Gödel numbering. I think an account of the theory of computation using the λ calculus as the primary model would be much more interesting.

Nevertheless, the book does succeed in pulling together a collection of topics that are *relevant* to computer science. The long-standing canonical reference[¶] does not include the topics of Chapters 8 and 9, for instance, and is generally less concerned with implementation issues (reduction machines, lazy evaluation, and explicit substitutions). With the help of a motivated instructor, this book could be the primary text for an effective course on the λ calculus.

Review of[‖]

Systems that Learn (second edition)
Authors: Jain, Osherson, Royer, Sharma
Publisher: The MIT Press, 1999
Hardcover: ISBN 0-262-100770
317 pages

Reviewer: Carl Smith

While this new edition shares much in common with the first edition, it is hardly fair to call it the "second edition." Firstly, only one of the four authors was among the triumvirate that produced the first edition. Secondly, although the new *Systems that Learn* is 50% larger than the first, it is not just an augmentation of more recent material. For sure, new material has been added, but more importantly, there has been a serious reorganization reflecting the codification of the fundamental core results in the intervening years. Furthermore, while the first edition focused primarily on language, as opposed to function, learning, the recent edition presents a balanced treatment of both threads. Like the first edition, there are numerous exercises, many of which point to extension of the material covered, making the book suitable for use as a graduate course text.

The obligatory introduction lays the groundwork for the technicalities to follow by formalizing somewhat the learning process. The first (of three) parts is rounded out by an in depth discussion of how to treat data, hypotheses and scientists as formal processes and some of the fundamental ramifications of doing so. Making these notions computable also received serious consideration here. The learning of languages and functions are considered in tandem. The presentation is coherent throughout, laying out the basic principles and paradigms considered in the rest of the book.

The second part of the book concerns various fundamental generalizations of the basic paradigm of learning. The discussion starts with various strategies for learning. These strategies revolve on around constraining the learning process in some way. For example, demanding that every hypothesis is consistent with all of the observed data produces an initially surprising constriction of what is learnable. The discussion of various constraints leads naturally to considering various criteria of learning. There are several notions of what it might mean for a scientist (modeled as an algorithm) to succeed at learning some function or language, given suitable examples. For example, does the learning happen only in the limit, or perhaps, after a fixed number of conjectures?

Another historically important issue that ramifies as an identification criteria is the proximity of the produced answer to the desired result. From the earliest days of the learning theory it was realized that science proceeds quite nicely utilizing only partially correct theories. The same phenomenon is also abundantly clear in human activities as well. Consequently, there has been much research on learning criteria where the learner is only required to come up with an "approximation" of the desired result. Different notions of "approximation" give rise to different identification criteria, most of which explained and discussed in the volume under review.

[¶]Barendregt, H.P. **The Lambda Calculus: Its Syntax and Semantics.** Elsevier: Amsterdam, 1984

The second part of the book is rounded out with a discussion of of various data environments. The early formal studies of learning assumed that the data arrived unstructured and unmolested. More recent studies reported on consider models of inaccurate data and data with some extra structure, as is the case when the data is delived in some regular order.

The third and final part of the book does what any book attempting to survey a research area must do, that is, consider the additional topics that seem to be important but have not yet been integrated into the fundamental core knowledge of the area. As is the nature of such adventures, there are some omissions, and the authors cannot be faulted for them. An entire chapter is devoted to team and probabilistic learning. Within the formal study of learning, it is rare when two different criteria turn out to capture the same collection of learnable phenomena. The deep relationships between learning by teams of algorithms and learning via probabilistic learning algorithms continues to stimulate much research. The presentation here is a excellent condensation of this line of inquiry.

The difficulty of learning persuaded many researchers to consider learning with some sort of additional information. Of course, there are a variety of ways to formalize the notion of "extra information." Among the techniques discussed are using bounds on the size of the answer sought and use of a known good approximation. A separate chapter is devoted to the use of an oracle during learning. Oracle learning gives the learning algorithm an arbitrary consultant to query. The results here focus on how the expressive power of the oracle trades off against the class of phenomena that is learnable.

Defining the complexity of learning has always been on onerous problem due to the fact the complexity of the learning algorithm, in the traditional complexity of algorithms sense, is much different from the intuitive notion of learning complexity. Many of the various and sundry notions of the complexity of learning are discussed in a chapter of the additional topics section.

The book concludes with a modern discussion of the enumeration technique. This technique dates to the earliest days of formal learning theory. The basic idea is to start with a list of possible answers, examine data to eliminate them one at a time, finally arriving at a potential answer that cannot be eliminated by the data. This final answer must be correct, since it is a program that agrees with all the data. A haunting question has been "Is this all there is to learning, the rest being just the complexity of the process?" The key to understanding the issues raised turned out to be one of transformations. The same issue seems to be at the core of the pedagogical technique of giving examples. If it is the case that some phenomenon can be learned, and all transformations of it can also be learned, then we say the original phenomenon can be learned *robustly*. In the classroom, we all hope that our students will learn from our examples to apply the techniques learned in new situations later in their lives. Are we teaching them robustly? This research area is fraught with difficulty and has been a catalyst to many elegant results. The third edition of *Systems that Learn* will undoubtedly contain more material on this topic.

Carl H. Smith
College Park, MD
November 4, 1999