible; we have developed an algorithm based on dynamic programming that runs in linear time, thus making experiments on large samples feasible. The sample S was generated using the target function in F_{100} that divides $[0, 1]$ into 100 segments of equal width $1/100$ and alternating label. In Figure 1 we plot $\epsilon(d)$ (which we can calculate exactly, since we have chosen the target function) when S consists of $m = 2000$ random examples (drawn from the uniform input distribution) corrupted by noise at the rate $\eta = 0.2$. For our current discussion it suffices to note that $\epsilon(d)$ does indeed experience a nontrivial minimum. Not surprisingly, this minimum occurs near (but

not exactly at) the target complexity of 100.

3 Three Algorithms for Model Selection

The first two model selection algorithms we consider are members of a general class that we shall informally refer to as *penalty-based* algorithms (and shall formally define shortly). The common theme behind these algorithms is their attempt to construct an approximation to $\epsilon(d)$ solely on the basis of the training error $\hat{\epsilon}(d)$ and the complexity d, often by trying to "correct" $\hat{\epsilon}(d)$ by the amount that it underestimates $\epsilon(d)$ through the addition of a "complexity penalty" term.

In Vapnik's *Guaranteed Risk Minimization* (GRM) [11], $\tilde{d}$ is chosen according to the rule [1]

$$\tilde{d} = \text{argmin}_d\{\hat{\epsilon}(d) + (d/m)(1 + \sqrt{1 + \hat{\epsilon}(d)m/d})\} \quad (1)$$

where for convenience but without loss of generality we have assumed that d is the Vapnik-Chervonenkis dimension [11, 12] of the class F_d; this assumption holds in the intervals model selection problem. The origin of this rule can be summarized as follows: it has been shown [11] (ignoring logarithmic factors) that for every d and for every $h \in F_d$, $\sqrt{d/m}$ is an upper bound on $|\hat{\epsilon}(h) - \epsilon(h)|$ and hence $|\hat{\epsilon}(d) - \epsilon(d)| \leq \sqrt{d/m}$. Thus, by simply adding $\sqrt{d/m}$ to $\hat{\epsilon}(d)$, we ensure that the resulting sum upper bounds $\epsilon(d)$, and if we are optimistic we might further hope that the sum is in fact a close approximation to $\epsilon(d)$, and that its minimization is therefore tantamount to the minimization of $\epsilon(d)$. The actual rule given in Equation (1) is slightly more complex than this, and reflects a refined bound on $|\hat{\epsilon}(d) - \epsilon(d)|$ that varies from d/m for $\hat{\epsilon}(d)$ close to 0 to $\sqrt{d/m}$ otherwise.

The next algorithm we consider, the *Minimum Description Length Principle* (MDL) [5, 6, 7, 1, 4] has rather different origins than GRM. MDL is actually a broad class of algorithms with a common information-theoretic motivation, each algorithm determined by the choice of a specific *coding* scheme for both functions and their training errors; this two-part code is then used to describe the training sample S. To illustrate the method, we give a coding scheme for the intervals model selection problem [2]. Let h be a function with exactly d alternations of label (thus, $h \in F_d$). To describe the behavior of h on the sample $S = \{\langle x_i, b_i \rangle\}$, we can simply specify the d inputs where h switches value (that is, the indices i such that $h(x_i) \neq h(x_{i+1})$) [3]. This takes $\log \binom{m}{d}$ bits; dividing by m to normalize, we obtain $(1/m) \log \binom{m}{d} \approx \mathcal{H}(d/m)$ [2], where $\mathcal{H}(\cdot)$ is the binary entropy function. Now given h, the labels in S can be described simply by coding the mistakes of h (that is, those indices i where $h(x_i) \neq f(x_i)$), at a normalized cost of $\mathcal{H}(\hat{\epsilon}(h))$. Technically, in the coding scheme just described we also need to specify the values of d and $\hat{\epsilon}(h) \cdot m$, but the cost of these is negligible. Thus, the version of MDL that we shall examine for the intervals model selection problem dictates the following choice of $\tilde{d}$:

$$\tilde{d} = \text{argmin}_d\{\mathcal{H}(\hat{\epsilon}(d)) + \mathcal{H}(d/m)\}. \quad (2)$$

In the context of model selection, GRM and MDL can both be interpreted as attempts to model $\epsilon(d)$ by transforming $\hat{\epsilon}(d)$ and d. More formally, a model selection algorithm of the form

$$\tilde{d} = \text{argmin}_d\{G(\hat{\epsilon}(d), d/m)\} \quad (3)$$

shall be called a *penalty-based* algorithm [4]. Notice that an ideal penalty-based algorithm would obey $G(\hat{\epsilon}(d), d/m) \approx \epsilon(d)$ (or at least $G(\hat{\epsilon}(d), d/m)$ and $\epsilon(d)$ would be minimized by the same value of d).

The third model selection algorithm that we examine has a different spirit than the penalty-based algorithms. In *cross validation* (CV) [9, 10], we use only a fraction $(1 - \gamma)$ of the examples in S to obtain the hypothesis sequence $\tilde{h}_1 \in F_1, \ldots, \tilde{h}_d \in F_d, \ldots$ — that is, $\tilde{h}_d$ is now $L(S', d)$, where S' consists of the first $(1 - \gamma)m$ examples in S. Here $\gamma \in [0, 1]$ is a parameter of the CV algorithm whose tuning we discuss briefly later. CV chooses $\tilde{d}$ according to the rule

$$\tilde{d} = \text{argmin}_d\{\hat{\epsilon}_{S''}(\tilde{h}_d)\} \quad (4)$$

where $\hat{\epsilon}_{S''}(\tilde{h}_d)$ is the error of $\tilde{h}_d$ on S'', the last γm examples of S that were withheld in selecting $\tilde{h}_d$. Notice that for CV, we expect the quantity $\epsilon(d) = \epsilon(\tilde{h}_d)$ to be (perhaps considerably) larger than in the case of GRM and MDL, because now $\tilde{h}_d$ was chosen on the basis of only $(1 - \gamma)m$ examples rather than all m examples. For this reason we wish to introduce the more general notation $\epsilon^{\gamma}(d) \equiv \epsilon(\tilde{h}_d)$ to indicate the fraction of the sample withheld from training. CV settles for $\epsilon^{\gamma}(d)$ instead of $\epsilon^0(d)$ in order to have an independent test set with which to directly estimate $\epsilon^{\gamma}(d)$.

4 A Controlled Experimental Comparison

Our results begin with a comparison of the performance and properties of the three model selection algorithms in a carefully controlled experimental setting — namely, the intervals model selection problem. Among the advantages of such controlled experiments, at least in comparison to empirical results on data of unknown origin, are our ability to exactly measure generalization error (since we know the target function and the distribution generating the data), and our ability to precisely study the effects of varying parameters of the data (such as noise rate, target function complexity, and sample size), on the performance of model selection algorithms. The experimental behavior we observe foreshadows a number of important themes that we shall revisit in our formal results.

We begin with Figure 2. To obtain this figure, a training sample was generated from the uniform input distribution

[1] Vapnik's original GRM actually multiplies the second term inside the argmin$\{\cdot\}$ above by a logarithmic factor intended to guard against worst-case choices from $VS(d)$. Since this prefactor renders GRM uncompetitive on the ensuing experiments, we consider this modified and quite competitive rule whose spirit is the same.

[2] Our goal here is simply to give one reasonable instantiation of MDL. Other coding schemes are obviously possible; however, several of our formal results will hold for essentially all MDL instantiations.

[3] In the full paper we justify our use of the sample points to describe h; it is quite similar to representing h using a grid of resolution $1/p(m)$ for some polynomial $p(\cdot)$.

[4] With appropriately modified assumptions, all of the formal results in the paper hold for the more general form $G(\hat{\epsilon}(d), d, m)$, where we decouple the dependence on d and m. However, the simpler coupled form will suffice for our purposes.

and labeled according to an intervals function over $[0, 1]$ consisting of 100 intervals of alternating label and equal width [5]; the sample was corrupted with noise at rate $\eta = 0.2$. In Figure 2, we have plotted the *true* generalization errors (measured with respect to the noise-free source of examples) ϵ_{GRM}, ϵ_{MDL} and ϵ_{CV} (using test fraction $\gamma = 0.1$ for CV) of the hypotheses selected from the sequence $\tilde{h}_1, \ldots, \tilde{h}_d, \ldots$ by each the three algorithms as a function of sample size m, which ranged from 1 to 3000 examples. As described in Section 2, the hypotheses $\tilde{h}_d$ were obtained by minimizing the training error within each class F_d. Details of the code used to perform these experiments will be provided in the full paper.

Figure 2 demonstrates the subtlety involved in comparing the three algorithms: in particular, we see that *none of the three algorithms outperforms the others for all sample sizes.* Thus we can immediately dismiss the notion that one of the algorithms examined can be said to be optimal for this problem in any standard sense. Getting into the details, we see that there is an initial regime (for m from 1 to slightly less than 1000) in which ϵ_{MDL} is the lowest of the three errors, sometimes outperforming ϵ_{GRM} by a considerable margin. Then there is a second regime (for m about 1000 to about 2500) where an interesting reversal of relative performance occurs, since now ϵ_{GRM} is the lowest error, considerably outperforming ϵ_{MDL}, which has temporarily leveled off. In both of these first two regimes, ϵ_{CV} remains the intermediate performer. In the third and final regime, ϵ_{MDL} decreases rapidly to match ϵ_{GRM} and the slightly larger ϵ_{CV}, and the performance of all three algorithms remains quite similar for all larger sample sizes.

Insight into the causes of Figure 2 is given by Figure 3, where for the same runs used to obtain Figure 2, we instead plot the quantities $\tilde{d}_{\text{GRM}}$, $\tilde{d}_{\text{MDL}}$ and $\tilde{d}_{\text{CV}}$, the value of $\tilde{d}$ chosen by GRM, MDL and CV respectively (thus, the "correct" value, in the sense of simply having the same number of intervals as the target function, is 100). Here we see that for small sample sizes, corresponding to the first regime discussed for Figure 2 above, $\tilde{d}_{\text{GRM}}$ is slowly approaching 100 from below, reaching and remaining at the target value for about $m = 1500$. Although we have not shown it explicitly, GRM is incurring nonzero training error throughout the entire range of m. In comparison, for a long initial period (corresponding to the first two regimes of m), MDL is simply choosing the shortest hypothesis that incurs no training error (and thus encodes both "legitimate" intervals and noise), and consequently $\tilde{d}_{\text{MDL}}$ grows in an uncontrolled fashion. More precisely, it can be shown that during this period $\tilde{d}_{\text{MDL}}$ is obeying $\tilde{d}_{\text{MDL}} \approx d_0 \equiv 2\eta(1-\eta)m + (1-2\eta)^2 s$, where s is the number of (equally spaced) intervals in the target function and η is the noise rate (so for the current experiment $s = 100$ and $\eta = 0.2$). This "overcoding" behavior of MDL is actually preferable, in terms of generalization error, to the initial "undercoding" behavior of GRM, as verified by Figure 2. Once $\tilde{d}_{\text{GRM}}$ approaches 100, however, the overcoding of MDL is a relative liability, resulting in the second regime. Figure 3 clearly shows that the transition from the second to the third regime (where approximate parity is achieved) is the direct result of a dramatic correction to $\tilde{d}_{\text{MDL}}$ from d_0 (defined above) to the target value of 100. Finally, $\tilde{d}_{\text{CV}}$ makes a more rapid but noisier approach to 100 than $\tilde{d}_{\text{GRM}}$, and in fact also overshoots 100, but much less dramatically than $\tilde{d}_{\text{MDL}}$. This more rapid initial increase again results in superior generalization error compared to GRM for small m, but the inability of $\tilde{d}_{\text{CV}}$ to settle at 100 results in slightly higher error for larger m. In the full paper, we examine the same plots of generalization error and hypothesis complexity for different values of the noise rate; here it must suffice to say that for $\eta = 0$, all three algorithms have comparable performance for all sample sizes, and as η increases so do the qualitative effects discussed here for the $\eta = 0.2$ case (for instance, the duration of the second regime, where MDL is vastly inferior, increases with the noise rate).

The behavior $\tilde{d}_{\text{GRM}}$ and $\tilde{d}_{\text{MDL}}$ in Figure 3 can be traced to the form of the *total penalty* functions for the two algorithms. For instance, in Figures 4, and 5, we plot the total MDL penalty $\mathcal{H}(\hat{\epsilon}(d)) + \mathcal{H}(d/m)$ as a function of d for the fixed sample sizes $m = 2000$ and 4000 respectively, again using noise rate $\eta = 0.20$. At $m = 2000$, we see that the total penalty has its *global* minimum at approximately 650, which is roughly the zero training error value d_0 discussed above (we are still in the MDL overcoding regime at this sample size; see Figures 2 and 3). However, by this sample size, a significant *local* minimum has developed near the target value of $d = 100$. At $m = 4000$, this local minimum at $d = 100$ has become the global minimum. The rapid transition of $\tilde{d}_{\text{MDL}}$ that marks the start of the final regime of generalization error is thus explained by the switching of the global total penalty minimum from d_0 to 100. In Figures 6, we plot the total GRM penalty, just for the sample size $m = 2000$. The behavior of the GRM penalty is much more controlled — for each sample size, the total penalty has a single-minimum bowl shape, with the minimum lying to the left of $d = 100$ for small sample sizes and gradually moving over $d = 100$ and sharpening there for large m; as Figure 6 shows, the minimum already lies at $d = 100$ by $m = 2000$, as confirmed by Figure 3.

A natural question to pose after examining Figures 2 and 3 is the following: is there a penalty-based algorithm that enjoys the best properties of both GRM and MDL? By this we would mean an algorithm that approaches the "correct" d value (whatever it may be for the problem in hand) more rapidly than GRM, but does so without suffering the long, uncontrolled "overcoding" period of MDL. An obvious candidate for such an algorithm is simply a modified version of GRM or MDL, in which we reason (for example) that perhaps the GRM penalty for complexity is too large for this problem (resulting in the initial reluctance to code), and we thus multiply the complexity penalty term in the GRM rule (the second term inside the argmin$\{\cdot\}$) in Equation (1) by a constant less than 1 (or analogously, multiply the MDL complexity penalty term by a constant greater than 1 to reduce overcoding). The results of an experiment on such a modified version of GRM are shown in Figures 7 and 8, where the original GRM performance is compared to a modified version in which the complexity penalty is multiplied by 0.5. Interestingly and perhaps unfortunately, we see that there is

[5] Similar results hold for a randomly chosen target function.

no free lunch: while the modified version does indeed code more rapidly and thus reduce the small m generalization error, this comes at the cost of a subsequent overcoding regime with a corresponding degradation in generalization error (and in fact a considerably slower return to $d = 100$ than MDL under the same conditions) [6]. The reverse phenomenon (reluctance to code) is experienced for MDL with an increased complexity penalty multiplier (details in the full paper).

Let us summarize the key points demonstrated by these experiments. First, none of the three algorithms dominates the others for all sample sizes. Second, the two penalty-based algorithms seem to have a bias either towards or against coding that is overcome by the inherent properties of the data asymptotically, but that can have a large effect on generalization error for small to moderate sample sizes. Third, this bias cannot be overcome simply by adjusting the relative weight of error and complexity penalties, without reversing the bias of the resulting rule and suffering increased generalization error for some range of m. Fourth, while CV is not the best of the algorithms for any value of m, it does manage to fairly closely track the best penalty-based algorithm for each value of m, and considerably beats both GRM and MDL in their regimes of weakness. We now turn our attention to our formal results, where each of these key points will be developed further.

5 A Bound on Generalization Error for Penalty-Based Algorithms

We begin our formal results with a bound on the generalization error for penalty-based algorithms that enjoys three features. First, it is general: it applies to practically any penalty-based algorithm, and holds for any model selection problem (of course, there is a price to pay for such generality, as discussed below). Second, for certain algorithms and certain problems the bound can give rapid rates of convergence to small error. Third, the form of the bound is suggestive of some of the behavior seen in the experimental results. We state the bound for the special but natural case in which the underlying learning algorithm L is training error minimization; in the full paper, we will present a straightforward analogue for more general L. Both this theorem and Theorem 2 in the following section are stated for the noise-free case; but again, straightforward generalizations to the noisy case will be included in the full paper.

Theorem 1 *Let $(\{F_d\}, f, D, L)$ be an instance of the model selection problem in which L performs training error minimization, and assume for convenience that d is the VC dimension of F_d. Let $G : [0,1] \times \Re \to \Re$ be a function that is continuous and increasing in both its arguments, and let $\epsilon_G(m)$ denote the expected generalization error of the penalty-based model selection algorithm $\tilde{d} = \text{argmin}_d\{G(\hat{\epsilon}(d), d/m)\}$ on a training sample of size m. Then* [7]

$$\epsilon_G(m) \leq R_G(m) + \sqrt{\tilde{d}/m} \tag{5}$$

where $R_G(m)$ approaches $\min_d\{\epsilon_{opt}(d)\}$ (which is the best generalization error achievable in any *of the classes F_d) as $m \to \infty$. The rate of this approach will depend on properties of G.*

Proof: For any value of d, we have the inequality

$$G\left(\hat{\epsilon}(\tilde{d}), \tilde{d}/m\right) \leq G\left(\hat{\epsilon}(d), d/m\right). \tag{6}$$

because $\tilde{d}$ is chosen to minimize $G(\hat{\epsilon}(d), d/m)$. Using the uniform convergence bound $|\epsilon(h) - \hat{\epsilon}(h)| \leq \sqrt{d/m}$ for all $h \in F_d$ and the fact that $G(\cdot,\cdot)$ is increasing in its first argument, we can replace the occurrence of $\hat{\epsilon}(\tilde{d})$ on the left-hand side of Equation (6) by $\epsilon(\tilde{d}) - \sqrt{\tilde{d}/m}$ to obtain a smaller quantity, and we can replace the occurrence of $\hat{\epsilon}(d)$ on the right-hand side by $\epsilon_{opt}(d) + \sqrt{d/m}$ to obtain a larger quantity. This gives

$$G\left(\epsilon(\tilde{d}) - \sqrt{\tilde{d}/m}, \tilde{d}/m\right) \leq G\left(\epsilon_{opt}(d) + \sqrt{d/m}, d/m\right). \tag{7}$$

Now because $G(\cdot,\cdot)$ is an increasing function of its second argument, we can further weaken Equation (7) to obtain

$$G\left(\epsilon(\tilde{d}) - \sqrt{\tilde{d}/m}, 0\right) \leq G\left(\epsilon_{opt}(d) + \sqrt{d/m}, d/m\right). \tag{8}$$

If we define $G_0(x) = G(x, 0)$, then since $G(\cdot,\cdot)$ is increasing in its first argument, $G_0^{-1}(\cdot)$ is well-defined, and we may write

$$\epsilon(\tilde{d}) \leq G_0^{-1}\left(G\left(\epsilon_{opt}(d) + \sqrt{d/m}, d/m\right)\right) + \sqrt{\tilde{d}/m}. \tag{9}$$

Now fix any small value $\tau > 0$. For this τ, let d' be the smallest value satisfying $\epsilon_{opt}(d') \leq \min_d\{\epsilon_{opt}(d)\} + \tau$ — thus, d' is sufficient complexity to almost match the approximative power of arbitrarily large complexity. Examining the behavior of $G_0^{-1}(G(\epsilon_{opt}(d') + \sqrt{d'/m}, d'/m))$ as $m \to \infty$, we see that the arguments approach the point $(\epsilon_{opt}(d'), 0)$, and so $G_0^{-1}(G(\epsilon_{opt}(d') + \sqrt{d'/m}, d'/m))$ approaches $G_0^{-1}(G(\epsilon_{opt}(d'), 0)) = \epsilon_{opt}(d') \leq \min\{\epsilon_{opt}(d)\} + \tau$ by continuity of $G(\cdot,\cdot)$, as desired. By defining

$$R_G(m) \equiv \min_d\left\{G_0^{-1}\left(G\left(\epsilon_{opt}(d) + \sqrt{d/m}, d/m\right)\right)\right\} \tag{10}$$

we obtain the statement of the theorem. □

Let us now discuss the form of the bound given in Theorem 1. The first term $R_G(m)$ approaches the optimal generalization error within $\bigcup F_d$ in the limit of large m, and the second term directly penalizes large complexity. These terms may be thought of as competing. In order for $R_G(m)$ to approach $\min_d\{\epsilon_{opt}(d)\}$ *rapidly* and not just asymptotically (that is, in order to have a fast *rate* of convergence), $G(\cdot,\cdot)$ should not penalize complexity too strongly, which is obviously at odds with the optimization of the term $\sqrt{\tilde{d}/m}$. For example, consider $G(\hat{\epsilon}(d), d/m) = \hat{\epsilon}(d) + (d/m)^\alpha$ for some power $\alpha > 0$. Assuming $d \leq m$, this rule is conservative (large penalty for complexity) for small α, and liberal (small penalty

[6] Similar results are obtained in experiments in which every occurrence of d in the GRM rule is replaced by an "effective dimension" $c_0 d$ for any constant $c_0 < 1$.

[7] We remind the reader that our bounds contain hidden logarithmic factors that we specify in the full paper.

for complexity) for large α. Thus, to make the term $\sqrt{\tilde{d}/m}$ small we would like α to be small, to prevent the choice of large $\tilde{d}$. However, $R_G(m) = \min_d\{\epsilon_{opt}(d) + \sqrt{d/m} + (d/m)^\alpha\}$, which increases as α decreases, thus encouraging large α (liberal coding).

Ideally, we might want $G(\cdot,\cdot)$ to balance the two terms of the bound, which implicitly involves finding an appropriately *controlled* but sufficiently *rapid* rate of increase in $\tilde{d}$. The tension between these two criteria in the bound echoes the same tension that was seen experimentally: for MDL, there was a long period of essentially uncontrolled growth of $\tilde{d}$ (linear in m), and this uncontrolled growth prevented any significant decay of generalization error (Figures 2 and 3). GRM had controlled growth of $\tilde{d}$, and thus would incur negligible error from our second term — but perhaps this growth was *too* controlled, as it results in the initially slow (small m) decrease in generalization error.

To examine these issues further, we now apply the bound of Theorem 1 to several penalty-based algorithms. In some cases the final form of the bound given in the theorem statement, while easy to interpret, is unnecessarily coarse, and better rates of convergence can be obtained by directly appealing to the proof of the theorem.

We begin with a *simplified* GRM variant (SGRM), defined by $G(\hat{\epsilon}(d), d/m) = \hat{\epsilon}(d) + \sqrt{d/m}$. For this algorithm, we observe that we can avoid weakening Equation (7) to Equation (8), because here $G(\epsilon(\tilde{d}) - \sqrt{\tilde{d}/m}, \tilde{d}/m) = \epsilon(\tilde{d})$. Thus the dependence on $\tilde{d}$ in the bound disappears entirely, resulting in

$$\epsilon_{\text{SGRM}}(m) \leq \min_d \left\{\epsilon_{opt}(d) + 2\sqrt{d/m}\right\}. \qquad (11)$$

This is not so mysterious, since SGRM penalizes strongly for complexity (even more so than GRM). This bound expresses the generalization error as the minimum of the sum of the best possible error within each class F_d and a penalty for complexity. Such a bound seems entirely reasonable, given that it is essentially the expected value of the empirical quantity we minimized to choose $\tilde{d}$ in the first place. Furthermore, if $\epsilon_{opt}(d) + \sqrt{d/m}$ approximates $\epsilon(d)$ well, then such a bound is about the best we could hope for. However, there is no reason in general to expect this to be the case. Bounds of this type were first given by Barron and Cover [1] in the context of density estimation.

As an example of the application of Theorem 1 to MDL we can derive the following bound on $\epsilon_{\text{MDL}}(m)$:

$$\begin{aligned}\epsilon_{\text{MDL}}(m) &\leq \min_d\{\mathcal{H}^{-1}(\mathcal{H}(\epsilon_{opt}(d) + \sqrt{d/m}) \\ &\qquad +\mathcal{H}(d/m))\} + \sqrt{\tilde{d}_{\text{MDL}}/m} \qquad (12)\\ &\leq \min_d\{\mathcal{H}(\epsilon_{opt}(d)) + 2\mathcal{H}(\sqrt{d/m})\} \\ &\qquad +\sqrt{\tilde{d}_{\text{MDL}}/m} \qquad (13)\end{aligned}$$

where we have used $\mathcal{H}^{-1}(y) \leq y$ and $\mathcal{H}(x+y) \leq \mathcal{H}(x) + \mathcal{H}(y)$. Again, we emphasize that the bound given by Equation (13) is vacuous without a bound on $\tilde{d}_{\text{MDL}}$, which we know from the experiments can be of order m. However, by combining this bound with an analysis of the behavior of $\tilde{d}_{\text{MDL}}$ for the intervals problem, we can give an accurate theoretical explanation for the experimental findings for MDL (details in the full paper).

As a final example, we apply Theorem 1 to a *variant* of MDL in which the penalty for coding is increased over the original, namely $G(\hat{\epsilon}(d), d/m) = \mathcal{H}(\hat{\epsilon}(d)) + 1/\lambda^2\mathcal{H}(d/m)$ where λ is a parameter that may depend on d and m. Assuming that we never choose $\tilde{d}$ whose total penalty is larger than 1 (which holds if we simply add the "fair coin hypothesis" to F_1), we have that $\mathcal{H}(d/m) \leq \lambda^2$. Since $\mathcal{H}(x) \geq x$, for all x, it follows that $\sqrt{\tilde{d}/m} \leq \lambda$. If λ is some decreasing function of m (say, m^α for some $0 < \alpha < 1$), then the bound on $\epsilon(\tilde{d})$ given by Theorem 1 decreases at a reasonable rate.

6 A Bound on the Additional Error of CV

In this section we state a general theorem bounding the additional generalization error suffered by cross validation compared to any *polynomial complexity* model selection algorithm M. By this we mean that given a sample of size m, algorithm M will never choose a value of $\tilde{d}$ larger than m^k for some fixed exponent $k > 1$. We emphasize that this is a mild condition that is met in practically every realistic model selection problem: although there are many documented circumstances in which we may wish to choose a model whose complexity is on the order of the sample size, we do not imagine wanting to choose, for instance, a neural network with a number of nodes *exponential* in the sample size. In any case, more general but more complicated assumptions may be substituted for the notion of polynomial complexity, and we discuss these in the full paper.

Theorem 2 *Let M be any polynomial complexity model selection algorithm, and let $(\{F_d\}, f, D, L)$ be any instance of model selection. Let $\epsilon_{\text{M}}(m)$ and $\epsilon_{\text{CV}}(m)$ denote the expected generalization error of the hypotheses chosen by M and CV respectively. Then*

$$\epsilon_{\text{CV}}(m) \leq \epsilon_{\text{M}}((1-\gamma)m) + O(\sqrt{\log(m)/\gamma m}). \qquad (14)$$

In other words, the generalization error of CV on m examples is at most the generalization error M on $(1-\gamma)m$ examples, plus the "test penalty term" $O(\sqrt{\log(m)/\gamma m})$.

Proof Sketch: Let $S = (S', S'')$ be a random sample of m examples, where $|S'| = (1-\gamma)m$ and $|S''| = \gamma m$. Let $d_{max} = ((1-\gamma)m)^k$ be the polynomial bound on the complexity selected by M, and let $\tilde{h}'_1 \in F_1, \ldots, \tilde{h}'_{d_{max}} \in F_{d_{max}}$ be determined by $\tilde{h}'_d = L(S', d)$. By definition of CV, $\tilde{d}$ is chosen according to $\tilde{d} = \text{argmin}_d\{\hat{\epsilon}_{S''}(\tilde{h}'_d)\}$. By standard uniform convergence arguments we have that $|\epsilon(\tilde{h}'_d) - \hat{\epsilon}_{S''}(\tilde{h}'_d)| = O(\sqrt{\log(m)/\gamma m})$ for all $d \leq d_{max}$ with high probability over the draw of S''. Therefore with high probability

$$\epsilon_{\text{CV}} = \min_d\{\epsilon(\tilde{h}'_d)\} + O(\sqrt{\log(m)/\gamma m}). \qquad (15)$$

But as we have previously observed, the generalization error of *any* model selection algorithm (including M) on input S'

is lower bounded by $\min_d\{\epsilon(\tilde{h}'_d)\}$, and our claim directly follows. □

Note that the bound of Theorem 2 does *not* claim $\epsilon_{\rm CV}(m) \leq \epsilon_{\rm M}(m)$ for all M (which would mean that cross validation is an optimal model selection algorithm). The bound given is weaker than this ideal in two important ways. First, and perhaps most importantly, $\epsilon_{\rm M}((1-\gamma)m)$ may be considerably larger than $\epsilon_{\rm M}(m)$. This could either be due to properties of the underlying learning algorithm L, or due to inherent *phase transitions* (sudden decreases) in the optimal information-theoretic learning curve [8, 3] — thus, in an extreme case, it could be that the generalization error that can be achieved within some class F_d by training on m examples is close to 0, but that the optimal generalization error that can be achieved in F_d by training on a slightly smaller sample is near $1/2$. This is intuitively the worst case for cross validation — when the small fraction of the sample saved for testing was critically needed for training in order to achieve nontrivial performance — and is reflected in the first term of our bound. Obviously the risk of "missing" phase transitions can be minimized by decreasing the test fraction γ, but only at the expense of increasing the test penalty term, which is the second way in which our bound falls short of the ideal. However, unlike the potentially unbounded difference $\epsilon_{\rm M}((1-\gamma)m) - \epsilon_{\rm M}(m)$, our bound on the test penalty can be decreased without any problem-specific knowledge by simply *increasing* the test fraction γ.

Despite these two competing sources of additional CV error, the bound has some strengths that are worth discussing. First of all, the bound holds for *any* model selection problem instance $(\{F_d\}, f, D, L)$. We believe that giving similarly general bounds for any penalty-based algorithm would be extremely difficult, if not impossible. The reason for this belief arises from the diversity of learning curve behavior documented by the statistical mechanics approach [8, 3], among other sources. In the same way that there is no universal learning curve behavior, there is no universal behavior for the relationship between the functions $\hat{\epsilon}(d)$ and $\epsilon(d)$ — the relationship between these quantities may depend critically on the target function and the input distribution (this point is made more formally in Section 7). CV is sensitive to this dependence by virtue of its target function-dependent and distribution-dependent estimate of $\epsilon(d)$. In contrast, by their very nature, penalty-based algorithms propose a *universal* penalty to be assigned to the observation of error $\hat{\epsilon}(h)$ for a hypothesis h of complexity d.

A more technical feature of Theorem 2 is that it can be combined with bounds derived for penalty-based algorithms using Theorem 1 to suggest how the parameter γ should be tuned. For example, letting M be the SGRM algorithm described in Section 5, and combining Equation (11) with Theorem 2 yields

$$\begin{aligned}\epsilon_{\rm CV}(m) &\leq \epsilon_{\rm SGRM}((1-\gamma)m) \\ &\quad + \sqrt{\log d_{\rm MAX}(m)/\gamma m} \qquad (16)\\ &\leq \min_d \left\{\epsilon_{opt}(d) + 2\sqrt{d/(1-\gamma)m}\right\} \\ &\quad + \sqrt{\log d_{\rm MAX}(m)/\gamma m} \qquad (17)\end{aligned}$$

If we knew the form of $\epsilon_{opt}(d)$ (or even had bounds on it), then in principle we could minimize the bound of Equation (19) as a function of γ to derive a recommended training/test split. Such a program is feasible for many specific problems (such as the intervals problem), or by investigating general but plausible bounds on the approximation rate $\epsilon_{opt}(d)$, such as $\epsilon_{opt}(d) \leq c_0/d$ for some constant $c_0 > 0$. We pursue this line of inquiry in some detail in the full paper. For now, we simply note that Equation (19) tells us that in cases for which the power law decay of generalization error within each F_d holds approximately, the performance of CV will be competitive with GRM or any other algorithm. This makes perfect sense in light of the preceding analysis of the two sources for additional CV error: in problems with power law learning curve behavior, we have a power law bound on $\epsilon_{\rm M}((1-\gamma)m) - \epsilon_{\rm M}(m)$, and thus CV "tracks" any other algorithm closely in terms of generalization error. This is exactly the behavior observed in the experiments described in Section 4, for which the power law is known to hold approximately.

7 Limitations on Penalty-Based Algorithms

Recall that our experimental findings suggested that it may sometimes be fair to think of penalty-based algorithms as being either conservative or liberal in the amount of coding they are willing to allow in their hypothesis, and that bias in either direction can result in suboptimal generalization that is not easily overcome by tinkering with the form of the rule. In this section we treat this intuition more formally, by giving a theorem demonstrating some fundamental limitations on the diversity of problems that can be effectively handled by any fixed penalty-based algorithm. Briefly, we show that there are (at least) two very different forms that the relationship between $\hat{\epsilon}(d)$ and $\epsilon(d)$ can assume, and that any penalty-based algorithm can perform well on only one of these. Furthermore, for the problems we choose, CV can in fact succeed on both. Thus we are doing more than simply demonstrating that no model selection algorithm can succeed universally for all target functions, a statement that is intuitively obvious. We are in fact identifying a weakness that is *special* to penalty-based algorithms. However, as we have discussed previously, the use of CV is not without pitfalls of its own. We therefore conclude the paper in Section 8 with a summary of the different risks involved with each type of algorithm, and a discussion of our belief that in the absence of detailed problem-specific knowledge, our overall analysis favors the use of CV.

Theorem 3 *For any sample size m, there are model selection problem instances $(\{F_d^1\}, f_1, D_1, L)$ and $(\{F_d^2\}, f_2, D_2, L)$ (where L performs empirical error minimization in both instances) and a constant γ independent of m such that for any penalty-based model selection algorithm G, either $\epsilon_G^1(m) \geq \min_d\{\epsilon_1(d)\} + \gamma$ or $\epsilon_G^2(m) \geq \min_d\{\epsilon_2(d)\} + \gamma$. Here $\epsilon_i(d)$ is the function $\epsilon(d)$ for instance $i \in \{1,2\}$, and $\epsilon_G^i(m)$ is the expected generalization error of algorithm G for instance i. Thus, on at least one of the two model selection problems, the generalization error of G is lower bounded away from the optimal value $\min_d\{\epsilon^i(d)\}$ by a constant independent of m.*

Proof Sketch: For notational convenience, in the proof we use $\hat{\epsilon}_i(d)$ and $\epsilon_i(d)$ ($i \in \{1,2\}$) to refer to the expected values of these functions. We start with a rough description of the properties of the two problems (see Figure 9): in Problem 1, the "right" choice of d is 0, any additional coding directly results in larger generalization error, and the training error, $\hat{\epsilon}_1(d)$, decays gradually with d. In Problem 2, a large amount of coding is required to achieve nontrivial generalization error, and the training error remains large as d increases until $d = m/2$, where the training error drops rapidly.

More precisely, we will arrange things so that the first model selection problem (Problem 1) has the following properties (1) The function $\hat{\epsilon}_1(d)$ lies between two linear functions with y-intercepts η_1 and $\eta_1(1-\eta_1)$ and common x-intercept $2\eta_1(1-\eta_1)m \leq m/2$; and (2) $\epsilon_1(d)$ is minimized at $d = 0$, and furthermore, for any constant c we have $\epsilon_1(cm) \geq c/2$. We will next arrange that the second model selection problem (Problem 2) will obey: (1) The function $\hat{\epsilon}_2(d) = a_1$ for $0 \leq d \leq 2\eta_1(1-\eta_1)m \leq m/2$, where $\eta_1(1-\eta_1) > a_1$; and (2) The function $\epsilon_2(d)$ is lower bounded by a_1 for $0 \leq d < m/2$, but $\epsilon_2(m/2) = 0$. In Figure 9 we illustrate the conditions on $\hat{\epsilon}(d)$ for the two problems, and also include hypothetical instances of $\hat{\epsilon}_1(d)$ and $\hat{\epsilon}_2(d)$ that are consistent with these conditions (and are furthermore representative of the "true" behavior of the $\hat{\epsilon}(d)$ functions actually obtained for the two problems we define momentarily).

We can now give the underlying logic of the proof using the hypothetical $\hat{\epsilon}_1(d)$ and $\hat{\epsilon}_2(d)$. Let $\tilde{d}_1$ denote the complexity chosen by G for Problem 1, and let $\tilde{d}_2$ be defined similarly. First consider the behavior of G on Problem 2. In this problem we know by our assumptions on $\epsilon_2(d)$ that if G fails to choose $\tilde{d}_2 \geq m/2$, $\epsilon_G \geq a_1$, already giving a constant lower bound on ϵ_G for this problem. This is the easier case; thus let us assume that $\tilde{d}_2 \geq m/2$, and consider the behavior of G on Problem 1. Referring to Figure 9, we see that for $0 \leq d \leq D_0$, $\hat{\epsilon}_1(d) \geq \hat{\epsilon}_2(d)$, and thus

$$\text{For } 0 \leq d \leq D_0, \;\; G(\hat{\epsilon}_1(d), d/m) \geq G(\hat{\epsilon}_2(d), d/m) \quad (18)$$

(because penalty-based algorithms assign greater penalties for greater training error or greater complexity). Since we have assumed that $\tilde{d}_2 \geq m/2$, we know that

$$\text{For } d < m/2, \;\; G(\hat{\epsilon}_2(d), d/m) \geq G(\hat{\epsilon}_2(\tilde{d}), \tilde{d}/m) \quad (19)$$

and in particular, this inequality holds for $0 \leq d \leq D_0$. On the other hand, by our choice of $\hat{\epsilon}_1(d)$ and $\hat{\epsilon}_2(d)$, $\hat{\epsilon}_1(\tilde{d}_2) = \hat{\epsilon}_2(\tilde{d}_2) = 0$. Therefore,

$$G(\hat{\epsilon}_1(\tilde{d}_2), \tilde{d}_2/m) = G(\hat{\epsilon}_2(\tilde{d}_2), \tilde{d}_2/m) \; . \quad (20)$$

Combining the two inequalities above (Equation 20 and Equation 21) with Equation 22, we have that

$$\text{For } 0 \leq d \leq D_0, \;\; G(\hat{\epsilon}_1(d), d/m) \geq G(\hat{\epsilon}_1(\tilde{d}_2), \tilde{d}_2/m) \quad (21)$$

from which it directly follows that in Problem 1, G *cannot* choose $0 \leq \tilde{d}_1 \leq D_0$. By the second condition on Problem 1 above, this implies that $\epsilon_G \geq \epsilon(D_0)$; if we arrange that $D_0 = cm$ for some constant c, then we have a constant lower bound on ϵ_G for Problem 1.

Due to space limitations, we defer the precise descriptions of Problems 1 and 2 for the full paper. However, in Problem 1 the classes F_d are essentially those for the intervals model selection problem, and in Problem 2 the F_d are based on parity functions. □

We note that although Theorem 3 was designed to create two model selection problems with the most disparate behavior possible, the proof technique can be used to give lower bounds on the generalization error of penalty-based algorithms under more general settings. In the full paper we will also argue that for the two problems considered, the generalization error of CV is in fact close to $\min_d\{\epsilon_i(d)\}$ (that is, within a small additive term that decreases rapidly with m) for *both* problems. Finally, we remark that Theorem 3 can be strengthened to hold for a *single* model selection problem (that is, a single function class sequence and distribution), with only the target function changing to obtain the two different behaviors. This rules out the salvation of the penalty-based algorithms via problem-specific parameters to be tuned, such as "effective dimension".

8 Conclusions

Based on both our experimental and theoretical results, we offer the following conclusions:

- Model selection algorithms that attempt to reconstruct the curve $\epsilon(d)$ solely by examining the curve $\hat{\epsilon}(d)$ often have a tendency to overcode or undercode in their hypothesis for small sample sizes, which is exactly the sample size regime in which model selection is an issue. Such tendencies are not easily eliminated without suffering the reverse tendency.
- There exist model selection problems in which a hypothesis whose complexity is close to the sample size should be chosen, and in which a hypothesis whose complexity is close to 0 should be chosen, but that generate $\hat{\epsilon}(d)$ curves with insufficient information to distinguish which is the case. The penalty-based algorithms cannot succeed in both cases, whereas CV can.
- The error of CV can be bounded in terms of the error of any other algorithm. The only cases in which the CV error may be dramatically worse are those in which phase transitions occur in the underlying learning curves at a sample size larger than that held out for training by CV.

Thus we see that both types of algorithms considered have their own Achilles' Heel. For penalty-based algorithms, it is an inability to distinguish two types of problems that call for drastically different hypothesis complexities. For CV, it is phase transitions that unluckily fall between $(1-\gamma)m$ examples and m examples. On balance, we feel that the evidence we have gathered favors use of CV in most common circumstances. Perhaps the best way of stating our position is as follows: given the general upper bound on CV error we have obtained, and the limited applicability of any fixed penalty-based rule demonstrated by Theorem 3 and the experimental results, the burden of proof lies with the practitioner who favors an penalty-based algorithm over CV. In other words, such a practitioner should have concrete evidence (experimental or theoretical) that their algorithm will outperform CV on the problem of interest. Such evidence

must arise from detailed problem-specific knowledge, since we have demonstrated here the diversity of behavior that is possible in natural model selection problems.

Acknowledgements

We give warm thanks to Yoav Freund and Ronitt Rubinfeld for their collaboration on various portions of the work presented here, and for their insightful comments. Thanks to Sebastian Seung and Vladimir Vapnik for interesting and helpful conversations. Dana Ron would like to thank the Eshkol Fellowship for its support. Part of this work was done while Yishai Mansour, Andrew Y. Ng, and Dana Ron were visiting AT&T Bell Laboratories.

References

[1] A. R. Barron and T. M. Cover. Minimum complexity density estimation. *IEEE Transactions on Information Theory*, 37:1034–1054, 1991.

[2] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley, 1991.

[3] D. Haussler, M. Kearns, H.S. Seung, and N. Tishby. Rigourous learning curve bounds from statistical mechanics. In *Proceedings of the Seventh Annual ACM Confernce on Computational Learning Theory*, pages 76–87, 1994.

[4] J. R. Quinlan and R. L. Rivest. Inferring decision trees using the minimum description length principle. *Information and Computation*, 80(3):227–248, 1989.

[5] J. Rissanen. Modeling by shortest data description. *Automatica*, 14:465–471, 1978.

[6] J. Rissanen. Stochastic complexity and modeling. *Annals of Statistics*, 14(3):1080–1100, 1986.

[7] J. Rissanen. *Stochastic Complexity in Statistical Inquiry*, volume 15 of *Series in Computer Science*. World Scientific, 1989.

[8] H. S. Seung, H. Sompolinsky, and N. Tishby. Statistical mechanics of learning from examples. *Physical Review*, A45:6056–6091, 1992.

[9] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society B*, 36:111–147, 1974.

[10] M. Stone. Asymptotics for and against cross-validation. *Biometrika*, 64(1):29–35, 1977.

[11] V. N. Vapnik. *Estimation of Dependences Based on Empirical Data*. Springer-Verlag, New York, 1982.

[12] V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16(2):264–280, 1971.

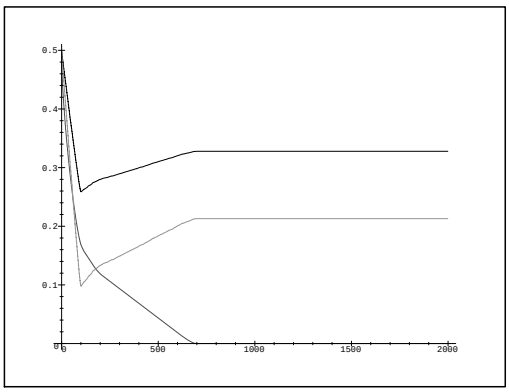

Figure 1: Experimental plots of the functions $\epsilon(d)$ (lower curve with local minimum), $\epsilon^{\eta}(d)$ (upper curve with local minimum) and $\hat{\epsilon}(d)$ (monotonically decreasing curve) versus complexity d for a target function of 100 alternating intervals, sample size 2000 and noise rate $\eta = 0.2$. Each data point represents an average over 10 trials. The flattening of $\epsilon(d)$ and $\epsilon^{\eta}(d)$ occurs at the point where the noisy sample can be realized with no training error.

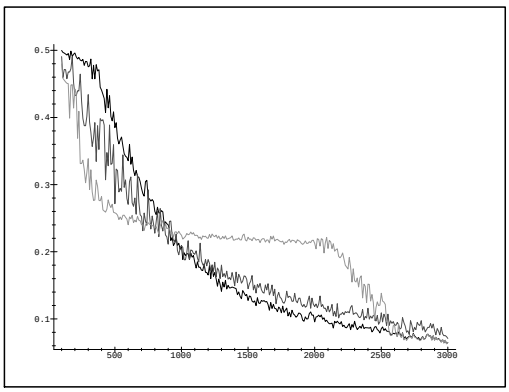

Figure 2: Experimental plots of generalization errors $\epsilon_{\mathrm{MDL}}(m)$ (most rapid initial decrease), $\epsilon_{\mathrm{CV}}(m)$ (intermediate initial decrease) and $\epsilon_{\mathrm{GRM}}(m)$ (least rapid initial decrease) versus sample size m for a target function of 100 alternating intervals and noise rate $\eta = 0.20$. Each data point represents an average over 10 trials.

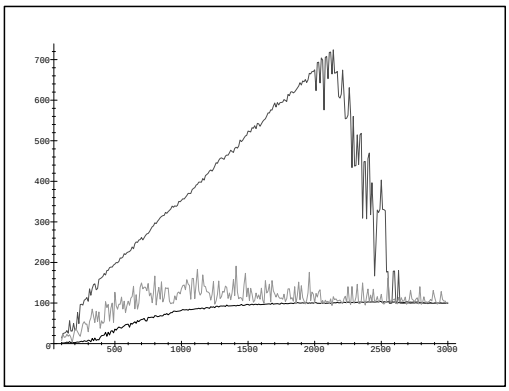

Figure 3: Experimental plots of hypothesis lengths $\tilde{d}_{\mathrm{MDL}}(m)$ (most rapid initial increase), $\tilde{d}_{\mathrm{CV}}(m)$ (intermediate initial increase) and $\tilde{d}_{\mathrm{GRM}}(m)$ (least rapid initial increase) versus sample size m for a target function of 100 alternating intervals and noise rate $\eta = 0.20$. Each data point represents an average over 10 trials.

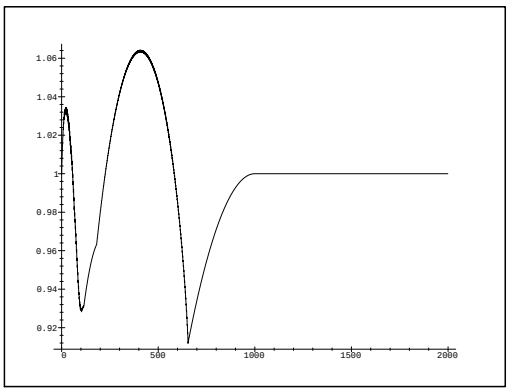

Figure 4: MDL total penalty $\mathcal{H}(\hat{\epsilon}(d)) + \mathcal{H}(d/m)$ versus complexity d for a single run on 2000 examples of a target function of 100 alternating intervals and noise rate $\eta = 0.20$. There is a local minimum at approximately $d = 100$, and the global minimum at the point of consistency with the noisy sample.

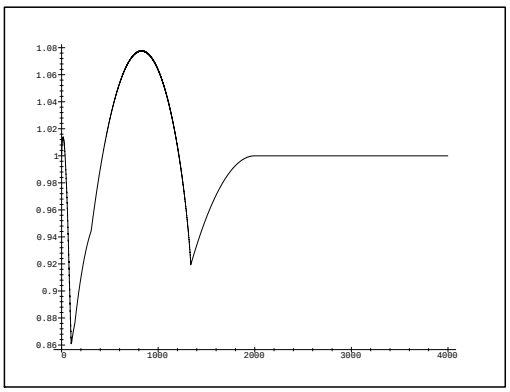

Figure 5: MDL total penalty $\mathcal{H}(\hat{\epsilon}(d)) + \mathcal{H}(d/m)$ versus complexity d for a single run on 4000 examples of a target function of 100 alternating intervals and noise rate $\eta = 0.20$. The global minimum has now switched from the point of consistency to the target value of 100.

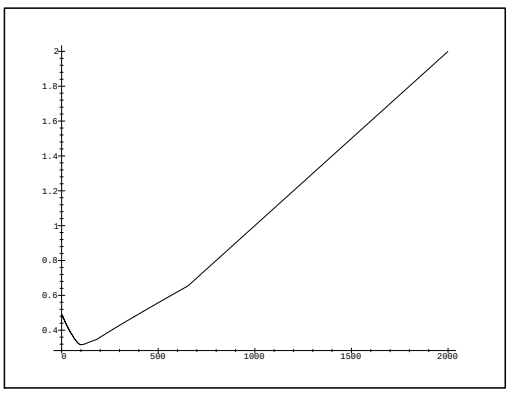

Figure 6: GRM total penalty $\hat{\epsilon}(d) + (d/m)(1 + \sqrt{1 + \hat{\epsilon}(d)m/d})$ versus complexity d for a single run on 2000 examples of a target function of 100 alternating intervals and noise rate $\eta = 0.20$.

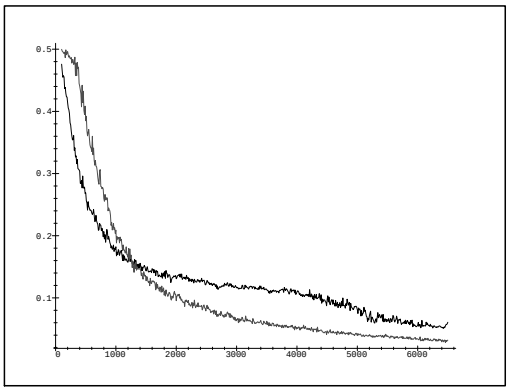

Figure 7: Experimental plots of generalization error $\epsilon_{\text{GRM}}(m)$ using complexity penalty multipliers 1.0 (slow initial decrease) and 0.5 (rapid initial decrease) on the complexity penalty term $(d/m)(1 + \sqrt{1 + \hat{\epsilon}(d)m/d})$ versus sample size m on a target of 100 alternating intervals and noise rate $\eta = 0.20$. Each data point represents an average over 10 trials.

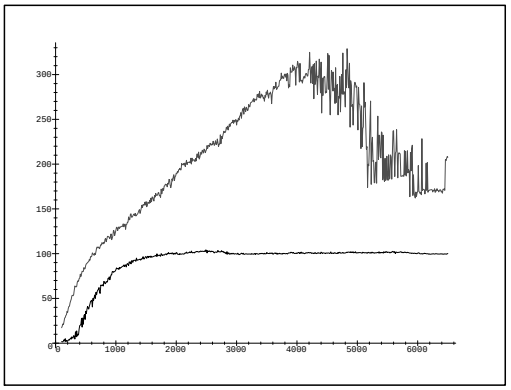

Figure 8: Experimental plots of hypothesis length $\tilde{d}_{\text{GRM}}(m)$ using complexity penalty multipliers 1.0 (slow initial increase) and 0.5 (rapid initial increase) on the complexity penalty term $(d/m)(1 + \sqrt{1 + \hat{\epsilon}(d)m/d})$ versus sample size m on a target of 100 alternating intervals and noise rate $\eta = 0.20$. Each data point represents an average over 10 trials.

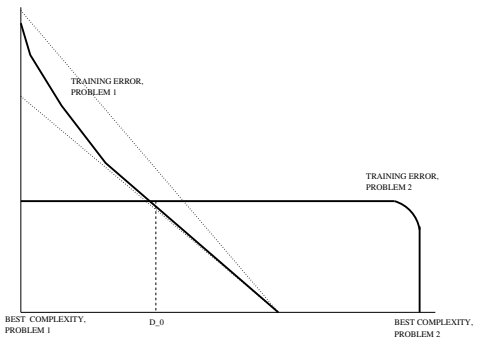

Figure 9: Figure illustrating the proof of Theorem 3. The dark lines indicate typical behavior for the two training error curves $\hat{\epsilon}_1(d)$ and $\hat{\epsilon}_2(d)$, and the dashed lines indicate the provable bounds on $\hat{\epsilon}_1(d)$.