PAC-learning and Asymptotic System Identification Theory.¹

Lennart Ljung

Dept. of Electrical Engineering, Linköping University
S-581 83 Linköping, Sweden,
Email: ljung@isy.liu.se.

Abstract

In this paper we discuss PAC-learning of functions from a traditional System Identification perspective. The well established asymptotic theory for the identified models' properties is reviewed from the PAC-learning perspective. The role of finite-dimensional, smooth parametrizations over compact parameter sets is spelled out. This also sets some limits for the interest of identification-theory type results in a learning-theory context.

1 Introduction

Learning theory and PAC learning has received considerable interest, [7], [8], [6] and [9]. Many issues are closely related to questions in system identification, but the relationships have not been extensively explored. We shall in this contribution look into one particular question in PAC learning and discuss it from a traditional system identification perspective. We will see what the asymptotic theory can offer and what it cannot.

A very attractive feature with the learning formalism is that it focuses on the “identification issues” and leaves questions of parameterizations and parameters aside. While parameters in most cases are just vehicles to achieve a good mapping (“system description”), they do play a prominent role in traditional system identification theory. It is thus of considerable interest to investigate the fundamental limitations of the estimation problem in a parameter-free context. It is then another matter that most of the time it will be very important to find good parameterizations for practical reasons.

A disclaimer: This contribution – regrettably – is based on rather superficial knowledge of the learning literature. I will discuss one PAC-learning question, as de-

scribed in [9], ported to a system identification environment using traditional analysis tools. The results themselves are implicit or explicit in the identification literature. We shall also be rather imprecise with assumptions and somewhat careless with remainder terms etc. The purpose of this paper is not to provide any strict results, but rather to give a perspective on a learning problem.

2 A Basic Inference Problem

Here is a typical problem in science and human learning: We are shown a collection of vector pairs

$$\{[y(t), \varphi(t)]; t = 1, \dots, N\} \quad (1)$$

Call this “the training set”. We are then shown $\varphi(N+1)$ and asked to name a corresponding value for $y(N+1)$. The variable t could be thought of as time, but could be anything. The vectors $y(t)$ and $\varphi(t)$ may take values in any sets (finite sets or subsets of $\mathbb{R}^n$ or anything else) and the dimension of $\varphi(t)$ could very well depend on t (and be unbounded).

This formulation covers most kinds of classification and model building problems. Conceptually, the problem amounts to learning a certain “pattern” from (1) and then applying that pattern to new data. This probably resembles human learning and the scientific process in general terms.

3 System Identification/Non-linear Regression

Turning to the system identification arena, $y(t)$ would typically be the output of a system and $\varphi(t)$ would be made up from previous inputs and outputs to the system. The preferred way of “pairing” y and φ is to seek a mapping $\hat{g}$

$$y = \hat{g}(\varphi) \quad (2)$$

that connects the two variables. The search for this mapping is carried out in terms of a parameterized family of mappings $g(\varphi, \theta)$ – usually called *the model structure* – and the archetypical search mechanism is to use

¹This work was supported in part by the Swedish Research Council for Engineering Sciences (TFR), which is gratefully acknowledged.

the least squares criterion evaluated for the “training set”:

$$\hat{\theta}_N = \arg \min_{\theta} \sum_{t=1}^N |y(t) - g(\varphi(t), \theta)|^2 \quad (3)$$

This gives the mapping to be used for future pairing:

$$\hat{y}(t) = \hat{g}_N(\varphi(t)) = g(\varphi(t), \hat{\theta}_N) \quad (4)$$

the hat on y marking its difference from the true output.

The question then is how good is (4)? To analyze such problems it is customary to assume that there is a “true” function g_0 that does the correct pairing, possibly with some randomness, like in

$$y(t) = g_0(\varphi(t)) + e(t) \quad (5)$$

where $e(t)$ is an unpredictable sequence (“white noise”). Questions of quality of (4) then relate to the question of how small is

$$g_0(\varphi) - \hat{g}_N(\varphi) \quad (6)$$

measured in some suitable norm.

Note that the essence of system identification – that φ is made up from past inputs and outputs – really has nothing to do with this general picture. From the current perspective we could regard the problem as a general non-linear regression.

4 PAC-Learning

From [9] we quote the following setup for *Function Learning*: Suppose that $(X, \mathcal{S})$ is a measurable space, $\mathcal{P}$ is a family of probability measures on $(X, \mathcal{S})$, and that $\mathcal{G}$ is a family of functions mapping X into $[0, 1]$, each of which is measurable with respect to $\mathcal{S}$. By a slight abuse of notation, let us agree to use the symbol $[0, 1]^X$ to denote the set of all *measurable* functions mapping X into $[0, 1]$, as opposed to the set of *all* functions mapping X into $[0, 1]$, which is the normal meaning. Given two measurable functions $a, b : X \rightarrow [0, 1]$ and a probability measure P on $(X, \mathcal{S})$, one can define

$$d_P(a, b) = \int_X |a(x) - b(x)| P(dx). \quad (7)$$

This is a pseudometric on $[0, 1]^X$, and equals the expected value of the difference $|a(x) - b(x)|$.

In function learning, there is a fixed but unknown target function $g_0 \in \mathcal{G}$, and a probability measure $P \in \mathcal{P}$ that may or may not be known. In the present context, an “algorithm” is an indexed family of maps $\{A_m\}$ where

$$A_m : (X \times [0, 1])^m \rightarrow \mathcal{G}. \quad (8)$$

Learning proceeds as follows: i.i.d. samples $x_1, x_2, \dots, x_N \in X$ are drawn in accordance with P , and for each sample x_i , an oracle returns the value $g_0(x_i)$. Define

$$\begin{aligned} \hat{g}_N(g_0; x) &:= A_N\{[(x_1, g_0(x_1)), \dots, (x_N, g_0(x_N))]\}, \text{ and} \\ r(N, \epsilon) &:= \sup_{P \in \mathcal{P}, g_0 \in \mathcal{G}} P^N\{x \in X^N : d_P[g_0, \hat{g}_N(g_0; x)] > \epsilon\}. \end{aligned} \quad (9)$$

(10)

Definition The algorithm $\{A_N\}$ is **probably approximately correct (PAC) to accuracy ϵ** if $r(N, \epsilon) \rightarrow 0$ as $N \rightarrow \infty$, and is **probably approximately correct (PAC)** if $r(N, \epsilon) \rightarrow 0$ as $N \rightarrow \infty$ for each $\epsilon > 0$. The function class $\mathcal{G}$ is **PAC-learnable to accuracy ϵ with respect to the family $\mathcal{P}$** if there exists an algorithm that is PAC to accuracy ϵ , and is **PAC-learnable with respect to the family $\mathcal{P}$** if there exists an algorithm that is PAC.

The relationship between PAC-learning and evaluating (6) is obvious. Often PAC learning is defined for “concepts” or sets, which corresponds to using functions g that are indicator functions.

5 Smooth Parameterizations

Let us return to (3). While this criterion makes sense for any parameterization $g(\varphi, \theta)$ the common algorithms and the traditional asymptotic theory work with *finite dimensional, smooth parameterizations over compact parameter sets*. This means that $g(\varphi, \theta)$ is assumed to be continuously differentiable, with a well defined derivative

$$\psi(\varphi, \theta) = \frac{\partial}{\partial \theta} g(\varphi, \theta) \quad (11)$$

This means – with some simplification – that the asymptotic properties follow those of a linear regression parameterization:

$$g(\varphi, \theta) = \sum_{k=1}^d \theta_k \psi_k(\varphi) = \theta^T \psi(\varphi) \quad (12)$$

with ψ independent of θ . (Asymptotically, we can always linearize g around the limit estimate θ^* and think of the linear regression regressors $\psi(\varphi)$ as $\psi(\varphi, \theta^*)$.) Of course, the asymptotic theory for estimating linear regressions is very well developed; see e.g. [1].

We may also note that, basically, any smooth, globally identifiable parameterization can be rewritten as a linear regression, according to [3].

System identification basically employs only smooth parameterizations over compact sets. However, certain

functions do not fit this. An example is the threshold function with the step at an unknown value:

$$g_0(\varphi) = \text{sat}(\varphi - \alpha) = \begin{cases} 0, & \text{if } \varphi < \alpha, \\ 1, & \text{if } \varphi \geq \alpha. \end{cases} \quad (13)$$

This is a basic function in the *perceptron* and thus a fundamental element in learning and neural networks. While (13) can be embedded in a smooth parameterization – like the familiar sigmoid – this is at the expense of compactness: the limit (13) is obtained as the scale (dilation) parameter tends to infinity. All this means that traditional asymptotic identification theory, based on smoothness, does not have much to offer for function families like (13).

6 Asymptotic Identification theory

For a finite-dimensional, smooth parameterization of g over a compact parameter set, the asymptotic properties of the estimate $\hat{\theta}_N$, defined by (3) are well known. We assume for simplicity that the sequence $\varphi(t), t = 1, \dots$ used in the training data is an ergodic stochastic process with probability measure P_e . We also assume that the outputs y are generated from φ according to (5), where $e(t)$ is white noise with variance λ . First we have

$$\hat{\theta}_N \rightarrow \theta^* = \arg \min_{\theta} E_e |g_0(\varphi) - g(\varphi, \theta)|^2 \text{ w.p. 1. as } N \rightarrow \infty \quad (14)$$

Here E_e denotes expectation over $\varphi(t)$ under the probability measure P_e :

$$E_e |g_0(\varphi) - g(\varphi, \theta)|^2 = \int_{\Phi} |g_0(\varphi) - g(\varphi, \theta)|^2 P_e(d\varphi) \quad (15)$$

The second basic result concerns the convergence rate: Provided θ^* is such a good model that $g(\varphi(t), \theta^*) - g_0(\varphi(t))$ can be neglected in comparison with the noise term e , then

$$\sqrt{N}(\hat{\theta}_N - \theta^*) \in \text{AsN}(0, H_e) \quad (16)$$

which denotes that the random variable to the left converges in distribution to the normal distribution with zero mean and covariance matrix H_e , where

$$H_e = \lambda [E_e \psi(\varphi(t)) \psi^T(\varphi(t))]^{-1} \quad (17)$$

with ψ defined by (11). Under some regularity conditions, we also have

$$\lim_{N \rightarrow \infty} N E_e (\hat{\theta}_N - \theta^*) (\hat{\theta}_N - \theta^*)^T = H_e \quad (18)$$

All basic questions about the asymptotic model properties can be answered from these results. Let us for example consider $\hat{g}_N(\varphi) = g(\varphi, \hat{\theta}_N)$. For any fixed φ , this is a random variable, with asymptotic distribution

$$\sqrt{N}(\hat{g}_N(\varphi) - g^*(\varphi)) \in \text{AsN}(0, \psi^T(\varphi) H_e \psi(\varphi)) \quad (19)$$

where $g^*(\varphi) = g(\varphi, \theta^*)$. Let us now focus on the following measure of fit:

$$V_N = \int_{\Phi} |g_0(\varphi) - g(\varphi, \hat{\theta}_N)|^2 P_v(d\varphi) \quad (20)$$

This corresponds to $d_{P_v}(g_0, \hat{g}_N)$ in the notation of (7). In the identification context, think of it as the prediction error variance – in addition to λ – that the model $\hat{G}_N$ would give, when tested on (validation) data $\varphi(t)$, which are independent of the estimation data and have a probability measure P_v . Note that V_N is a random variable, due to its dependence on $\hat{\theta}_N$. From the asymptotic results above, we can derive its asymptotic properties. Let us concentrate on the “variance contribution” to V_N :

$$W_N = \int_{\Phi} |g^*(\varphi) - g(\varphi, \hat{\theta}_N)|^2 P_v(d\varphi) \quad (21)$$

Asymptotically,

$$W_N = (\theta^* - \hat{\theta}_N)^T R_v (\theta^* - \hat{\theta}_N) \\ R_v = E_v \psi(\varphi) \psi^T(\varphi)$$

where E_v denotes expectation with respect to the probability measure P_v . In view of (16)-(17) we thus know the distribution of W_N – it is “ χ^2 -like”. Notice that the distribution of W_N will depend on P_e (the probability measure under which the data were collected), P_v (the evaluation measure), and on the true function g_0 as well as of the model structure $g(\varphi, \theta)$ (since these determine ψ).

A *rather remarkable* (but well known) simplification occurs when the estimation and evaluation measures P_v and P_e coincide. Then $R_v = H_e^{-1}$ and $N \times W_N / \lambda$ becomes χ^2 distributed with d degrees of freedom, where $d = \dim(\theta)$. This also means that

$$E W_N = \lambda \frac{d}{N} \quad (22)$$

The distribution of the error W_N thus suddenly becomes independent both of $P_e = P_v$ and of the true function g_0 . Moreover it depends on the model structure only in terms of the number of employed parameters. All (asymptotic) results regarding the distribution of the model error norm W_N become uniform in $P_e = P_v$ and g_0 and will only depend on λ, d and N .

In particular, the PAC-measure $r(N, \epsilon)$ defined by (10) is just an $N\epsilon/\lambda$ -quantile of the $\chi^2(d)$ -distribution, for any function family $\mathcal{F}$ that can be smoothly parameterized by d parameters and any family of measures $\mathcal{P}$ that guarantee the ergodicity required for our results. These function classes/model structures are thus PAC-learnable even of the oracle returns answers with an arbitrary degree of white noise disturbances.

7 Non-smooth parameterizations

The results of the previous section are probably not very exciting from a learning perspective. Many concepts on learning, like Definition 1, are phrased for “noise-free data” (or oracles that give exact information). Function classes that can be smoothly parameterized by finitely many (say d) parameters over a compact set, are then trivially learnable: After having seen d correct function values, the correct values of these parameters can be computed, and the function is exactly known. (Well, to be correct, exactly known in the range space defined by the d training samples, but if these are “persistently exciting”, which is the generic case, this gives the complete function.)

Concept learning deals with indicator functions of the kind (13). While the criterion (3) still makes sense, it is clear that the location of the step is estimated only using the values of φ that are the (two in the scalar case) nearest neighbors to the step. The estimate will then not be formed by averaging many observations, which is the basis of the traditional asymptotic analysis. Also the conditions for the usual Cramer-Rao bound fail in this situation.

One can of course think of smoothly parameterized functions that well approximate the indicator function, like a sigmoid with a steep derivative. Then the asymptotic theory of the previous section can be applied. The crucial player $\psi(\varphi, \theta^*)$ (the derivative with respect to the location parameter) will then be a function with support only in a small region around the step. This confirms that the asymptotic properties of the estimate will depend only on the probability measure of φ very close to the step. As the step becomes more steep, ψ will become more delta-function like, and the variable H_e in (17) will tend to zero. The convergence rate of a location parameter estimate in a step function can thus be faster than the traditional $1/\sqrt{N}$.

8 Non-parametric methods/infinite dimensional parameterizations

For a real life mapping, it is of course restrictive to assume that it can be found in a finitely parameterized family of functions. We would then work with infinite function expansions like

$$g(\varphi, \theta) = \sum_{k=1}^{\infty} \theta_k g_k(\varphi, \theta) \quad (23)$$

If g_k are independent of θ we have a linear regression. Clearly, such function expansions give us considerable flexibility in describing even quite complicated mappings, including those, like (13), that cannot be smoothly parameterized with a finite amount of

bounded parameters. In practice, for any finite data set, we will of course have to truncate the expansion at some point. However, it is reasonable to let the number of used parameters, say d , be dependent on the number of data N : $d(N)$. According to the asymptotic theory, the variance contribution to the model error when $P_v = P_e$, and when the bias error is small compared to the noise e is

$$EW_N = \lambda \frac{d(N)}{N}$$

(See (22)). This gives the basic guideline for how to let $d(N)$ increase with N : Choose the number of parameters, so that EW_N balances the bias error $E|g_0(\varphi) - g^*(\varphi)|^2$. The result will depend on the noise level λ and how important the tail of the expansion (23) is to approximate the true mapping. This could be a rather complex decision, and we refer to [5] and [2] and to [4] in the linear identification case for a deeper discussion. Again, in the noise-free case $\lambda = 0$ this decision becomes trivial: For a smoothly parameterized structure, say a linear regression we may use $d = N$, and the issue of retrieving the true mapping is a function approximation problem – which of course is difficult enough.

9 Conclusions

The meeting between traditional asymptotic identification theory and PAC-learning, described in this contribution, can be characterized as polite, but not very productive nor exciting. The model structures in common use in identification are PAC-learnable (using e.g. the least squares criterion (3)), even when an arbitrary amount of white noise is corrupting the information. This does not come as a surprise, since they have been successfully used for a long time. For other model structures (function families $\mathcal{F}$), that cannot be finitely, smoothly parameterized over compact parameter sets, traditional identification theory does not seem to have much to offer.

I suppose that the really exciting question is whether learning theory can help us create a theory of system identification, void of parameters, and whether this would actually change the way we estimate our models.

References

- [1] N.R. Draper and H. Smith. *Applied Regression Analysis, 2nd ed.* Wiley, New York, 1981.
- [2] A. Juditsky, H. Hjalmarsson, A. Benveniste, B. Delyon, L. Ljung, J. Sjöberg, and Q. Zhang. Nonlinear black-box modeling in system identification: Mathematical foundations. *Automatica*, 31(12):1724–1750, 1995.

- [3] L. Ljung and T. Glad. *Modeling of Dynamic Systems*. Prentice Hall, Englewood Cliffs, 1994.
- [4] L. Ljung and Z.D. Yuan. Asymptotic properties of black-box identification of transfer functions. *IEEE Trans. Automatic Control*, AC-30:514–530, 1985.
- [5] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.Y. Glorennec, H. Hjalmarsson, and A. Juditsky. Nonlinear black-box modeling in system identification: A unified overview. *Automatica*, 31(12):1691–1724, 1995.
- [6] L.G. Valiant. A theory of the learnable. *Comm. ACM*, 27(11):1134–1142, 1984.
- [7] V.N. Vapnik. *Estimation of Dependences Based on Empirical Data*. Springer-Verlag, 1982.
- [8] V.N. Vapnik and A. Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theor. Probability Appl.*, 56:269–280, 1971.
- [9] M. Vidyasagar. *Learning, Generalization and Neural Networks*. To appear, 1996.