Some General Results on Overfitting in Machine Learning

Antony Van der Mude

AT&T
Room 2F-319, 101 Crawfords Corner Road
Holmdel, New Jersey 07733
vandermude@att.com

Abstract

Overfitting has always been a problem in machine learning. Recently a related phenomenon called "oversearching" has been analyzed. This paper takes a theoretical approach using a very general methodology covering most learning paradigms in current use. Overfitting is defined in terms of the "expressive accuracy" of a model for the data, rather than "predictive accuracy". The results show that even if the learner can identify a set of best models, overfitting will cause it to bounce from one model to another. Overfitting is ameliorated by having the learner bound the search space, and bounding is equivalent to using an accuracy (or bias) more restrictive than the problem accuracy. Also, Ramsey's Theorem shows that every data sequence has an situation where either consistent overfitting or underfitting is unavoidable. We show that oversearching is simply overfitting where the resource used to express a model is the search space itself rather than a more common resource such as a program that executes the model. We show that the smallest data sequence guessing a model defines a canonical resource. There is an equivalence in the limit between any two resources to express the same model space, but it may not be effectively computable.

Introduction

For as long as machine learning has been attempted, it has been known that overfitting has been a problem. For example, in his pioneering paper on machine learning of checkers playing, Samuel notes "A second defect [with learning-by-generalization] seemed to be connected with the too frequent introduction of new terms into the scoring polynomial and the tendency for these new terms to assume dominant positions on the basis of insufficient evidence" [Samuel 59]. Overfitting can be thought of as the incorporation of "noise" in the data into the induced model. What is meant by noise is any relationship that is found during training that is not found in subsequent testing. It is usually thought that overfitting arises from the selection of a model that is more detailed than the data admits. It has also been found that overfitting can occur simply if one searches long enough - this is termed "oversearching" [Quinlan and Cameron-Jones 95].

This paper is an attempt to derive some general principles about overfitting. There have been many successful results, both theoretical and experimental that relate to overfitting ([Quinlan 86], [Kibler and Langley 88], [Pfahringer 95]). This paper begins with a definition of learning that is sufficiently general to encompass all types of learning that are usually studied. From this very abstract definition, the derived results will apply to most types of learning.

The basic paradigm of machine learning used here is one where the learning machine, presented with $\mathbf{t}$ items of data, searches a space of models and selects the best model $\mathbf{M}$ based on some evaluation function that quantifies the relative accuracy of each model. We consider a learning problem to consist of a set of cases. Each case is manifested by a set of data sequences. For convenience in the proofs, the data sequences are assumed to be all infinite. For each case, many different models can fit the data. An example of a problem is the set of all regular expressions that involve the star operator. Any data sequence for each case has the same elements - the infinite set of strings that make up the regular set. The set of best models for this case consists of all finite state automata that accept the strings generated by the regular expression.

How well a model fits the data depends on a particular problem's definition - it is just some sort of logical predicate that defines the model's accuracy on the data. Although for most problems, the cases partition both the space of data sequences and the space of models into discrete sets, this assumption is not needed for this paper.

Several criteria define successful learning. Examples are Probabilistically Approximately Correct learning (PAC learning) and Identification in the Limit. The formalism we use does not treat these as any more than subtypes of a more general view of learning. Identification in the Limit offers a restrictive criterion that applies to the process of learning as a whole. In PAC learning, we can either say that each case has data sequences where noisy data leads to errors, or that the accuracy criterion has a stochastic quality to it, or both. Since we are discussing the localized learning behavior

for the most part, we do not usually need to select a particular global criterion. In those situations where we do, we have chosen a weak form of learning - a generalization of Behaviorally Correct Identification [Case and Smith 1983] - where correct behavior simply means that the model fits that data according to the arbitrary accuracy function.

Given some problem and a learning criterion, the determination of overfitting or underfitting is a hard concept to formalize in a general sense. Typically, overfitting is considered to happen because of fitting data with a model which, although it succeeds at fitting the data seen so far, (the "training set") it is not as good at fitting the whole data sequence. This focuses on the "predictive accuracy" of the model. The problem with this view is that the input to the learning procedure is usually the initial segment of an infinite number of possible data sequences. This means that if a learning system chooses some model **M** for the data, it may be perfectly acceptable for some situations, but be considered overfitting (or even underfitting) in the context of another data sequence.

As an example of this problem, let the data sequence begin with the values <16, 256, 2, 4>. A natural presumption is that an acceptable model for the data is the rule 2^n, assuming that 8, 32, 64 and 128 will come later. If the value 6 comes along, one concludes that this initial guess had overfit the data, leading to a corrected guess of **2n**. Similarly, this initial guess underfits if no other value less than 256 ever appears. Then, the guess of 2^n is better, leading to the prediction that the next smallest value is 65536. Therefore, the single guess made by the learning machine can be interpreted in three different ways, depending on the future data presentation.

This paper will instead use a concept of overfitting that compares the relative accuracy of an arbitrary model against some ideal accuracy defined relative to the initial segment of the data seen so far. This can be considered an intrinsic measure of overfitting; an "expressive accuracy" rather than predictive. The problem is that there is no obvious general method to compute this ideal accuracy. There is usually no single best value that can be used to derive this ideal accuracy for all learning problems. Even an optimum accuracy measure (such as an average) that can be derived from a family of models, cannot be defined in a way that is generally applicable to all learning problems. I will derive my results without reference to a particular method of computing the reference point by which over or underfitting is determined. Thus, these results will be applicable to any definition that says that model **M** overfits data D^t if the accuracy of the model on the data is better than some reference value, and underfits if it is not as accurate.

The results are divided into two parts. The first part derives some properties of overfitting and search functions. These results apply to any arbitrary model space. In the second section we show that the model space can be expressed in terms of any arbitrary computational resource. A resource is any parameter, such as the number of lines of program code, the number of branches in a search tree, the amount of time that a learning program runs. Thus, a resource is anything that can be expressed as a finite set of units, each of which may improve the accuracy of the model in some particular situation. In this viewpoint, oversearching becomes overfitting if the model space is redefined in terms of the location of the model in the search space instead of the number of rules in the model.

Definitions

Problem Definition

A Learning Problem is defined to be a four-tuple $P=<C_P, D_P, M_P, A_P>$, where $C \in C_P$ is a problem case, $D^\infty \in D_P(C)$ is a data sequence from the space of possible data sequences, $M \in M_P(C)$ is a model for the data selected from the space of models, $A_P(M|D^\infty) \rightarrow A$, is an accuracy measure that defines the relative accuracy of the models on a given data sequence. The set of cases C_P is assumed to be infinite.

An input data sequence D^∞ is assumed to be infinite and without repetitions. Also, any two sequences D^∞ and D'^∞ that are composed of the same values in a different order must be part of the same case $D_P(C)$. This is for convenience in the proofs. It does not result in a loss of generality, since features such as the data set size or presentation order can be easily expressed in a framework that equates different presentation orders. For example, to construct problems where some cases are either a finite or infinite set of values **S**, the data sequence can be cylinderized - that is, each $D^\infty \in D_P$ is the set **SXN**. The input being presented to the learning machine at time **t** is D^t - the first **t** elements of D^∞.

In general, we do not assume a problem has data sequences that are recursively enumerable [Van der Mude 86]. In practice, this assumption is true for machine learning - actually, most applications are sets of total recursive functions of low computational complexity.

The problem accuracy compares the fit of models to data. The term 'bias' is sometimes used to describe this measure [Mitchell 80]. The accuracy measure is a function whose range **A** has a total ordering, such as the integers, rationals or reals. There is a zero value representing an exact match, but there may be no upper limit to the accuracy measure. The smaller the value $A_P(M|D^\infty) \in A$ is, the better the fit of the model to the data. We shall not presume that a learning machine can compute A_P. In fact, it may be possible that A_P is not effectively computable at all.

For all $D^\infty \in D_P$, define $M_P(D^\infty)$ to be the set of all **M** where $A_P(M|D^\infty)$ is minimal. The set of models $M_P(C)$ is defined to be the union of $M_P(D^\infty)$ for all $D^\infty \in D_P(C)$. For all **C** and **M**, if $M \notin M_P(C)$ then for every $D^\infty \in D_P(C)$ there is a $M' \in M_P(C)$ where $A_P(M'|D^\infty) < A_P(M|D^\infty)$. For the

purposes of this paper, we will assume that every model can be expressed as a finite set of integers. Other equivalent expressions are a character string or an integer for each model.

We shall assume that the accuracy function A_P is approximable by some computable function **A**. In the limit, $A(MID^t)$ approaches $A_P(MID^\infty)$, at least for all $M \in M_P(D^\infty)$. No special definition is given beyond the following general requirements that define a well-behaved accuracy function. If the accuracy $A_P(MID^\infty)=0$ then $A(MID^t)$ must get arbitrarily close to zero. That is, for all **t** if $A(MID^t)>A_P(MID^\infty)$, then there is a time **s** such that for all $r>s$, $A(MID^t)>A(MID^r)$. The symmetrical relation exists if $A(MID^t)<A_P(MID^\infty)$. Also, the relative ordering of the accuracies must match: that is, for all D^∞ and for all **M** and **M′**, if $A_P(MID^\infty)<A_P(M'ID^\infty)$ there is a time **t** such that for all $s>t$, $A(MID^t)<A(M'ID^t)$.

A computable noise-free problem **P** is one where there is an effective procedure $I_P(M)$ to generate indices of recursively enumerable sets such that for all $C \in C_P$ and all $D^\infty \in D_P(C)$ there is a $M \in M_P(C)$ where $I_P(M)=D^\infty$ and $A_P(MID^\infty)=0$. That is, every **M** is a model for an recursively enumerable data sequence and every data sequence in the problem has a model. A problem that has noise, on the other hand, would contain cases where the data cannot be exactly fit by any model, thus the accuracy can never really go to zero for that case.

Most learning functions search a space of models that provides alternatives that fit the data seen so far better than any previous guess. This characteristic of a model space will be termed the refinement property: given any model, either the model is completely accurate, or it can be improved (refined) in some way [Laird 88]. More formally, the refinement property holds if, for all D^t and **M**, either $A(MID^t)=0$ or else there exists an **M′** such that $A(M'ID^t)<A(MID^t)$.

Learning Machine and Learning Criterion

Let **P** be a problem and $L(D^t)$ be a learning machine which outputs a value in M_P. **L** is defined for all $D^t \subset D^\infty$ and for any $D^\infty \in D_P$. A learning machine $L(D^t)$ is termed a search function if it selects the best model from a finite set $S(D^t)$ - the model **M** with the minimum goodness measure $G(MID^t)$. The goodness value may be the accuracy $A(MID^t)$, but this is not necessary.

Some theorems consider the overall learning criterion. We shall use the weak variant of Identification in the limit, Behaviorally Correct Identification. BC identification occurs if there is a time **t** such that for all $s>t$, $L(D^s) \in M_P(D^\infty)$.

Over/Under fitting

Over and underfitting is some sort of comparison of the accuracy of an arbitrary model **M** on some initial data sequence D^t compared against the accuracy of the best models $M_P(D^\infty)$ on D^t. As discussed above, there is no single acceptable way to combine the accuracies of all of these models. For the purposes of this paper, we shall not select a method. We shall instead define over and underfitting to be defined relative to a cutoff value $F(D^t)$. The function is defined relative to the finite set D^t, without knowledge of which data sequence D^∞ it is part of. The function **F** may or may not be effectively computable. The learning machine $L(D^t)=M$ overfits if $A(MID^t)<F(D^t)$ and underfits if $A(MID^t)>F(D^t)$.

If **P** has noise, the best models may not have zero accuracy, in contrast to the noise free problem where every data sequence has a model that exactly matches it. With noisy data, it is possible for the accuracy, even for one of the best models, to bounce between over and underfitting. This situation can occur also with noise-free data, depending upon how $F(D^t)$ is defined. The learning function **L** chronically overfits D^∞ if there is a **t** for all time $s>t$, $L(D^s)$ overfits.

General Results

The concept of a well-behaved accuracy reasonably matches the kind of behaviors we expect when considering learning problems. For example, it is possible to show that a well-behaved accuracy can be constructed for any computable noise-free problem.

Theorem 1.1: Let **P** be a computable noise-free problem. Then there exists a total function **A** that is a well behaved accuracy function.

Proof: Let any noise-free problem **P** be given. By the definition, for all $C \in C_P$ and all $D^\infty \in D_P(C)$ there is a $M \in M_P(C)$ where $I_P(M)=D^\infty$ and $A_P(MID^\infty)=0$. Compute **A** as follows: Let any input D^t and model **M** be given. For the given **M** enumerate $I_P(M)=D'^\infty$ for **t** steps. Call this D'^t. **A** will return the value $1/s$ where **s** is the first point where the two sequences D^t and D'^t differ. If they do not differ, return $1/(1+r)$ where **r** is the largest value in D^t. To show the accuracy is well behaved, we need to show the following. In the limit, $A(MID^t)$ approaches $A_P(MID^\infty)$, at least for all $M \in M_P(D^\infty)$. If the accuracy $A_P(MID^\infty)=0$ then $A(MID^t)$ will get arbitrarily close to zero. For all D^∞ and for all **M** and **M′**, if $A_P(MID^\infty)<A_P(M'ID^\infty)$ there is a time **t** such that for all $s>t$, $A(MID^t)<A(M'ID^t)$. For all D^t and **M**, either $A(MID^t)=0$ or else there exists an **M′** such that $A(M'ID^t)<A(MID^t)$. As $t \to \infty$ the accuracy for $M \in M_P(D^\infty)$ goes to zero, and every other model stops at some point where the first difference occurs. ■

This argument works for problems with partial recursive data sequences as well as total recursive data. Although it is much harder to make a generalization about noisy data sequences and data sequences where there is a complicated relationship between the data sequence for the model and the data sequences for each case, we shall always assume that the accuracy function is approximable by some computable function.

Overfitting is usually considered to happen because a model is chosen for some data sequence that matches the data well up to that point, but is not the best model for the data overall. But overfitting can occur even if you have focused on a set of models all of which are good choices for the problem. The problem is that if you have the minimum accuracy as the criterion for choice, then overfitting is inevitable. Even if a learning machine limits itself to only a finite number of correct choices, overfitting can occur.

Theorem 1.2: Let any problem P be given. Let $L(D^t)$ be a learning machine whose domain is finite subsets D^t of sequences $D^{\sim} \in D_P$ and whose range is in M_P. Assume $L(D^t)$ and $F(D^t)$ are defined such that for $D^{\sim} \in D_P(C)$ there exists a time t such that for all $r>t$ there is a set $S \subset M_P$ such that for all $M \in S$, $A(L(D^r)|D^r) \leq A(M|D^r)$ and there are two $M' \in S$ and $M'' \in S$ where $A(M'|D^r) < F(D^r)$ and $A(M''|D^r) > F(D^r)$. Then L chronically overfits $D^{\sim}$.

Proof: Obvious. Since at least one model M exists where $A(M|D^r) < F(D^r)$ for every $r>t$, then the learning machine will choose one of these overly accurate models. ■

The natural corollary to this observation is that chronic overfitting does not occur if the set of models as a whole is sometimes more, sometimes less accurate than the cutoff. This condition can arise if there is no real difference between the models in the set - they are just different expressions of the same property - and the F function differs from the accuracy of this unique property. Where the problem has only one data sequence for each case, it may be possible for the accuracy and the F function to match. This is not true for some problems where the matching of model to data is stochastic in nature. Then there can be two models whose explanation of the data may have different underlying causes, but the same effect in the long run. This would yield a different relative accuracy at different times on the same data, but the same accuracy in the limit and Theorem 1.2 would apply.

Corollary 1.3: Let any problem P and $L(D^t)$ be a learning machine whose domain is finite subsets D^t of sequences $D^{\sim} \in D_P$ and whose range is in M_P. Assume $L(D^t)$ and $F(D^t)$ are defined such that for $D^{\sim} \in D_P(C)$ there exists a time t such that for all $r>t$ there is a set $S(D^r) \subset M_P$ such that for all $M \in S(D^r)$, $A(L(D^r)|D^r) \leq A(M|D^r)$, but it is not true that L chronically overfits $D^{\sim}$, then there are an infinite number of times r such that for all $M \in S(D^r)$ we have $A(M|D^r) \geq F(D^r)$.

A conclusion that can be drawn from this simple observation is that any search-based learning method will not effectively avoid overfitting on a problem with many different answers if all it does is limit the search space. Another conclusion is that something more than just the accuracy must be factored in to avoid overfitting.

Let us turn to the situation where there is only one good model under different names. That is, every $M \in M_P(C)$ has the same accuracy for the same data at the same time $A(M|D^t)$. It would be nice to set $F(D^t)=A(M|D^t)$. But if the problem has the refinement property it is not possible to assume that $F(D^t)=A(M|D^t)$, since the data sequence D^t can be a subset of any number of data sequences $D^{\sim}$ in any number of $D_P(C)$, and each of the $M \in M_P(C)$ could have a different accuracy for the same data $A(M|D^t)$.

Theorem 1.4: Let P be a problem with the refinement property where for all C and all $M \in M_P(C)$, $M' \in M_P(C)$ and $D^{\sim} \in D_P$ and t, $A(M|D^t)=A(M'|D^t)$. For all $D^{\sim} \in D_P$ and t if $F(D^t) \neq 0$ then there are an infinite number of C where $M \in M_P(C)$ and $D'^{\sim} \in D_P(C)$ such that $D^t \subset D'^{\sim}$ but $F(D^t) \neq A(M|D^t)$.

Proof: Let any problem P be given. For all D^t and M, either $A(M|D^t)=0$ or else there exists an M' such that $A(M'|D^t) < A(M|D^t)$. If $F(D^t) \neq 0$ then there are either an infinite number of C where $M \in M_P(C)$ and $D'^{\sim} \in D_P(C)$ such that $D^t \subset D'^{\sim}$ and $A(M|D^t)=0$ or $A(M|D^t)=F(D^t)$ and there exists an M' such that $A(M'|D^t) < A(M|D^t)$ and $A(M'|D^t) \neq F(D^t)$. ■

The best we can assume is that for all $D^{\sim} \in D_P(C)$ where $M \in M_P(C)$, it is true that for all t there is an $s>t$ and an $r>t$ where $F(D^s) \leq A(M|D^s)$ and $F(D^r) \geq A(M|D^r)$, so that M neither chronically overfits nor underfits.

Assume that we have a learning machine based on search that chronically overfits. Can it be fixed? Search means that the learning machine selects some finite set of models $S(D^t)$ and chooses one of these as the best guess - best being the minimization of some function G. We can adjust the search space to keep out models that are artificially accurate (i.e., that overfit). But we can alternatively adjust the minimization function so that no model chosen by G is more accurate than the best one. The G function matches A for the model selected by the learning machine; there exist cases where a model is eliminated from contention by setting $G(M|D^t) > A(M|D^t)$. The following theorem shows that both methods are equivalent. The theorem that follows deals with a single data sequence D^t. Since the construction is uniform in the definitions of the learning machine, it shows that both methods are effectively equivalent for patching a chronically overfitting learning machine.

Theorem 1.5: Let P be any problem. Let L and S be such that for any $D^{\sim} \in D_P(C)$ and t there is a finite set $S(D^t) \subset M_P$ such that for all $M \in S(D^t)$, $A(L(D^t)|D^t) \leq A(M|D^t)$ and $L(D^t)$ overfits. There is an L' and S' such that there is a finite set $S'(D^t) \subset S(D^t) \subset M_P$ such that for all $M \in S$, $A(L'(D^t)|D^t) \leq A(M|D^t)$ and L' does not overfit iff there is an L'' and G such that $G(L''(D^t)|D^t) = A(L''(D^t)|D^t)$, $G(L''(D^t)|D^t) \leq G(M|D^t)$ for all $M \in S(D^t)$, and for all $M' \in M_P$, if $A(M'|D^t) < F(D^t)$ then

$G(M'|D^t)>A(M'|D^t)$ and L'' does not overfit.

Proof: Let L and S be such that for $D^\infty \in D_P(C)$ there exists a time t such that for all $r>t$ there is a finite set $S(D^r)\subset M_P$ such that for all $M\in S(D^r)$, $A(L(D^r)|D^r)\leq A(M|D^r)$. We shall use S in the construction of the other machines.

$\rightarrow$: Assume there is an L' and S' such that there is a finite set $S'(D^t)\subset S(D^t)\subset M_P$ such that for all $M\in S$, $A(L'(D^t)|D^t)\leq A(M|D^t)$. Construct L'' as follows: Given D^t, compute $S(D^t)$ and $S'(D^t)$ For all $M\in S'(D^t)$, $G(M|D^t)=A(M|D^t)$. For all $M'\in(S(D^t)-S'(D^t))$, $G(M'|D^t)=\max\{A(M'|D^t) \mid M'\in S'(D^t)\}$. If the set $S'(D^t)$ is enumerated first, then the M returned by $L''(D^t)$ is the first one where $G(M|D^t)=A(L'(D^t)|D^t)$, so $L'(D^t)=L''(D^t)$. By the construction of G, $G(L''(D^t)|D^t)=A(L''(D^t)|D^t)$, and for all $M'\in M_P$, if $A(M'|D^t)<F(D^t)$ then $G(M'|D^t)>A(M'|D^t)$. Since L' does not overfit and $L'(D^t)=L''(D^t)$, L'' does not overfit either.

$\leftarrow$: Assume there is an L' and G such that for all $M\in S(D^t)$, $G(L'(D^t)|D^t)\leq G(M|D^t)$. Construct L'' as follows: Given D^t, compute $S(D^t)$. Define $S'(D^t)$ as follows: Given any $M\in S(D^t)$, add M to $S'(D^t)$ iff $G(M|D^t)\leq A(M|D^t)$. By the definition of G, there exists an $M\in S(D^t)$ where $G(M|D^t)=A(M|D^t)$, so $S'(D^t)$ is not empty. By the definition of G, given any $M'\in M_P$, if $A(M'|D^t)<F(D^t)$ then $G(M'|D^t)>A(M'|D^t)$, so M' is not added to $S'(D^t)$. Since L' does not overfit and $L'(D^t)=L''(D^t)$, L'' does not overfit either. ■

The previous theorem showed how to fix a learning machine when it overfits. The question is, does the learning machine actually do anything like learn after it is fixed? We shall presume that the finite set of candidates that are selected by the function S is constantly growing in size. Once a model is chosen, it remains in the set of candidates and is judged on the basis of its accuracy. The next theorem states that Behaviorally Correct identification is possible if and only if S grows to include alternative models slower than the best model for the data converges to the optimum accuracy.

Theorem 1.6: Let P be a problem and L and S be such that for $D^\infty \in D_P(C)$ and t there is a finite set $S(D^t)\subset M_P$ such that for all $M\in S(D^t)$, $A(L(D^t)|D^t)\leq A(M|D^t)$. Also, for all D^s and D^t if $s<t$, then $S(D^s)\subset S(D^t)$.

For all C and all $D^\infty \in D_P(C)$, L BC-identifies D^∞ iff there is a $M\in M_P(D^\infty)$ and t such that $M\in S(D^t)$ and for all $s>t$ and all $M'\notin M_P(D^\infty)$ where for all $M''\in M_P(D^\infty)$ and $M''\in S(D^t)$ we have $A(M'|D^s)<A(M''|D^t)$ then $M\notin S(D^s)$.

Proof: Obvious . ■

We can show that the function G in Theorem 1.5 will preserve BC identification if, although $A(M|D^t)<G(M|D^t)$, if it ever happens that $G(M|D^t)=A(M|D^t)$ then this equivalence must be true for M from then on.

Theorem 1.7: Let P be a problem and L and S be such that for $D^\infty \in D_P$ and t there is a finite set $S(D^t)\subset M_P$ such that for all $s<t$, $S(D^s)\subset S(D^t)$ and for all $M\in S(D^t)$, $A(L(D^t)|D^t)\leq A(M|D^t)$. Let L BC-identify P.

Let G and L' be defined so that for all $D^\infty \in D_P$ and t, $L'(D^t)\in S(D^t)$ and for all $M\in S(D^t)$, $G(L'(D^t)|D^t)\leq G(M|D^t)$ and $A(M|D^t)\leq G(M|D^t)$. If for all $D^\infty \in D_P$ and t, $G(L'(D^t)|D^t)=A(L'(D^t)|D^t)$, and if $G(M|D^t)=A(M|D^t)$ implies that for all $s\geq t$ $G(M|D^s)=A(M|D^s)$ and there is an $M'\in M_P(D^\infty)$ and r where $G(M'|D^r)=A(M'|D^r)$ then L' which uses G in place of A, must BC-identify P also.

Proof: By the construction in Theorem 1.5, define $S'(D^t)$ from G and $S(D^t)$ as follows: Given any $M\in S(D^t)$, add M to $S'(D^t)$ iff $G(M|D^t)=A(M|D^t)$. Then for all $s<t$, $S'(D^s)\subset S'(D^t)$. By the definition of G, there exists an $M\in M_P(D^\infty)$ and t such that $M\in S'(D^t)$. Since L BC-identifies P, then by Theorem 1.6 we have for all $s>t$ and all $M\notin M_P(D^\infty)$ where for all $M'\in M_P(D^\infty)$ and $M'\in S(D^t)$ such that $A(M|D^s)<A(M'|D^t)$ then $M\notin S(D^s)$. This does not change in the construction of $S'(D^t)$ so L' must BC-identify P also. ■

We conclude this section with a proof that shows that for any problem, for any case and any data sequence for that case, there are situations where uniform underfitting or overfitting are inevitable. This assumes that there is noise in the data, so even the best model toggles between over and under fitting as more data is given. Since the chance of an exact match is vanishingly small, it can arbitrarily assigned to be either an overfit or underfit.

This theorem is an application of Ramsey theory. That is, for every problem complicated enough, there is a finite data sequence where every n-element subset of this larger set is consistently overfit or underfit:

Ramsey's Theorem (Finite Version): For all $k, l, r \in \omega$ there exists $n(k,l,r)\in\omega$ such that if $n\geq n(k,l,r)$ and $X:[n|k]\rightarrow[r]$ is any r-coloring of the k-element subsets of $[n]$, then some l-subset of $[n]$ has all its k-element subsets with the same color.

See [Graham 81] for a good introduction to this field. In applying this theorem, r is two. We shall arbitrarily assign exact matches to be either overfit or underfit.

Theorem 1.8: Let any noisy problem be given. For any values k and l and any case C and $D^\infty \in D_P(C)$ there is a sufficiently large n where there is a set D^l where $D^l\subset D^n\subset D^\infty$ and every data presentation $D^k\subset D^l$ underfits or every such D^k overfits.

Proof: This is immediate by Ramsey's Theorem, where the mapping $X:[n|k]\rightarrow[r]$ is the function F. ■

Resources

The original inspiration for this paper was the Quinlan and Cameron-Jones paper on oversearching. To quote from their abstract: "when learning classifiers, more extensive search for rules is shown to lead to lower predictive accuracy on many of the real-world domains investigated." Overfitting usually occurs when "the construction of theories more complex than can be justified by the data leads to poor predictive performance". This situation is the result of Theorem 1.6 in the previous section. Using Behaviorally Correct identification as the criterion for learning, the learning system does not chronically overfit if the search space does not include poor predictors as fast as the best model for the data converges to the optimum accuracy.

To apply the results in the previous section to oversearching requires an understanding of the notion of what a model space is. Although models for the data are often considered to be composed of a finite set of rules (for a classifier), a set of states (for a state machine), or even a set of program statements, a model space can actually be constructed using any arbitrary resource. The representation itself is usually considered as the definition of the model space. Here we are considering a model in some "pure" form that can be manifested in any way, as long as the accuracy of model to data remains unchanged

The oversearching phenomenon can be understood as an overfitting result, where the space of classifiers are expressed using the search space of the learning machine as the resource. Consider the overfitting phenomenon when a learning system guesses a model that is too precise - it incorporates unimportant variability in the data as part of the model. Oversearching is analogous; the unimportant variability arises from the effort to search for better models. The similarity is hidden because the models are expressed in terms of the same resource (number of lines of code or whatever), but the results are analyzed in terms of two different resources.

The following theorem shows that any machine that minimizes a function over a finite but increasing search space solves some sort of learning problem as defined in this paper. This establishes a canonical problem definition based on describing a model in the model space by the first data sequence that causes the learning machine to guess that model. This canonical form will be used as the basis from which any other expression of a model space can be expressed. This use of the first initial data segment as a canonical form is similar to its use as the basis of the order independence result in Identification in the Limit [Blum and Blum 75]. For the purposes of this theorem, the "pure" form of the model is some integer expressed in some undefined form.

Theorem 2.1: Let L, G and S be such that for all finite input sets D^t there is a finite set $S(D^t)$ such that $L(D^t)\in S(D^t)$ and for all integers $m\in S(D^t)$, $G(L(D^t)|D^t)\leq G(m|D^t)$. The function S is defined from finite sets to finite sets in such a way that if $D^t\subseteq D^r$ then $S(D^t)\subseteq S(D^r)$. Let some method of computing a limit value be given. There exists a problem $P=<C_P,D_P,M_P,A_P>$ where for every D^∞ such that there is a M and x such that in the limit as $t\to\infty$ $G(L(D^t)|D^t)=G(M|D^t)=x$, there is an C where $D^\infty\in D_P(C)$ and L BC-identifies P.

Proof: Let L be given, where $L(D^t)$ is computed by minimizing $G(M|D^t)$ for $M\in S(D^t)$ and S grows as given. Define $P=<C_P,D_P,M_P,A_P>$ as follows. M_P is the range of L. Choose some enumeration of the finite sequences. The value $M\in M_P$ is a representation of the first finite sequence D^t in the enumeration where $L(D^t)=m$, for some integer m. For every infinite sequence of integers D^∞ where the function $L(D^t)$ returns a value for all $D^t\subset D^\infty$, define $A_P(M|D^\infty)$ to be the limit of $G(M|D^t)$ as $t\to\infty$ for all $D^t\subset D^\infty$ if such a value exists. If there is no limit value for M on D^∞ but there is some $A_P(M'|D^\infty)$ defined so far then set $A_P(M|D^\infty)$ arbitrarily to some larger value than $A_P(M'|D^\infty)$. For every D^∞ where there is some $A_P(M|D^\infty)$ defined, and there is some time t where for all $s\geq t$ if $A_P(L(D^s)|D^\infty)=A_P(M|D^\infty)$, include D^∞ as a case $C\in C_P$ (i.e. $C=D^\infty$) and include $M'\in M_P(C)$ if $A_P(M|D^\infty)=A_P(M'|D^\infty)$. Each of these cases are BC-identified by L.■

With the canonical problem definition of Theorem 2.1 and the relationship of problem to BC-identification in Theorem 1.7 we have the basic relationship of problem to learning machine, in that the first finite data sequence where a value is in the range of the learning machine becomes the resource used to express the model. We now want to interrelate different learning machines. We can show that for two learning machines that are solving the same problem (except that the model spaces are expressed by the use of two different resources) and there are duplications of models in the space, then the model spaces can be put in one-to-one correspondence with each other in the limit.

Theorem 2.2: Let machines L and L' be such that for all M in the range of L there is an infinite number of M' in the range of L' such that for all finite sets D^t, $G(M|D^t)=G'(M'|D^t)$ and the same holds in the other direction - for all M' in the range of L' there is an infinite number of M in the range of L such that for all finite sets D^t, $G(M|D^t)=G'(M'|D^t)$. Then there is a one-to-one function f from the range of L to the range of L' and a function $g(M,t)$ such that for all M there is a t such that for all $s\geq t$, $g(M,s)=f(M)$.

Proof: The function $g(M,t)$ is defined as follows. For every M in the range of L, enumerate the elements M' in the range of L' not currently assigned via g to any $M''<M$. Check $G(M|D^s)$ against $G'(M'|D^s)$ for all $s<t$ and go to the next element in the range of L' if the values are different. If they are the same, then $g(M,t)=M'$. The function is total, since for every M and t a corresponding

M' can be found, since there are an infinite number that match. Eventually, every M will stabilize on a value for some t. Then, the next greater value will stabilize and so on. ■

Even though the values of the correspondence function are computable in the limit, there exist situations where they are not computable by any total function. This is a consequence of the very abstract and general definitions we are using in this paper. Because I have not restricted how the mapping from model names to accuracy is defined, I can have problems of arbitrary complexity. This is true even for classes where equality is computable, such as the identification of finite sets.

Theorem 2.3: Let machines L and L' be such that for all M in the range of L there is an infinite number of M' in the range of L' such that for all finite sets D^t, $G(M|D^t)=G'(M'|D^t)$ and the same holds in the other direction - for all M' in the range of L' there is an infinite number of M in the range of L such that for all finite sets D^t, $G(M|D^t)=G'(M'|D^t)$. Then there is a one-to-one function f from the range of L to the range of L' and a function $g(M,t)$ such that for all M there is a t such that for all $s \geq t$, $g(M,s)=f(M)$, but the function f is not effectively computable.

Proof: Define the problem $P=<C_P,D_P,M_P,A_P>$ as follows. C_P are finite sets. D_P are cylinders of finite sets. that is, for every finite set S in C_P, every element of SXN appears in the data sequence $D^\infty \in D_P(S)$. $M_P(S)$ is just the listing of the elements of the set S, with the number of models in each case made infinite by tagging each model with any integer. $A_P(M|D^\infty)$ is defined to be the difference between M and D^∞. $A_P(M|D^t)$ is defined similarly. The learning machine L is simply the function that extracts the set S from the data SXN in D^t. The value that is used as a tag may is simply the number t for input D^t.

Define a new problem $P'=<C_P,D_P,M'_P,A_P>$ that is virtually identical to P except that the model space is defined as $M \in M_P$ if M is the set of state transitions for some Turing Machine TM_i. If W_i is the recursively enumerable set that is defined to be the domain of TM_i, then $TM_i \in M_P(S)$ if $W_i=S$.

A learning machine L' can be defined as follows: for $L'(D^t)$, let z be the maximum value for some $<z,y>$ in D^t for some value y. Let v be the maximum of the two values z and t. For each value i, run TM_i from 0 to v for t steps. Halt on the first W_i that matches $\{ x \mid <x,y> \in D^t \}$ for as much of W_i as can be computed in t steps, for the range 0 to v. Both machines learn their associated problems in the limit. There is also a one-to-one function f from the range of L to the range of L' but the function f is not effectively computable, otherwise the Halting Problem would be solvable. ■

Taken together, the three results show that: (1) the data sequences can be used as a canonical resource to represent the model space, (2) any other resource is equivalent in the limit, and (3) resources in general have no effective correspondence. Thus, the Quinlan and Cameron-Jones result on oversearching is one where the relationship between model size and search space resources is not a simple equivalence.

Conclusions

The results given in this paper are very general. There are many ways to strengthen these results if other assumptions are added. For example, the concept of refinement is a very powerful one that has been studied in machine learning [Laird 88], but not used to its full effect here. For example, refinement can be exploited as a way to arrive at a metric for the overfitting function F. An example is the use of the Vapnik-Chervonenkis dimension as a metric for the accuracy (inductive bias) [Haussler 88].

This paper employs Ramsey's theorem to show that overfitting is inevitable. The way that the theorem is applied is curious and somewhat unexpected. Ramsey's theorem is usually interpreted as if the r-coloring of the k-element subsets of $[n]$ refers to a model of resource size k fitting a data sequence of size n where r defines the types of relationships the models can have to the data. This is not the way that Ramsey's theorem was used here. Instead, r is a measure of the accuracy of the model. Ramsey's Theorem has been the inspiration for a rich branch of number theory that could be expected to yield significant results that apply to machine learning in general, not just in analyzing overfitting. It is useful because the results apply to the data being presented regardless of the particular nature of the learning problem, so they define some universal properties of data presentation that are unavoidable.

We have defined learning problems to be composed of cases that partition the set of data and of models, where the learning function finds a minimum of a function over a finite domain. Although this seems to be as general as learning can get, it is possible to generalize this paradigm in a number of ways.

For example, the cases do not have to be a partition of the problem. The association of models and data to a case could be probabilistic, for example. This might mean that the best G measure is found empirically, instead of being predefined. This means using machine learning to find best machine learning algorithm, creating a bootstrap effect of using metalearning to minimize overfitting through a whole family of related learning problems.

This paper has been an attempt to give a theoretical explanation for experimental results on overfitting. The intriguing results on oversearching provided the impetus to look in a critical manner at the basic definition of what it means to overfit data. It is hoped that a theoretical analysis of the overfitting and oversearching will serve to point the way to new experimental tests, such as the creation of a taxonomy of resources that can be used to

express model spaces or the development of general methods of defining criteria of overfitting that capture the varying notions of expressive accuracy.

References

Blum, Lenore; and Blum, Manuel. 1975. Toward a Mathematical Theory of Inductive Inference, *Information and Control* 28(2):125-155.

Case, John; and Smith, Carl H.. 1983. Comparison of Identification Criteria for Machine Inductive Inference. *Theoretical Computer Science* 25:193-220.

Graham, Ronald L. 1981. *Rudiments of Ramsey Theory.* Regional Conference Series in Mathematics; no 45, Providence: American Mathematical Society.

Haussler, David. 1988. Quantifying Inductive Bias: AI Learning Algorithms and Valiant's Learning Framework. *Artificial Intelligence* 36:177-221.

Kibler, Dennis; and Langley, Pat. 1988. Machine Learning as an Experimental Science. In Proceedings of the Third European Working Session on Learning. San Mateo: Morgan Kaufmann.

Laird, Philip D. 1988. *Learning From Good and Bad Data*, Norwell MA: Kluwer.

Mitchell, T. M. 1980. The Need for Biases in Learning Generalizations, Technical Report CBM-TR-117, Dept. of Computer Science, Rutgers Univ., New Brunswick, NJ.

Pfahringer, Bernhard. 1995. A New MDL Measure for Robust Rule Induction (Extended Abstract). In Proceedings 8th European Conference on Machine Learning: ECML-95, Heraclion, Crete. 331-334. Berlin: Springer-Verlag.

Quinlan, J. R. 1986. Induction of Decision Trees. *Machine Learning* 1:81-106.

Quinlan, J. R.; and Cameron-Jones, R. M. 1995. Oversearching and Layered Search in Empirical Learning. In Proceedings Fourteenth International Joint Conference on Artificial Intelligence, Montreal. 1019-1024.

Samuel, A. L. 1959. Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal* 3(3).

Van der Mude, A. 1986. Only the Complements of Recursively Enumerable Sets are Reliably Identified in the Limit, Technical Report CBM-TR-168, Dept. of Computer Science, Rutgers University, New Brunswick, NJ.