
Understanding Generalization Requires Universal Induction

Aram Eftekar
AIXI Labs
aebtekar@alumni.cmu.edu

Marcus Hutter
Google DeepMind & ANU
www.hutter1.net

Danica J. Sutherland
UBC & Amii
dsuth@cs.ubc.ca

Abstract

Classical statistical theory is insufficient to explain the successes of general-purpose AI models, because it depends on handcrafted inductive biases that it cannot justify. No Free Lunch theorems force any learner that beats chance on some environments to underperform on others. Moreover, lifting these theorems to meta-learning precludes using experience to learn “from scratch” which kinds of environments occur in practice. Thus, any method that makes meaningful predictions necessarily begins with an inductive bias not justified by data. Choosing to bias toward short programs yields Solomonoff induction (SI), which competes with all computable learners. Starting from such a generic prior, however, costs huge “constants” compared to specialized methods that exploit background information. We relativize SI to an *information vantage point*, granting its short programs access to all pre-existing code and data. This reframes the inductive bias: instead of seeking some absolute notion of simplicity, we favor *accessibility* with respect to our vantage point. The relativized SI is sample-optimal on finite data: any learner that outperforms it necessarily contains inaccessible information about the data. While SI is incomputable and hence not a practical algorithm, it provides a formal optimum for inference in the limit of infinite compute. We argue that algorithmic information theory, which underlies SI, is necessary to explain the generalization behavior of modern (and future) AI systems.

1 Introduction

While neural networks have come to dominate machine learning, much about their impressive generalization performance seems to defy analysis from classical statistical theory. Learning is about *inductive* inference; unlike *deductive* inferences which follow from strict logical rules, induction extrapolates from data to make predictions. Sometimes, we extrapolate general patterns that hold with such consistency that they feel like strict rules: a data source appears to sample i.i.d. (until one day the distribution changes); the correct interpolation of a set of points is a smooth curve (until one day it turns out to be discontinuous); the Sun rises each day (until one day its fusion fuel is depleted). These heuristics are not guaranteed to hold, and there are exceptions. While ad hoc heuristics suffice for simpler statistical problems – a particular density estimation problem might really be from a given Hölder class [Tsy09, Section 1.2] – a self-contained theory of learning that can describe general-purpose AI systems will need to derive its own inductive heuristics.

Ad hoc inductive biases are commonly justified empirically, by pointing to earlier instances in which they were successful. In other words, a part of the learning algorithm is itself inductively meta-learned from a lifetime of experience; in turn, our lifetime learning algorithm is informed by billions of years of Darwinian evolution [Gol+26]. Justifying the biases this way is circular, but suggests a promising case study: since it seems implausible that our microbial ancestors billions of years ago “knew” much, evolution may serve as a model of learning from a state of true ignorance.

Naive attempts to build a theory of learning from true ignorance run into No Free Lunch theorems. In statistical learning terms, we have no *a priori* way to guarantee low generalization error: we can only bound either the estimation *or* approximation error: For example, a model class for which uniform convergence holds guarantees low estimation error (that our algorithm does not substantially overfit), but then cannot guarantee low approximation error (that it does not underfit). Whether these explanations of estimation error even apply to practical neural networks remains unclear after much investigation; there are meaningful negative results [NTS15; Zha+17; NK19; Jia+20; Gas+24], but also some positive indications [DR17; Zho+19; Lot+24a; Lot+24b]. If they apply, why should these classes also have low approximation error? Alternatively, “soft-preference” techniques, such as structural risk minimization [VC74; Vap91] in a universal hypothesis class, can ensure zero approximation error [also see Wil25]. However, their estimation error is dependent on the choice of soft preference. Choosing a model class or a soft preference amounts to an inductive bias.

Despite No Free Lunch, it is an empirical fact that life and machines alike are often able to learn well. One might ask: why care about theory? Kawaguchi et al. [KKB22] identify three “practical roles” for generalization theory: (1) provide guarantees on expected risk, (2) provide guarantees on the generalization gap, and (3) provide insights to guide model selection. Learning theory addresses role (2) for “small” hypothesis classes, showing that test error is near training error. (The use of a validation set can, in some ways, make any class effectively small [cf. KKB22, Section 4].) As to roles (1) and (3), however, No Free Lunch theorems present extreme challenges. We can only say anything if we assume the function we are trying to learn lies in some “concept class.” Why should it? This is a challenge both for the philosophy of learning theory and for understanding what AI systems can, or should, do.

In Section 2, we expand on how No Free Lunch theorems preclude the possibility of learning without an inductive bias (and in fact, a meta-inductive bias, which cannot itself be learned via experience nor evolution [Wol23]). Theoretical explanations of learning are possible by assuming structure *a priori*; however, we now see generalist AI models deployed on increasingly arbitrary tasks. What sort of inductive bias would account for their generalization performance on new problems?

With the advent of the attention mechanism [Vas+17] and “reasoning” modes [ElK+24; Guo+25; Sne+25], leading model classes are rapidly moving toward supporting general, even universal forms of computation [PBM21; MS24; Yan+25; LW26]. Solomonoff induction is a logical extreme in this direction, corresponding to Bayesian inference with a weighted mixture prior over all possible computer programs [Sol64]. Due to the halting problem, Solomonoff induction is not computable, but it provides an idealization of learning in the limit of unbounded computation; it is a central object in algorithmic information theory [LV19; HQC24]. However, it necessarily chooses prior weights for each of the infinitely many possible computer programs; what justifies that prior?

Taking inspiration from Schurz [Sch19], we argue in Section 3 that while there may be no good prior in an absolute sense, there is a sample-optimal prior in the sense of regret against all *accessible* alternatives. This motivates an inductive bias in favor of accessibility to a given *information vantage point*, which Section 4 formalizes as a universal computer equipped with an oracle. We prove (Theorem 3) that oracle Solomonoff induction competes with every accessible predictor. While there are always predictors that do better on the particular data we end up seeing, they cannot have been accessible to us: our regret is bounded by the predictor’s (algorithmic) mutual information with the data, which is the extent to which it “knew the answer” in advance.

In Section 5, we discuss evidence supporting this point of view as a reasonable description of modern AI systems. Appendices A and B cover additional details of oracle Solomonoff induction, Appendix C discusses what some key results in statistical learning theory (do not) say in relation to No Free Lunch, and Appendix D gives implications for the problem of Boltzmann brains in cosmology. Overall, we argue that **to understand generalization in modern AI systems, it is necessary to invoke the conceptual toolkit of algorithmic information theory.**

2 No Free Lunch in learning theory

No Free Lunch theorems have an extensive history, dating back to Hume’s problem of induction [Hu1748; Put63; Goo83; Wol96; SL05; Ada+19; Bel20; SG21; Wol23]. Here, we focus on a straightforward formulation for finite input and output sets, whose proof is immediate by counting.

Proposition 1. Consider a function $f^* : \mathbb{Z}_n \rightarrow \mathbb{Z}_k$, and let $D_{\text{train}} \subset \mathbb{Z}_n$ be a set of $|D_{\text{train}}| = m < n$ observations. Then, there are exactly k^{n-m} functions $f : \mathbb{Z}_n \rightarrow \mathbb{Z}_k$ consistent with these observations, in the sense that $f(x) = f^*(x)$ for all $x \in D_{\text{train}}$. At every unobserved $x' \in \mathbb{Z}_n \setminus D_{\text{train}}$, for every $y' \in \mathbb{Z}_k$, exactly k^{n-m-1} of the functions consistent with f^* on D_{train} have $f(x') = y'$.

As a consequence of Proposition 1, no matter how a prediction algorithm uses the training dataset $(x, f^*(x))_{x \in D_{\text{train}}}$ to predict $f^*(x')$, it will fail for the vast majority of possible target functions f^* . To better appreciate the situation, let's analyze it from both a Bayesian and a frequentist perspective.

Bayesian and frequentist no free lunches To avoid inductive bias, suppose the Bayesian statistician begins with a uniform prior over all k^n possible functions. After observing the training dataset $(x, f^*(x))_{x \in D_{\text{train}}}$, the posterior belief is a uniform distribution on the k^{n-m} functions consistent with the data. No generalization is achieved, because the belief on non-training points $x' \notin D_{\text{train}}$ remains uniform, assigning probability $k^{n-m-1}/k^{n-m} = 1/k$ to each possible value of $f^*(x')$.

Similarly, suppose the frequentist statistician samples a random hypothesis \hat{f} uniformly¹ among the k^{n-m} functions that satisfy $\hat{f}(x) = f^*(x)$ for all $x \in D_{\text{train}}$, to test on a hold-out set $D_{\text{test}} \subset \mathbb{Z}_n \setminus D_{\text{train}}$. Since $\{\hat{f}(x) : x \in D_{\text{test}}\}$ is uniformly distributed, the test rejects \hat{f} with overwhelming probability. We can follow up by sampling new hypotheses to test, but if all we do is sample at random until we obtain satisfactory performance on D_{test} , the resulting hypothesis will fail to generalize, remaining uniformly distributed on all $x' \notin D_{\text{train}} \cup D_{\text{test}}$.

We note an important difference between the two methodologies. The Bayesian statistician sets an explicit inductive bias at the start, via a prior. Thereafter, Bayes' rule determines optimal posterior inferences. This makes Bayesian methods readily automatable, perhaps contributing to their historical popularity in AI research. On the other hand, the frequentist statistician delivers an inductive bias in an online fashion, providing guidance in the form of hypotheses to test one after another. This is more amenable to experimental science: when human scientists are unable to encode their beliefs into an explicit prior, they benefit from the ability to make a sequence of guesses.

In these naive forms, neither methodology is self-contained: they are formally agnostic to the question of which inductive bias is most appropriate, outsourcing this choice to human design. With a good Bayesian prior or a good method of hypothesis generation, much better inferences become possible.

A concrete example Let $n = k = 19$, so that we want to learn a function $f^* : \mathbb{Z}_{19} \rightarrow \mathbb{Z}_{19}$. We observe $f^*(4) = 4$, $f^*(8) = 8$, and $f^*(11) = 11$. What is $f^*(0)$? Without further constraints, it can be anything: by Proposition 1, there are 19^{16} functions compatible with the data, with exactly 19^{15} of them assigning each possible value to $f(0)$.

Nonetheless, it feels intuitive to guess the function $f_1(x) := x$, which predicts $f_1(0) = 0$. What makes f_1 a better guess than, say,

$$f_2(x) := x + (x - 4)(x - 8)(x - 11) = x^3 - 4x^2 + 13x + 9 \pmod{19},$$

which also matches the data but predicts $f_2(0) = 8$? We might think of f_2 as representing a weaker inductive bias, in the sense of being derived by fitting the data to a large hypothesis class

$$\mathcal{H}_0 := \{x \mapsto a_3x^3 + a_2x^2 + a_1x + a_0 \pmod{19} : a_i \in \mathbb{Z}_{19}\}.$$

\mathcal{H}_0 contains 19^4 elements, enough to *overfit* our small dataset. Hence, observing $f_2 = f^*$ on $D_{\text{train}} = \{4, 8, 11\}$ should not give us confidence that f_2 will continue to fit future data. Since f_1 belongs to the much smaller hypothesis class

$$\mathcal{H}_1 := \{x \mapsto a_1x + a_0 \pmod{19} : a_i \in \mathbb{Z}_{19}\},$$

we expect it not to drastically overfit. The issue with this analysis is that f_2 is a member of not only \mathcal{H}_0 , but also some smaller classes. In particular, the following has exactly the same size as \mathcal{H}_1 :

$$\mathcal{H}_2 := \{x \mapsto x^3 - 4x^2 + a_1x + a_0 \pmod{19} : a_i \in \mathbb{Z}_{19}\}.$$

¹Alternatively, one could choose an "arbitrary" consistent hypothesis, which might perform better (or worse) than the random choice. Typical frequentist analyses therefore assume either a random choice, a *worst-case* choice (naturally only worse), or perhaps some particular tie-breaking rule (which would correspond to a choice of some non-uniform inductive bias).

Choosing \mathcal{H}_2 may feel like “cheating,” since we constructed it by taking the leading terms $x^3 - 4x^2$ from f_2 , which depended on the particular observations on D_{train} . (A more extreme choice would be the singleton hypothesis class $\{f_2\}$, a common mistake made by students in beginning learning theory courses.) But, regardless of how we the authors selected it, what is it for you the reader that makes \mathcal{H}_2 feel “tailored” to the training data while \mathcal{H}_1 does not? If we remove our intuitive preference for f_1 and \mathcal{H}_1 , the situation is formally symmetric. If the true function is f_1 , then fitting to \mathcal{H}_1 performs well while fitting to \mathcal{H}_2 performs poorly; if the true function is f_2 , the reverse is true.

We visualize the space of possible functions in Figure 1. Learning theory [VC68; Val84; SB14; MRT18; Bac24], roughly, says that learning from a “small” hypothesis class (such as \mathcal{H}_1 or \mathcal{H}_2) probably yields hypotheses with a small generalization gap: their training performance is about the same as their test performance. Large classes, such as \mathcal{H}_0 , lack this property: when using them, we may overfit the training data. Small classes are vulnerable to a different problem: if they contain no good approximation to the true function, then we *underfit*, i.e. the training performance is itself poor. Indeed, \mathcal{H}_2 underfits most random samples (of three or more points) labeled by f_1 , and conversely \mathcal{H}_1 underfits most samples labeled by f_2 .

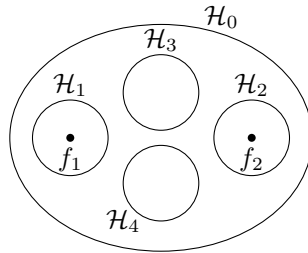


Figure 1: Both the “simple” function f_1 and the “complex” function f_2 belong to small hypothesis classes. The same can be said of any function in the large class \mathcal{H}_0 . To generalize well, we should choose a class that is both small and contains (a good approximation of) the true function.

Learning theory recommends choosing a small class to prevent overfitting, but does not tell us *which* small class prevents *underfitting*. No Free Lunch says that no algorithm generalizes well on most functions, so no hypothesis class simultaneously satisfies both constraints for most functions. We are forced to favor certain kinds of functions *a priori*. Why do we feel inclined to choose \mathcal{H}_1 over the myriad equally small classes such as \mathcal{H}_2 ? Low polynomial degree seems like a reasonable heuristic, but is not a general rule; it is disastrous for fitting other “simple” functions such as $f_3(x) := \lfloor x/10 \rfloor$.

Meta-learning an inductive bias Vitally, past experience cannot serve as a root justification for inductive biases: even if we (generously) assume we identify the full labeling function in each problem, meta-learning (learning to learn) is itself a learning problem [Wol23].

Corollary 2. *Consider a sequence of n' prediction problems, with the i th revealing a function $g_i : \mathbb{Z}_n \rightarrow \mathbb{Z}_k$. After observing g_1, \dots, g_m with $m < n'$, there remain $k^{(n'-m)n}$ consistent meta-mappings $i \mapsto g_i$, exactly $k^{(n'-m-1)n}$ of which predict each of the k^n possible choices for g_{m+1} .*

Proof. Uniquely label each function $g : \mathbb{Z}_n \rightarrow \mathbb{Z}_k$ by a number $\text{enc}(g) \in \mathbb{Z}_{k^n}$. Now consider the problem of learning the meta-function $f : \mathbb{Z}_{n'} \rightarrow \mathbb{Z}_{k^n}$ given by $f(i) := \text{enc}(g_{i+1})$. The result follows from applying Proposition 1 with $n, k, D_{\text{train}}, x'$ replaced by n', k^n, \mathbb{Z}_m, m , respectively. \square

As suggested in Section 1, let’s consider how Nature might learn $f^* = f_1$ in a toy model of Darwinian evolution. Imagine a population of creatures, each genetically wired to compute some function $f : \mathbb{Z}_{20} \rightarrow \mathbb{Z}_{20}$. For simplicity, we assume an infinite population, providing enough initial variation so that no mutations are needed. On each day, the creatures are tested on a single input x ; a creature f survives iff $f(x) = f_1(x) = x$. On the first three days, suppose the inputs are 4, 8, 11. Then, all surviving creatures satisfy $f(4) = 4, f(8) = 8, f(11) = 11$. Do the creatures adapt under selection? Specifically, given a new input on day four, is the survival rate better than chance?

The answer depends on the initial population. This setting is formally identical to our Bayesian analysis above, with natural selection taking the role of Bayesian updates, while the initial and

subsequent populations represent the prior and posterior beliefs, respectively. If the initial creatures are uniformly sampled from the set of all functions, then the generalization performance – the survival rate on day four – is only as good as chance. Without an explicit nonuniform bias at the outset, we can never learn one.

3 Accessibility as a universal inductive bias

In any realistic evolutionary setting, the initial distribution would not be uniform over all possible functions. In biology, perhaps even in the primordial soup where life began, molecules can arrange themselves into simple computing machines [Joh+22]. Thus, functions that can be computed by simple circuits, such as $f_1(x) := x$ or $f_3(x) := \lfloor x/10 \rfloor$, are sampled more frequently than others. This results in good generalization performance when testing on such functions.

The inductive bias toward simplicity is commonly known as Occam’s razor, after the medieval philosopher William of Ockham. Its mathematical formalization as Solomonoff induction was proposed by Solomonoff [Sol64], and improved by Levin [Lev71]. Rathmanner and Hutter [RH11] advocate for it as a “gold standard” for induction, although this view is challenged by McGregor [McG14], Herrmann [Her20], Neth [Net23], and Sterkenburg [Ste26].

One major issue is that when simplicity is measured by description length, it depends on the choice of description language. In particular, *any* prior distribution $\nu(x)$ can be interpreted as a simplicity bias: the self-delimiting code of length $\lceil \log \frac{1}{\nu(x)} \rceil$ is shortest at the peaks of ν . Even Solomonoff’s *universal* measures can be distributed arbitrarily on sequences of any prescribed finite length. Thus, “simplicity” appears to be a catch-all term for whatever heuristic we choose.

Instead of seeking simplicity in absolute terms, Schurz [Sch19] argues that reality forces a meta-inductive bias toward inductive methods that are *accessible* to us. To clarify his point, note that the source code of a learning algorithm is itself a kind of data. We cannot magically materialize the code for the best possible learner – perhaps one that hardcodes the unseen test set. Instead, we must derive a good learner using algorithms already in our possession, perhaps in our brain. The totality of information in our possession forms our *information vantage point*: if someone offers us a computer, a language, a heuristic, or a novel learning algorithm, we add it to our store of information. Programs that are short, derivable by running a short program, or derivable from a program already in our possession, are more accessible because we have better odds of randomly stumbling upon them.

Thus, among the k^n functions in Proposition 1, we only compete with a smaller set of accessible “experts” [CL06], representing all methods that we can realistically access. These include the simple “always predict 1,” as well as established methods such as “run an online AutoML system to do evolutionary hyperparameter search over neural network architectures optimized by stochastic gradient methods, and use the predictor with the best cross-validation score.” Even intelligently searching for better methods will, by definition, only ever yield accessible predictors. We will see that predicting according to an accessibility prior is competitive with every accessible predictor, and therefore near-optimal among all predictors that we can conceivably use.

For the primordial soup, the most accessible functions are those which occur most frequently as molecules are stochastically rearranged. The physical Church-Turing thesis implies these will be short programs in some language determined by the laws of Nature [Deu13; Özk15; JW18; KW20; Wol24; EH25]. As evolution selects for genetic information relevant to survival, future generations start from a richer information vantage point. Eventually, genetically modern humans appeared, whose information vantage point is further shaped by cultural development and life experience.

Informally, we measure the inaccessibility of a piece of information (i.e., a string) by the length of the shortest “generalized pointer” (i.e., another string) to it. For example, here are some ways to point to the contents of Shakespeare’s *Hamlet*: (1) give the contents verbatim; (2) give the contents in a compressed format that takes advantage of predictable structure, such as grammatical rules; (3) give the location of a hard copy at the local library; (4) instruct oneself to recite the contents from memory. Pointer (1) is the longest, (2) is shorter, (3) is shorter still if we have a good library, and (4) is the shortest for someone who has memorized *Hamlet*.

We will formalize the information vantage point as a universal computer U equipped with an oracle q , which can be queried for background information such as library books and open-source code. If U represents a standard home computer and q represents a snapshot of the World Wide Web, then a

short program p can download a Python interpreter, packages, open-source repositories, and datasets, and adapt them with a custom Python script, to make a state-of-the-art predictor.

In the spirit of No Free Lunch theorems, we should note that more information is not automatically better. The legendary Library of Babel [Bor41] is said to contain all possible 410-page books. This gigantic library is useless for the purposes of accessibility, since a set of directions pointing to one of its books would be as complex as the actual text – unless the library is designed to privilege certain books, e.g., placing *Hamlet* on an altar while leaving most other books on unremarkable shelves.

4 Solomonoff induction

We now develop the formal theory of fully general optimal online learning with respect to an information vantage point. Our results are self-contained, deviating from standard presentations only where needed to make the vantage points explicit.

4.1 Bayesian mixture predictors

To balance simplicity with generality, we focus on optimizing the log loss (also called cross-entropy) in the setting of online binary sequence prediction. We make no distributional assumptions: the sequence $x \in \mathbb{B}^*$ ($\mathbb{B} := \{0, 1\}$) may be chosen by an adversary. As a special case, chunks of x may be sampled i.i.d., perhaps from a continuously parametrized distribution [Hut04, Section 3.7.2].

We denote the length of x by $|x|$, its t th bit by x_t , and its first $t - 1$ bits by $x_{<t}$. At each step t , a predictor ν predicts a conditional semi-probability distribution $\nu(x_t \mid x_{<t})$, i.e. a function satisfying $\sum_{x'_t \in \{0,1\}} \nu(x'_t \mid x_{<t}) \leq 1$, with each term being non-negative. Equivalently, ν can be expressed as a Bayesian prior given by the sequential *semimeasure*² $\nu(x) := \prod_{t=1}^{|x|} \nu(x_t \mid x_{<t})$.

The log loss over the entire sequence $x \in \mathbb{B}^*$ is given by

$$L(\nu, x) := \log \frac{1}{\nu(x)} = - \sum_{t=1}^{|x|} \log \nu(x_t \mid x_{<t}). \quad (1)$$

Suppose we have a countable set of “expert” predictors $\{\nu_i\}_{i=1}^{\infty}$, with prior weights $w_i > 0$ satisfying $\sum_i w_i \leq 1$. The weights may represent an expert’s prior credibility or accessibility. We form the mixture-of-experts predictor

$$\xi(x) := \sum_{i=1}^{\infty} w_i \nu_i(x). \quad (2)$$

Since $\xi \geq w_i \nu_i$, the predictor ξ is competitive against every expert ν_i , its *regret* being given by

$$L(\xi, x) - L(\nu_i, x) = \log \frac{\nu_i(x)}{\xi(x)} \leq \log \frac{1}{w_i}.$$

This bound is independent of x , but limited by our set of experts.

Unfortunately, Putnam [Put63] showed that no computable predictor can achieve finite regret against *all* computable experts [see also Sol09, Section 3; Ste26]. This is because an adversary can always select x_t to be the bit that the predictor thinks is least likely, given $x_{<t}$. Since the predictor is computable, a computable expert can simulate it to anticipate x_t , thus outperforming the predictor. Intuitively, this means every computable predictor loses to an expert with a comparable compute budget, which in turn is included in a mixture predictor with a much higher compute budget. Thus, the joint optimization of runtime and sample efficiency necessarily presents a tradeoff [Sch02; Ven+11; FLH16; Nak21; Fin+26]. While its analysis seems difficult, we can already gain insights from an idealization [Wei07] that optimizes only sample efficiency, in the limit of infinite compute.

There are several known constructions of limit-computable predictors that dominate all computable experts [Hut04, Section 2.4.3; HM07]. They appear to have broadly similar properties, so for simplicity we focus on the original Solomonoff-Levin approach [Sol64; Lev71; Sol78]. Their universal mixture specializes (2) with a suitable choice of w_i and ν_i . To introduce it, we need some concepts from algorithmic information theory [LV19; HQC24].

²We use semimeasures to allow prediction by general programs, some of which get stuck in an infinite loop after sampling a partial sequence. Thus, there may be some probability that there is no next bit after $x_{<t}$ [LV19, Chapter 4; WH25].

4.2 Elements of algorithmic probability

Computers encode data and algorithms alike as binary strings. Given a finite string $x \in \mathbb{B}^*$, and a finite or infinite string $y \in \mathbb{B}^* \cup \mathbb{B}^\infty$, their concatenation $x_1x_2 \dots x_{|x|}y_1y_2 \dots$ is denoted xy , and we write $x \sqsubseteq xy$ to say that x is a prefix of xy . For $x, y \in \mathbb{B}^\infty$, their interleaving $x_1y_1x_2y_2 \dots$ is denoted $x \oplus y$. The infinite string of zeros is denoted $\mathbf{0} := 0^\infty$. Natural numbers $n \in \mathbb{N}$ are encoded as self-delimiting strings $\bar{n} \in \mathbb{B}^*$ with $|\bar{n}| = O(\log n)$, so that e.g., each pair $(n, x) \in \mathbb{N} \times \mathbb{B}^*$ is uniquely decodable from the string $\bar{n}x$.

We fix a universal reference computer U . If we want to run multiple computers, the other computers can be represented as interpreters that simulate them on U .

It will be convenient to take U to be a *monotone Turing machine* with four binary data tapes. It has two read-only one-way input tapes, one of whose contents we call an *oracle* and the other a *program*; a two-way read-write work tape; and a one-way write-only output tape.³ Conceptually, the oracle serves as a hardware abstraction layer over all accessible data: library books, websites, open-source algorithms, and interpreters all become uniformly available as queries against the tape. For simplicity, we treat its contents as static. The tapes are infinitely long; this makes it possible, for example, to package a randomized algorithm with its own infinite source of random bits.

Every cell of the work tape is initially set to 0. The output is taken to contain only the bits that are eventually printed, ignoring the output tape's unwritten suffix. Let $U^q(p) \in \mathbb{B}^* \cup \mathbb{B}^\infty$ denote the output of U , given the oracle $q \in \mathbb{B}^\infty$ and program $p \in \mathbb{B}^\infty$. Since the output tape is one-way, its limiting value is well-defined regardless of whether U halts.

We choose U to be *universal* in the sense that there exists an effective enumeration of all such monotone Turing machines $\{T_i\}_{i \in \mathbb{N}}$, such that for all $i \in \mathbb{N}$ and $p, q \in \mathbb{B}^\infty$,

$$U^q(\bar{i}p) = T_i^q(p). \quad (3)$$

Thus, we think of \bar{i} as an interpreter for T_i in the language of U . Note that if q includes the code for an interpreter, we can choose T_i to be a computer that runs the interpreter at a given address of its oracle tape; this allows \bar{i} to describe only an address instead of directly providing code.

We associate a semimeasure to each program $p \in \mathbb{B}^\infty$, and to each oracle $q \in \mathbb{B}^\infty$. For $x \in \mathbb{B}^* \cup \mathbb{B}^\infty$,

$$\begin{aligned} \mu_p(x) &:= \mathbb{P}_{\alpha \sim \lambda}(x \sqsubseteq U^\alpha(p)), \\ M^q(x) &:= \mathbb{P}_{\alpha \sim \lambda}(x \sqsubseteq U^q(\alpha)). \end{aligned}$$

Here, λ denotes the Lebesgue measure on \mathbb{B}^∞ , meaning that the bits of $\alpha \in \mathbb{B}^\infty$ are independent unbiased coin flips. Conceptually, μ_p is a semimeasure *programmed* by p : $\mu_p(x)$ gives the fraction of "random seeds" α for which the output of program p starts with x . Meanwhile, M^q is a *universal* semimeasure with *access* to q : $M^q(x)$ gives the fraction of random programs α whose output starts with x , given the ability to inspect q . The difference stems from the asymmetry between p and q in (3), which implies that for $i \in \mathbb{N}$,

$$\mu_{\bar{i}\mathbf{0}}(x) = \mathbb{P}_{\alpha \sim \lambda}(x \sqsubseteq U^\alpha(\bar{i}\mathbf{0})) = \mathbb{P}_{\alpha \sim \lambda}(x \sqsubseteq T_i^\alpha(\mathbf{0})).$$

Thus, $\{\mu_{\bar{i}\mathbf{0}}(x)\}_{i \in \mathbb{N}}$ enumerates the semimeasures sampled by monotone machines; this is exactly the set of lower-semicomputable semimeasures [LV19, Theorem 4.5.2], which properly includes all computationally feasible predictors.

In contrast, the Solomonoff-Levin semimeasure M^q has a positive probability of sampling from every machine T_i , and can therefore be viewed as a universal mixture [HQC24, Theorem 3.8.8]. Concretely, since $\mathbb{P}_{\alpha \sim \lambda}(\bar{i} \sqsubseteq \alpha) = 2^{-|\bar{i}|}$, we can take the mixture to be (2) with $w_i := 2^{-|\bar{i}|}$ and $\nu_i(x) := \mathbb{P}_{\alpha \sim \lambda}(x \sqsubseteq T_i^q(\alpha))$. In particular, setting $i = \iota_U = O(1)$ such that $T_{\iota_U}^q(p) := U^p(q)$, we obtain $\mu_{\bar{\iota}_U p}(x) = M^p(x) \geq 2^{-|\bar{\iota}_U|} \mu_p(x)$. We can write the inequality more compactly as

$$M^p(x) \stackrel{\times}{\geq} \mu_p(x), \quad (4)$$

where the relational operators $\stackrel{\times}{\geq}, \stackrel{\times}{\leq}$ hide a multiplicative factor that depends only on U (and $\stackrel{\times}{\geq}$ means that both $\stackrel{\times}{\leq}$ and $\stackrel{\times}{\leq}$ hold). Similarly, we use $\stackrel{\pm}{\leq}, \stackrel{\pm}{\geq}, \stackrel{\pm}{=}$ to hide an additive term that depends only on U .

³If the Turing machine formalism is unfamiliar, it will suffice to think of U as a computer that reads from two input streams and writes to one output stream. The streams ensure that we only consider inputs and outputs at non-negative addresses, and that outputs cannot be overwritten. The work tape functions as the working random-access memory (RAM); since inputs can be written to the work tape, they too are effectively random-access.

Since the hidden constants are universal, independent of specific learning algorithms or data, U can be chosen in such a way as to make all the constants in this paper small [cf. LV19, Section 3.9].

The information vantage point U^q can be thought of as a specialized computer, obtained by equipping the reference computer U with variable data q . Classical Solomonoff induction predicts according to the Bayesian prior M^0 , which may suffer high regret against a specialized expert that takes background information into account. By placing the background information in q , we will see that *oracle Solomonoff induction* using M^q has low regret against any expert that is accessible to us in practice. In particular, we never regret using the “wrong” universal computer: if we had access to a more suitable computer, it would be encoded within q .

4.3 Sample-optimal online learning with oracle Solomonoff induction

Shannon information theory studies the entropy and mutual information of random variables. It is readily applied to settings that sample repeatedly from stationary or ergodic distributions, for which aggregate codelengths concentrate near their expectations [CT06, Chapter 3].

Algorithmic information theory defines analogous quantities for individual samples without associated distributions [GV04; LV19]. Some argue this makes algorithmic information theory relevant to more general (e.g., non-ergodic) settings [Kol83; EH25, Section IV.B]. For our analysis, a useful analogue to the Shannon conditional entropy $H(X | Q)$ (for random variables Q, X) is the *a priori complexity* of $x \in \mathbb{B}^* \cup \mathbb{B}^\infty$ relative to $q \in \mathbb{B}^\infty$, given by

$$KM^q(x) := \log \frac{1}{M^q(x)}. \quad (5)$$

The logarithm’s base amounts to a choice of units [Fra05]. If taken in base 2, KM^q is expressed in bits, and is typically slightly less than the prefix Kolmogorov complexity [HQC24, Theorem 3.8.7].

In Shannon theory, the conditional mutual information is $I(P; X | Q) := H(X | Q) - H(X | P, Q)$ for random variables P, Q, X . Analogously, we define an asymmetric notion of algorithmic mutual information between $p \in \mathbb{B}^\infty$ and $x \in \mathbb{B}^*$, relative to $q \in \mathbb{B}^\infty$, by

$$MI^q(p : x) := KM^q(x) - KM^{p \oplus q}(x) = \log \frac{M^{p \oplus q}(x)}{M^q(x)}. \quad (6)$$

Recall that $p \oplus q$ is the interleaving of p and q , providing access to both strings. Other definitions of algorithmic mutual information appear in the literature [Gác21, Section 3.1; Ver21], but MI^q is simple and useful for our purposes. Appendix A establishes some intuitive properties: Proposition 4 shows that $0 \stackrel{\pm}{\leq} MI^q(p : x) \stackrel{\pm}{\leq} \min(KM^q(p), KM^q(x))$, and Theorem 5 proves that information cannot be created *ex nihilo*, neither by a deterministic nor a randomized process.

The following bound states that if a predictor μ_p substantially outperforms M^q , it must contain additional information about x that is not in q . Since information cannot be created, there is no way to invent such a predictor. It is in this sense that M^q is sample-optimal.

Theorem 3 (Regret of oracle Solomonoff induction). *For all $p, q \in \mathbb{B}^\infty$, the regret of a universal predictor M^q against another predictor μ_p , on any data $x \in \mathbb{B}^*$, is given by*

$$L(M^q, x) - L(\mu_p, x) \stackrel{\pm}{\leq} MI^q(p : x) - MI^p(q : x) \stackrel{\pm}{\leq} MI^q(p : x).$$

Proof. Using Equations (1) and (4),

$$L(M^q, x) - L(\mu_p, x) = \log \frac{\mu_p(x)}{M^q(x)} \stackrel{\pm}{\leq} \log \frac{M^p(x)}{M^q(x)} = \log \frac{M^{p \oplus q}(x)}{M^q(x)} - \log \frac{M^{p \oplus q}(x)}{M^p(x)}.$$

Using the definition (6), the first term is $MI^q(p : x)$. Since $M^{p \oplus q}(x) \cong M^{q \oplus p}(x)$, the second term is $\pm -MI^p(q : x)$, which is $\stackrel{\pm}{\leq} 0$ by Proposition 4. \square

Combining Theorem 3 and Proposition 4 yields the weaker bound $L(M^q, x) - L(\mu_p, x) \stackrel{\pm}{\leq} KM^q(p)$, which generalizes Corollary 3.8.10 of [HQC24]. Theorem 3 goes further by clarifying what type of information is relevant to the regret. For example, suppose $p = r\alpha$ consists of source code r for a randomized algorithm, along with an infinite source of random bits α , neither of which is tailored to the data x . Then $KM^q(p)$ is infinite, while $MI^q(p : x)$ is small.

Although oracle Solomonoff induction is formally Bayesian, Theorem 3 provides a distribution-free frequentist guarantee. This suggests an *interpretation* of probabilities [Gil00; HH16, Part IV]: no matter what x the world presents, a sample-optimal learner may model its subjective uncertainty using M^q [RH11]. Our position does not take a side in the Bayesian-frequentist divide [May18; SH19], but may help illuminate why it exists. Given the framing from Section 2, where Bayesian inference fixes its inductive bias up front and frequentist inference accumulates it online, an idealized learner with unlimited resources can afford to commit to a universal prior, and is naturally Bayesian. A resource-bounded learner cannot, so its inductive bias must accumulate over time. Any realistic attempt to approximate M^q would find a never-ending sequence of candidate programs; therefore, we can expect practical methods to acquire a mix of Bayesian and frequentist character.

5 Neural network generalization

Some of the most famous results in deep learning are “universal representation theorems,” which say that neural networks can approximate arbitrary functions [Cyb89; Hor91; KL20]. Referring back to Figure 1, this is like saying neural networks are a large class; they will fit the training data, but uniform convergence will not apply. While this is frequently brought up as “the reason” that neural networks work well in practice, this is misleading: equivalent universal approximation results are shared by Gaussian kernel spaces [SC08, Section 4.6] and even histograms [RF10, Section 7.4].

In order to generalize well, what we actually need is a universal inductive bias – ideally one that approaches something like M^q in a limit of large model size and compute. There are some indications that modern models may correspond to a simplicity bias in universal computing terms. Such arguments have recently been made to explain various empirical results in practical networks [Del+24; Gol+24; Huh+24; Hua+24], as well as in toy models [RS24].

Why does this happen? In addition to the various mechanisms suggested by the previous papers, Buzaglo et al. [Buz+24] prove that wide feedforward networks have disproportionately large regions in parameter space corresponding to functions also computed by much narrower networks. Thus, a random search for interpolating solutions is more likely to correspond to a narrow network, supporting an earlier empirical finding that random search generalizes well [Chi+23]. This is not quite a bias toward short programs, but since small feedforward networks can directly encode small binary circuits [Par94], it is closely related. Adding recurrence and an external memory turns circuits into universal Turing machines; transformers with chain of thought move toward that [PBM21; MS24; Yan+25; LW26]. (Whether the parameter space bias extends to favoring short programs, and what consequences that has for practical optimization methods, remain unclear.)

In another direction, Grau-Moya et al. [Gra+24] generate data from random programs with a computation time bound, in order to train a neural network to approximate Solomonoff induction in-context. Of course, due to No Free Lunch, such an approach can only generalize to unseen inputs if the learner already has a built-in pseudo-universal bias. Since their approach *does* generalize across domains, it seems the built-in bias is close enough to learn a reasonable approximation to M^q . (Shaw et al. [Sha+26] have similar aims in relation to the Minimum Description Length principle [Wal05; GR19].)

In a related vein, some have argued that trained large language models behave similarly in-context to Bayesian inference [Xie+22; WS25] or even specifically Solomonoff induction [WM25]. While the in-context claim is separate from our main point about generalization from training, if true it further supports that the inductive bias of these models is somewhat aligned with M^q .

6 Discussion

Goldblum et al. [Gol+24] recently argued that algorithmic information theory explains how to beat No Free Lunch, because “low-complexity structure shared by real-world datasets and machine learning models enables broad generalization across domains and sample sizes with a single model class.” We do not disagree, but we ask: low complexity in terms of *what*, and *why* [Ste26]? Any argument based on empirical success – as theirs is – reverts back to the problem of meta-induction as in Corollary 2. Our accessibility framing provides a crisper account of optimal learning: while we cannot be sure that M^q will predict accurately, Theorem 3 guarantees that no conceivable alternative does better.

In itself, this choice brings prescriptive value: generalist AI systems should probably aim to look something like M^q . It is also of descriptive value: there is reason to think that current systems already approximate M^q to some extent. Both directions help us understand what our methods *should* do in challenging settings, such as out-of-distribution generalization and in-context learning. Models of agency based on optimal induction, such as AIXI, can help us anticipate future agentic capabilities and behaviors, and identify safety risks before they occur [HQC24, Chapter 15; Meu+25; EC26].

To the extent that real models fail to match M^q – due to computational bounds or other differences – we expect that M^q will serve as a useful baseline from which to understand any deviations. Algorithmic information theory is a mathematical discipline replete with concepts relevant to data and learning, accounting for factors such as accessibility, model size, time bounds, and even novelty [FLH16; VS16; LV19; Zaf25; EWJ25; Fin+26; EC26]. We believe its applications are severely underexplored, and its relevance grows rapidly with advancing AI capabilities.

Acknowledgments and Disclosure of Funding

The authors would like to thank Cole Wyeth for particularly helpful feedback, as well as David Wolpert, David Quarel, Tom Sterkenburg, Daniel Herrmann, Francesca Zaffora Blando, Nathan Srebro, Henrik Marklund, Alex Infanger, and Lily Stelling for productive conversations.

This work was supported in part by the Canada CIFAR AI Chairs program and the AI Safety Tactical Opportunities Fund (AISTOF).

References

- [Ada+19] Stavros P. Adam, Stamatios-Aggelos N. Alexandropoulos, Panos M. Pardalos, and Michael N. Vrahatis. “No Free Lunch Theorem: A Review.” *Approximation and Optimization: Algorithms, Complexity and Applications*. Springer, 2019, pages 57–82.
- [Alq24] Pierre Alquier. “User-friendly Introduction to PAC-Bayes Bounds.” *Foundations and Trends® in Machine Learning* 17.2 (Jan. 2024), pages 174–303. arXiv: 2110.11216 [stat.ML].
- [Bac24] Francis Bach. *Learning Theory from First Principles*. MIT Press, 2024. URL: https://www.di.ens.fr/~fbach/ltfp_book.pdf.
- [BE02] Olivier Bousquet and André Elisseeff. “Stability and Generalization.” *Journal of Machine Learning Research* 2 (2002), pages 499–526.
- [Bel20] Gordon Belot. “Absolutely no free lunches!” *Theoretical Computer Science* 845 (2020), pages 159–180. arXiv: 2005.04791 [cs.LG].
- [Ben82] Charles H Bennett. “The thermodynamics of computation—a review.” *International Journal of Theoretical Physics* 21 (1982), pages 905–940.
- [BKZ20] Olivier Bousquet, Yegor Klochkov, and Nikita Zhivotovskiy. “Sharper Bounds for Uniformly Stable Algorithms.” *Conference on Learning Theory*. 2020.
- [Bor41] Jorge Luis Borges. “La Biblioteca de Babel.” *El Jardín de Senderos que se Bifurcan*. Buenos Aires: Editorial Sur, 1941. Translated as “The Library of Babel.” *Collected Fictions*. Translated by Andrew Hurley. London: Penguin Books, 1998.
- [BT87] John D. Barrow and Frank J. Tipler. *The Anthropic Cosmological Principle*. New York, NY, USA: Oxford University Press, 1987.
- [Buz+24] Gon Buzaglo, Itamar Harel, Mor Shpigel Nacson, Alon Brutzkus, Nathan Srebro, and Daniel Soudry. “How Uniform Random Weights Induce Non-uniform Bias: Typical Interpolating Neural Networks Generalize with Narrow Teachers.” *International Conference on Machine Learning*. 2024, pages 5035–5081. arXiv: 2402.06323 [cs.LG].
- [Car20] Sean M. Carroll. “Why Boltzmann Brains Are Bad.” *Current Controversies in Philosophy of Science*. Edited by Shamik Dasgupta and Brad Weslake. Routledge, 2020, pages 7–20. arXiv: 1702.00850 [hep-th].
- [Chi+23] Ping-yeh Chiang, Renkun Ni, David Yu Miller, Arpit Bansal, Jonas Geiping, Micah Goldblum, and Tom Goldstein. “Loss Landscapes are All You Need: Neural Network Generalization Can Be Explained Without the Implicit Bias of Gradient Descent.” *International Conference on Learning Representations*. 2023.

- [CL06] Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge: Cambridge University Press, 2006.
- [Coh+26] Alon Cohen, Liad Erez, Steve Hanneke, Tomer Koren, Yishay Mansour, Shay Moran, and Qian Zhang. “Sample Complexity of Agnostic Multiclass Classification: Natarajan Dimension Strikes Back.” *ACM Symposium on Theory of Computing*. 2026. arXiv: 2511.12659 [cs.LG].
- [CT06] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. 2nd edition. Hoboken, NJ, USA: Wiley-Interscience, 2006.
- [Cyb89] George Cybenko. “Approximation by Superpositions of a Sigmoidal Function.” *Mathematics of Control, Signals and Systems* 2.4 (1989), pages 303–314.
- [Del+24] Grégoire Delétang, Anian Ruoss, Paul-Ambroise Duquenne, Elliot Catt, Tim Genewein, Christopher Mattern, Jordi Grau-Moya, Li Kevin Wenliang, Matthew Aitchison, Laurent Orseau, Marcus Hutter, and Joel Veness. “Language Modeling Is Compression.” *International Conference on Learning Representations*. 2024. arXiv: 2309.10668 [cs.LG].
- [Deu13] David Deutsch. “What is Computation? (How) Does Nature Compute?” *A Computable Universe: Understanding and Exploring Nature as Computation*. Edited by Hector Zenil. World Scientific, 2013, pages 551–565.
- [DH10] Rodney G. Downey and Denis R. Hirschfeldt. *Algorithmic Randomness and Complexity*. Springer, 2010.
- [DR17] Gintare Karolina Dziugaite and Daniel M. Roy. “Computing Nonvacuous Generalization Bounds for Deep (Stochastic) Neural Networks with Many More Parameters than Training Data.” *Conference on Uncertainty in Artificial Intelligence*. 2017. arXiv: 1703.11008 [cs.LG].
- [DS14] Amit Daniely and Shai Shalev-Shwartz. “Optimal Learners for Multiclass Problems.” *Conference on Learning Theory*. 2014. arXiv: 1405.2420 [cs.LG].
- [EC26] Aram Eftekar and Michael K. Cohen. “Golden Handcuffs make safer AI agents” (2026). arXiv: 2604.13609 [cs.LG].
- [EEP05] Andre Elisseeff, Theodoros Evgeniou, and Massimiliano Pontil. “Stability of Randomized Learning Algorithms.” *Journal of Machine Learning Research* 6.3 (2005), pages 55–79.
- [EH25] Aram Eftekar and Marcus Hutter. “Foundations of Algorithmic Thermodynamics.” *Physical Review E* 111.1 (2025), page 014118. arXiv: 2308.06927 [cond-mat.stat-mech].
- [ElK+24] Ahmed El-Kishky, Daniel Selsam, Francis Song, Giambattista Parascandolo, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ilge Akkaya, Ilya Sutskever, Jason Wei, Jonathan Gordon, Karl Cobbe, Kevin Yu, Lukas Kondraciuk, Max Schwarzer, Mostafa Rohaninejad, Noam Brown, Shengjia Zhao, Trapit Bansal, Vineet Kosaraju, Wenda Zhou, et al. *Learning to Reason with LLMs*. Sept. 12, 2024. URL: <https://openai.com/index/learning-to-reason-with-llms/>.
- [EWJ25] Aram Eftekar, Yuhao Wang, and Dominik Janzing. “Toward universal laws of outlier propagation.” *Proceedings of the Conference on Uncertainty in Artificial Intelligence*. 2025, pages 1167–1183. arXiv: 2502.08593 [cs.LG].
- [Fin+26] Marc Finzi, Shikai Qiu, Yiding Jiang, Pavel Izmailov, J. Zico Kolter, and Andrew Gordon Wilson. “From Entropy to Epiplexity: Rethinking Information for Computationally Bounded Intelligence” (2026). arXiv: 2601.03220 [cs.LG].
- [FLH16] Daniel Filan, Jan Leike, and Marcus Hutter. “Loss Bounds and Time Complexity for Speed Priors.” *International Conference on Artificial Intelligence and Statistics*. PMLR. 2016, pages 1394–1402. arXiv: 1604.03343 [cs.LG].
- [Fra05] Michael P Frank. “The indefinite logarithm, logarithmic units, and the nature of entropy” (2005). arXiv: physics/0506128 [physics].
- [Gác21] Peter Gács. “Lecture Notes on Descriptive Complexity and Randomness” (2021). arXiv: 2105.04704 [cs.IT].
- [Gas+24] Michael Gastpar, Ido Nachum, Jonathan Shafer, and Thomas Weinberger. “Fantastic Generalization Measures Are Nowhere to Be Found.” *International Conference on Learning Representations*. 2024. arXiv: 2309.13658 [cs.LG].

- [Gil00] Donald Gillies. *Philosophical Theories of Probability*. Routledge, 2000.
- [Gol+24] Micah Goldblum, Marc Finzi, Keefer Rowan, and Andrew Gordon Wilson. “Position: The No Free Lunch Theorem, Kolmogorov Complexity, and the Role of Inductive Biases in Machine Learning.” *International Conference on Machine Learning*. Position Paper Track. 2024. arXiv: 2304.05366 [cs.LG].
- [Gol+26] Judah Goldfeder, Philippe Wyder, Yann LeCun, and Ravid Shwartz Ziv. “AI Must Embrace Specialization via Superhuman Adaptable Intelligence” (2026). arXiv: 2602.23643 [cs.AI].
- [Goo83] Nelson Goodman. *Fact, Fiction, and Forecast*. 4th edition. Harvard University Press, 1983.
- [GR19] Peter Grünwald and Teemu Roos. “Minimum description length revisited.” *International Journal of Mathematics for Industry* 11.01 (2019), page 1930001. arXiv: 1908.08484 [stat.ME].
- [Gra+24] Jordi Grau-Moya, Tim Genewein, Marcus Hutter, Laurent Orseau, Grégoire Delétang, Elliot Catt, Anian Ruoss, Li Kevin Wenliang, Christopher Mattern, Matthew Aitchison, and Joel Veness. “Learning Universal Predictors” (2024). arXiv: 2401.14953 [cs.LG].
- [Guo+25] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, et al. “DeepSeek-R1 Incentivizes Reasoning in LLMs through Reinforcement Learning.” *Nature* 645.8081 (Sept. 2025), pages 633–638. arXiv: 2501.12948 [cs.CL].
- [GV04] Peter Grünwald and Paul M. B. Vitányi. “Shannon Information and Kolmogorov Complexity” (2004). arXiv: cs/0410002 [cs.IT].
- [Her20] Daniel A. Herrmann. “PAC Learning and Occam’s Razor: Probably Approximately Incorrect.” *Philosophy of Science* 87.4 (2020), pages 685–703.
- [HH16] Alan Hájek and Christopher Hitchcock. *The Oxford Handbook of Probability and Philosophy*. Oxford University Press, 2016.
- [HM07] Marcus Hutter and Andrej Muchnik. “On semimeasures predicting Martin-Löf random sequences.” *Theoretical Computer Science* 382.3 (2007), pages 247–261. arXiv: 0708.2319 [cs.IT].
- [Hor91] Kurt Hornik. “Approximation Capabilities of Multilayer Feedforward Networks.” *Neural Networks* 4.2 (1991), pages 251–257.
- [HQC24] Marcus Hutter, David Quarel, and Elliot Catt. *An Introduction to Universal Artificial Intelligence*. Chapman and Hall/CRC, 2024.
- [Hu1748] David Hume. *An Enquiry Concerning Human Understanding*. London: A. Millar, 1748. Reprinted as Tom L. Beauchamp, editor. *An Enquiry Concerning Human Understanding*. Oxford Philosophical Texts. Oxford: Oxford University Press, 2007.
- [Hua+24] Yuzhen Huang, Jinghan Zhang, Zifei Shan, and Junxian He. “Compression Represents Intelligence Linearly.” *Conference on Language Modeling*. 2024. arXiv: 2404.09937 [cs.CL].
- [Huh+24] Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. “The Platonic Representation Hypothesis.” *International Conference on Machine Learning*. Position Paper Track. 2024. arXiv: 2405.07987 [cs.LG].
- [Hut04] Marcus Hutter. *Universal Artificial Intelligence: Sequential Decisions Based on Algorithmic Probability*. Springer, 2004.
- [Hut10] Marcus Hutter. “A Complete Theory of Everything (will be subjective).” *Algorithms* 3.4 (2010), pages 329–350. arXiv: 0912.5434 [cs.IT].
- [Jia+20] Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. “Fantastic Generalization Measures and Where to Find Them.” *International Conference on Learning Representations*. 2020. arXiv: 1912.02178 [cs.LG].
- [Joh+22] Iain G. Johnston, Kamaludin Dingle, Sam F. Greenbury, Chico Q. Camargo, Jonathan P. K. Doye, Sebastian E. Ahnert, and Ard A. Louis. “Symmetry and simplicity spontaneously emerge from the algorithmic nature of evolution.” *Proceedings of the National Academy of Sciences* 119.11 (2022), e2113883119.

- [JW18] Dominik Janzing and Paweł Wocjan. “Does Universal Controllability of Physical Systems Prohibit Thermodynamic Cycles?” *Open Systems & Information Dynamics* 25.03 (2018), page 1850016. arXiv: 1701.01591 [cond-mat.stat-mech].
- [KKB22] Kenji Kawaguchi, Leslie Pack Kaelbling, and Yoshua Bengio. “Generalization in Deep Learning.” *Mathematical Aspects of Deep Learning*. Cambridge University Press, 2022. arXiv: 1710.05468 [stat.ML].
- [KL20] Patrick Kidger and Terry Lyons. “Universal Approximation with Deep Narrow Networks.” *Conference on Learning Theory*. 2020. arXiv: 1905.08539 [cs.LG].
- [Kol06] Vladimir Koltchinskii. “Local Rademacher Complexities and Oracle Inequalities in Risk Minimization.” *The Annals of Statistics* 34.6 (2006), pages 2593–2656. arXiv: 0708.0083 [math.ST].
- [Kol83] Andrey N. Kolmogorov. “Combinatorial foundations of information theory and the calculus of probabilities.” *Russian Mathematical Surveys* 38.4 (1983), page 29.
- [KW20] Artemy Kolchinsky and David H. Wolpert. “Thermodynamic Costs of Turing Machines.” *Physical Review Research* 2.3 (2020), page 033312. arXiv: 1912.04685 [cond-mat.stat-mech].
- [Lev71] Leonid A. Levin. PhD thesis. Moscow State University, 1971. Translated as “Some theorems on the algorithmic approach to probability theory and information theory.” *Annals of Pure and Applied Logic* 162.3 (2010), pages 224–235.
- [Lev84] Leonid A. Levin. “Randomness Conservation Inequalities; Information and Independence in Mathematical Theories.” *Information and Control* 61.1 (1984), pages 15–37.
- [Lot+24a] Sanae Lotfi, Marc Anton Finzi, Yilun Kuang, Tim G. J. Rudner, Micah Goldblum, and Andrew Gordon Wilson. “Non-Vacuous Generalization Bounds for Large Language Models.” *International Conference on Machine Learning*. 2024, pages 32801–32818. arXiv: 2312.17173 [stat.ML].
- [Lot+24b] Sanae Lotfi, Yilun Kuang, Marc Anton Finzi, Brandon Amos, Micah Goldblum, and Andrew Gordon Wilson. “Unlocking Tokens as Data Points for Generalization Bounds on Larger Language Models.” *Advances in Neural Information Processing Systems*. 2024. arXiv: 2407.18158 [stat.ML].
- [LV19] Ming Li and Paul Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. 4th edition. Springer, 2019.
- [LW26] Qian Li and Yuyi Wang. “Constant Bit-size Transformers Are Turing Complete.” *Advances in Neural Information Processing Systems*. 2026. arXiv: 2506.12027 [cs.CC].
- [May18] Deborah G. Mayo. *Statistical Inference as Severe Testing: How to Get Beyond the Statistics Wars*. Cambridge University Press, 2018.
- [McG14] Simon McGregor. “Natural Descriptions and Anthropic Bias: Extant Problems in Solomonoff Induction.” *Conference on Computability in Europe*. Springer. 2014, pages 293–302.
- [Men21] Shahar Mendelson. “Extending the Scope of the Small-Ball Method.” *Studia Mathematica* 256 (2021), pages 147–167. arXiv: 1709.00843 [stat.ML].
- [Meu+25] Alexander Meulemans, Rajai Nasser, Maciej Wołczyk, Marissa A. Weis, Seijin Kobayashi, Blake Richards, Guillaume Lajoie, Angelika Steger, Marcus Hutter, James Manyika, Rif A. Saurous, João Sacramento, and Blaise Agüera y Arcas. “Embedded Universal Predictive Intelligence: A Coherent Framework for Multi-Agent Learning” (2025). arXiv: 2511.22226 [cs.AI].
- [MRT18] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. 2nd edition. MIT Press, 2018. URL: <https://cs.nyu.edu/~mohri/mlbook/>.
- [MS24] William Merrill and Ashish Sabharwal. “The Expressive Power of Transformers with Chain of Thought.” *International Conference on Learning Representations*. 2024. arXiv: 2310.07923 [cs.LG].
- [Mül20] Markus P. Müller. “Law without law: from observer states to physics via algorithmic information theory.” *Quantum* 4 (2020), page 301. arXiv: 1712.01826 [quant-ph].
- [Mül26] Markus P. Müller. “Algorithmic idealism: what should you believe to experience next?” *Foundations of Physics* 56.1 (2026), page 11. arXiv: 2412.02826 [physics.hist-ph].

- [Nak21] Preetum Nakkiran. “Turing-Universal Learners with Optimal Scaling Laws” (2021). arXiv: 2111.05321 [cs.LG].
- [Nat89a] Balas K. Natarajan. “On Learning Sets and Functions.” *Machine Learning* 4 (1989), pages 67–97.
- [Nat89b] Balas K. Natarajan. *Some Results on Learning*. Technical report CMU-RI-TR-89-06. Pittsburgh, PA: Carnegie Mellon University, Feb. 1989. URL: <https://publications.ri.cmu.edu/some-results-on-learning>.
- [Net23] Sven Neth. “A Dilemma for Solomonoff Prediction.” *Philosophy of Science* 90.2 (2023), pages 288–306. arXiv: 2206.06473 [cs.AI].
- [Nie09] André Nies. *Computability and Randomness*. Volume 51. Oxford University Press, 2009.
- [NK19] Vaishnavh Nagarajan and J. Zico Kolter. “Uniform Convergence May Be Unable to Explain Generalization in Deep Learning.” *Advances in Neural Information Processing Systems*. 2019. arXiv: 1902.04742 [cs.LG].
- [NTS15] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. “In Search of the Real Inductive Bias: On the Role of Implicit Regularization in Deep Learning.” *International Conference on Learning Representations*. Workshop Track. 2015. arXiv: 1412.6614 [cs.LG].
- [Özk15] Eray Özkural. “Ultimate Intelligence Part I: Physical Completeness and Objectivity of Induction.” *International Conference on Artificial General Intelligence*. Springer. 2015, pages 131–141.
- [Par94] Ian Parberry. *Circuit Complexity and Neural Networks*. MIT Press, 1994.
- [PBM21] Jorge Pérez, Pablo Barceló, and Javier Marinkovic. “Attention is Turing-Complete.” *Journal of Machine Learning Research* 22.75 (2021), pages 1–35.
- [Put63] Hilary Putnam. “‘Degree of Confirmation’ and Inductive Logic.” *The Philosophy of Rudolf Carnap*. Edited by Paul Arthur Schilpp. Open Court, 1963.
- [RF10] H. L. Royden and P. M. Fitzpatrick. *Real Analysis*. 4th edition. Prentice Hall, 2010.
- [RH11] Samuel Rathmanner and Marcus Hutter. “A Philosophical Treatise of Universal Induction.” *Entropy* 13.6 (2011), pages 1076–1136. arXiv: 1105.5721 [cs.LG].
- [RS24] Yi Ren and Danica J. Sutherland. *Understanding Simplicity Bias towards Compositional Mappings via Learning Dynamics*. 2024. arXiv: 2409.09626 [cs.LG].
- [SB14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014. URL: <https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/copy.html>.
- [SC08] Ingo Steinwart and Andreas Christmann. *Support Vector Machines*. Springer, 2008. DOI: 10.1007/978-0-387-77242-4.
- [Sch02] Jürgen Schmidhuber. “The Speed Prior: A New Simplicity Measure Yielding Near-Optimal Computable Predictions.” *Conference on Computational Learning Theory*. Springer. 2002, pages 216–228.
- [Sch19] Gerhard Schurz. *Hume’s Problem Solved: The Optimality of Meta-Induction*. MIT Press, 2019.
- [SG21] Tom F. Sterkenburg and Peter D. Grünwald. “The no-free-lunch theorems of supervised learning.” *Synthese* 199.3 (2021), pages 9979–10015. arXiv: 2202.04513 [cs.LG].
- [SH19] Jan Sprenger and Stephan Hartmann. *Bayesian Philosophy of Science*. Oxford University Press, 2019.
- [Sha+10] Shai Shalev-Shwartz, Ohad Shamir, Nathan Srebro, and Karthik Sridharan. “Learnability, Stability and Uniform Convergence.” *Journal of Machine Learning Research* 11 (2010), pages 2635–2670.
- [Sha+26] Peter Shaw, James Cohan, Jacob Eisenstein, and Kristina Toutanova. “Bridging Kolmogorov Complexity and Deep Learning: Asymptotically Optimal Description Length Objectives for Transformers.” *International Conference on Learning Representations*. 2026. arXiv: 2509.22445 [cs.LG].
- [Sim+10] Andrea De Simone, Alan H. Guth, Andrei Linde, Mahdiyar Noorbala, Michael P. Salem, and Alexander Vilenkin. “Boltzmann Brains and the Scale-Factor Cutoff Measure of the Multiverse.” *Physical Review D* 82.6 (2010), page 063520. arXiv: 0808.3778 [hep-th].

- [SL05] Steven A. Sloman and David Lagnado. “The Problem of Induction.” *The Cambridge Handbook of Thinking and Reasoning*. Edited by Keith J. Holyoak and Robert G. Morrison. Cambridge University Press, 2005, pages 95–116.
- [Sne+25] Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. “Scaling LLM Test-Time Compute Optimally Can Be More Effective than Scaling Parameters for Reasoning.” *International Conference on Learning Representations*. 2025. arXiv: 2408.03314 [cs.LG].
- [Sol09] Ray J. Solomonoff. “Algorithmic Probability: Theory and Applications.” *Information Theory and Statistical Learning*. Edited by Frank Emmert-Streib and Matthias Dehmer. Berlin/Heidelberg, Germany: Springer, 2009, pages 1–23.
- [Sol64] Ray J. Solomonoff. “A Formal Theory of Inductive Inference. Parts I and II.” *Information and Control* 7 (1964), 1–22 and 224–254.
- [Sol78] Ray J. Solomonoff. “Complexity-based induction systems: comparisons and convergence theorems.” *IEEE Transactions on Information Theory* 24.4 (1978), pages 422–432.
- [Ste26] Tom F. Sterkenburg. “Solomonoff Induction” (2026). arXiv: 2603.20274 [cs.FL].
- [Tsy09] Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer Series in Statistics. Dordrecht: Springer, 2009. DOI: 10.1007/b13794.
- [Val84] Leslie G. Valiant. “A Theory of the Learnable.” *Communications of the ACM* 27.11 (Nov. 1984), pages 1134–1142. DOI: 10.1145/1968.1972.
- [Vap91] Vladimir Vapnik. “Principles of Risk Minimization for Learning Theory.” *Advances in Neural Information Processing Systems*. Volume 4. 1991, pages 831–838.
- [Vas+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. “Attention Is All You Need.” *Advances in Neural Information Processing Systems*. 2017. arXiv: 1706.03762 [cs.CL].
- [VC68] Vladimir N. Vapnik and Alexey Ya. Chervonenkis. *Doklady Akademii Nauk* 181.4 (1968), page 781. Translated as “On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities.” Translated from the Russian by B. Seckler. *Theory of Probability and Its Applications* 16.2 (1971), page 264. DOI: 10.1137/1116025.
- [VC74] Vladimir N. Vapnik and Alexey Ya. Chervonenkis. *Theory of Pattern Recognition*. Moscow: Nauka, 1974.
- [Ven+11] Joel Veness, Kee Siong Ng, Marcus Hutter, William Uther, and David Silver. “A Monte-Carlo AIXI Approximation.” *Journal of Artificial Intelligence Research* 40 (2011), pages 95–142. arXiv: 0909.0801 [cs.AI].
- [Ver21] Nikolay Vereshchagin. “Proofs of Conservation Inequalities for Levin’s Notion of Mutual Information of 1974.” *Theoretical Computer Science* 856 (2021), pages 14–20. arXiv: 1911.05447 [cs.IT].
- [Vov01] Volodya Vovk. “Competitive on-line statistics.” *International Statistical Review* 69.2 (2001), pages 213–248.
- [VS16] Nikolay Vereshchagin and Alexander Shen. “Algorithmic statistics: forty years later.” *Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday*. Springer, 2016, pages 669–737. arXiv: 1607.08077 [cs.CC].
- [Wai19] Martin J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge University Press, 2019.
- [Wal05] Christopher Stewart Wallace. *Statistical and Inductive Inference by Minimum Message Length*. Springer, 2005.
- [Wei07] Michael Weisberg. “Three Kinds of Idealization.” *The Journal of Philosophy* 104.12 (2007), pages 639–659.
- [WH25] Cole Wyeth and Marcus Hutter. “Value under ignorance in universal artificial intelligence.” *International Conference on Artificial General Intelligence*. Springer. 2025, pages 338–349. arXiv: 2512.17086 [cs.AI].
- [Wil25] Andrew Gordon Wilson. “Position: Deep Learning is Not So Mysterious or Different.” *International Conference on Machine Learning*. Position Paper Track. 2025. arXiv: 2503.02113 [cs.LG].
- [WM25] Jun Wan and Lingrui Mei. “Large Language Models as Computable Approximations to Solomonoff Induction” (2025). arXiv: 2505.15784 [cs.LG].

- [Wol23] David H. Wolpert. “The Implications of the No-Free-Lunch Theorems for Meta-Induction.” *Journal for General Philosophy of Science* 54.3 (2023), pages 421–432. arXiv: 2103.11956 [cs.LG].
- [Wol24] David H. Wolpert. “Implications of Computer Science Theory for the Simulation Hypothesis” (2024). arXiv: 2404.16050 [cs.LO].
- [Wol96] David H. Wolpert. “The lack of a priori distinctions between learning algorithms.” *Neural computation* 8.7 (1996), pages 1341–1390.
- [WRS25] David H. Wolpert, Carlo Rovelli, and Jordan Scharnhorst. “Disentangling Boltzmann Brains, the Time-Asymmetry of Memory, and the Second Law.” *Entropy* 27.12 (2025), page 1227. arXiv: 2507.10959 [physics.hist-ph].
- [WS25] Tomoya Wakayama and Taiji Suzuki. *In-Context Learning Is Provably Bayesian Inference: A Generalization Theory for Meta-Learning*. 2025. arXiv: 2510.10981 [stat.ML].
- [Xie+22] Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. “An Explanation of In-context Learning as Implicit Bayesian Inference.” *International Conference on Learning Representations*. 2022. arXiv: 2111.02080 [cs.CL].
- [Yan+25] Chenxiao Yang, Nathan Srebro, David McAllester, and Zhiyuan Li. “PENCIL: Long Thoughts with Short Memory.” *International Conference on Machine Learning*. 2025. arXiv: 2503.14337 [cs.LG].
- [Zaf25] Francesca Zaffora Blando. “Bayesian merging of opinions and algorithmic randomness.” *The British Journal for the Philosophy of Science* 76.4 (2025), pages 921–952.
- [Zha+17] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. “Understanding Deep Learning Requires Rethinking Generalization.” *International Conference on Learning Representations*. 2017. arXiv: 1611.03530 [cs.LG].
- [Zho+19] Wenda Zhou, Victor Veitch, Morgane Austern, Ryan P. Adams, and Peter Orbanz. “Non-Vacuous Generalization Bounds at the ImageNet Scale: A PAC-Bayesian Compression Approach.” *International Conference on Learning Representations*. 2019. arXiv: 1804.05862 [stat.ML].
- [ZSS20] Lijia Zhou, Danica J. Sutherland, and Nathan Srebro. “On Uniform Convergence and Low-Norm Interpolation Learning.” *Advances in Neural Information Processing Systems*. 2020. arXiv: 2006.05942 [stat.ML].

A Additional properties of algorithmic mutual information

Our first result is analogous to the Shannon theory inequality for random variables P, Q, X :

$$0 \leq I(P; X | Q) \leq \min(H(P | Q), H(X | Q)).$$

Proposition 4 (Information bounds). *For all $p, q \in \mathbb{B}^\infty$ and $x \in \mathbb{B}^*$,*

$$0 \stackrel{+}{\leq} MI^q(p : x) \stackrel{+}{\leq} \min(KM^q(p), KM^q(x)).$$

Proof. By considering the machine $T_i^{p \oplus q}(\alpha) := U^q(\alpha)$ that ignores p , we obtain $M^{p \oplus q}(x) \geq 2^{-|i|} M^q(x) \stackrel{\times}{\cong} M^q(x)$. Hence from the definition (6), $0 \stackrel{+}{\leq} MI^q(p : x) \leq KM^q(x)$.

For the remaining inequality, consider the machine $T_i^q(\alpha \oplus \beta) := U^{U^q(\beta) \oplus q}(\alpha)$, where the right-hand size is taken to stop if it tries to access a non-existent index of $U^q(\beta)$. Since i is a universal constant,

$$\begin{aligned} M^q(x) &= \mathbb{P}_{\gamma \sim \lambda}(x \sqsubseteq U^q(\gamma)) \\ &\stackrel{\times}{\geq} \mathbb{P}_{\alpha, \beta \sim \lambda}(x \sqsubseteq U^q(i(\alpha \oplus \beta))) \\ &= \mathbb{P}_{\alpha, \beta \sim \lambda}(x \sqsubseteq T_i^q(\alpha \oplus \beta)) \\ &\geq \mathbb{P}_{\beta \sim \lambda}(U^q(\beta) = p) \cdot \mathbb{P}_{\alpha \sim \lambda}(x \sqsubseteq U^{p \oplus q}(\alpha)) \\ &= M^q(p) \cdot M^{p \oplus q}(x). \end{aligned}$$

By rearranging and taking logarithms, $MI^q(p : x) \stackrel{+}{\leq} KM^q(p)$. □

Next, we show that no deterministic or randomized method can reliably create mutual information with x . Levin [Lev84] and Vereshchagin [Ver21] proved similar results for symmetric notions of mutual information, under the name *conservation of randomness*; Ebtekar and Hutter [EH25] frame one such result as a generalization of the second law of thermodynamics.

Theorem 5 (No information ex nihilo). *Let $r \in \mathbb{B}^*$ be such that the randomly generated string $p := U^q(r\beta)$ is infinitely long for almost all $\beta \sim \lambda$. Then, for all $x \in \mathbb{B}^*$,*

$$\mathbb{E}_p \left[2^{MI^q(p;x)} \right] \stackrel{\times}{\leq} 2^{|r|},$$

where MI^q here is expressed in units of bits.

Proof. Consider again the machine $T_i^q(\alpha \oplus \beta) := U^{U^q(\beta) \oplus q}(\alpha)$. A random string $\gamma \sim \lambda$ has probability $2^{-|\bar{i}|-|r|}$ of having the form $\bar{i}(\alpha \oplus r\beta)$ for some $\alpha, \beta \in \mathbb{B}^\infty$. Since i is constant,

$$\begin{aligned} M^q(x) &= \mathbb{P}_{\gamma \sim \lambda} (x \sqsubseteq U^q(\gamma)) \\ &\stackrel{\times}{\geq} 2^{-|r|} \mathbb{P}_{\alpha, \beta \sim \lambda} (x \sqsubseteq U^q(\bar{i}(\alpha \oplus r\beta))) \\ &= 2^{-|r|} \mathbb{E}_{\beta \sim \lambda} \mathbb{P}_{\alpha \sim \lambda} (x \sqsubseteq T_i^q(\alpha \oplus r\beta)) \\ &= 2^{-|r|} \mathbb{E}_p \mathbb{P}_{\alpha \sim \lambda} (x \sqsubseteq U^{p \oplus q}(\alpha)) \\ &= 2^{-|r|} \mathbb{E}_p M^{p \oplus q}(x). \end{aligned}$$

Rearranging yields

$$\mathbb{E}_p \left[2^{MI^q(p;x)} \right] = \frac{\mathbb{E}_p M^{p \oplus q}(x)}{M^q(x)} \stackrel{\times}{\leq} 2^{|r|}. \quad \square$$

Suppose we want to design a predictor μ_p that substantially outperforms M^q . According to Theorem 3, this requires $MI^q(p : x)$ to be high. Let's imagine what the process of designing a suitable program p might look like. According to the physical Church-Turing thesis, the design process must be controlled by a pre-existing algorithm, presumably inside our brain, making it accessible from our information vantage point. We can model any accessible method as using very little new code r , which uses the information vantage point q and random source β to generate $p := U^q(r\beta)$. Theorem 5 says that such a process tends to produce low $MI^q(p : x)$.

B Additional considerations regarding oracle Solomonoff induction

B.1 Fresh restarting $M^{y^0}(x)$ vs lifelong learning $M^0(yx)$

Suppose we want to predict $x \in \mathbb{B}^*$, using some prior information $y \in \mathbb{B}^*$. We can take either (1) a *fresh restarting* approach, directly predicting x according to the semimeasure $M^{y^0}(x)$ that accesses y via the oracle; or (2) a *lifelong learning* approach, predicting the concatenation yx according to the semimeasure $M^0(yx)$, so that the predictor sees y before x .

According to Theorem 3, these two approaches optimize different objectives against different classes of experts. While both can access y while predicting x , a lifelong learner with sufficient experience must converge to the conclusion supported by y [Sol78], whereas a freshly restarted learner must give substantial weight to other alternatives.

For example, suppose y is just a long sequence of zeros. In order to achieve bounded regret against a straightforward “always zero” expert, the next-bit predictions must converge to zero as y gets longer. There is no need to hedge strongly against the possibility of seeing $x_1 = 1$, because there is no accessible expert that performs well on a long string of zeros, and also changes its prediction to 1 at a very specific time. On the other hand, the fresh restarting learner must compete with experts that are *only* evaluated on x , and must therefore hedge against both possibilities for x_1 !

More generally, given a long sequence consisting of multiple episodes, predicting each new episode $e^{(i)} \in \mathbb{B}^*$ with a freshly instantiated universal semimeasure $M^{e^{(1)} \dots e^{(i-1)}}(e^{(i)})$ allows us to bound the regret separately per episode. However, if we want to minimize the total loss across all episodes, then the lifelong learning Solomonoff induction $M^0(e^{(1)} \dots e^{(i)})$ is superior. In either case, the oracle should represent the information available at the start of an evaluation.

B.2 The reference universal computer

Fundamentally, the information vantage point consists of not only the oracle q , but the entire relativized machine U^q . In the finite case $q = y\mathbf{0}$, the oracle can be “hardcoded” into the description of another universal computer V , such that $V^{\mathbf{0}} = U^q$. Our “oracle Solomonoff induction” is then just classical Solomonoff induction with respect to V .

Nonetheless, it is useful to separate out a “minimal” U to treat as a universal reference. Intuitively, the idea is as follows: while each person on Earth likely has a different vantage point, they are also likely to share some common knowledge. Provided that everyone can agree on a “Schelling point” machine U , we retain the freedom to write programs in our preferred languages, which translate onto U via interpreters that we include in our personal instantiation of q . U might be chosen to be accessible from physics [Deu13; Özk15; JW18; KW20; Wol24; EH25], and/or tailored to minimize the hidden constants in our regret bounds [cf. LV19, Section 3.9]. By agreeing on U , we ensure that the hidden constants have small universal bounds independent of q, p, x . The oracle q is itself not a universal constant, so we explicitly account for its influence in our bounds.

B.3 Other loss functions

An important limitation of Theorem 3 is that it only applies to the log loss. The *strong convexity* of the log loss enables the predictor to hedge when it is uncertain [Vov01]. In contrast, many real settings require commitment to a specific action.

An important case is the 0-1 loss. Instead of probabilistic predictions, it demands a series of discrete predictions that the next bit x_i will be either 0 or 1, and the loss is simply the number of mistakes. Given a Bayesian prior ν , it seems reasonable to choose the prediction that minimizes expected loss. For the 0-1 loss, this is the maximum a posteriori (MAP) estimate, so that

$$L_{0-1}(\nu, x) := \sum_{i=1}^{|x|} \mathbb{1}(\nu(x_{<i}x_i) \leq \nu(x_{<i}\neg x_i)).$$

Unfortunately, the adversarial example of Putnam [Put63] (cf. Section 4.1) returns in full force to produce a sequence x on which this strategy – and in fact, *any* strategy – predicts every bit incorrectly. If the prior is M^q , no computable expert can predict this x ; nonetheless, an always-0 or always-1 “expert” would guess correctly at least half of the time, resulting in a large regret. Since these experts are simple, Proposition 4 implies $MI^q(p : x) \pm 0$, so the bound in Theorem 3 does not hold.

On the other hand, this x does not seem like a “natural” sequence that could arise, for example, by sampling from a computable distribution. Its construction depends on the incomputable semimeasure M^q , which is related to the *Turing jump* q' of q [Nie09; DH10]. We speculate that it may be possible to bound the regret for general loss functions in terms of $MI^q(p \oplus q' : x)$, leaving the analysis to future work.

C No Free Lunch and complexity measures

No Free Lunch means that any generalization bound we prove must depend on an inductive bias. How can we understand the inductive bias implicit in some classical statistical learning results?

C.1 Uniform convergence

We will first review some relevant textbook results. More details are available from many sources, such as the recent book by Bach [Bac24, Section 4.5].

Consider a learning problem where we observe independent data $Z = (z_1, \dots, z_m) \sim \mathcal{D}^m$ – i.e., the z_i are i.i.d. from \mathcal{D} – and wish to identify a hypothesis $h \in \mathcal{H}$ having small risk $L_{\mathcal{D}}(h) = \mathbb{E}_{z \sim \mathcal{D}} \ell(h, z)$, often based on the empirical risk $L_Z(h) = \frac{1}{m} \sum_{i=1}^m \ell(h, z_i)$. In typical prediction settings, we have $z = (x, y)$ and $\ell(h, (x, y))$ might be $\mathbb{1}(h(x) \neq y)$, $\|h(x) - y\|^2$, or $-\log(h(x) \cdot y)$.

A primary tool of statistical learning theory is *uniform convergence*, based on the trivial inequality

$$L_{\mathcal{D}}(\hat{h}) \leq L_Z(\hat{h}) + \sup_{h \in \mathcal{H}} [L_{\mathcal{D}}(h) - L_Z(h)]. \quad (7)$$

If we have a reasonable upper bound on $\sup_{h \in \mathcal{H}} [L_{\mathcal{D}}(h) - L_Z(h)]$, then “what you see is what you get”: predictors h with low training loss $L_Z(h)$ will actually generalize well (have small $L_{\mathcal{D}}(h)$).

For an empirical risk minimizer $\hat{h} \in \operatorname{argmin}_{h \in \mathcal{H}} L_Z(h)$, we further have for all $h^* \in \mathcal{H}$ that

$$L_Z(\hat{h}) \leq L_Z(h^*) = L_{\mathcal{D}}(h^*) + (L_Z(h^*) - L_{\mathcal{D}}(h^*)).$$

For any fixed h^* , we can apply a simple concentration inequality⁴ (such as Hoeffding’s) to show that $L_Z(h^*) - L_{\mathcal{D}}(h^*)$ is small. Then, if we choose that fixed h^* to have loss (arbitrarily close to) $\inf_{h \in \mathcal{H}} L_{\mathcal{D}}(h)$, we know $L_Z(\hat{h})$ will not be much bigger than $\inf_{h \in \mathcal{H}} L_{\mathcal{D}}(h)$. If we also have uniform convergence, meaning the last term in (7) is known to be small, then \hat{h} will be nearly as good as the optimal predictor from \mathcal{H} .

It is worth noting that uniform convergence (7) gives only an upper bound on the generalization performance. While uniform convergence occurs iff the problem is uniformly learnable for binary classification [e.g. SB14, Theorem 6.7], the same is not true in more general settings. There are explicit constructions, for example, in multiclass classification [Nat89b, Section 6], mean estimation with missing data [SB14, Exercise 13.2], and high-dimensional linear classification [NK19, Section 3.1] and regression [ZSS20, Sections 3.1 and 3.2], where learning is possible but uniform convergence does not hold.⁵

C.2 Rademacher complexity

To show uniform convergence, we can use *Rademacher complexity*. Letting $\sigma_1, \dots, \sigma_m$ follow a Rademacher distribution $\operatorname{Uniform}(\{-1, +1\})$, the (expected) Rademacher complexity of a set of real-valued functions \mathcal{F} is

$$\operatorname{Rad}_{\mathcal{D}^m}(\mathcal{F}) := \mathbb{E}_{z_1, \dots, z_m \sim \mathcal{D}} \mathbb{E}_{\sigma_1, \dots, \sigma_m} \left[\sup_{f \in \mathcal{F}} \frac{1}{m} \sum_{i=1}^m \sigma_i f(z_i) \right].$$

⁴Many sources instead bound $\sup_{h \in \mathcal{H}} |L_{\mathcal{D}}(h) - L_Z(h)|$, and so do not need separate control of $L_Z(h^*) - L_{\mathcal{D}}(h^*)$. While perhaps conceptually simpler, the description here makes clear that we only need to upper-bound $L_Z - L_{\mathcal{D}}$ for h^* , while we need to upper-bound $L_{\mathcal{D}} - L_Z$, or equivalently lower-bound L_Z , for all hypotheses. Thus asking for an upper bound everywhere is potentially wasteful. While with Rademacher complexity the gap in the bound is only a small constant, some methods [e.g. Men21] can give much tighter lower bounds than upper bounds. The hardness result of Nagarajan and Kolter [NK19], used to argue that “uniform convergence may be unable to explain generalization in deep learning,” is also fundamentally about cases where $L_Z - L_{\mathcal{D}}$ is high; no such result is possible in the one-sided case [ZSS20, footnote 5].

⁵The examples of Nagarajan [Nat89a, Theorem 6.7], Shalev-Shwartz and Ben-David [SB14, Exercise 13.2], and Zhou et al. [ZSS20, Section 3.1] apply to one-sided uniform convergence, as well as two-sided. While Zhou et al. left the one-sided failure as a conjecture, that conjecture is true when $d_S \geq 1$, as we now show.

The proof is identical to the two-sided case once we know $\mathbb{E} \lambda_{\max}(\Sigma - \hat{\Sigma})$ from their Proposition B.2 (which they denote ρ) is $\Omega(\sqrt{\lambda_n/n})$. To show this, fix a unit vector $u \in \mathbb{R}^{d_S}$ and let $q = -X_J^T X_S u / \|X_J^T X_S u\| \in \mathbb{R}^{d_J}$. Consider representing $\Sigma - \hat{\Sigma}$ in a basis beginning with the orthonormal vectors $(u, 0)$ and $(0, q)$; the upper-left 2×2 submatrix in that basis is

$$A := \begin{bmatrix} u^T (I_{d_S} - \frac{1}{n} X_S^T X_S) u & u^T (-\frac{1}{n} X_S^T X_J) q \\ q^T (-\frac{1}{n} X_J^T X_S) u & q^T (\frac{\lambda_n}{d_J} I_{d_J} - \frac{1}{n} X_J^T X_J) q \end{bmatrix} = \begin{bmatrix} 1 - \frac{1}{n} \|X_S u\|^2 & \frac{1}{n} \|X_J^T X_S u\| \\ \frac{1}{n} \|X_J^T X_S u\| & \frac{\lambda_n}{d_J} - \frac{1}{n} \|X_J q\|^2 \end{bmatrix}.$$

The largest eigenvalue of $\Sigma - \hat{\Sigma}$ is at least as large as $\lambda_{\max}(A)$. Let $Y = \|X_S u\|$. Since $X_J X_J^T \rightarrow \lambda_n I$ as $d_J \rightarrow \infty$, we have $\|X_J X_S u\|^2 \rightarrow \lambda_n$ and $\|X_J q\|^2 = \|X_J X_J^T X_S u\|^2 / \|X_J X_S u\|^2 \rightarrow \lambda_n$, with all convergences almost sure; thus

$A \rightarrow M := \begin{bmatrix} 1 - \frac{1}{n} Y^2 & \frac{1}{n} \sqrt{\lambda_n} Y \\ \frac{1}{n} \sqrt{\lambda_n} Y & -\frac{1}{n} \lambda_n \end{bmatrix}$. As $\lambda_{\max} \left(\begin{bmatrix} \alpha & \beta \\ \beta & \gamma \end{bmatrix} \right) = \frac{1}{2} (\alpha + \gamma + \sqrt{(\alpha - \gamma)^2 + 4\beta^2}) \geq \frac{1}{2} (\alpha + \gamma) + |\beta|$, we have

$$\lambda_{\max}(\Sigma - \hat{\Sigma}) \geq \lambda_{\max}(A) \rightarrow \lambda_{\max}(M) \geq \frac{1}{2} \left(1 - \frac{1}{n} Y^2 \right) - \frac{\lambda_n}{2n} + \frac{\sqrt{\lambda_n}}{n} Y.$$

Since u is a unit vector and X_S is standard normal, Y^2 is $\chi^2(n)$; hence $(1 - \frac{1}{n} Y^2)$ has mean zero. Y is $\chi(n)$, with mean $\sqrt{n-1} + O(1/n) = \sqrt{n} + O(1/\sqrt{n})$; thus $\mathbb{E} \sqrt{\lambda_n} Y/n = \sqrt{\frac{\lambda_n}{n}} (1 + O(1/n))$. Recalling that in the problem setup it was assumed $\lambda_n = o(n)$, we have shown that

$$\mathbb{E} \lambda_{\max}(M) \geq \sqrt{\frac{\lambda_n}{n}} \left(-\frac{1}{2} \sqrt{\frac{\lambda_n}{n}} + 1 + O\left(\frac{1}{n}\right) \right) = \Omega\left(\sqrt{\frac{\lambda_n}{n}}\right).$$

Following the proof of their Theorem 3.2, we thus have $\lim_{n \rightarrow \infty} \lim_{d_J \rightarrow \infty} \mathbb{E} \sup_{\|w\| \leq \frac{1}{\hat{w}_{\min}}} [L_{\mathcal{D}}(w) - L_S(w)] = \infty$.

This definition arises from a technique called symmetrization, which yields [Bac24, Proposition 4.2]

$$\mathbb{E}_{Z \sim \mathcal{D}^m} \sup_{f \in \mathcal{F}} \left[\mathbb{E}_{z \sim \mathcal{D}} [f(z)] - \frac{1}{m} \sum_{i=1}^m f(z_i) \right] \leq 2 \text{Rad}_{\mathcal{D}^m}(\mathcal{F}). \quad (8)$$

One can also use “desymmetrization” to show a closely related lower bound, although the form is slightly more complex [Kol06, Section 2.2; Wai19, Proposition 4.12].

Plugging in $\mathcal{F} = \{z \mapsto \ell(h, z) : h \in \mathcal{H}\}$ gives exactly a bound on $\mathbb{E}_{Z \sim \mathcal{D}^m} \sup_{h \in \mathcal{H}} [L_{\mathcal{D}}(h) - L_Z(h)]$. We can further obtain high-probability bounds, if ℓ is bounded, by McDiarmid’s inequality [Bac24, Section 4.4.1].

Finally, we often bound the complexity of a loss class using the complexity of the hypothesis class. If $h(x) \in \mathbb{R}$ and ℓ is G -Lipschitz in its second argument, then Talagrand’s contraction lemma [Bac24, Proposition 4.3] implies that $\text{Rad}(\mathcal{F}) \leq G \text{Rad}(\mathcal{H})$; thus

$$\mathbb{E}_{S \sim \mathcal{D}^m} \sup_{h \in \mathcal{H}} [L_{\mathcal{D}}(h) - L_Z(h)] \leq 2G \text{Rad}(\mathcal{H}).$$

Let’s turn for a moment to the problem of learning a linear function in some fixed feature space, $\mathcal{H}_{\mathcal{W}} = \{x \mapsto \langle w, \phi(x) \rangle : w \in \mathcal{W}\}$. Choosing $\phi(x) = x$ covers standard linear predictors. Choosing ϕ to map into some Hilbert space (here assumed to be over the reals for simplicity) corresponds to learning in a reproducing kernel Hilbert space (RKHS).

The usual bound for the Rademacher complexity of $\mathcal{H}_{\mathcal{W}}$ is

$$\begin{aligned} \text{Rad}_{\mathcal{D}^m}(\mathcal{H}_{\mathcal{W}}) &= \mathbb{E} \sup_{w \in \mathcal{W}} \frac{1}{m} \sum_{i=1}^m \sigma_i \langle w, x_i \rangle \\ &= \mathbb{E} \sup_{w \in \mathcal{W}} \left\langle \frac{1}{m} \sum_{i=1}^m \sigma_i x_i, w \right\rangle \\ &\leq \mathbb{E} \left\| \frac{1}{m} \sum_{i=1}^m \sigma_i x_i \right\| \left(\sup_{w \in \mathcal{W}} \|w\| \right) \quad \text{by Cauchy-Schwarz} \\ &\leq \frac{1}{\sqrt{m}} \sqrt{\mathbb{E}_{x \sim \mathcal{D}} \|x\|^2} \left(\sup_{w \in \mathcal{W}} \|w\| \right) \quad \text{by Jensen and } \mathbb{E}[\sigma_i \sigma_j] = \mathbb{1}(i = j). \end{aligned} \quad (9)$$

Since $\sup_{w \in \mathcal{W}} \|w\|$ appears so naturally here, it is tempting to think that it plays a vital role: low-norm functions are easy to learn, while high-norm functions are hard. This almost seems like a “built-in” inductive bias for learning functions from $\mathcal{H}_{\mathcal{W}}$.

This intuition is incorrect.

Proposition 6 ([Bac24, Exercise 4.9]). *Let \mathcal{H} be any set of functions $\mathcal{X} \rightarrow \mathbb{R}$, $f : \mathcal{X} \rightarrow \mathbb{R}$ any function, and write $\mathcal{H} + \{f\} := \{x \mapsto h(x) + f(x) : h \in \mathcal{H}\}$. Then $\text{Rad}_{\mathcal{D}^m}(\mathcal{H} + \{f\}) = \text{Rad}_{\mathcal{D}^m}(\mathcal{H})$.*

Proof. Since f does not interact with h and $\mathbb{E} \sigma_i = 0$, we have

$$\mathbb{E} \left[\sup_{h' \in \mathcal{H} + \{f\}} \sum_i \sigma_i h'(x_i) \right] = \mathbb{E} \sup_{h \in \mathcal{H}} \left[\sum_i \sigma_i h(x_i) + \sum_i \sigma_i f(x_i) \right] = \mathbb{E} \sup_{h \in \mathcal{H}} \left[\sum_i \sigma_i h(x_i) \right]. \quad \square$$

In particular, translating the class is irrelevant: for any arbitrary w_0 , we have $\text{Rad}_{\mathcal{D}^m}(\mathcal{H}_{\mathcal{W}}) = \text{Rad}_{\mathcal{D}^m}(\mathcal{H}_{\mathcal{W} - \{w_0\}})$. Thus,

$$\text{Rad}_{\mathcal{D}^m}(\mathcal{H}_{\mathcal{W}}) = \inf_{w_0} \text{Rad}_{\mathcal{D}^m}(\mathcal{H}_{\mathcal{W} - \{w_0\}}) \leq \frac{1}{\sqrt{m}} \sqrt{\mathbb{E} \|x\|^2} \left(\inf_{w_0} \sup_{w \in \mathcal{W}} \|w - w_0\| \right).$$

That is, it is not the maximum norm but the *radius* of \mathcal{W} that matters. While our previous upper bound (9) did indeed prefer low-norm predictors to high-norm ones, that was an artifact of our proof. In fact, we could have subtracted w_0 in (9) in the first place, “naturally” yielding the radius.

This is exactly the picture of Figure 1: the Rademacher bound only cares about the size of each hypothesis class. It does *not* tell you which functions to prefer. By Proposition 6, this is not specific to linear classes; *any* function class can be translated by *any* function without modifying the Rademacher complexity.

C.3 Combinatorial dimensions for classification

For binary classifiers, where both upper and lower bounds for learnability are controlled by the VC dimension [VC68; SB14, Theorem 6.7], an exactly analogous result holds. There is no such thing as an “inherently simple” binary classifier; it only matters how flexible the hypothesis class is.

Proposition 7 ([MRT18, Exercise 3.25]). *Define the operation $a \text{ xor } b := \mathbb{1}(a \neq b)$ for $a, b \in \{0, 1\}$. Write $\mathcal{H} \text{ xor } f := \{x \mapsto h(x) \text{ xor } f(x) : h \in \mathcal{H}\}$. Then, $\text{VCdim}(\mathcal{H} \text{ xor } f) = \text{VCdim}(\mathcal{H})$.*

Proof. Let $h_1, \dots, h_N \in \mathcal{H}$ be hypotheses achieving the vectors of labels $y_1, \dots, y_N \in \{0, 1\}^{|X|}$ for the points in a set X . Write y for the vector of labels achieved by f on X . Then $h_1 \text{ xor } f, \dots, h_N \text{ xor } f$ achieve the labels $y_1 \text{ xor } y, \dots, y_N \text{ xor } y$, with xor elementwise. As xor is a bijection, the number of labelings in each set is the same; thus \mathcal{H} shatters X iff $\mathcal{H} \text{ xor } f$ does. \square

The same is true for multiclass classification, where learnability is jointly characterized by the Natarajan and DS dimensions [Coh+26].

Proposition 8. *Let \mathcal{H} be a class of functions from \mathcal{X} to $\mathcal{Y} \subseteq \mathbb{N}$. For each $x \in \mathcal{X}$, let π_x be an arbitrary bijection on \mathcal{Y} , and let $\pi \circ \mathcal{H} = \{x \mapsto \pi_x(h(x)) : h \in \mathcal{H}\}$. Then the Natarajan dimension [Nat89a] of \mathcal{H} and $\pi \circ \mathcal{H}$ are equal; the same is true for the DS dimension [DS14].*

Proof. The proof proceeds exactly as that of Proposition 7, but using the concepts of Natarajan shattering or DS shattering instead of VC shattering. \square

C.4 PAC-Bayes

Another major technique for proving generalization is PAC-Bayes bounds; Alquier [Alq24] gives a recent overview. In these bounds, there is a *prior* and a *posterior* distribution over predictors, although these need not be related by the rules of Bayesian inference. There are many forms of PAC-Bayes bounds, but in general, they bound the random variable $L_{\mathcal{D}}(h) - L_Z(h)$, for h sampled from the posterior distribution, in terms of the KL divergence of the posterior from the prior.

Such bounds also satisfy a similar invariance to the actual form of the hypotheses, as the KL divergence is invariant to any invertible transformation. If we transform the prior and posterior in the same way, we obtain the same bound on the expected generalization gap.

C.5 Stability

Probably the only other widely-used technique for proving generalization bounds is *algorithmic stability*; this can apply outside of settings with uniform convergence. For instance, the most common definition is uniform stability:

Definition 9 ([BE02; EEP05]). *When $Z = (z_1, \dots, z_m)$, let Z_{-i} denote $(z_1, \dots, z_{i-1}, z_{i+1}, \dots, z_m)$. A (possibly randomized) learning algorithm \mathcal{A} is $\beta(m)$ -uniformly stable if for all $m \geq 1$,*

$$\sup_{\substack{Z \in \mathcal{Z}^m, i \in [m] \\ z, z' \in \mathcal{Z}}} |\mathbb{E}_{\mathcal{A}} \ell(\mathcal{A}(Z_{-i}), z) - \mathbb{E}_{\mathcal{A}} \ell(\mathcal{A}(Z), z)| \leq \beta(m).$$

We say an algorithm is uniformly stable if $\lim_{m \rightarrow \infty} \beta(m) = 0$.

Uniformly stable algorithms bear guarantees on $\mathbb{E}_{\mathcal{A}} [L_{\mathcal{D}}(\mathcal{A}(Z)) - L_Z(\mathcal{A}(Z))]$ in terms of $\beta(m)$ [BE02; EEP05; BKZ20]. Once again, nothing in this definition privileges any particular form of hypothesis; it only matters how stable the algorithm’s output is to small changes in the training set. The same is true for various other notions of stability [Sha+10].

D Epistemic Boltzmann brains

Cosmology is the branch of physics that studies the large-scale structure of the Universe. One criterion for evaluating a proposed model, is whether it predicts an overabundance of so-called *Boltzmann brains*. These are observers that arise from chance fluctuations in a thermal background, instantiating a moment of subjective experience – complete with illusory memories of an orderly

past – before dissipating back into equilibrium. In models where Boltzmann brains (BBs) vastly outnumber ordinary observers (OOs) like ourselves, it is argued that by simple counting, we must conclude that we are almost certainly BBs. Since this conclusion would completely undermine the scientific method, it is argued that such models should be rejected [BT87; Sim+10; Car20].

In this section, we argue a contrary view: the problem of BBs is not specific to particular models of the cosmos, but is in fact a kind of No Free Lunch theorem. Even if our Universe contains no BBs at all, **without an inductive bias, we are forced to conclude that we are probably Boltzmann brains**. On the other hand, Hutter [Hut10] and Müller [Mül20; Mül26] show that with a Solomonoff-style inductive prior, even observers that know their Universe is full of BBs will view themselves as OOs, and make ordinary inferences. Hence, there is no reason to judge a model by its prevalence of BBs.

We make four versions of the argument, first from a position of total epistemic ignorance, adding more generous assumptions at each subsequent iteration.

D.1 Total epistemic ignorance

Ideally, we should be able to make inferences about our surrounding world without any prior assumptions. Through our biological senses, we take a sequence of observations, which we hope to extrapolate into predictions about future observations. Unfortunately, without an inductive bias, the No Free Lunch Theorem applies exactly as stated in Proposition 1. That is, no matter what observations we have accumulated so far, it remains equally plausible to receive any sequence of future observations. Thus, with overwhelmingly high subjective probability, we should expect any apparent structure in our past to dissolve immediately, just as for a BB.

D.2 Knowledge of dynamics

Now we add the assumption that we know the dynamical laws of the physical Universe, but not the particular state that it is in. Wolpert et al. [WRS25] argue that concluding we are BBs from such an assumption would be circular, because a BB cannot scientifically infer the laws of physics. Nonetheless, it should not *hurt* us to assume we are endowed with some additional knowledge, as if by divine inspiration.

Without an inductive bias, we may take a maximum entropy prior over the possible states of the Universe. If we do, then our subjectively held belief distribution is identical to that of a Universe at large-scale equilibrium, also known as *heat death*. By the second law of thermodynamics, the time-evolution of our belief distribution remains at maximum entropy at all future times. Thus, even if the Universe is objectively quite orderly, our belief state is exactly that of a BB, unable to make useful inferences about the state of our surroundings. Before expanding on the details of this argument, we add an even more generous assumption.

D.3 Control of dynamics

Now we add the assumption that we can choose the dynamical laws at will, subject to obeying the second law of thermodynamics, i.e., that entropy cannot decrease. In other words, while the previous argument assumed we were divinely inspired by knowledge of physics, here we assume that we are divinely endowed with the ability to control physics as we see fit. We will see that, even granting such unrealistic powers, it remains impossible to make useful inferences about our surroundings.

At first, this may seem absurd. If we want to learn the state of our environment, surely we can just look at it and record its value to our memory; or failing that, we apply our physics-bending powers to *make* the environment match our beliefs! The issue is that the second law of thermodynamics does not play well with maximum entropy priors.

For example, consider an ordered pair (m_t, e_t) representing the state of a memory and an environment variable, respectively, at time t . For simplicity, we model the states as discrete. In order to measure e_t , we might like to implement the assignment operation $m_{t+1} := e_t$, which can also be represented as a map

$$(m, e) \mapsto (e, e).$$

However, this map is irreversible, and decreases the entropy of the joint system. Nature forbids irreversible operations in isolation; in reality, such measurements are only possible by dumping

the excess entropy into a third system that is known to have low entropy [Ben82]. One valid implementation first initializes the memory to zero by dissipating into a heat bath, after which we can implement the reversible (i.e., injective) map

$$(0, e) \mapsto (e, e).$$

Unfortunately, a maximum entropy prior assigns equal probability to every possible joint state of the Universe. Since every transformation that obeys the second law preserves this distribution, there is no way to be confident that we have set the memory to zero.

D.4 Control of dynamics *and* initialized memory

It seems strange to imagine not knowing the state of our own memory, since by definition the memory should be used to record our beliefs. In this final version of the argument, we start with some memory of a fixed size, say 1 GB, containing known initial values, a large fraction of which are zeros. Our prior is maximum entropy on everything in the Universe except for this memory, resulting in a total entropy that is 1 GB less than the maximum possible.

In this case, we can indeed perform some measurements, after which our memory's state becomes correlated with the environment. However, by the second law of thermodynamics, our belief distribution will never be more than 1 GB away from heat death. Converted to physical units, this gap is less than $10^{-13} \text{ J K}^{-1}$ [Fra05; EH25], so our observations can never assure us of large-scale structure sufficient to extract a macroscopically useful quantity of work. Normally, we make large-scale inferences by generalizing from smaller amounts of data; however, the near-maximum entropy prior forces us to conclude the smallest possible correlation consistent with our measurements. Once again, we are effectively trapped as epistemic BBs.

As a final note, Wolpert et al. [WRS25] propose that we should believe a maximum entropy distribution on the trajectory of the Universe, conditional on two snapshots of its macrostate: the Big Bang, and the present day. We take the position that their procedure is unjustified, since there is no direct way to obtain those snapshots. Instead, scientific inferences about the Universe are more readily justified via (approximate) universal induction on real observations [Hut10; Mül20; Mül26].