Distribution of Environments in Formal Measures of Intelligence: Extended Version

Bill Hibbard
December 2008

Abstract

This paper shows that a constraint on universal Turing machines is necessary for Legg's and Hutter's formal measure of intelligence to be unbiased. Their measure, defined in terms of Turing machines, is adapted to finite state machines. A No Free Lunch result is proved for the finite version of the measure, and this motivates a less abstract measure.

Introduction

Goertzel's keynote at AGI-08 described theory as one useful direction for artificial intelligence research (Goertzel 2008). Schmidhuber, Hutter and Legg have produced a number of recent results to formally define intelligence and idealized intelligent agents. In particular, Legg and Hutter have developed a formal mathematical model for defining and measuring the intelligence of agents interacting with environments (Legg and Hutter 2006). Their model includes weighting distributions over time and environments. The point of this paper is to argue that a constraint on the weighting over environments is required for the utility of the intelligence measure.

The first section of this paper describes Legg's and Hutter's measure and demonstrates the importance of the weighting over environments. Their measure is defined in terms of Turing machines and the second section investigates how the measure can be adapted to a finite model of computing. The third section proves an analog of the No Free Lunch Theorem for this finite model. The final section uses these results to argue for a less abstract model for weighting over environments.

A Formal Measure of Intelligence

Legg and Hutter used reinforcement learning as a framework for defining and measuring intelligence (Legg and Hutter 2006). In their framework an agent interacts with its environment at a sequence of discrete times, sending action a_i to the environment and receiving observation o_i and reward r_i from the environment at time i. These are members of finite sets A, O and R respectively, where R is a set of rational numbers between 0.0 and 1.0. The environment is defined by a probability measure:

$$\mu(o_k r_k \mid o_1 r_1 a_1 \ldots o_{k-1} r_{k-1} a_{k-1})$$

and the agent is defined by a probability measure:

$$\pi(a_k \mid o_1 r_1 a_1 \ldots o_{k-1} r_{k-1} a_{k-1}).$$

The value of agent π in environment μ is defined by the expected value of rewards:

$$V_\mu^\pi = \mathbf{E}(\sum_{i=1}^{\infty} w_i r_i)$$

where the $w_i \geq 0.0$ are a sequence of weights for future rewards subject to $\sum_{i=1}^{\infty} w_i = 1$ (Legg and Hutter combined the w_i into the r_i). In reinforcement learning the w_i are often taken to be $(1-\gamma)\gamma^{i-1}$ for some $0.0 < \gamma < 1.0$. Note $0.0 \leq V_\mu^\pi \leq 1.0$.

The intelligence of agent π is defined by a weighted sum of its values over a set E of computable environments. Environments are computed by programs, finite binary strings, on some prefix universal Turing machine (PUTM) U. The weight for $\mu \in E$ is defined in terms of its Kolmogorov complexity:

$$K(\mu) = \min \{ |p| : U(p) \text{ computes } \mu \}$$

where $|p|$ denotes the length of program p. The intelligence of agent π is:

$$V^\pi = \sum_{\mu \in E} 2^{-K(\mu)} V_\mu^\pi.$$

The value of this expression for V^π is between 0.0 and 1.0 because of Kraft's Inequality for PUTMs (Li and Vitányi 1997):

$$\sum_{\mu \in E} 2^{-K(\mu)} \leq 1.0.$$

Legg and Hutter state that because $K(\mu)$ is independent of the choice of PUTM up to an additive constant that is independent of μ, we can simply pick a PUTM. They do caution that the choice of PUTM can affect the relative intelligence of agents and discuss the possibility of limiting PUTM complexity. But in fact a constraint on PUTMs is necessary to avoid intelligence measures biased toward specific environments:

Proposition 1. Given $\mu \in E$ and $\varepsilon > 0$ there exists a PUTM U_μ such that for all agents π:

$$V_\mu^\pi / 2 \leq V^\pi < V_\mu^\pi / 2 + \varepsilon$$

where V^π is computed using U_μ.

Proof. Fix a PUTM U_0 that computes environments. Given $\mu \in E$ and $\varepsilon > 0$, fix an integer n such that $2^{-n} < \varepsilon$. Then construct a PUTM U_μ that computes μ given the program "1", fails to halt (alternatively, computes μ) given a program starting with between 1 and n 0's followed by a 1, and computes $U_0(p)$ given a program of n+1 0's followed by p. Now define K using U_μ. Clearly:

$$2^{-K(\mu)} = 1/2$$

And, applying Kraft's Inequality to U_0:

$$\sum_{\mu' \neq \mu} 2^{-K(\mu')} \leq 2^{-n} < \varepsilon.$$

So:

$$V^{\pi} = V_{\mu}^{\pi} / 2 + X$$

Where

$$X = \sum_{\mu' \neq \mu} 2^{-K(\mu')} V_{\mu'}^{\pi} \text{ and } 0 \leq X < \varepsilon. \ \square$$

In addition to the issue of weighting over environments, there are other interesting issues for an intelligence measure:

1. It is not clear what weighting of rewards over time is best. V_{μ}^{π} is defined using the reinforcement learning expression for the value of the state at the first time step. But an intelligent agent generally needs time to learn a novel environment, suggesting that V_{μ}^{π} should be defined by the value of the state at a later time step, or even its limit as time increases to infinity. On the other hand, speed of learning is part of intelligence and the expression for the value at the first time step rewards agents that learn quickly.

2. The expression for V^{π} combines weighting over both environments and time, which can lead to unintuitive results. Lucky choices of actions at early, heavily weighted, time steps in simple, heavily weighted, environments, may count more toward an agent's intelligence than good choices of actions in very difficult, but lightly weighted, environments. As environment complexity increases, agents will require longer times to learn good actions. Thus, given a distribution of time weights that is constant over all environments, even the best agents will be unable to get any value as environment complexity increases to infinity. It would make sense for different environments to have different time weight distributions.

3. If PUTM programs were answers (as in Solomonoff Induction, where an agent seeks programs that match observed environment behavior) then weighting short programs more heavily would make sense, since shorter answers are better (according to Occam's razor). But here they are being used as questions and longer programs pose more difficult questions so arguably should be weighted more heavily. But if the total weight over environments is finite and the number of environments is infinite, then it is inevitable that environment weight must approach zero as environment complexity increases to infinity. On the other hand, shorter programs are more probable, determined for example by frequency of occurrence as substrings of sequences of random coin flips, and we may wish to weight environments by probability of occurrence.

4. Whatever PUTM is used to compute environments, all but an arbitrarily small ε of an agent's intelligence is determined by its value in a finite number of environments.

5. As Legg and Hutter state, AIXI (Hutter 2004) has maximal intelligence by their measure. However, given a positive integer n, there exist an environment μ_n, based on a finite table of AIXI's possible behaviors during the first n time steps, and an agent π_n, such that μ_n gives AIXI reward 0 at each of those time steps and gives π_n reward 1 at each of those time steps. If most time weight occurs during the first n time steps and we apply Proposition 1 to μ_n (clearly resulting in a different PUTM than used to define AIXI), then π_n could have higher measured intelligence than AIXI (only possible because of the different PUTMs).

A Finite Model

Wang makes a convincing argument that finite and limited resources are an essential component of a definition of intelligence (Wang 1995). Lloyd estimates that the universe contains no more than 10^{90} bits of information and can have performed no more than 10^{120} elementary operations during its history (Lloyd 2002), in which case our universe is a finite state machine (FSM) with no more than $2^{(10^{90})}$ states. Adapting Legg's and Hutter's intelligence measure to a finite computing model would be consistent with finite physics, and can also address several of the issues listed in the previous section. Let's reject the notion that finite implies trivial on the grounds that the finite universe is not trivial.

As before, assume the sets A, O and R of actions, observations and rewards are finite and fixed. A FSM is defined by a mapping:

$$f: S(n) \times A \rightarrow S(n) \times O \times R$$

where $S(n)=\{1,2,3,\ldots,n\}$ is a set of states and "1" is the start state (we assume deterministic FSMs so this mapping is single-valued). Letting s_i denote the state at time step i, the timing is such that $f(s_i,a_i) = (s_{i+1},o_i,r_i)$. Because the agent π may be nondeterministic its value in this environment is defined by the expected value of rewards:

$$V_f^{\pi} = \mathbf{E}(\sum_{i=1}^{M(n)} w_{n,i} r_i)$$

where the $w_{n,i} \geq 0.0$ are a sequence of weights for future rewards subject to $\sum_{i=1}^{M(n)} w_{n,i} = 1$ and $M(n)$ is a finite time limit depending on state set size. Note that different state set sizes have different time weights, possibly giving agents more time to learn more complex environments.

Define $F(n)$ as the set of all FSMs with the state set $S(n)$. Define:

$$F = \bigcup_{n=L}^{H} F(n)$$

as the set of all FSMs with state set size between L and H. Define weights W_n such that $\sum_{n=L}^{H} W_n = 1$, and for $f \in F(n)$ define $W(f) = W_n / |F(n)|$. Then $\sum_{f \in F} W(f) = 1$ and we define the intelligence of agent π as:

$$V^{\pi} = \sum_{f \in F} W(f) V_f^{\pi}.$$

The lower limit L on state set size is intended to avoid domination of V^π by the value of π in a small number of environments, as in Proposition 1. The upper limit H on state size means that intelligence is determined by an agent's value in a finite number of environments. This avoids the necessity for weights to tend toward zero as environment complexity increases. In fact, the weights W_n may be chosen so that more complex environments actually have greater weight than simpler environments.

State is not directly observable so this model counts multiple FSMs with identical behavior. This can be regarded as implicitly weighting behaviors by counting numbers of representations.

No Free Lunch

The No-Free-Lunch Theorem (NFLT) tells us that all optimization algorithms have equal performance when averaged over all finite environments (Wolpert and Macready 1997). It is interesting to investigate what relation this result has to intelligence measures that average agent performance over environments.

The finite model in the previous section lacks an important hypothesis of the NFLT: that the optimization algorithm never makes the same action more than once. This is necessary to conclude that the ensembles of rewards are independent at different times. The following constraint on the finite model achieves the same result:

Definition. An environment FSM satisfies the No Repeating State Condition (NRSC) if it can never repeat the same state. Such environments must include one or more final states (successor undefined) and a criterion of the NRSC is that every path from the start state to a final state has length $\geq M(n)$, the time limit in the sum for V_f^π (this is only possible if $M(n) \leq n$).

Although the NRSC may seem somewhat artificial, it applies in the physical universe because of the second law of thermodynamics (under the reasonable assumption an irreversible process is always occurring somewhere). Now we show a No Free Lunch result for the finite model subject to the NRSC:

Proposition 2. In the finite model defined in the previous section, assume that $M(n) \leq n$ and restrict F to those FSMs satisfying the NRSC. Then for any agent π, $V^\pi = (\sum_{r\in R}\ r) / |R|$, the average reward. Thus all agents have the same measured intelligence.

Proof. Given an agent π, calculate:

$$V^\pi = \sum_{f\in F} W(f)\ V_f^\pi =$$

$$\sum_{n=L}^{H} \sum_{f\in F(n)} W(f)\ V_f^\pi =$$

$$\sum_{n=L}^{H} (W_n / |F(n)|) \sum_{f\in F(n)} V_f^\pi =$$

$$\sum_{n=L}^{H} (W_n / |F(n)|) \sum_{f\in F(n)} \mathbf{E}(\sum_{i=1}^{M(n)} w_{n,i}\ r_{f,i}) =$$

$$\sum_{n=L}^{H} (W_n / |F(n)|) \sum_{i=1}^{M(n)} w_{n,i} \sum_{f\in F(n)} \mathbf{E}(r_{f,i}).$$

where $r_{f,i}$ denotes the reward to the agent from environment f at time step i.

To analyze $\sum_{f\in F(n)}\ \mathbf{E}(r_{f,i})$, define $P(s,a|i,f)$ as the probability that in a time sequence of interactions between agent π and environment f, π makes action a and f is in state s at time step i. Also define $P(r|i,f)$ as the probability that f makes reward r at time step i. Note:

(1) $\sum_{a\in A} \sum_{s\in S} P(s,a|i,f) = 1$

Let f^R denote the R-component of a map f:$S(n)\times A\rightarrow S(n)\times O\times R$. For any $s\in S$ and $a\in A$, partition $F(n)$ into the disjoint union $F(n) = \bigcup_{r\in R} F(s,a,r)$ where $F(s,a,r) = \{\ f\in F(n) \mid f^R(s,a) = r\}$. Define a deterministic probability:

$$P(r|f,s,a) \quad = 1 \text{ if } f\in F(s,a,r)$$
$$= 0 \text{ otherwise.}$$

Given any two reward values $r_1,r_2\in R$ (here these do not denote the rewards at the first and second time steps) there is a one-to-one correspondence between $F(s,a,r_1)$ and $F(s,a,r_2)$ as follows: $f_1\in F(s,a,r_1)$ corresponds with $f_2\in F(s,a,r_2)$ if $f_1 = f_2$ everywhere except:

$$f_1^R(s,a) = r_1 \neq r_2 = f_2^R(s,a).$$

(Changing a reward value does not affect whether a FSM satisfies the NRSC.) Given such f_1 and f_2 in correspondence, because of the NRSC f_1 and f_2 can only be in state s once, and because they are in correspondence they will interact identically with the agent π before reaching state s. Thus:

(2) $P(s,a|i,f_1) = P(s,a|i,f_2)$

Because of the one-to-one correspondence between $F(s,a,r_1)$ and $F(s,a,r_2)$ for any $r_1,r_2\in R$, and because of equation (2), the value of $\sum_{f\in F(s,a,r)}\ P(s,a|i,f)$ is independent of r and we denote it by $Q(i,s,a)$. We use this and equation (1) as follows:

$$|F(n)| = \sum_{f\in F(n)} 1 =$$

$$\sum_{f\in F(n)} \sum_{a\in A} \sum_{s\in S} P(s,a|i,f) =$$

$$\sum_{a\in A} \sum_{s\in S} \sum_{f\in F(n)} P(s,a|i,f) =$$

$$\sum_{a\in A} \sum_{s\in S} \sum_{r\in R} \sum_{f\in F(s,a,r)} P(s,a|i,f) =$$

$$\sum_{a\in A} \sum_{s\in S} \sum_{r\in R} Q(i,s,a) =$$

$$\sum_{a\in A} \sum_{s\in S} |R|\ Q(i,s,a).$$

So for any $r\in R$:

(3) $\sum_{a\in A} \sum_{s\in S} \sum_{f\in F(s,a,r)} P(s,a|i,f) =$

$$\sum_{a\in A} \sum_{s\in S} Q(i,s,a) =$$

$$|F(n)| / |R|.$$

Now we are ready to evaluate $\sum_{f\in F(n)}\ \mathbf{E}(r_{f,i})$:

$$\sum_{f\in F(n)}\ \mathbf{E}(r_{f,i}) =$$

$\sum_{f\in F(n)} \sum_{r\in R} r\, P(r|i,f) =$

$\sum_{f\in F(n)} \sum_{r\in R} r \sum_{a\in A} \sum_{s\in S} P(r|f,s,a)\, P(s,a|i,f) =$

$\sum_{r\in R} r \sum_{a\in A} \sum_{s\in S} \sum_{f\in F(n)} P(r|f,s,a)\, P(s,a|i,f) =$

$\sum_{r\in R} r \sum_{a\in A} \sum_{s\in S} \sum_{f\in F(s,a,r)} P(s,a|i,f) =$ (by 3)

$\sum_{r\in R} r\, |F(n)| / |R| = |F(n)| \,(\sum_{r\in R} r) / |R|.$

Plugging this back into the expression for V^{π}:

$V^{\pi} = \sum_{n=L}^{H} (W_n / |F(n)|) \sum_{i=1}^{M(n)} w_{n,i} \sum_{f\in F(n)} \mathbf{E}(r_{f,i}) =$

$\sum_{n=L}^{H} (W_n / |F(n)|) \sum_{i=1}^{M(n)} w_{n,i}\, |F(n)| \,(\sum_{r\in R} r) / |R| =$

$\sum_{n=L}^{H} (W_n / |F(n)|)\, |F(n)| \,(\sum_{r\in R} r) / |R| =$

$(\sum_{r\in R} r) / |R|.$ □

By letting $L = H$ in the finite model, Proposition 2 applies to a distribution of environments defined by FSMs with the same state set size.

It would be interesting to construct a PUTM in Legg's and Hutter's model for which all agents have the same measured intelligence within an arbitrarily small ε. It is not difficult to construct a PUTM, somewhat similar to the one defined in the proof of Proposition 1, that gives equal weight to a set of programs defining all FSMs with state set size n satisfying the NRSC, and gives arbitrarily small weight to all other programs. The difficulty is that multiple FSMs will define the same behavior and only one of those FSMs will be counted toward agent intelligence, since Legg's and Hutter's measure sums over environment behaviors rather than over programs. But if their measure had summed over programs, then a PUTM could be constructed for which an analog of Proposition 2 could be proved.

A Revised Finite Model

According to current physics the universe is a FSM satisfying the NRSC. If we measure agent intelligence using a distribution of FSMs satisfying the NRSC in which all FSMs with the same number of states have the same weight, then Proposition 2 shows that all agents have the same measured intelligence. This is a distribution of environments in which past behavior of environments provides no information about their future behavior. For a useful measure of intelligence, environments must be weighted to enable agents to predict the future from the past.

It is easy to construct single environments against which different agents have different performance, so Proposition 1 implies that a weighting of environments based on program length is capable of defining different performance measures for different agents. However, we want to constrain an intelligence measure to ensure that it is based on performance against a large number of environments rather than a single environment.

This suggests a distribution of environments based on program length but less abstract than Kolmogorov complexity, in order to avoid a distribution of environments as constructed in the proof of Proposition 1. So revise the finite model of the previous sections to specify environments in an ordinary programming language, with static memory allocation and no recursion so environments are FSMs. Lower and upper limits on environment program length ensure that the model includes only a finite number of environments. For nondeterministic FSMs the language may include an oracle for truly random numbers.

Because the physical world satisfies the NRSC its behavior never repeats precisely (theoretically behavior could repeat precisely in the part of the universe sensed by a human, although in practice it doesn't). But human agents learn to predict future behavior in the world by recognizing current behavior as similar to previously observed behaviors, and making predictions based on those previous behaviors. Similarity can be recognized in sequences of values from unstructured sets such as {0, 1}, but there are more ways to recognize similarity in sequences of values from sets with metric and algebraic structures such as numerical sets. Our physical world is described largely by numerical variables, and the best human efforts to predict behaviors in the physical world use numerical programming languages.

So revise the finite model to define the sets A and O of actions and observations using numerical values (finitely sampled in the form of floating point or integer variables), just as rewards are taken from a numerical set R. Short environment programs that mix numerical and conditional operations will generally produce observations and rewards as piecewise continuous responses to agent actions, enabling agents to predict based on similarity of behaviors. Including primitives for numerical operations in environment programs has the effect of skewing the distribution of environments toward similarity with the physical world.

The revised finite model is a good candidate basis for a formal measure of intelligence. But the real point of this paper is that distributions over environments and time pose complex issues for formal intelligence measures. Ultimately our definition of intelligence depends on the intuition we develop from using our minds in the physical world, and the key to a useful formal measure is the way its weighting distribution over environments abstracts from our world.

References

Goertzel, B. Review of Past and Present AGI Research. Keynote address to *Artificial General Intelligence 2008*. http://www.agi-08.org/slides/goertzel.ppt

Hutter, M. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. Springer, Berlin. 2004. 300 pages.

Legg, S. and M. Hutter. Proc. A Formal Measure of Machine Intelligence. *15th Annual Machine Learning Conference of Belgium and The Netherlands (Benelearn 2006)*, pages 73-80.

http://www.idsia.ch/idsiareport/IDSIA-10-06.pdf

Li, M. and P. Vitányi, *An Introduction to Kolmogorov Complexity and Its Applications, 2nd ed.*. Springer, New York, 1997. 637 pages.

Lloyd, S. Computational Capacity of the Universe. Phys.Rev.Lett. 88 (2002) 237901.
http://arxiv.org/abs/quant-ph/0110141

Wang, P. Non-Axiomatic Reasoning System --- Exploring the essence of intelligence. PhD Dissertation, Indiana University Comp. Sci. Dept. and the Cog. Sci. Program, 1995.
http://www.cogsci.indiana.edu/farg/peiwang/
PUBLICATION/wang.thesis.ps

Wolpert, D. and W. Macready, No Free Lunch Theorems for Optimization. *IEEE Transactions on Evolutionary Computation* **1**, 67. 1997.
http://ic.arc.nasa.gov/people/dhw/papers/78.pdf