Identifying Languages from Stochastic Examples

Dana Angluin
(but presented by Jeff Heinz)
heinz@udel.edu

University of Delaware

SIG Theory
April 22, 2009

Why We are Here

> Turning to theoretical results on learning, it seems that statistical learners may be more powerful than non-statistical learners. For example, while Gold's famous results showed that neither finite state nor context free languages can be learnt from positive examples alone (Gold, 1967), it turns out that *probabilistic* context free languages can be learnt from positive examples alone (Horning, 1969). [emphasis in original]
>
> (Johnson and Riezler *Cognitive Science* 26:3, 2002)

Their Explanation

> Informally, a class of languages may be statistically learnable even though its categorical counterpart is not because the statistical learning framework makes stronger assumptions about the training data (i.e. is it is distributed according to some probabilistic grammar from the class) and accepts a weaker criterion for successful learning (convergence in probability).
>
> (Johnson and Riezler *Cognitive Science* 26:3, 2002)

This talk

- What are these stronger assumptions about the possible distributions that determine the presentation of the learning data?
- What does “convergence in probability” really contribute to learning power?
- Are these results due to the “stronger assumptions” about the training data limited to classes with probabilistic grammars?

This talk

- What are these stronger assumptions about the possible distributions that determine the presentation of the learning data?
- What does “convergence in probability” really contribute to learning power? Not much, I think.
- Are these results due to the “stronger assumptions” about the training data limited to classes with probabilistic grammars?

This talk

- What are these stronger assumptions about the possible distributions that determine the presentation of the learning data?
- What does “convergence in probability” really contribute to learning power? Not much, I think.
- Are these results due to the “stronger assumptions” about the training data limited to classes with probabilistic grammars? I don’t think so.

Presentations of Data - Functions

1. The set of all total functions $f : N \to N$ is $F[N, N]$.
2. For any $f \in F[N, N]$, a *complete presentation* of f is

$$\langle x_0, f(x_0)\rangle, \langle x_1, f(x_1)\rangle, \ldots$$

   such that for all $x \in N$ there exists $i \in N$ such that $x_i = x$.
3. Let $\phi_0, \phi_1, \ldots$ be a enumeration for all partial recursive functions.

Presentations of Data - Languages

1. A *language* is a subset of N.
2. The *characteristic function* is a total function with domain N and codomain $\{0,1\}$, denoted $[x \in L]$.
3. A *complete presentation* of a language L is a complete presentation of $[x \in L]$.
4. A *positive presentation* of a language L is a sequence $x_0, x_1, \ldots$ such that for all $i \in N$, $x_i \in L \cup \{*\}$ and for all $x \in L$, there is $i \in N$ such that $x_i = x$.
5. Let $W_0, W_1, \ldots$ be an enumeration for recursively enumerable subsets of N.

Inductive Inference Machines

1. An *inductive inference machine* M is a Turing machine that running on an input sequence σ. We let $M[\sigma]$ denotes the empty, finite, or infinite sequence of numbers output by M.
2. A *probabilistic inductive inference machine* M is like the one above except there is also a coin tape which consists of an infinite sequence of Hs and Ts. M may read from this tape, at which point the tape head advances. M cannot write to the coin tape. The next-state function of M may depend on what is read.

Criteria of Identification: EX and TXTEX

1. Finite sequences *converge* to their last element. Infinite sequences *converge* to the element i iff all but finitely many elements in the sequence are i.
2. An inductive inference machine M *EX-identifies* $f \in F[N, N]$ if and only if for every complete presentation σ of f, $M[\sigma]$ is a non-empty sequence which converges to some i such that $\phi_i = f$.
3. An inductive inference machine M *TXTEX-identifies* $L \subseteq N$ if and only if for every positive presentation σ of L, $M[\sigma]$ is a non-empty sequence which converges to some i such that $W_i = f$.

Probabilistic Criteria of Identification

- Pitt (1985) observes the set of coin tapes for which a probabilistic inductive inference machine M converges is a measurable set. So the probability that M converges to f (or L) on σ is well-defined.
- A probabilistic inductive inference machine M *EX-identifies* $f \in F[N, N]$ *with probability* p if and only if for every complete presentation σ of f, the probability that M EX-identifies f on σ is at least p.
- A probabilistic inductive inference machine M *TXTEX-identifies* $L \subseteq N$ *with probability* p if and only if for every positive presentation σ of L, the probability that M TXTEX-identifies f on σ is at least p.

EX and TXTEX Identifiable Classes

- EX denotes those classes of functions $C \subseteq F[N, N]$ for which there exists an inductive inference machine M which EX-identifies in the limit every function $f \in C$.
- TXTEX denotes those classes of languages $\mathcal{L}$ for which there exists an inductive inference machine M which TXTEX-identifies in the limit every language $L \in \mathcal{L}$.
- $EX_{prob(p)}$ denotes those classes of functions $C \subseteq F[N, N]$ for which there exists a probabilistic inductive inference machine M which EX-identifies in the limit every function $f \in C$ with probability p.
- $TXTEX_{prob(p)}$ denotes those classes of languages $\mathcal{L}$ for which there exists an inductive inference machine M which TXTEX-identifies in the limit every language $L \in \mathcal{L}$ with probability p.

Results due to Pitt (1985)

Theorem 1. For all $p > 1/2$, $EX_{prob(p)} = EX$.

Theorem 2. For all $p > 2/3$, $TXTEX_{prob(p)} = TXTEX$.

- Conclusion: If we require TXTEX-identification (EX-identification) with probability greater than 2/3 (1/2), extra random information like coin tosses is no help.

Probability Distributions and the Draw Oracle

1. Let X be a nonempty finite of countable set. A *distribution* on X is a function $D : X \to [0, 1]$ such that $\sum_{x \in X} D(x) = 1$.
2. If X is any nonempty countable set and D is any distribution on X then $DRAW(D)$ is a oracle that is called with no input and returns an element of X according to D. Each call is an independent event.

EX Identification with input from DRAW(D)

1. A distribution is *complete* if and only if $D(X) > 0$ for all $x \in N$.
2. If $f \in F[N, N]$ and D is a distribution on N then D and f determine a distribution on $N \times N$:

$$D[f](\langle x, y \rangle) = D(x) \text{ if } f(x) = y$$

$$D[f](\langle x, y \rangle) = 0 \text{ if } f(x) \neq y$$

Lemma 5. If D is a complete distribution on N and $f \in F[N, N]$ then the sequence of values returned by an infinite sequence of calls to $DRAW(D[f])$ is a complete presentation with probability 1.

Equivalence of $EX_{draw(p)}$ and $EX_{prob(p)}$

- $EX_{draw(p)}$ denotes those classes of functions $C \subseteq F[N, N]$ such that for every complete distribution D on N, there exists an inductive inference machine M that EX-identifies all $f \in C$ when run with oracle $DRAW(D[f])$ as input with probability at least p.

Theorem 6. $EX_{draw(p)} = EX_{prob(p)}$.

proof sketch. For any $C \in EX_{draw(p)}$, there is some inductive inference machine M which identifies C for any complete distribution D. Angluin shows there is a probabilistic inductive inference machine M' which simulates M when run on a particular distribution D.

Likewise, For any $C \in EX_{prob(p)}$, there is some probabilistic inductive inference machine M which identifies C. Angluin shows there is an inductive inference machine M' which, when run on any complete distribution D, simulates M.

TXTEX Identification with input from DRAW(D)

1. The *support of* D is

$$S(D) = \{x \in X : D(x) > 0\}$$

   i.e. the elements of X with nonzero probability.

2. A distribution D on $N \cup \{*\}$ is *admissible* if and only if $S(D) - \{*\} = L$.

Lemma 8. If L is any language and D any distribution on $N \cup \{*\}$ admissible for L, then the sequence of values returned by an infinite sequence of calls to $DRAW(D)$ is a positive presentation with probability 1.

Equivalence of $TXTEX_{draw(p)}$ and $TXTEX_{prob(p)}$

- $TXTEX_{draw(p)}$ denotes those classes of languages $\mathcal{L}$ such that for every complete distribution D on N, there exists a probabilistic inductive inference machine M that TXTEX-identifies all $L \in \mathcal{L}$ when run with oracle $DRAW(D)$ as input with probability at least p.

Theorem 9. $TXTEX_{draw(p)} = TXTEX_{prob(p)}$.

proof sketch. As before. For any $C \in TXTEX_{draw(p)}$, there is some inductive inference machine M which identifies C for any complete distribution D. Angluin shows there is a probabilistic inductive inference machine M' which simulates M when run on a particular distribution D.

Likewise, For any $C \in TXTEX_{prob(p)}$, there is some probabilistic inductive inference machine M which identifies C. Angluin shows there is an inductive inference machine M' which, when run on any complete distribution D, simulates M.

Interim Discussion

- If the presentation of the data is drawn from some complete distribution, *and nothing about the distribution is known*, then it is the same as tossing coins.

Identifying Distributions

- The next results in the paper are about *identifying distributions*.

How does identifying distributions relate to identifying functions or languages?

- Is the problem of identifying languages or functions when samples are drawn from some distribution the same as identifying a distribution? Or are two distinct notions being conflated?
- For example, is a PCFG uniquely determined by the distribution it generates over strings?

Approximately Computable Distributions

1. A distribution D on N is said to be *approximately computable* if and only if there is a total recursive function f such that for every $x \in N$, and every positive rational number ϵ, $f(x, \epsilon)$ is a rational number r such that $|D(x) - r| \leq \epsilon$.
2. The sequence of distributions $D_0, D_1, \ldots$ is said to be *uniformly approximately computable* if and only if there is a total recursive function f such that for every $i \in N$, for every $x \in N$, and every positive rational number ϵ, $f(i, x, \epsilon)$ is a rational number r such that $|D(x) - r| \leq \epsilon$.

A Distance Measure for Distributions

- Define $d_*(D_1, D_2) = sup\{|D_1(x) - D_2(x)| : x \in N\}$
- Observe d_* is a distance metric.
  1. $d_*(D_1, D_2) = d_*(D_2, D_1)$.
  2. $d_*(D_1, D_2) = 0$ iff $D_1 = D_2$.
  3. $d_*(D_1, D_3) \leq d_*(D_1, D_2) + d_*(D_2, D_3)$.

Approximately Computable distances between distributions

1. An oracle X *represents* a distribution D on N if and only if whenever X is called with $x \in N$ and a positive rational number ϵ, the output of X is a rational number r such that $|r - D(x)| \leq \epsilon$.
2. If $d(D_1, D_2)$ is a distance metric on distributions, then $d(D_1, D_2)$ is *approximately computable* if and only if
   (a) there are two oracles X and Y where X represents D_1 and Y represents D_2
   (b) there is Turing Machine $M^{X,Y}(\epsilon)$ that calls on oracles X and Y such that for any positive rational number ϵ, the output of $M^{X,Y}(\epsilon)$ is a rational number r such that $|r - d(D_1, D_2)| \leq \epsilon$.

Properties of $d_*(D_1, D_2)$

Lemma 13. $d_*(D_1, D_2)$ is approximately computable.

- Let $D\langle n\rangle$ denote the empirical distribution after drawing n samples, for $n \geq 1$. In other words

$$D\langle n\rangle(x) = \frac{|\{0 \leq i \leq n-1 : x_i = x\}|}{n}$$

Lemma 14. Let D be any distribution on N. Let $a > 1$ and let $I(n) = \sqrt{6a(\log n)/n}$. Then with probability 1, $d_*(D, D\langle n\rangle) \leq I(n)$ for all but finitely many values of n.

Finitely Approachable Distributions

Let $\Delta = D_0, D_1, D_2, \ldots$ be a sequence of distributions on N.

1. If D is any distribution on N then let

$$approach_\Delta(D) = inf\{d_*(D, D_i) : i \in N\}$$

2. If for D there exists D_i such that $D_*(D, D_i) = approach_\Delta(D)$ then D is *finitely approachable by* Δ.
3. If $approach_\Delta(D) = 0$ and D is finitely approachable by Δ, then $D = D_i$.

Criteria for Identifying Distributions and Main Result

- A machine M EX-identifies the distribution D if and only if the probability is 1 that $M[Draw(D)]$ is a non-empty sequence of indices that converges to some i such that $D = D_i$.

Theorem 15. Let $\Delta = D_0, D_1, D_2, \ldots$ be a sequence of uniformly approximately computable sequences of distributions on N. There exists and inductive inference machine M that EX-identifies any D_i from Δ.

Proof Sketch of Theorem 15.

proof sketch. Angluin defines an inductive inference machine M that uses Δ. M works in stages $n = 1, 2, \ldots$. At stage n, M request an input and forms the empirical distribution $D\langle n\rangle$. For each i, $0 \leq i \leq n-1$, M approximates $d_*(D_i, D\langle n\rangle)$ to within $I(n)$. Let $e_i\langle n\rangle$ denote this approximation. M outputs the least $i < n$ such that $e_i\langle n\rangle \leq 2I(n)$ if there is such an i and then goes on to stage $n+1$.

This works because

1. M is an effective procedure
2. It can be shown there exists N such that for all $n > N$, if D_i is the target distribution,
   - 2.1 then $e_i\langle n\rangle \leq 2I(n)$.
   - 2.2 then for all $j < i$ and for all $n > N$, $e_j\langle n\rangle \geq 3I(n)$.

Scope of Theorem 15

1. Recall that $W_0, W_1, \ldots$ is an enumeration for recursively enumerable subsets of N.
2. Angluin shows how to define a uniformly approximately computable sequence of distributions $E_0, E_1, \ldots$ on $N \cup \{*\}$ such that for all $i \in N$, $L(E_i) = W_i$.
3. In other words, by Theorem 15, there is is EX-identifiable computable sequence of distributions whose associated languages are all the r.e. sets.

Conclusion Part I

1. Horning's (1969) result is essentially a corollary to the above.
2. Angluin suggests that the result (Theorem 15) has an analog from Gold 1967 that all the r.e. sets are identifiable in the limit from positive data if those positive presentations are required to be generated by primitive recursive functions.

Conclusion Part II

> The key to this result is to concentrate on modelling the functions presenting the text, which, being primitive recursive, are an enumerable class of total functions. Similarly, our result on identifying distributions, concentrates on an enumerable, computationally tractable class of "generators", namely, uniformly approximately computable sequences of distributions. The analogy between these results suggests there is a great power in attempting to model "how" a behavior is produced, as well as "what" behavior is produced.
>
> (Angluin, p. 21)

- "how" a behavior is produced = the generating functions
- "what" behavior is produced = the target language, function, etc.