THE INFORMATIONAL COMPLEXITY OF LEARNING

Perspectives on Neural Networks
and Generative Grammar

THE INFORMATIONAL COMPLEXITY OF LEARNING

Perspectives on Neural Networks
and Generative Grammar

PARTHA NIYOGI

Massachusetts Institute of Technology
Cambridge, MA

Springer Science+Business Media, LLC

Library of Congress Cataloging-in-Publication Data

A C.I.P. Catalogue record for this book is available
from the Library of Congress.

ISBN 978-1-4613-7493-0 ISBN 978-1-4615-5459-2 (eBook)
DOI 10.1007/978-1-4615-5459-2

Copyright © 1998 Springer Science+Business Media New York
Originally published by Kluwer Academic Publishers in 1998
Softcover reprint of the hardcover 1st edition 1998

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher, Springer Science+Business Media, LLC.

Printed on acid-free paper.

Contents

List of Figures	ix
Foreword	xv
Preface	xix
Acknowledgments	xxi
1. INTRODUCTION	1
1.1 The Components of a Learning Paradigm	3
1.1.1 Concepts, Hypotheses, and Learners	3
1.1.2 Generalization, Learnability, Successful learning	6
1.1.3 Informational Complexity	7
1.2 Parametric Hypothesis Spaces	11
1.3 Technical Contents and Major Contributions	13
1.3.1 A Final Word	19
2. GENERALIZATION ERROR FOR NEURAL NETS	21
2.1 Introduction	21
2.2 Definitions and Statement of the Problem	24
2.2.1 Random Variables and Probability Distributions	24
2.2.2 Learning from Examples and Estimators	25
2.2.3 The Expected Risk and the Regression Function	26
2.2.4 The Empirical Risk	28
2.2.5 The Problem	28
2.2.6 Bounding the Generalization Error	30
2.2.7 A Note on Models and Model Complexity	33
2.3 Stating the Problem for Radial Basis Functions	34
2.4 Main Result	36
2.5 Remarks	36
2.5.1 Observations on the Main Result	36
2.5.2 Extensions	37
2.5.3 Connections with Other Results	39
2.6 Implications of the Theorem in Practice: Putting In the Numbers	40
2.6.1 Rate of Growth of n for Guaranteed Convergence	40
2.6.2 Optimal Choice of n	41
2.6.3 Experiments	45

2.7	Conclusion	50
2-A	Notations	50
2-B	A Useful Decomposition of the Expected Risk	56
2-C	A Useful Inequality	56
2-D	Proof of the Main Theorem	57
2-D.1	Bounding the approximation error	58
2-D.2	Bounding the estimation error	60
2-D.3	Bounding the generalization error	72
3.	ACTIVE LEARNING	75
3.1	A General Framework For Active Approximation	77
3.1.1	Preliminaries	77
3.1.2	The Problem of Collecting Examples	80
3.1.3	In Context	83
3.2	Example 1: A Class of Monotonically Increasing Bounded Functions	86
3.2.1	Lower Bound for Passive Learning	87
3.2.2	Active Learning Algorithms	88
3.2.2.1	Derivation of an optimal sampling strategy	88
3.2.3	Empirical Simulations, and other Investigations	94
3.2.3.1	Distribution of Points Selected	94
3.2.3.2	Classical Optimal Recovery	95
3.2.3.3	Error Rates and Sample Complexities for some Arbitrary Functions: Some Simulations	97
3.3	Example 2: A Class of Functions with Bounded First Derivative	100
3.3.1	Lower Bounds	102
3.3.2	Active Learning Algorithms	105
3.3.2.1	Derivation of an optimal sampling strategy	105
3.3.3	Some Simulations	110
3.3.3.1	Distribution of points selected	110
3.3.3.2	Error Rates:	113
3.4	Conclusions, Extensions, and Open Problems	115
3.5	A Simple Example	117
3.6	Generalizations	119
3.6.1	Localized Function Classes	119
3.6.2	The General ϵ -focusing strategy;	120
3.6.3	Generalizations and Open Problems	122
4.	LANGUAGE LEARNING	125
4.1	Language Learning and The Poverty of Stimulus	126
4.2	Constrained Grammars—Principles and Parameters	128
4.2.1	Example: A 3-parameter System from Syntax	129
4.2.2	Example: Parameterized Metrical Stress in Phonology	132
4.3	Learning in the Principles and Parameters Framework	134
4.4	Formal Analysis of the Triggering Learning Algorithm	137
4.4.1	Background	138
4.4.2	The Markov formulation	139
4.4.2.1	Parameterized Grammars and their Corresponding Markov Chains	139

4.4.2.2	Markov Chain Criteria for Learnability	140
4.4.2.3	The Markov chain for the 3-parameter Example	143
4.4.3	Derivation of the transition probabilities for the Markov TLA structure	145
4.4.3.1	Formalization	145
4.4.3.2	Additional Properties of the Learning System	147
4.5	Characterizing Convergence Times for the Markov Chain Model	148
4.5.1	Some Transition Matrices and Their Convergence Curves	148
4.5.2	Absorption Times	152
4.5.3	Eigenvalue Rates of Convergence	153
4.5.3.1	Eigenvalues and Eigenvectors	153
4.5.3.2	Representation of T^k	154
4.5.3.3	Initial Conditions and Limiting Distributions	155
4.5.3.4	Rate of Convergence	156
4.5.3.5	Transition Matrix Recipes:	156
4.6	Exploring Other Points	157
4.6.1	Changing the Algorithm	157
4.6.2	Distributional Assumptions	159
4.6.3	Natural Distributions—CHILDES CORPUS	160
4.7	Batch Learning Upper and Lower Bounds: An Aside	162
4.8	Conclusions, Open Questions, and Future Directions	164
4-A	Unembedded Sentences For Parametric Grammars	167
4-B	Memoryless Algorithms and Markov Chains	167
4-C	Proof of Learnability Theorem	168
4-C.1	Markov state terminology	168
4-C.2	Canonical Decomposition	169
4-D	Formal Proof	170
5.	LANGUAGE CHANGE	173
5.1	Introduction	173
5.2	Language Change in Parametric Systems	181
5.3	Example 1: A Three Parameter System	182
5.3.1	Starting with Homogeneous Populations:	183
5.3.1.1	$\mathcal{A} = \text{TLA}; P_i = \text{Uniform}; \text{Finite Sample} = 128$	183
5.3.1.2	$\mathcal{A} = \text{Greedy, No S.V.}; P_i = \text{Uniform}; \text{Finite Sample} = 128$	186
5.3.1.3	$\mathcal{A} = \text{a) R.W. b) S. V. only}; P_i = \text{Uniform}; \text{Finite Sample} = 128$	187
5.3.1.4	Rates of Change	188
5.3.2	Non-homogeneous Populations: Phase-Space Plots	192
5.3.2.1	Phase-Space Plots: Grammatical Trajectories	193
5.3.2.2	Issues of Stability	194
5.4	Example 2: The Case of Modern French:	196
5.4.1	The Parametric Subspace and Data	197
5.4.2	The Case of Diachronic Syntax Change in French	198
5.4.3	Some Dynamical System Simulations	199
5.4.3.1	Homogeneous Populations [Initial–Old French]	199
5.4.3.2	Heterogeneous Populations (Mixtures)	201
5.5	Conclusions	203

6. CONCLUSIONS	207
6.1 Emergent Themes	208
6.2 Extensions	210
6.3 A Concluding Note	212
References	213

List of Figures

1.1	The space of possibilities. The various factors which affect the informational complexity of learning from examples.	10
1.2	The structure of a Hyper Basis Function Network (same as regularization network).	12
1.3	Parametric difference in phrase structure between English and Bengali on the basis of the parameter p_2 .	14
1.4	Analysis of the English sentence “with one hand” according to its parameterized X -bar grammar.	15
1.5	Analysis of the Bengali sentence “ek haath diye” a literal translation of “with one hand” according to its parameterized X -bar grammar. Notice the difference in word order.	16
2.1	This figure shows a picture of the problem. The outermost circle represents the set F . Embedded in this are the nested subsets, the H_n ’s. f_0 is an arbitrary target function in $\mathcal{F}$, f_n is the closest element of H_n and $\hat{f}_{n,l}$ is the element of H_n which the learner hypothesizes on the basis of data.	33
2.2	Bound on the generalization error as a function of the number of basis functions n keeping the sample size l fixed. This has been plotted for a few different choices of sample size. Notice how the generalization error goes through a minimum for a certain value of n . This would be an appropriate choice for the given (constant) data complexity. Note also that the minimum is broader for larger l , that is, an accurate choice of n is less critical when plenty of data is available.	42
2.3	The bound on the generalization error as a function of the number of examples for different choices of the rate at which network size n increases with sample size l . Notice that if $n = l$, then the estimator is not guaranteed to converge, i.e., the bound on the generalization error diverges. While this is a distribution free-upper bound, we need distribution-free lower bounds as well to make the stronger claim that $n = l$ will never converge.	44

- 2.4 This figures shows various choices of (l, n) which give the same generalization error. The x -axis has been plotted on a log scale. The interesting observation is that there are an infinite number of choices for number of basis functions and number of data points all of which would guarantee the same generalization error (in terms of its worst case bound). 45
- 2.5 The generalization error as a function of number of examples keeping the number of basis functions (n) fixed. This has been done for several choices of n . As the number of examples increases to infinity the generalization error asymptotes to a minimum which is not the Bayes error rate because of finite hypothesis complexity (finite n). 46
- 2.6 The generalization error, the number of examples (l) and the number of basis functions (n) as a function of each other. 47
- 2.7 The generalization error is plotted as a function of the number of nodes of an RBF network (10) trained on 100 data points of a function of the type (16) in 2 dimensions. For each number of parameters 10 results, corresponding to 10 different local minima, are reported. The continuous lines above the experimental data represents the bound $\frac{a}{n} + b[(nk \ln(nl) - \ln \delta)/l]^{1/2}$ of eq. (14), in which the parameters a and b have been estimated empirically, and $\delta = 10^{-6}$. 48
- 2.8 Everything is as in figure (6), but here the dimensionality is 6 and the number of data points is 150. As before, the parameters a and b have been estimated empirically and $\delta = 10^{-6}$. Notice that this time the curve passes through some of the data points. However, we recall that the bound indicated by the curve holds under the assumption that the *global* minimum has been found, and that the data points represent different *local* minima. Clearly in the figure the curve bounds the best of the local minima. 49
- 2.9 If the distance between $I[f_n]$ and $I[\hat{f}_{n,l}]$ is larger than 2ϵ , the condition $I_{\text{emp}}[\hat{f}_{n,l}] \leq I_{\text{emp}}[f_n]$ is violated. 57
- 3.1 An arbitrary data set fitted with cubic splines 78
- 3.2 A depiction of the situation for an arbitrary data set. The set $\mathcal{F}_{\mathcal{D}}$ consists of all functions lying in the boxes and passing through the datapoints (for example, the dotted lines). The approximating function h is a linear interpolant shown by a solid line. 89
- 3.3 Zoomed version of interval. The maximum error the approximation scheme could have is indicated by the shaded region. This happens when the adversary claims the target function had the value y_i throughout the interval. 89

3.4	The situation when the interval C_i is sampled yielding a new data point. This subdivides the interval into two subintervals and the two shaded boxes indicate the new constraints on the function.	91
3.5	How the CLA chooses its examples. Vertical lines have been drawn to mark the x -coordinates of the points at which the algorithm asks for the value of the function.	95
3.6	The dotted line shows the density of the samples along the x -axis when the target was the monotone-function of the previous example. The bold line is a plot of the derivative of the function. Notice the correlation between the two.	96
3.7	The situation when a function $f \in \mathcal{F}$ is picked, n sample points (the x 's) are chosen and the corresponding y values are obtained. Each choice of sample points corresponds to a choice of the a 's. Each choice of a function corresponds to a choice of the b 's.	97
3.8	Error rates as a function of the number of examples for the arbitrary monotone function shown in a previous figure.	98
3.9	Four other monotonic functions on which simulations have been run comparing random, uniform, and active sampling strategies.	99
3.10	This figure plots the log of the error (L_1 error) against N the number of examples for each of the 4 monotonic functions shown in fig.	101
3.11	Construction of a function satisfying Lemma 2.	103
3.12	An arbitrary data set for the case of functions with a bounded derivative. The functions in $\mathcal{F}_{\mathcal{D}}$ are constrained to lie in the parallelograms as shown. The slopes of the lines making up the parallelogram are d and $-d$ appropriately.	105
3.13	A zoomed version of the i th interval.	106
3.14	Subdivision of the i th interval when a new data point is obtained.	107
3.15	A figure to help the visualization of Lemma 4. For the x shown, the set $\mathcal{F}_{\mathcal{D}}$ is the set of all values which lie within the parallelogram corresponding to this x , i.e., on the vertical line drawn at x but within the parallelogram.	108
3.16	Four functions with bounded derivative considered in the simulations. The uniform bound on the derivative was chosen to be $d = 10$.	111
3.17	How CLA-2 chooses to sample its points. Vertical lines have been drawn at the x values where the CLA queried the oracle for the corresponding function value.	112
3.18	How CLA-2 chooses to sample its points. The solid line is a plot of $ f'(x) $ where f is Function-1 of our simulation set. The dotted line shows the density of sample points (queried by CLA-2) on the domain.	113

3.19	Results of Simulation B. Notice how the sampling strategy of the active learner causes better approximation (lower rates) for the same number of examples.	114
3.20	Variation with epsilons.	115
4.1	Analysis of an English sentence. The parameter settings for English are spec-first, and comp-final.	130
4.2	analysis of the Bengali translation of the English sentence of the earlier figure. The parameter settings for Bengali are spec-first, and comp-first.	131
4.3	Depiction of stress pattern assignment to words of different syllable length under the parameterized bracketing scheme described in the text.	134
4.4	The space of possible learning problems associated with parameterized linguistic theories. Each axis represents an important dimension along which specific learning problems might differ. Each point in this space specifies a particular learning problem. The entire space represents a class of learning problems which are interesting.	136
4.5	The 8 parameter settings in the GW example, shown as a Markov structure. Directed arrows between circles (states, parameter settings, grammars) represent possible nonzero (possible learner) transitions. The target grammar (in this case, number 5, setting [0 1 0]), lies at dead center. Around it are the three settings that differ from the target by exactly one binary digit; surrounding those are the 3 hypotheses two binary digits away from the target; the third ring out contains the single hypothesis that differs from the target by 3 binary digits. Note that the learner can either stay in the same state or step in or out one ring (binary digit) at a time, according to the single-step learning hypothesis; but some transitions are not possible because there is no data to drive the learner from one state to the other under the TLA. Numbers on the arcs denote transition probabilities between grammar states; these values are not computed by the original GW algorithm. The next section shows how to compute these values, essentially by taking language set intersections.	144
4.6	Convergence as function of number of examples. The horizontal axis denotes the number of examples received and the vertical axis represents the probability of converging to the target state. The data from the target is assumed to be distributed uniformly over degree-0 sentences. The solid line represents TLA convergence times and the dotted line is a random walk learning algorithm (RWA). Note that random walk actually converges <i>faster</i> than the TLA in this case.	151

- 4.7 Convergence rates for different learning algorithms when L_1 is the target language. The curve with the slowest rate (large dashes) represents the TLA. The curve with the fastest rate (small dashes) is the Random Walk (RWA) with no greediness or single value constraints. Random walks with exactly one of the greediness and single value constraints have performances in between these two and are very close to each other. 158
- 4.8 Rates of convergence for TLA with L_1 as the target language for different distributions. The y -axis plots the probability of converging to the target after m samples and the x -axis is on a log scale, i.e., it shows $\log(m)$ as m varies. The solid line denotes the choice of an “unfavorable” distribution characterized by $a = 0.9999$; $b = c = d = 0.000001$. The dotted line denotes the choice of $a = 0.99$; $b = c = d = 0.0001$ and the dashed line is the convergence curve for a uniform distribution, the same curve as plotted in the earlier figure. 161
- 5.1 A simple illustration of the state space for the 3-parameter syntactic case. There are 8 grammars, a probability distribution on these 8 grammars, as shown above, can be interpreted as the linguistic composition of the population. Thus, a fraction P_1 of the population have internalized grammar, g_1 , and so on. 179
- 5.2 Percentage of the population speaking languages L_1 and L_2 as it evolves over the number of generations. The plot has been shown only upto 20 generations, as the proportions of L_1 and L_2 speakers do not vary significantly thereafter. Notice the “S” shaped nature of the curve (Kroch, 1989, imposes such a shape using models from population biology, while we obtain this as an emergent property of our dynamical model from different starting assumptions). Also notice the region of maximum change as the V2 parameter is slowly set by increasing proportion of the population. L_1 and L_2 differ only in the V2 parameter setting. 185
- 5.3 Percentage of the population speaking languages L_5 and L_2 as it evolves over the number of generations. Notice how the shift occurs over a space of 4 generations. 186
- 5.4 Time evolution of grammars using greedy algorithm with no single value. 188

- 5.5 Time evolution of linguistic composition for the situations where the learning algorithm used is the TLA (with greediness dropped, corresponding to the dotted line), and the Random Walk (solid line). Only the percentage of people speaking L_1 (-V2) and L_2 (+V2) are shown. The initial population is homogeneous and speaks L_1 . The percentage of L_1 speakers gradually decreases to about 11 percent. The percentage of L_2 speakers rises to about 16 percent from 0 percent. The two dynamical systems (corresponding to S.V. and R.W.) converge to the same population mix. However, the trajectory is not the same—the rates of change are different, as shown in this plot. 189
- 5.6 Time evolution of linguistic composition for the situations where the learning algorithm used is the TLA (with single-value dropped). Only the percentage of people speaking L_2 (+V2) is shown. The initial population is homogeneous and speaks L_1 . The maturational time (number, N , of sentences the child hears before internalizing a grammar) is varied through 8, 16, 32, 64, 128, 256, giving rise to the six curves shown in the figure. The curve which has the highest initial rate of change corresponds to the situation where 8 examples were allowed to the learner to develop its mature hypothesis. The initial rate of change decreases as the maturation time N increases. The value at which these curves asymptote also seems to vary with the maturation time, and increases monotonically with it. 191
- 5.7 The evolution of L_2 speakers in the community for various values of p (a parameter related to the sentence distributions P_i , see text). The algorithm used was the TLA, the initial population was homogeneous, speaking only L_1 . The curves for $p = 0.05, 0.75$, and 0.95 have been plotted as solid lines. 193
- 5.8 Subspace of a Phase-space plot. The plot shows $(\pi_1(t), \pi_2(t))$ as t varies, i.e., the proportion of speakers speaking languages L_1 and L_2 in the population. The initial state of the population was homogeneous (speaking language L_1). The algorithm used was the TLA with the single-value constraint dropped. 194
- 5.9 Subspace of a Phase-space plot. The plot shows $(\pi_1(t), \pi_2(t))$ as t varies for different initial conditions (non-homogeneous populations). The algorithm used by the learner is the TLA with single-value constraint dropped. 195
- 5.10 Evolution of speakers of different languages in a population starting off with speakers only of Old French. 200
- 5.11 Tendency to lose V2 as a result of new word orders introduced by Modern French source in our Markov Model. 202

From Talmudic times to today, whenever people have pondered the nature of intelligence, two topics arise again and again: learning and language. These two abilities perhaps constitute the very essence of what it means to be human. As Chomsky notes, these two abilities can also be cast generally as a puzzle dubbed “Plato’s Problem”: How do we come to know so much about the world, given that we are provided so little information about it? — what modern linguists and psychologists call “the poverty of the stimulus.” So for example, children come into this world not knowing whether they will be born in Beijing or New Delhi, yet, on the briefest exposure to the local languages — literally, perhaps, just hundreds of example sentences — they come into full possession of “knowledge” of Chinese or Hindi. How is this possible?

In this book, Partha Niyogi provides a modern solution to Plato’s problem — one of the first formal, computational answers, yielding fresh insights into both *how* learning works (Part 1) and *how* human language is learned (Part 2). The results are important not just in themselves — in the field known as computational learning theory — but range well beyond, to questions particular about human language learning and how many sentences it takes to learn German or Japanese; what neural nets can and cannot do depending on their size; and, more generally, the trade-off between the complexity of one’s theory and the data required by a machine (or a child) to learn it.

Such generality does not come as a surprise: An answer to Plato’s problem presupposes not only that we understand *what* we know about the world, but also *how much* data we need to learn it. This is Niyogi’s contribution: constructing a theory of the *informational complexity* of learning. He sharpens Plato’s question by introducing the modern armamentarium of computational learning theory — itself building on the insights of computational complexity, and the work of Solomonoff, Chaitin, Kolmogorov, and Vapnik: How can we come to know so much about the world *given that we have so few data samples, time and computational energy*?

In Part 1 of this book Niyogi tackles this question from an abstract mathematical viewpoint, laying the foundations for an application to language learning in Part 2. Just as it was impossible before the advent of recursive function

theory to even properly talk about how language could “make infinite use of finite means”, there’s a strong sense in which only recently have we found the right tools to talk about the informational complexity of learning. One might mark the beginning of the modern era from Kolmogorov, Chaitin, and Solomonoff’s work on so-called “program size” complexity — how many lines of code does it take to compute some algorithm — but can be more precisely pinpointed within the last 10–15 years, with the rise of Valiant’s computational learning theory, and, especially, Vapnik’s earlier and far-reaching generalization of that approach. As Vapnik notes, this paradigm shift itself can be regarded from a statistical point of view and efforts to overcome the limitations of Fisher’s classical statistical methods from the 1930s. Fisher too engaged in attacking Plato’s problem: for Fisher, a central question was how to find the “best” function fitting some set of data — and how many data samples it would take — given that one knew *almost everything* about the function, up to a very few parameters, like its mean, standard deviation, and so forth — what Fisher dubbed discriminant analysis.

Vapnik — and Niyogi in this book — goes far beyond Fisher’s classical paradigm to answer the question in modern computational dress: how many data examples will it take to find the “best” function to approximate a target function, given that we know *almost nothing* about the function, except general properties of the model class from which it is drawn — for instance, that they are all straight lines. This too is a learning problem, indeed, a learning problem *par excellence*. To address it, Niyogi focuses on the thorny issue that Fisher, and every learning theorist since, has wrestled with: the so-called *bias-variance* dilemma, the familiar problem of balancing the number of variables one uses to fit data versus the amount of data one needs to properly estimate all those parameters. If one picks too narrow a class of models, then one runs the risk of not getting to the right answer at all — this is one type of error, *bias*. So, a common response to modeling the world — be it interest rates, speech, or the weather — is simply to dump more parameters into one’s theory, add more variables to an econometric model, or add more nodes to a neural net. But then one runs a second, just as dangerous risk, and encounters a second sort of error: so many variables that it takes too much data to estimate them all with some degree of confidence, *variance*.

Put another way, there are *two* important sources of learning error; recognition and proper treatment of this fact is a key difference that separates Fisher’s past from Niyogi’s present. First, because data is limited, one might not pick the best function — a familiar kind of estimation error. But second, even if we pick the best function, it might not be the correct one, because the class of functions we use might not let us get close to the right answer. If we are trying to learn Chinese, then we could go wrong in two ways: we might not get enough Chinese example sentences, or the class of languages we try to use to approximate Chinese might include only Japanese and German.

Is there a way to resolve this dilemma, and avoid being gored by the variance/bias horns? Niyogi shows how. Part 1 develops an explicit mathematical (and computational) way to trade-off extra model complexity against the num-

ber of examples one needs to learn a model. Often there's an optimal balance between the two — like the fairy-tale about Goldilocks and the Three Bears, a model that's neither too big, nor too small, but “just right.” Without the tools that Niyogi has provided, we might never understand how to balance model complexity against sample complexity. Importantly, though Niyogi uses a *particular* model class, radial basis functions (RBFs) to “learn” functions, the details are not crucial to his results. They apply more generally to a large class of approximation methods — including multilayer perceptrons. Niyogi's findings (obtained in collaboration with F. Girosi) should therefore be read by those in the neural net community, more generally, by any who build learning machines.

In Part 2, Niyogi turns his conceptual insights about sample complexity to the specific case of language learning — demonstrating the generality of his approach. Here, the class of models to learn are possible grammars (languages), parameterized along one of a small number of (discontinuous) dimensions: English sentences come along in the form verb-object, as in *ate an apple*, while Japanese sentences come in the form object-verb, as in “apple ate” (*ringo tabeta*). We may imagine a single binary “switch” setting this range of variation. By combining twenty or so such spaces, we can fill the space of possible syntactic variation in human language — this is the so-called “principles and parameters” approach recently developed by Chomsky. On current linguistic accounts, this picture simplifies learning, because a child can simply decide how to flip its language switches on the basis of the sentences it hears. But does this really help? Niyogi again shows that one requires a formal, informational complexity analysis. The language space can be modeled as a Markov chain, with the states denoting various parameter combinations. Importantly, one inherits the results known for Markov chains, where transitions between states model the child moving from language hypothesis to hypothesis. In particular, one can analytically determine when a language learning system of this type is learnable, but, more importantly, whether it is *feasibly* learnable — whether it will take 100 or 1 million examples to learn. In this way, Niyogi has added an important new criterion to that of learnability *tout court*, and so a new constraint on linguistic theories. Not only should (human) languages be learnable, they should be learnable from just a few examples, as seems to be the case. Niyogi's formalization makes explicit predictions about this informational complexity, and shows that, indeed, some otherwise reasonable linguistic parameterizations fail this new learnability litmus test. In this sense, the mathematical theory developed in Part 1 does real work for us: by adding the criterion of sample complexity, we add a new tool for empirical investigation into the nature of language, and so a way to answer concretely Plato's problem. One could not ask for more.

Robert C. Berwick
Tomaso Poggio
Massachusetts Institute of Technology

Preface

In many ways, the ability to learn is recognized as a critical, almost central component of intelligence. This monograph is a treatment of the problem of learning from examples. As with any scientific enterprise, we have a certain point of view. First, we limit ourselves to inductive learning where the learner has access to finite data and has to generalize to the infinite set. We ignore, therefore, transductive or deductive forms of learning. The investigations are in a statistical setting with a primary focus on the informational complexity of learning.

In any work of this sort, it is necessary to make certain choices. A natural one perhaps is the adoption of the statistical theory of learning as a framework for analysis. This allows us to pose basic questions of learnability and informational complexity that potentially cut across different domains. Less obvious choices are the domains that we have chosen to investigate — function learning with neural networks and language learning with generative grammars. They present an interesting contrast. Neural networks are real-valued, infinite-dimensional, continuous mappings. Grammars are boolean-valued, finite-dimensional, discrete mappings. Furthermore, researchers in neural computation and generative linguistics rarely communicate. They typically go to different conferences, publish in different journals, and it is often believed that the theoretical underpinnings and technical contents of the two fields are fundamentally at odds with each other.

This monograph is an attempt to bridge the gap. By asking the same question — how much information do we need to learn — of both kinds of learning problems, we highlight their similarities and differences. Function learning requires neural networks with bounded capacity (finite VC dimension); language learning requires grammars with constrained rules (universal grammar). There is a learnability argument at the heart of the modern approach to linguistics, and fundamentally, as we shall see, parameter setting in the principles and parameters framework of modern linguistic theory is little different from parameter estimation in neural networks.

The structure of the monograph is as follows. In chapter 1, the basic framework for analysis is developed. This is a non-technical and accessible chapter where the problem of learning from examples is discussed, the notion of informational complexity is introduced and intuitions about it are developed. Thereafter, we have four very technical chapters, each dealing with a specific learning problem.

Chapter 2 considers the problem of learning functions in a Sobolev space using radial basis function networks (a kind of neural network). It is shown

how the generalization error (a measure of how well the learner has learned) depends upon the size of the network and the number of examples that the learner has access to. Formal bounds on generalization error are developed. Chapter 3 considers the situation where the learner is no longer a passive recipient of data but can actively select data of its own choosing. The effect of this on the informational complexity of learning is studied for monotonic functions and functions with a bounded derivative. Chapter 4 considers language learning in the principles and parameters framework of Chomsky. It is shown how language acquisition reduces to parameter estimation and the number of examples needed to estimate such parameters is calculated. Chapter 5 considers a population of learners each of whom is trying to attain a target grammar. A model of language change is derived. It is shown how such a model characterizes the evolutionary consequences of language learning. By comparing with historically observed language change phenomena, it allows us to pose an evolutionary criterion for the adequacy of learning theories. This is the first formal model of language change and its derivation rests crucially on an analysis of the informational complexity of language learning developed in the previous chapter. I have deliberately included an abstract with each chapter to allow the reader the possibility of skipping the details if he or she wishes to read the chapters non-sequentially.

As with all interdisciplinary pieces of work, I have to acknowledge an intellectual debt to many fields. The areas of approximation theory and statistics, particularly the part of empirical process theory beautifully worked out by Vapnik and Chervonenkis, play an important role in chapter 2. Ideas from adaptive integration and numerical analysis play an important role in chapter 3. Chapters 4 and 5 have evolved from an application of the computational perspective to the analysis of learning paradigms that are considered worthwhile in linguistic theory. My decision of what is linguistically worthwhile has been greatly influenced by scholarly works in the Chomskyan tradition. Here, there is some use of Markov chain theory and dynamical systems theory.

In all of this, I have brought to bear well known results and techniques from different areas of mathematics to formally pose and answer questions of interest in human and machine learning; questions previous unposed or unanswered or both. In this strict sense, there is little new mathematics here; though an abundant demonstration of its usefulness as a research tool to gain insight in the cognitive or computer sciences. This reflects my purpose and intended audience for this book — all people interested in computational aspects of human or machine learning and its interaction with natural language

Partha Niyogi
Hoboken, NJ

Acknowledgments

This book arose out of a doctoral dissertation submitted to the Electrical Engineering and Computer Science department at MIT. Thanks go first and foremost to my thesis committee. Tommy Poggio supported me throughout, provided the kind of intellectual freedom that is rare in these times and constantly reassured me of the usefulness of theoretical analyses. Bob Berwick introduced me to linguistic theory, widened the scope of this work considerably, and has been a friend at all times. Vladimir Vapnik followed the learning-theoretic part of this work quite closely. We have had several long discussions and his advice, encouragement and influence on this work is profound. Ron Rivest first taught me formal machine learning and was wonderfully supportive throughout. I cannot thank Federico Girosi enough. He has spent countless selfless hours discussing a variety of subjects and has always helped me retain my perspective.

Numerous people, at MIT and elsewhere, have touched my life in various ways. I especially wish to thank Sanjoy Mitter, Victor Zue, Ken Stevens, Morris Halle, Patrick Winston, David Lightfoot, Amy Weinberg, Noam Chomsky, and Kah Kay Sung.

All of this research was conducted at the Artificial Intelligence Lab and the Center for Biological and Computational Learning at MIT. This was sponsored by a grant from the National Science Foundation under contract ASC-9217041 (this award includes funds from ARPA provided under the HPCC program); and by a grant from ARPA/ONR under contract N00014-92-J-1879. Additional support has been provided by Siemens Corporate Research Inc., Mitsubishi Electric Corporation and Sumitomo Metal Industries. Support for the A.I. Laboratory's artificial intelligence research was provided by ARPA contract N00014-91-J-4038.

To my family
and
to Parvati Krishnamurty