Variations of the Garrabrant-inductor

Sune K. Jakobsen

The purpose of this note is to point out some interesting variations of the logical inductor defined by Garrabrant et al. [1] and to state some open problems. As pointed out in Section 7.3 of [1], the ideas of aggregating different algorithms' "knowledge" by making them trade against each other can be generalized to other classes of algorithms than just polynomial time algorithms, e.g. to linear time or exponential time algorithms. In this note I will suggest various other changes that will make the inductor have better properties than the Garrabrant inductor, at the cost of being even slower.[1]

1 r-traders

Garrabrant traders, as defined in [1] takes the day number, n, as input and outputs a trading strategy, T_n. Since each T_n is the output of a polynomial time algorithm, it can only involve polynomially many sentences. With Garrabrant traders we get theorems like

Theorem 1 (Theorem 4.2.1, first half)**.** *Let $\overline{\phi}$ be an e.c. sequence of theorem. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n 1.$$

That is, the probabilities that the logical inductor assigns to a sequence theorems are guaranteed to go to 1, but this guarantee only holds if we consider one theorem per day. There are exponentially many sentences of a given length, so if we want something to hold for all/most theorems of length n, we would have to wait exponentially many days.

To solve this problem, we define an *r-trader* to be a trader that takes input (n, r) and outputs an n-trading strategy $T_{n,r}$. On day n we run each trader on (n, r) for all $r \in \{0,1\}^{\leq n}$.[2] The trader's strategy on day n is then $T_n = \sum_r T_{n,r}$.

For Garrabrant traders it is possible to let the trading strategy re-compute what it did on previous days. This is not possible for r-traders, so to help the traders, we let the trading strategies depend (in an expressible way) on the infimum and supremum on the trader's plausible assessments (as defined in [1] after Definition 3.5.1) and on the trader's holdings as well as on $\mathbb{P}_{\leq n}$.

[1] It is trivial to define a logical inductor with better properties if you are willing to make it slower: just give each trader more time. A Garrabrant inductor with exponential time traders would have properties even stronger than those in Theorem 2 of this note. However, I hope the variations suggested here are philosophically more interesting than just giving the traders more time, and that the ideas might be useful when designing practical algorithms inspired by logical inductors.

[2] Or we could instead say $\{0,1\}^{\leq p(n)}$, where p is a polynomial that depends on the trader.

Definition 1. Let $\mathcal{S} \subset \{0,1\}^n$ denote the set of sentences (with some fixed encoding) and let $\mathcal{S}_n \subset \mathcal{S}$ be the set of sentences of length at most n.

A random variable Φ is *efficiently samplable from n bits* if there is a polynomial time algorithm A such that if U_n is uniformly distributed on $\{0,1\}^n$ we have $\Phi \sim A(U_n)$. A sequence of random variables $\overline{\Phi}$ is *efficiently samplable* if there is an algorithm A such that for each n we have $\Phi_n \sim A(U_n)$.

This definition generalizes e.c. sequences: if $\overline{\phi}$ is an e.c. sequence of theorems, it means that there exists a polynomial time algorithm A that on input n returns ϕ_n. We can now define an algorithm B that on input u returns $A(|u|)$, so we can consider $\overline{\phi}$ to be an efficiently samplable sequences of (degenerate) random variables.

We will now show that if a market is not exploitable by r-traders we can generalize Theorem 4.2.1 to efficiently sampable sequences of random variables.

Theorem 2 (Variation of first half of Theorem 4.2.1)**.** *Let $\overline{\mathbb{P}}$ be a market that is not exploitable by r-traders and let $\overline{\Phi}$ be an efficiently samplable sequences of random variables supported on theorems. Then for any $\epsilon > 0$ and for sufficiently large n we have*

$$\Pr[\mathbb{P}_n(\Phi_n) \geq 1 - \epsilon] \geq 1 - \epsilon$$

Here the probability Pr is taken over the randomness in Φ_n, while the $\mathbb{P}_n(\phi)$ returns the logical inductor's probability estimate for particular values ϕ of Φ_n. The theorem says that for a random instance ϕ of Φ_n, the logical inductor will most likely be almost correct on day n. By using the ideas of Garrabrant et al. we can add r-traders to the market and get a logical inductor that that is not exploitable by r-traders and hence have the above property.

Proof. (Sketch) Assume that $\overline{\mathbb{P}}$ is not exploitable by any r-traders and consider an r-trader that acts as follows. On input (n, r) with $|r| \neq n$ it returns the empty trade and on input (n, r) with $|r| = n$ it computes $A(r)$. Now if the price of $A(r)$ stocks is below $1 - \epsilon$ and the lower bound on the plausible assessment of the trader is no worse than -1, the trader buys $2^{-|r|} = 2^{-n}$ stocks in $A(r)$. By construction, the trader can at most spend 1 on each day, so -2 is a lower bound on its worst plausible assessment on any day.

For each dollar it spends, it will eventually get at least $\frac{1}{1-\epsilon} > 1 + \epsilon$ back. Since the trader, by assumption, does not exploit the market, it can only spend a finite total amount. This could happen in two ways: either the restriction that the trader does not buy if it has a plausible assessments below -1 stops the trader from buying infinitely many times or only finitely many times. If infinitely many times, let M be the total amount spend on stocks in the limit. Since the trader reaches the limit of -1, we must have $M \geq 1$. By definition of limit, there will be a time n where the trader has already spent $M - \epsilon$ on stocks. Each dollar will eventually pay back more than $1 + \epsilon$, and by then the trader will have spend at most ϵ more. This gives the trader a worst plausible assessment of $(M - \epsilon)\epsilon - \epsilon \geq M\epsilon - \epsilon \geq 0$ so the trader cannot hit the limit of -1 again. Contradiction.

This shows that there is some last time $n_0 - 1$ where the -1 limit prevents the trader from buying stocks. We have already seen that the trader will only spend a finite amount on stocks, so we must now have

$$\infty > \sum_{n \geq n_0} \sum_{r \in \{0,1\}^n} 2^{-n} \mathbb{1}_{\mathbb{P}_n(A(r)) < 1-\epsilon} = \sum_{n \geq n_0} \Pr(\mathbb{P}_n(\Phi_n) < 1 - \epsilon).$$

Here $\mathbb{1}_P$ is the indicator function of P. This shows that in particular, the probabilities must go to 0.

In this proof sketch we use discontinuous (non-expressible) trading strategies. It is easy to approximate them sufficiently well by expressible strategies as in [1]. □

I would think that all the theorems given in Garrabrant et al. for sequences $\overline{\phi}$ can be generalized in similar ways if we introduce r-traders to the market, but I have not checked.

2 Restricted r-traders

When we allow r-traders we get a very strong logical inductor. To see an example of how strong it is, let $h_n : \{0,1\}^{2n} \to \{0,1\}^{2n}$ be a sequence of one-way permutations, and for $x \in \{0,1\}^n, y \in \{0,1\}^{2n}$ let $\phi_{x,y}$ denote the sentence

$$\phi_{x,y} := \text{“}\exists r \in \{0,1\}^n : h_n(x \circ r) = y\text{”}.$$

Although it is difficult to determine whether such a sentence is true when presented to you, it is easy to sample such true sentences: simply sample x and r and compute y. So Theorem 2 tells us that a logical inductor that is constructed using r-traders, and hence not exploitable by r-traders, will learn to recognize true sentences $\phi_{x,y}$. This is fine if we want our logical inductor to be as “efficient” as possible (when efficiency is measured only by considering how quickly $\overline{\mathbb{P}}$ converges measured in “days” rather than in actually running time). However, if we want a realistic model for how fast a practical approximation of a logical inductor converges, we want it *not* to assign accurate probabilities to $\phi_{x,y}$ too quickly: practical algorithms will (under the standard cryptographic assumption that one-way permutations exists) not be able to make good guesses of the truth value of a given $\phi_{x,y}$ in a reasonable time. If we can define a logical inductor that captures this difference, it might be useful for understanding cryptography and understanding the P vs NP problem.

To capture the fact that some samplable theorems are hard to prove without the randomness used to generate them, we might use restricted r-traders instead. A *restricted r-trader* is a polynomial time algorithm that takes as input (n, ϕ) and outputs a trading strategy $T_{n,\phi}$ that only trades in ϕ stocks. On day n, the restricted r-trader is run on each input (n, ϕ) with $|\phi| \leq n$, and all the outputs are added up. Notice that the trade on ϕ is still allowed to depend in an expressible way on the prices of other stocks. Like for the r-traders, we also allow restricted r-traders to output trading strategies that depend in an expressible way on the trader's holdings and on the infimum and supremum on plausible assessments of the trader's worth. We now get a different variation of Theorem 4.2.1, but to state it, we first need a definition.

Definition 2. A sequence $\overline{\Phi}$ of random variables is *efficiently computable* or *e.c.* all the density function of each Φ_n only takes rational values, has support in $\{0,1\}^{\leq n}$ and if there is an algorithm A that on input n and $x \in \{0,1\}^{\leq n}$ returns $\Pr(\Phi_n = x)$.

Theorem 3 (Variation of first half of Theorem 4.2.1)**.** *Let $\overline{\mathbb{P}}$ be a market that is not exploitable by restricted r-traders and let $\overline{\Phi}$ be an e.c. sequences of random* *variables taking values in the theorems in $\mathcal{S}$. Then for any $\epsilon > 0$ and sufficiently large n we have*

$$\Pr[\mathbb{P}_n(\Phi_n) \geq 1 - \epsilon] \geq 1 - \epsilon.$$

Proof. Consider a restricted r-trader that acts as follows: on input (n, ϕ) it computes $A(n, \phi)$. If the trader's plausible assessments are bounded below by -1, and the price of ϕ stock are below $1 - \epsilon$, the trader buys $A(n, \phi)$ stocks in ϕ. The rest of this proof is similar to that of Theorem 2. □

Unlike Theorem 2 this theorem no longer says that the logical inductor will assign accurate probabilities to $\phi_{x,y}$. Notice however that the statement of Theorem 2 might still hold for the logical inductor made out of restricted r-traders: it is difficult to prove upper bound on how fast logical inductors learn, so I have not been able to prove that a logical inductor build from restricted r-traders will not be able to get accurate probabilities for the $\phi_{x,y}$'s quickly.

We know that for some enumerations of the traders Garrabrant inductor does not have the properties given in Theorem 2 and Theorem 3: you can enumerate the traders in a way that ensure that at most $n^{\log(n)}$ stocks have been traded on day n and there are e.c. sequences of random variables supported on theorems with a support that is larger than $n^{\log(n)}$.

3 Labeled stocks

Instead of just having one stock per sentence ϕ, we can have infinitely many stocks (ϕ, r) per sentence, one for each $r \in \{0, 1\}^*$. We say that the stock (ϕ, r) is for ϕ and has label r. We now consider a variant of the restricted r-trader, which on input (n, r) can trade on all stocks with the label r. Again the trader is allowed to base its trading strategy on the prices of stocks with other labels. Once a sentence ϕ has been proven by $\overline{D}$, the values of all the stocks (ϕ, r) is set to 1. If the market is functioning perfectly, the price of (ϕ, r) should not depend on r. However, because all the traders have bounded computational power and each sentence can have infinitely many different labels, we could hope that the price of the stock might depend on the label. For example, that $(\phi_{x,y}, r)$ is valued at 1 when $h_n(x \circ r) = y$ but $(\phi_{x,y}, \lambda)$ is still valued at 2^{-n} for the empty sting λ. If there is a price difference (or if we can modify the definition of a logical inductor to create a price difference in some natural way) this might be useful for cryptography.

4 Value of Information and Option traders

Suppose you are offered a bet on a decidable sentence ϕ. You can pay \$0.60 for the bet and then if ϕ is true you get \$1 back. You have run your logical inductor for n days, and it outputs $\mathbb{P}_n(\phi) = 0.3$. If you cannot run the logical inductor for longer, you should not bet, but what if you can run the logical inductor for n more days at some price p? Then you should run it if and only if you think $\max(\mathbb{P}_{2n}(\phi) - 0.6, 0)$ is in expectation more than p.

You can write down a sentence that encodes the logical inductor itself and make the logical inductor estimate this expectation: $\mathbb{E}_n \max(\mathbb{P}_{2n}(\phi) - 0.6, 0)$. However, this is an extremely complicated sentence, because it contains an encoding of a logical inductor. Since the inductor has a long running time, it will take a long time for $\overline{D}$ to prove theorems about it, so for "astronomically small" n (values that can actually occur in this universe) $\mathbb{E}_n \max(\mathbb{P}_{2n}(\phi)-0.6, 0)$ is probably not useful. To get a better expectation of $\max(\mathbb{P}_{2n}(\phi) - 0.6, 0)$, we could add call and put options to the market. For each sentence ϕ, each rational number $x \in [0, 1]$ and each natural number n we can add a call options that gives the owner the right to buy one stock for sentence ϕ at time n for the price x and a put option that gives the owner the right to sell stocks for ϕ at time n for the price x. We define option-traders to be like Garrabrant traders, except trading strategies can also depend on the prices of options and can also involve buying and selling options. If a trader has a negative amount of some options, the plausible assessments have to take this into account in the natural way. For simplicity, we assume that call options (ϕ, n, x) are transformed into stocks for ϕ on day n if and only if $\phi^{*n} > x$ for the price of x, and similarly put options are also exercised automatically.[3] This creates a discontinuity, but if only exercise the options *after* day n and not as part of finding the market equilibrium on day n this is not a problem.

We should now have the following recurring unbiasedness theorem, modelled on Theorem 4.3.6 in Garrabrant et al.

Conjecture 4. *Let $\overline{\phi}, \overline{m}, \overline{x}$ be a e.c. sequences and let $(\phi, m, x)^{*n}$ denote the price on day n of a call option for one ϕ stock to be bought at price x on day m. Let $\overline{w}$ be a $\mathbb{P}$-generable divergent weighting. Then the sequence*

$$\frac{\sum_{i \le n} w_i \left((\phi_i, m_i, x_i)^{*i} - \max(\mathbb{P}_{m_i}(\phi_i) - x_i, 0)\right)}{\sum_{i \le n} w_i}$$

has 0 *as a limit point.*

I think you can build a proof of this based on the proof of Theorem 4.5.9 of Garrabrant et al.

If you care about the Value of Information about Value of Information, you can also add higher order options, but having traders tradering higher order options will probably make the logical inductor slower at getting accurate values of $\mathbb{P}(\phi)$'s.

5 Open problems

Is there a natural logical inductor that captures the fact that some theorems can be hard to prove if presented to you, but easy to generate in a way that ensures

[3] Automatic execution of options could force some traders to go bankrupt (go over the budget, and be eliminated by the Budgeter algorithm on all future days). We could let trader choose whether to exercise the options themselves, but this raises many other questions. E.g. are the sellers and buyers of options paired up, or are sold options just exercised at the average rate at which bough options are exercised? Can traders exercise their option before the expiry day? If a trader buys and sells the same option, should they cancel out, or should we allow them not to chancel out in case the trader wants to exercise his options some day and the buyers of his options does not? Should a trader be allowed to exercise their option after they go bankrupt? If not, this might make the price of options artificially low (although probably not in the limit), as some buyers of options will not be able to exercise their options later on. I didn't want to think about all these questions, so I just assume that options are exercised automatically. I think that the answers to all these questions will not make a difference in the limit, but the will make some difference in the running time.

that you only generate true sentences? This is morally equivalent to: Is there a logical inductor that has the property of Theorem 3 but not of Theorem 2? Equivalently, is there a logical inductor that are exploitable by r-traders, but not by restricted r-traders? This might be useful for cryptography, especially if we could prove this distinction, for example conditioned on the existence of one-way permutations.

We have various parameters to play around to get such a logical inductor: running time of traders, restriction on the traders, what information the trades can depend on (bounds on plausible assessments? holdings?), are the traders required to distribute its budget between the different r's in a way that does not depend on $\mathbb{P}_{\leq n}$, the enumeration of the traders, the total budget of the different traders, etc. Unfortunately, it seems difficult to prove upper bounds on how fast the logical inductor is learning, so it will be difficult to prove that a logical inductor does not have the property of Theorem 2.

It would also be interesting to get a working definition of the entropy of an LUV (logically uncertain variable). For example, if $x, r \in \{0,1\}^n, y \in \{0,1\}^{2n}$ with $y = h_n(x \circ r)$ but only y is given, then x is a LUV. If h_n is a one-way permutation it should intuitively have entropy n, but can we capture this? There exists various definition of computational entropy, e.g. HILL entropy and Yao entropy [2], but here the setting is different: you see a pseudo-random output X and want to estimate the computational entropy of the random variable. Still, there might be connections between the entropy of an LUV and computational notions of entropy.

References

[1] Scott Garrabrant, Tsvi Benson-Tilsen, Andrew Critch, Nate Soares, and Jessica Taylor. Logical induction. *arXiv preprint arXiv:1609.03543*, 2016.

[2] Stephan Krenn, Krzysztof Pietrzak, Akshay Wadia, and Daniel Wichs. A counterexample to the chain rule for conditional hill entropy. Cryptology ePrint Archive, Report 2014/678, 2014. http://eprint.iacr.org/2014/678.