New paper: "Logical induction"

September 12, 2016 | Nate Soares (https://intelligence.org/author/nate/) | Papers (https://intelligence.org/category/papers/)

(https://arxiv.org/abs/1609.03543)MIRI is releasing a paper introducing a new model of deductively limited reasoning: "**Logical induction** (https://arxiv.org/abs/1609.03543)," authored by Scott Garrabrant, Tsvi Benson-Tilsen, Andrew Critch, myself, and Jessica Taylor. Readers may wish to start with the abridged version (https://intelligence.org/files/LogicalInductionAbridged.pdf).

Consider a setting where a reasoner is observing a deductive process (such as a community of mathematicians and computer programmers) and waiting for proofs of various logical claims (such as the *abc* conjecture, or "this computer program has a bug in it"), while making guesses about which claims will turn out to be true. Roughly speaking, our paper presents a computable (though inefficient) algorithm that outpaces deduction, assigning high subjective probabilities to provable conjectures and low probabilities to disprovable conjectures long before the proofs can be produced.

This algorithm has a large number of nice theoretical properties. Still speaking roughly, the algorithm learns to assign probabilities to sentences in ways that respect any logical or statistical pattern (https://intelligence.org/2016/04/21/two-new-papers-uniform/) that can be described in polynomial time. Additionally, it learns to reason well about its own beliefs and trust its future beliefs while avoiding paradox. Quoting from the abstract:

> These properties and many others all follow from a single *logical induction criterion*, which is motivated by a series of stock trading analogies. Roughly speaking, each logical sentence φ is associated with a stock that is worth \$1 per share if φ is true and nothing otherwise, and we interpret the belief-state of a logically uncertain reasoner as a set of market prices, where $\mathbb{P}_n(\varphi)$=50% means that on day n, shares of φ may be bought or sold from the reasoner for 50¢. The logical induction criterion says (very roughly) that there should not be any polynomial-time computable trading strategy with finite risk tolerance that earns unbounded profits in that market over time.

This criterion is analogous to the "no Dutch book" criterion used to support other theories of ideal reasoning, such as Bayesian probability theory and expected utility theory. We believe that the logical induction criterion may serve a similar role for reasoners with deductive limitations, capturing some of what we mean by "good reasoning" in these cases.

The logical induction algorithm that we provide is theoretical rather than practical. It can be thought of as a counterpart to Ray Solomonoff's theory of inductive inference, which provided an uncomputable method for ideal management of *empirical* uncertainty but no corresponding method for reasoning under uncertainty about logical or mathematical sentences.[1] Logical induction closes this gap.

Any algorithm that satisfies the logical induction criterion will exhibit the following properties, among others:

1. *Limit convergence* and *limit coherence*: The beliefs of a logical inductor are perfectly consistent in the limit. (Every provably true sentence eventually gets probability 1, every provably false sentence eventually gets probability 0, if φ provably implies ψ then the probability of φ converges to some value no higher than the probability of ψ, and so on.)

2. *Provability induction*: Logical inductors learn to recognize any pattern in theorems (or contradictions) that can be identified in polynomial time.

    - Consider a sequence of conjectures generated by a brilliant mathematician, such as Ramanujan, that are difficult to prove but keep turning out to be true. A logical inductor will recognize this pattern and start assigning Ramanujan's conjectures high probabilities well before it has enough resources to verify them.

    - As another example, consider the sequence of claims "on input n, this long-running computation outputs a natural number between 0 and 9." If those claims are all true, then (roughly speaking) a logical inductor learns to assign high probabilities to them as fast as they can be generated. If they're all false, a logical inductor learns to assign them low probabilities as fast as they can be generated. In this sense, it learns inductively to predict how computer programs will behave.

    - Similarly, given any polynomial-time method for writing down computer programs that halt, logical inductors learn to believe that they will halt roughly as fast as the source codes can be generated. Furthermore, given any polynomial-time method for writing down computer programs that *provably* fail to halt, logical inductors learn to believe that they will fail to halt roughly as fast as the source codes can be generated. When it comes to computer programs that fail to halt but for which there is no proof of this fact, logical inductors will learn not to anticipate that the program is going to halt anytime soon, even though they can't tell whether the program is going to halt in

the long run. In this way, logical inductors give some formal backing to the intuition of many computer scientists that while the halting problem is undecidable in full generality, this rarely interferes with reasoning about computer programs in practice.[2]

3. *Affine coherence*: Logical inductors learn to respect logical relationships between different sentences' truth-values, often long before the sentences can be proven. (E.g., they will learn for arbitrary programs that "this program outputs 3" and "this program outputs 4" are mutually exclusive, often long before they're able to evaluate the program in question.)

4. *Learning pseudorandom frequencies*: When faced with a sufficiently pseudorandom sequence, logical inductors learn to use appropriate statistical summaries. For example, if the Ackermann(*n*,*n*)th digit in the decimal expansion of π is hard to predict for large *n*, a logical inductor will learn to assign ~10% subjective probability to the claim "the Ackermann(*n*,*n*)th digit in the decimal expansion of π is a 7."

5. *Calibration* and *unbiasedness*: On sequences that a logical inductor assigns ~30% probability to, if the average frequency of truth converges, then it converges to ~30%. In fact, on any subsequence where the average frequency of truth converges, there is no efficient method for finding a bias in the logical inductor's beliefs.

6. *Scientific induction*: Logical inductors can be used to do sequence prediction, and when doing so, they dominate the universal semimeasure.

7. *Closure under conditioning*: Conditional probabilities in this framework are well-defined, and conditionalized logical inductors are also logical inductors.[3]

8. *Introspection*: Logical inductors have accurate beliefs about their own beliefs, in a manner that avoids the standard paradoxes of self-reference.

    - For instance, the probabilities on a sequence that says "I have probability less than 50% on the *n*th day" go extremely close to 50% and oscillate pseudorandomly, such that there is no polynomial-time method to tell whether the *n*th one is slightly above or slightly below 50%.

9. *Self-trust*: Logical inductors learn to trust their future beliefs more than their current beliefs. This gives some formal backing to the intuition that real-world probabilistic agents can often be reasonably confident in their future reasoning in practice, even though Gödel's incompleteness theorems place strong limits on reflective reasoning in full generality.[4]

The above claims are all quite vague; for the precise statements, refer to the paper (https://intelligence.org/files/LogicalInduction.pdf).

Logical induction was developed by Scott Garrabrant in an effort to solve an open problem we spoke about (https://intelligence.org/2016/04/21/two-new-papers-uniform/) six months ago. Roughly speaking, we had formalized two different desiderata for good reasoning under logical uncertainty: the ability to recognize patterns in what is provable (such as mutual exclusivity relationships between claims about computer programs), and the ability to recognize statistical patterns in sequences of logical claims (such as recognizing that the decimal digits of π seem pretty pseudorandom). Neither was too difficult to achieve in isolation, but we were surprised to learn that our simple algorithms for achieving one seemed quite incompatible with our simple algorithms for achieving the other. Logical inductors were born of Scott's attempts to achieve both simultaneously.[5]

I think there's a good chance that this framework will open up new avenues of study in questions of metamathematics, decision theory, game theory, and computational reflection that have long seemed intractable. I'm also cautiously optimistic that they'll improve our understanding of decision theory and counterfactual reasoning, and other problems related to AI value alignment (https://intelligence.org/technical-agenda/).[6]

We've posted a talk online that helps provide more background for our work on logical induction:[7]

Edit: For a more recent talk on logical induction that goes into more of the technical details, see here (https://www.youtube.com/watch?v=UOddW4cXS5Y).

"Logical induction (https://intelligence.org/files/LogicalInduction.pdf)" is a large piece of work, and there are undoubtedly still a number of bugs. We'd very much appreciate feedback: send typos, errors, and other comments to errata@intelligence.org (mailto:errata@intelligence.org).[8]

1. While impractical, Solomonoff induction gave rise to a number of techniques (ensemble methods) that perform well in practice. The differences between our algorithm and Solomonoff induction point in the direction of new ensemble methods that could prove useful for managing logical uncertainty, in the same way that modern ensemble methods are useful for managing empirical uncertainty. ↩
2. See also Calude and Stay's (2006) "Most Programs Stop Quickly or Never Halt. (http://arxiv.org/abs/cs/0610153)" ↩
3. Thus, for example one can make a logical inductor over Peano arithmetic by taking a logical inductor over an empty theory and conditioning it on the Peano axioms. ↩
4. As an example, imagine that one asks a logical inductor, "What's your probability of φ, given that in the future you're going to think φ is likely?" Very roughly speaking, the inductor will answer, "In that case φ would be likely," even if it currently thinks that φ is quite unlikely. Moreover, logical inductors do this in a way that avoids paradox. If φ is "In the future I will think φ is less than 50% likely," and in the present you ask, "What's your probability of φ, given that in the future you're going to believe it is ≥50% likely?" then its answer will be "Very low." Yet if you ask "What's your probability of φ, given that in the future your probability will be *extremely close* to 50%?" then it will answer, "Extremely close to 50%." ↩
5. Early work towards this result can be found at the Intelligent Agent Foundations Forum (https://agentfoundations.org/item?id=270). ↩
6. Consider the task of designing an AI system to learn the preferences of a human (e.g., cooperative inverse reinforcement learning (http://arxiv.org/abs/1606.03137)). The usual approach would be to model the human as a Bayesian reasoner trying to maximize some reward function, but this severely limits our ability to model human irrationality and miscalculation even in simplified settings. Logical induction may help us address this problem by providing an idealized formal model of limited reasoners who don't know (but can eventually learn) the logical implications of all of their beliefs.

   Suppose, for example, that a human agent makes an (unforced) losing chess move. An AI system programmed to learn the human's preferences from observed behavior probably shouldn't conclude that the human *wanted* to lose. Instead, our toy model of this dilemma should allow that the human may be resource-limited and may not be able to deduce the full implications of their moves; and our model should allow that the AI system is aware of this too, or can learn about it. ↩
7. Slides from then relatively nontechnical portions (https://intelligence.org/files/LogicalInductionSlidesA.pdf); slides from the technical portion (https://intelligence.org/files/LogicalInductionSlidesB.pdf). For viewers who want to skip to the technical content, we've uploaded the talk's middle segment as a shorter stand-alone video: link (https://www.youtube.com/watch?v=QF-eCscwf38). ↩
8. The intelligence.org version (https://intelligence.org/files/LogicalInduction.pdf) will generally be more up-to-date than the arXiv version (https://arxiv.org/abs/1609.03543). ↩

14 Comments

Login ▾

G

Join the discussion…

LOG IN WITH

OR SIGN UP WITH DISQUS (?)

Name

5 Share

Best Newest Oldest

Alexey Feigin

7 years ago

This paper is really really important and I wouldn't have noticed it unless told by a friend. I would appreciate it if there was a less noisy RSS feed for MIRI. Something like just the monthly newsletter with 1 most important thing highlighted.

1 0 Reply • Share ›

Rob Bensinger Mod → Alexey Feigin

7 years ago

Does https://intelligence.org/ca... work for this purpose? We may also experiment with highlighting a key news item or two in the newsletter.

0 0 Reply • Share ›

Alexey Feigin → Rob Bensinger

7 years ago

Thanks for replying! Yes, the newsletter RSS feed is serviceable, but if you could highlight the most important item of the month or two in the newsletter content that would be great.

0 0 Reply • Share ›

O

Oongawa

7 years ago edited

Confused by the definition of belief states as having finite support, assigning 0 probability to sentences they've never considered before. Why isn't this trivial to exploit, by just coming up with new true sentences each day, buying them for nothing, and selling them when the price goes up?

1 0 Reply • Share ›

T

Tsvi Benson-Tilsen → Oongawa

7 years ago

You can visualize a logical inductor (that uses belief states) as having a quickly-expanding bubble of "considered sentences". That bubble grows fast enough that, for any particular trader, *eventually* it is the case that for all future times the bubble will engulf all the sentences traded on by that trader.

A little more precisely, if you fix a polytime trader T, there may be some finite number of days on which that trader has free reign to buy large numbers of shares at price 0. But for any logical inductor P_n, eventually P_n will start assigning (possibly) non-zero probabilities to the sentences that T trades on. So even though, at all times n, there are sentences with P_n(phi) = 0, it is also the case that for every polytime trader T, there are only finitely many times n at which T (e.g.) buys a full share at price 0. If we consider just the specific algorithm LIA given in the paper, on some day k, T appears as S_k in the enumeration used by TradingFirm. At that point MarketMaker starts actually selecting prices on sentences traded by T, rather than defaulting to 0 by omission.

3 0 Reply • Share ›

B

Benjamin Rhodes

7 years ago edited

Is MIRI aware of the Probabilistic Numerics research community? Their work is much more applied, but it seems like there might be interesting connections with logical induction. This short blog post gives a flavour of what they are up to http://probabilistic-numeri...

0 0 Reply • Share ›

Nicholas Robinson → Benjamin Rhodes

6 years ago

Following up on this. Hopefully someone at MIRI sees new comments on the site? I'd appreciate understanding how this MIRI work relates to probabilistic numerics (they might be very different but superficially they sounds similar) http://probabilistic-numeri...

0 0 Reply • Share ›

T **Tsvi Benson-Tilsen** → Nicholas Robinson
6 years ago

I'm not familiar with the details of this research field, but: it is related on some level, although as Benjamin said, it is much more applied. More specifically, decision theory research tends to ask for models of reasoning under logical uncertainty about *adversarial* processes, such as other powerful reasoners / agents (which can be specified and reasoned about formally). This is probably in contrast to the problems dealt with by probabilistic numerics, which are "merely very computationally difficult problems", and not actively adversarial.

1 0 Reply • Share ›

 **renurenu**
7 years ago

Any planned peer reviews of this article? The title seems awfully generic given that logical induction is a huge topic, with so many systems out there, so this would be the first point any reviewer is likely to remark.

0 0 Reply • Share ›

 **Rob Bensinger** Mod → renurenu
7 years ago

Yep, the paper is being submitted for publication. I don't know what systems you have in mind when you say "so many systems"; there are many approaches to inductive reasoning, but not (yet) many approaches to inductive reasoning about logical or mathematical facts. "Mathematical induction" might have been a better name for this concept, but unfortunately it's already taken! (And isn't a form of induction in the relevant sense.)

0 0 Reply • Share ›

 **renurenu** → Rob Bensinger
7 years ago edited

Wait, what about the whole field of inductive logic programming (and its applications to mathematical tasks)?
Great the paper is submitted to a journal!

0 0 Reply • Share ›

T **Tsvi Benson-Tilsen** → renurenu
7 years ago

The intention of the name is to contrast to logical deduction (producing very confident logical assertions via theorem proving); logical induction instead produces varying confidence in hard-to-compute logical assertions by learning inductively from the truth values of easier-to-compute logical facts. This is related to ILP in that they both search for programs that explain some logical facts, but distinct in that logical induction is concerned with the theoretical computational complexity of very general (e.g. first-order) logical reasoning.

0 0 Reply • Share ›

 **FeepingCreature**
7 years ago

Good stuff!
Just skimming, but saw "beliefs accurate beliefs" on page 67.

0 0 Reply • Share ›

 **Rob Bensinger** Mod → FeepingCreature
7 years ago

Thanks, Feeping!

0 0 Reply • Share ›