Evolutionary Meta-Learning in Neural Networks as a Neutral Testing Ground for Nativism and Empiricism

Daniel Grimmer

Faculty of Philosophy Departement, Yale University, CT, 06520, USA

Abstract

The recent engineering success of neural networks is often seen as favoring Empiricism over Nativism. Indeed, Empiricist accounts of epistemology (i.e., tabula rasa initialization followed by general-purpose learning from massive amounts of sensory data) align closely with the standard way of training neural networks. Crucially, however, this training regime imposes a broadly Empiricist epistemology by engineering fiat, rather than fairly testing it against a Nativist alternative. We argue that this issue can be remedied with a change of training regime; our *Evolutionary Meta-Learning* framework allows one to use neural networks to efficiently simulate the evolutionary pressures that shaped the human mind. We present a first-principles derivation showing that the dynamics of a Darwinian Lineage Simulation (DLS) are formally equivalent to a noisy Stochastic Gradient Ascent (SGA) up a log-fitness landscape. Furthermore, we demonstrate that the Baldwin Effect—modeled here as an evolutionary pressure towards rapid learning so as to minimize childhood mortality—is well captured by the MAML++ meta-objective function. This framework offers a neutral testing ground for Nativist and Empiricist adaptation strategies while also aligning both approaches with Sutton's Bitter Lesson. Without hand-coding or brute stipulation, adaptations are chosen by the evolutionary process itself as it navigates its way up an ever-changing fitness landscape.

[0] Email: daniel.grimmer@yale.edu
Orcid ID: 0000-0002-8449-3775

1 Introduction

While the ongoing conflict between Nativists and Empiricists has a long history (see Sec. 4), one of its current sites of conflict is the following question: Are artificial neural networks a good scientific model of human cognition? That is, are these systems sufficiently like us such that we can learn about ourselves through them by analogy? This pursuit of human-analogous AI (valuable to scientists and philosophers alike) ought to be starkly contrasted with the engineer's goal of achieving high-performance AI. While a surprising alignment between Sutton's Bitter Lesson and our present aims will be discussed briefly at the end of this section, we must set aside the engineer's perspective for now; we should not assume that machines optimized for engineering purposes will automatically serve our scientific and philosophical goals.

To better articulate this point, consider the often-made distinction between a submarine and a fish.[1] The question becomes: Is a submarine sufficiently analogous to a fish such that we can learn about fish by building and studying submarines? No. While submarines do manage to solve a fish-like task, they do so in ways that thoroughly undermine the scientific viability of the submarine-fish analogy. A submarine is clearly well-suited to the goals of its engineers (i.e., getting around underwater) and this in turn is very helpful to the marine biologist (it helps her observe the fish). But submarines are a complete failure *as a scientific model of fish*. Typically this is where the story ends, with our marine biologist dismissing the usefulness of the submarine-fish analogy, lamenting the amount of funding and attention that it has received, and grumpily returning to her original methods of studying fish (which, admittedly, is sometimes aided by the use of submarines).

But why end the story here? Our marine biologist could instead modify and redeploy these technological advancements to her own ends by constructing a *robotic fish*. Namely, she could do so with the explicit goal of bolstering whichever aspects of the robot-fish analogy would enable her to study fish via robots. For instance, suppose that she is interested in the evolutionary history of a certain fin shape. To this end, she might design a robot fish whose fins could be subjected to analogous evolutionary pressures. This would not be bio-mimicry for its own sake, but a rigorous exploration of convergent evolution and functional isomorphism.

This continuation of the parable provides us with a nuanced answer to our original question: No, artificial neural networks (as received from the engineers) are not good analogical models of human cognition simply because they have not been designed and calibrated to function as such. Good scientific models do not come cheap, they require scientific labor. They do not simply fall off of the engineering tree! Importantly, the issue here is not with the scientific viability of AI-human analogies in principle, but rather with the scientists and philosophers who try to use *off-the-shelf AI systems* as models of human cognition. Instead, the scientific community ought to design their own suite of AI systems which are purpose-built to enable us

[1]The Submarine vs Fish distinction originates with Dijkstra (1984) although it is being redeployed here with some modification.

to study human cognition by analogy.

Here, then, is the question which we should be asking: Is there some way of modifying and redeploying the AI engineer's current tool set (neural networks and machine learning) so as to construct a scientifically useful model of human cognition, especially its evolutionary history? We shall answer definitively: Yes, *when properly trained* (i.e., via evolutionary meta-learning) artificial neural networks can be used to efficiently and faithfully simulate the same kinds of evolutionary pressures which shaped human cognition. Importantly, this proposal does not in any way favor Empiricism over Nativism (or vice versa). Indeed, as we shall soon discuss, the Evolutionary Meta-Learning (EML) framework introduced in this paper actually provides a neutral testing ground for the Nativism-Empiricism debate.

Before introducing meta-learning, however, allow us to briefly review how neural networks typically learn within a single training run (think, within one lifetime). The networks overall connectivity is specified as some fixed architecture (e.g., FNN, CNN, LLM, etc.) with some random initialization for its parameter settings, θ_0, e.g., its initial weights and biases. The network's parameters are then repeatedly updated, $\theta_n \mapsto \theta_{n+1}$, in accordance with some learning algorithm, $\mathcal{A}$, so as to optimize some objective function, $\mathcal{L}(\theta)$.[2] A concrete example of a learning algorithm would be pre-conditioned gradient ascent with a time-varying learning rate:[3]

$$\mathcal{A} : \theta_n \mapsto \theta_{n+1} = \theta_n + \eta_n \, P \, \nabla\mathcal{L}(\theta_n). \tag{1}$$

At each step of training, θ_n moves in the direction which most increases $\mathcal{L}$ (namely, the $\nabla\mathcal{L}(\theta_n)$ direction) modified by some pre-conditioning matrix, P, and scaled by some learning rate, η_n. Biologically, this update step, $\theta_n \mapsto \theta_{n+1}$, is ontogenic in the sense that it concerns the development of the network within one training run. Notice that the network's ultimate performance, $\mathcal{L}(\theta_N)$, depends explicitly upon its final parameters, θ_N, but also implicitly upon its architecture as well as a great many so-called hyperparameters, including: its initialization, θ_0, as well as the details of its learning algorithm, $\mathcal{A}$, e.g., its pre-conditioning matrix, P, and its time-varying learning rate, η_n.

Meta-learning is a collection of machine learning techniques which allow a neural network to improve its performance *across a series of training runs* (think, across several generations). In particular, if one were to restart the above-described training process, one could make targeted changes to the network's architecture as well as its hyperparameters (θ_0 and $\mathcal{A}$) with the goal of improving the network's overall learning outcome. For simplicity, this paper will be focused upon meta-learning θ_0 and $\mathcal{A}$ within a fixed architecture. The core results of this paper can, however, be extended to scenarios in which the architecture too is meta-learned,

[2]It is a pedagogical simplification to introduce the inner loop in terms of maximizing an explicit objective function, $\mathcal{L}$. See footnote 4 for further discussion.

[3]It is important to note that our sign-convention in this paper is opposite to most of the optimization community; namely, our objective functions, $\mathcal{L}$, and meta-objective functions, $\mathcal{M}$, are quantities to be *maximized*, e.g., log-probabilities of survival and log-fitness ratings. Hence, we speak of *Gradient Ascent* rather than Gradient Descent. Where appropriate the term "optimize" shall be used as a neutral stand-in for "maximize/minimize".

e.g., through a Neural Architecture Search (NAS). Setting this possibility aside, one can "meta-update" both θ_0 and $\mathcal{A}$ before each training runs in accordance with some meta-learning algorithm, $\mathcal{A}_{\text{Meta}}$, so as to optimize some meta-objective function, $\mathcal{M}(\theta_0, \mathcal{A})$.[4] A salient example (relevant to our later discussion in Sec. 3) is the MAML++ meta-objective, defined as:[5]

$$\mathcal{M}(\theta_0, \mathcal{A}) = \sum_n \mathcal{L}(\theta_n). \tag{2}$$

In this case, the meta-objective is to modify the network's hyperparameters (i.e., θ_0 and the details of $\mathcal{A}$) so as to holistically improve *its entire learning trajectory*, not just its endpoint. The simplest possible meta-learning algorithm, $\mathcal{A}_{\text{Meta}}$, would be to move the network's next initialization, θ_0, in the direction which most increases the meta-objective, $\mathcal{M}$ (namely, the $\nabla_{\theta_0}\mathcal{M}(\theta_0, \mathcal{A})$ direction). The details of $\mathcal{A}$ could be similarly meta-updated. Biologically, this meta-update step is phylogenic in the sense that it concerns the development of the network across a series of training runs (think, across several generations). This whole process (i.e., a training run followed by a reset and a meta-update of the hyperparameters) is traditionally called *the outer loop*. By contrast, the many ontogenic update steps within each training run are traditionally called *the inner loops*.

The above-described techniques are called "meta-learning" because they allow a neural network to "learn how to learn" across a series of training runs. A closely related phenomena is well-known to evolutionary biologists and cognitive scientists, namely the Baldwin effect;[6]

[4]We have been forced into a slight mischaracterization here by our pedagogical choice to introduce both the inner loop and the outer loop in terms of optimizing an explicit objective function, $\mathcal{L}(\theta)$, and meta-objective function, $\mathcal{M}(\theta_0, \mathcal{A})$, respectively. Importantly, the dynamics of some meta-learning process is fully specified by its meta-learning algorithm, $\mathcal{A}_{\text{Meta}}$, *whether or not one can even articulate* an objective function, $\mathcal{L}$, for the inner loop or a meta-objective function, $\mathcal{M}$, for the outer loop. Properly introduced, the inner loop *simply is* an application of the learning algorithm, $\mathcal{A}$, to θ_n in response to new data (with no objective $\mathcal{L}$ in sight). Similarly, the outer loop *simply is* a training run followed by a reset and an application of $\mathcal{A}_{\text{Meta}}$ to meta-update θ_0 and $\mathcal{A}$, regardless of whether an explicit meta-objective is formulated. Interestingly, however, the evolutionary dynamics considered in Sec. 2 does pick out an explicit meta-objective function for the outer loop, namely, the log-fitness rating, $\mathcal{M} = \log \mathcal{F}_\lambda$. Connecting this fact to the long-standing debates about teleological notions in biology is, unfortunately, beyond the scope of this paper.

[5]Model Agnostic Meta-Learning (MAML) was introduced in Finn et al. (2017) and optimizes for the network's performance after N steps, $\mathcal{L}(\theta_N)$, e.g., performance on a "final exam". By contrast, MAML++ was introduced in Antoniou et al. (2019) and optimizes the network's performance cumulatively across its entire learning trajectory, e.g., performance on regular assignments. Whereas MAML++ was initially proposed as an engineering heuristic for stabilizing training, the results of Sec. 3 will allow us to see it as a natural consequence of the constant threat of failure-induced mortality.

[6]This term was coined by Simpson (1953) as a consolidation of the independent proposals of Baldwin (1896), Morgan (1896), and Osborn (1897). A cornerstone result by Hinton and Nowlan (1987) introduced the Baldwin effect to computer scientists. Importantly, Ackley and Littman (1992) introduced the concept of Evolutionary Reinforcement Learning (ERL) in which neural networks evolve within a complex simulated environment. More recently, Fernando et al. (2018) have applied Baldwinian Meta-Learning to deep neural networks. Moreover, Wong et al. (2026) have recently leveraged Baldwinian evolution to achieve breakthroughs in getting neural networks to internalize and simulate the laws of physics. Interestingly, both of these recent papers draw an *explicit contrast* between their Baldwinian methods and gradient-based meta-learning methods such as MAML and MAML++. By contrast, Sec. 3 will provide a model of child mortality for which the MAML++ meta-objective function *literally is* the shape of the fitness landscape that underwrites the Baldwin effect.

Indeed, all species are subject to a strong evolutionary pressure for their young to be able to quickly learn fitness-enhancing skills. Across many generations the individuals who are better at learning such skills will be selected for. The species as a whole will thereby "learn how to learn". But can the Baldwin effect really be framed as a meta-learning process in the machine-learning sense? If so, what meta-learning algorithm does Darwinian evolution employ? And what meta-objective function, if any, does it optimize?

These questions are answered in Secs. 2 and 3 where we provide a first principles derivation of the above-described meta-learning techniques from purely biological considerations (i.e., mutation and selection). In particular, Sec. 2 will introduce Darwinian Lineage Simulations (DLS) and show that the path followed by each Darwinian lineage is formally equivalent to a noisy Stochastic Gradient Ascent ($\mathcal{A}_{\text{Meta}} = \text{noisy SGA}$) up the log-fitness landscape, $\mathcal{M} = \log\mathcal{F}_\lambda$. Sec. 3 will then characterize the shape of this fitness landscape by positing a simple model of child mortality (i.e., every task failure results in a chance of instant death). According to this model the fitness landscape is exactly the MAML++ meta-objective introduced above, $\log\mathcal{F}_\lambda = \sum_n \mathcal{L}(\theta_n)$, where $\mathcal{L}(\theta)$ is the child's instantaneous log-probability of survival such that $\log\mathcal{F}_\lambda$ is their total log-probability of surviving childhood.

This Evolutionary Meta-Learning (EML) framework enables us to use neural networks to efficiently and faithfully simulate the same kinds of evolutionary pressures that shaped human cognition. Importantly, it also offers a *neutral testing ground* to adjudicate between Nativist and Empiricist adaptation strategies. Indeed, both kinds of adaptations are possible within this framework (see Sec. 3.3 for further discussion). Crucially (as Eq. (15) will show), the choice between these various adaptation strategies is ultimately made by the topography of the fitness landscape itself; it is not a brute stipulation about our model of cognition.

This brings us to yet another problem with using artificial neural networks (as they are typically trained) as scientific models of human cognition. Namely, the standard training regime implicitly adopts a broadly Empiricist epistemology as a matter of engineering fiat. Indeed, although the choice of architecture and random initialization scheme can both be given a Nativist gloss, the overall focus is still on setting the conditions for general-purpose learning upon massive amounts of data.[7] Importantly, the issue here is not *what* is stipulated per se but rather *that* it is stipulated. Let us suppose (momentarily) than an Empiricist account of epistemology is broadly correct. Even in this case, however, it should arrive in our best model of human cognition after a fair contest with a Nativist alternative and not via brute stipulation.

Concretely, if adaptations which seek out a general-purpose learning method are broadly

[7]There are two important ways in which the standard training regime does not begin from a pure tabula rasa. Firstly, there are different ways to randomly initialize the network's parameters (e.g., Xavier or Kaiming) so as to stabilize both forward and back propagation of signals in deep neural networks. These stable initializations schemes are necessary for the learning algorithm, $\mathcal{A}$, to begin doing its work. Secondly, one's choice of architecture can already encode innate concepts into the network. A prime example of this is the fact that the Convolutional Neural Network (CNN) architecture pre-loads the network with a notion of locality and translation invariance. Indeed, two of the biggest paradigm shifts in AI research were "unlocked" by finding the right architecture for the task, i.e. CNNs and LLMs.

preferred over Nativist adaptations, then some evolutionary explanation is required for this fact. For instance, are such Empiricist adaptations favored due to the complexity of the environment or its variability? Moreover, suppose that human intelligence is primarily mediated by some general-purpose learning mechanism, $\mathcal{A}^*$. Can the standard training regime teach us anything about this $\mathcal{A}^*$? What is it like? What are its evolutionary origins? How is it realized in neurons? The standard training regime is ill-equipped to answer these question because in addition to stipulating Empiricism over Nativism, it also stipulates a particular *hand-coded* learning algorithm, typically Adam or SGD. Contrast this kind of "Stipulated Empiricism" with the *Evolved Empiricism* enabled by the Evolutionary Meta-Learning (EML) framework. Namely, a general-purpose learning algorithm can be sought out by updating $\mathcal{A}$ across many outer loops. Thus, the Empiricist has much to gain by adopting the EML framework. They might: 1) properly validate themselves against the Nativists on neutral ground, and 2) develop a deeper understanding of how evolution has shaped the general-purpose learning mechanisms upon which the success of their view depends.

This critique of Stipulated Empiricism reveals a perhaps surprising misalignment between current engineering practice and the "Bitter Lesson" famously articulated by Sutton (2019). Sutton argues that the history of AI teaches us that general-purpose methods which scale with compute/data (such as search and learning) will ultimately outperform methods that don't (e.g., having a human hand-code their domain-specific expertise). Importantly, however, the standard training regime described above violates the spirit of the Bitter Lesson by relying on a *hand-coded* learning algorithm (e.g., Adam or SGD). The engineer who hard-codes the optimizer is violating the Bitter Lesson in the same way as the Symbolicist AI researcher who hand-coded grammar rules: They are both relying on human design rather than techniques that scale with compute/data. Properly applied, the Bitter Lesson points away from Stipulated Empiricism and towards Evolved Empiricism. In this light, the EML framework is not merely a tool for cognitive scientists; it is also the logical conclusion of the engineer's quest for scale. [8]

The way in which the EML framework aligns the incentives of the engineer (tools that scale with compute/data) and the incentives of the scientist/philosopher (a scientifically valuable model of cognition) persists *even if one assumes a broadly Nativist epistemology*. Indeed, when properly understood the Bitter Lesson does not recommend against innate structures; it recommends against *hand-coded* innate structures. One of the major contributions of this pa-

[8]Indeed, Evolved Empiricism is effectively the second pillar of Clune's (2020) recommended path to Artificial General Intelligence (AGI), namely via AI-generating algorithms. But what about the other two pillars? For the sake of simplicity, this paper has set aside the first pillar (meta-learning architectures) with our focus instead being on meta-learning an initialization, θ_0, within a fixed architecture. Importantly, however, the EML framework (and Scalable Nativism) could, in principle, be expanded to include a Neural Architecture Search (NAS). Clune's (2020) third pillar (automatically generating effective learning environments) is the most technically challenging and least explored. In the context of EML, this pillar becomes Niche Construction: where the network plays a substantial role in determining its own fitness environment. This final application of the Bitter Lesson removes human expertise from the process entirely, since the AI would then be giving itself learning tasks to accelerate its progress along the other two pillars. Thus the logic of the Bitter Lesson ultimately leads to a complete removal of human expertise from the design loop, recursively self-improving AI, and thereby the *Bitter Singularity*.

per is to establish *Scalable Nativism* by giving Nativists a new set of Bitter-Lesson-Compatible tools. Concretely, the EML framework is a scalable general purpose method for constructing domain-specific innate structures by leveraging massive amounts of compute/data. Rather than having some expert human hand-code their domain-specific knowledge (a hallmark of GOFAI), Nativists can instead evolve innate structures in silico by identifying the relevant fitness pressures and then running an evolutionary simulation. This shift from expert to evolution has a deep historical analog; Whereas early Nativists suggested a theological origin for our innate cognitive structures (e.g., via God's expertise) later Nativists have postulated an evolutionary origin.[9]

Let us briefly estimate the amount of linguistic data which Darwinian evolution has had access to in shaping our capacity for language acquisition. One can imagine an unbroken parent-to-child baton-pass of linguistic experience extending from the dawn of behavioral modernity to the present lasting approximately $100,000$ years. Assuming a daily intake of roughly $20,000$ words this amounts to 730 billion words total. It is a remarkable coincidence that the recent success of LLMs came just as they achieved this evolutionary scale of data intake; GPT-3's training scale was approximately 300 billion tokens (Brown et al.; 2020) whereas current training scales are often in the trillions of tokens. Whereas the engineers have already deployed such an evolutionary scale of compute/data towards ontogenic development, we propose that scientists should build their own custom AI systems with this level of phylogenic development. Concretely, by leveraging "Bitter Lesson" scales of data across many outer loops it should be possible to achieve "Poverty of Stimulus" levels of data efficiency for the inner loops.[10]

Unfortunately, the current benchmarks for modeling language acquisition are not aligned with this evolutionary perspective. For instance, consider the BabyLM Challenge (Charpentier et al.; 2026) which articulates one of their goals as follows: "Second, improving our ability to train LMs on the same kinds and quantities of data that humans learn from [merely 100M words] hopefully will give us greater access to plausible cognitive models of humans and help us understand what allows humans to acquire language so efficiently." Importantly, however, the BabyLM Challenge provides no guidelines to enforce that submissions are in any way analogous to human cognition; Submarines are here allowed to enter what should ostensibly be a Robot Fish competition. In fact, by restricting the total data allowance to a strictly ontogenic

[9]See Samet et al. (2024) for an overview of this history. See also Chomsky (1998, pg.40) regarding Plato's solution to the Meno problem: "To render Plato's answer intelligible, we have to provide a mechanism by which our knowledge is remembered from an earlier existence. ... In modern terms, that means reconstructing Platonic 'remembrance' in terms of the genetic endowment, which specifies the initial state of the language faculty".

[10]For a powerful statement of this point by an early AI pioneer, see Moravec (1988): "Encoded in the large, highly evolved sensory and motor portions of the human brain is a billion years of experience about the nature of the world and how to survive in it. The deliberate process we call reasoning is, I believe, the thinnest veneer of human thought, effective only because it is supported by this much older and much more powerful, though usually unconscious, sensorimotor knowledge. We are all prodigious Olympians in perceptual and motor areas, so good that we make the difficult look easy. Abstract thought, though, is a new trick, perhaps less than 100 thousand years old. We have not yet mastered it. It is not all that intrinsically difficult; it just seems so when we do it." See also Mu (2025).

scale (100M words) they implicitly disallow evolutionary approaches to language acquisition. This framing of the challenge commits all of the above-discussed sins: It is disconnected from evolutionary biology, it violates the Bitter Lesson by forbidding the use of large amounts of compute/data. It thus forces it participants either stipulate a broadly Empiricist epistemology, or to hand-code Nativist structures. One can improve the scientific value of this challenge as follows: Enforce an evolutionary approach with only the inner loops being subjected to a poverty-of-stimulus condition. The challenge would then be to achieve human levels of language acquisition on unseen languages by designing biologically plausible architectures, fitness environments, and evolutionary dynamics (e.g., the EML framework).[11]

The remainder of this paper proceeds as follows. Sec. 2 introduces Darwinian Lineage Simulations and proves that they are formally equivalent to a noisy Stochastic Gradient Ascent up the log-fitness landscape. Sec. 3 will then provide a model of some Baldwin-like evolutionary pressures and thereby derive the MAML++ meta-objective function from purely biological considerations. Sec. 4 concludes by providing some further discussion of historical framing and significance of this work.

2 Darwinian Lineages Evolve via Stochastic Gradient Ascent

This section will briefly introduce a new formalism for simulating how mutation and selection pressures guide a Darwinian lineage up an ever-changing fitness landscape. As we shall see, the dynamics of a *Darwinian Lineage Simulation* (DLS) is formally equivalent to a noisy Stochastic Gradient Ascent (SGA) up the log-fitness landscape. This result allows us to perform biologically faithful simulations of evolution using the extremely efficient tools of machine learning.

The results of this section also constitute a first-principles derivation of the meta-learning framework introduced in Sec. 1. Namely, *from purely biological considerations*, we find that the outer loop's meta-learning algorithm is a noisy version of Stochastic Gradient Ascent, $\mathcal{A}_{\text{Meta}} =$ noisy SGA, with respects to a certain meta-objective, $\log \mathcal{F}_\lambda$, the log of a stochastic fitness landscape. This connection will be made explicit in Sec. 3.2 where a model of child mortality will be put forward which directly relates the fitness of a neural network to its initialization, θ_0, as well as the details of its learning algorithm, $\mathcal{A}$.

The model organisms considered in a Darwinian Lineage Simulation have two notable properties: they have a continuous genotype, $\phi \in \mathbb{R}^N$, and they undergo asexual reproduction. Just like any other Darwinian organism, a population with a distribution of different genotypes, $P_g(\phi)$, will be affected by the mechanisms of mutation and selection from one generation to the next, $P_{g+1}(\phi)$. In each generation, only some survive to maturity; of those, only some will

[11] A harder but more revealing challenge would be to study in silico how the cultural evolution of language relates to the biological evolution of our language faculty. Such a co-evolutionary story can be found in Deacon (1997).

reproduce, having a variable number of children. We can combine all of these considerations into a multiplicative fitness rating, $\mathcal{F}(\phi)$, for an individual with genotype, ϕ. Soon we will add some stochastic elements to $\mathcal{F}$ to capture both intra- and inter-generational variability (denoted η and λ). Later in Sec. 3 we will provide a concrete model of this stochastic fitness function. For now, however, let us continue on abstractly. If there were no mutations arising between generations then the initial distribution of the next generation, $P_{g+1}(\phi)$, would be the product of $\mathcal{F}(\phi)$ and $P_g(\phi)$, at least up to normalization.

Importantly, however, mutations do occur such that whenever an individual reproduces (asexually) their continuous genotype, ϕ, is randomly modified in some way before being received by each of their children. We shall model the mutation profile for this species using a Gaussian distribution, $\mathcal{N}(\phi; 0, \mu^2\, I)$, centered at zero and with a constant mutation rate of μ in all directions. These mutations provide the genetic variation upon which selection can then act. Combining these selection and mutation effects together one finds the following generational update rule:

Non-Stochastic Generational Update Rule:

$$P_{g+1}(\phi) \propto (\mathcal{F}(\phi)\; P_g(\phi)) \, * \, \mathcal{N}(\phi; 0, \mu^2\, I). \tag{3}$$

The $\propto$ symbol indicates proportionality (the formula on the right is not properly normalized). The symbol $*$ indicates the convolution operation.

Before moving on, let us make a quick improvement to this generational update rule by adding intra- and inter-generational stochasticity. Different individuals within the same generation will be subjected to different fitness pressures; Identical twins might, by random chance, encounter different amounts of food and predators. To capture this intra-generational variability let us add an index η to the fitness function. Similarly, we shall add an index λ to the fitness function to capture inter-generational variability. Consider our running example for this paper: a species of bird where each generation is born into a different environment and so must learn to identify a different set of predators than the previous generation. For instance, whereas one generation might need to identify alligators against a murky water background, the next generation could be tasked with identifying brown snakes against a dry red dirt background. Alternatively, consider a linguistic example where although your parent's generation had to learn German, your generation had to learn English. We can easily generalize Eq. (3) to account for both kinds of variability (η and λ). We simply have to use the stochastic fitness function, $\mathcal{F}_{\lambda,\eta}(\phi)$, and average over the intra-generational variability η as follows:

Stochastic Generational Update Rule:

$$\begin{aligned} P_{g+1}(\phi) &\propto \int p(\eta|\phi,\lambda)\; (\mathcal{F}_{\lambda,\eta}(\phi)\; P_g(\phi)) \, * \, \mathcal{N}(\phi; 0, \mu^2\, I)\, \mathrm{d}\eta, \\ &\propto \left(\overline{\mathcal{F}_\lambda}(\phi)\; P_g(\phi)\right) \, * \, \mathcal{N}(\phi; 0, \mu^2\, I). \end{aligned} \tag{4}$$

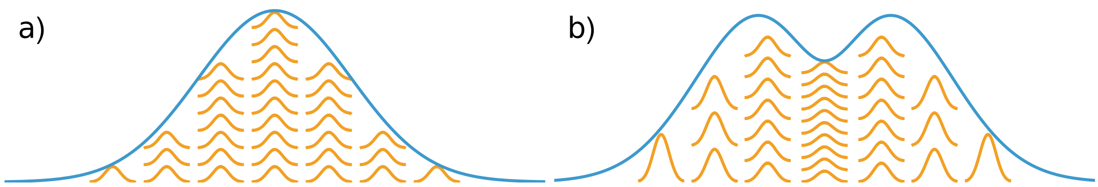

Figure 1: a) A total population (blue) can be be viewed as a "super-distribution" of sub-populations (orange). b) Each sub-populations can be evolved individually (orange) and then recombine back into the evolved total population (blue).

We average over η because these intra-generational variations each have their own chance, $p(\eta|\phi,\lambda)$, of being felt by the individuals in this generation.[12] We do not average over λ because (by definition) inter-generational variations are felt in common across an entire generation. Comparing this update rule with our previous one, Eq. (3), we see that it is exactly the same except with the old fitness function, $\mathcal{F}(\phi)$, being replaced by a new λ-indexed averaged fitness function, $\overline{\mathcal{F}_\lambda}(\phi) := \int p(\eta|\phi,\lambda)\ \mathcal{F}_{\lambda,\eta}(\phi)\ \mathrm{d}\eta$. To reduce clutter in our notation, we shall now drop the line over $\mathcal{F}_\lambda$ and implicitly assume that we have always already averaged over intra-generational variations in the fitness function.[13]

It is, unfortunately, difficult to track the genotype distribution of the full population through repeated applications of Eq. (4). Fortunately, however, Fig. 1a illustrates how we can greatly simplify this task (without approximation) by dividing the total population (blue) into a "super-distribution" of sub-populations (orange). Importantly, this kind of decomposition is also possible when the total population is non-Gaussian. Such a decomposition into sub-populations is useful because the generational update equation is linear with respect to $P_g(\phi)$, at least up to normalization. It follows from this linearity that one can apply Eq. (4) to all of the sub-populations individually and then recombine them back together. The result of this process is illustrated in Fig. 1b. If the sub-populations are each weighted by their average fitness when recombined, then the overall result is exactly the same as simply evolving the total population once according to Eq. (4).

Moreover, one could decompose the total population into sub-populations and then evolve them all separately across twenty generations. Each sub-population would thereby generate its own *Darwinian lineage*. Recombining these lineages (when appropriately weighted by their relative fitness ratings) would yield the exact same result as applying Eq. (4) twenty times to the full population. In this way we can recover the dynamics of the full population's genotype distribution by following an ensemble of Darwinian lineages.

Let us then turn our attention to one of these subpopulations and its resulting Darwinian

[12]The possible dependence of η on ϕ is one way in which this model could capture Niche Construction: where genes influence the shape of the fitness landscape that they evolve in. While nothing that follows depends upon this possibility, it is relevant for the third pillar of Clune's (2020) path to AGI, see footnote 8.

[13]If one wants to later reintroduce the line over $\mathcal{F}_\lambda$ a bit of caution is required because $\log(\overline{\mathcal{F}_\lambda})(\phi) \geq \overline{\log(\mathcal{F}_\lambda)}(\phi)$ by Jensen's inequality. The two expressions converge, however, in the limit where the average fitness $\overline{\mathcal{F}_\lambda}(\phi)$ is large compared to the typical intra-generational variations in fitness due to η.

lineage. In particular, let us choose a sub-population whose genotype distribution, $P_g(\phi)$, is an isotropic Gaussian distribution with mean, ϕ_g, and variance, $\sigma_g^2 I$. Assuming that σ_g^2 is sufficiently small, we can efficiently track the genotype distribution from one generation to the next via the following update rule:[14]

Stochastic Generational Update Rule:

$$\phi_{g+1} = \phi_g + \sigma_g^2 \nabla \log(\mathcal{F}_\lambda)(\phi_g), \tag{5}$$

$$\sigma_{g+1}^2 = \sigma_g^2 + \mu^2. \tag{6}$$

Notice that at each generational step, the gradient of the log-fitness function, $\nabla \log(\mathcal{F}_\lambda)$, is evaluated at a different genotypic location (ϕ_g changes) and upon a different fitness landscape (the λ-index changes). Concurrently, the population's variance steadily increases by μ^2 generation after generation due to accumulated mutations. In direct alignment with Fisher's (1930) Fundamental Theorem of Natural Selection, this variance acts as the (meta-)learning rate of evolution, effectively accelerating the evolutionary process. See Fig. 2.

This constant increase of population variance ultimately undermines the small-variance assumption which was used in deriving the above update rule. To continue tracking the lineage efficiently, we must introduce a method of "down-sampling" to gain control over the variance of our tracked population. Fortunately, we have already discussed a means of achieving this goal. Fig. 1a illustrates the fact that any Gaussian population can be viewed as a Gaussian "super-distribution" of Gaussian sub-populations. Hence we can reduce the variance of our tracked population by randomly selecting one of these smaller sub-populations to follow. Quantitatively, we can pick a parameter $\beta_g^2 \geq 0$ to reduce the population's variance by as $\sigma_{g+1}^2 = \sigma_g^2 + \mu^2 - \beta_g^2$. Importantly, one can still recover the full population dynamics by simply running an ensemble of these down-sampled lineage simulations and then appropriately recombining them (i.e., weighted by there respective fitness ratings).

Because the mean of a randomly chosen sub-population may differ from the mean of the total population, randomly down-sampling in this way introduces a noise term, ξ_g, into our generational update rule. Incorporating this down-sampling yields the definitive Darwinian

[14]This result follows directly from unpacking the technical condition for σ_g to be considered small, which comes in two parts. Firstly, we require $\log(\mathcal{F}_\lambda)(\phi)$ to be approximately quadratic within a radius of several σ_g of ϕ_g. Together with our assumption that $P_g(\phi)$ is Gaussian it then follows from the form of Eq. (4) that $P_{g+1}(\phi)$ is approximately Gaussian as well. It is then easy to calculate the mean and variance of $P_{g+1}(\phi)$ in terms of H_g, the Hessian of $\log(\mathcal{F}_\lambda)$ at ϕ_g. Secondly, we need for σ_g^2 to be small with respects to this Hessian such that $(I - \sigma_g^2 H_g)^{-1} \approx I$ is approximately the identity matrix. Under this further approximation, the mean and variance of $P_{g+1}(\phi)$ are given by Eqs. (5) and (6) as claimed.

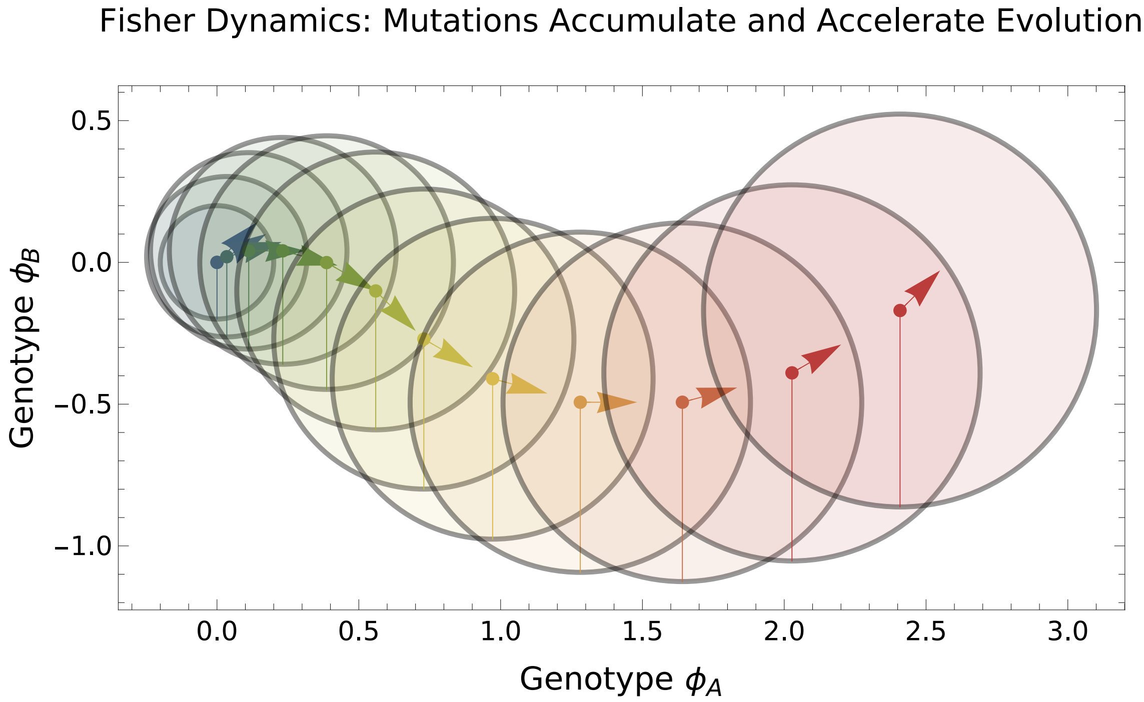

Figure 2: The trajectory of a Darwinian lineage is plotted in a 2D genotype space (ϕ_A, ϕ_B) without down-sampling. Each shaded disk represents the population's genotype distribution, $P_g(\phi)$, at a specific generation. The point at the center of each disk represents the mean genotype, ϕ_g, whereas the radius represents its variance, σ_g. The arrows indicate the gradient of the log-fitness function, $\nabla \log(\mathcal{F}_\lambda)(\phi_g)$, for each generation. Notice that even as these gradients remain the same size, the genotype's rate of change, $\Delta\phi_g = \sigma_g^2 \nabla \log(\mathcal{F}_\lambda)(\phi_g)$, is increasing. This is because (in alignment with Fisher's Theorem) the population's current amount of variation, σ_g^2, acts as its meta-learning rate. Beginning from and individual (without down-sampling) the population's variance grows as $\sigma_g = \sqrt{g}\,\mu$ where μ is the mutation rate. At generation $g = 0$, there is no variance and hence no selection pressure. Subsequently, however, mutations accumulate and accelerate the evolutionary process. This acceleration continues at least until σ_g becomes comparable to the scale of the second derivative of $\log(\mathcal{F}_\lambda)$. The need to control the variance of the population that we are tracking motivates the down-sampling procedure introduced below.

Lineage Simulation (DLS) update rule:

DLS Update Rule (with Random Down-Sampling):

$$\phi_{g+1} = \phi_g + \sigma_g^2 \nabla \log(\mathcal{F}_\lambda)(\phi_g) + \xi_g \qquad \text{where } \xi_g \sim \mathcal{N}(0, \beta_g^2 I), \tag{7}$$

$$\sigma_{g+1}^2 = \sigma_g^2 + \mu^2 - \beta_g^2. \tag{8}$$

By setting the noise scale β_g appropriately, we gain full control over the variance of the tracked population and can thereby guarantee that our small variance assumption is always satisfied. As promised, this result demonstrates that a Darwinian Lineage Simulation is formally equivalent to a noisy Stochastic Gradient Ascent ($\mathcal{A}_{\text{Meta}}$ = noisy SGA) up an ever-changing log-fitness landscape, $\mathcal{M}(\theta_0, \mathcal{A}) = \log \mathcal{F}_\lambda(\phi)$. This will constitute the outer loop of the meta-learning process described in Sec. 1.

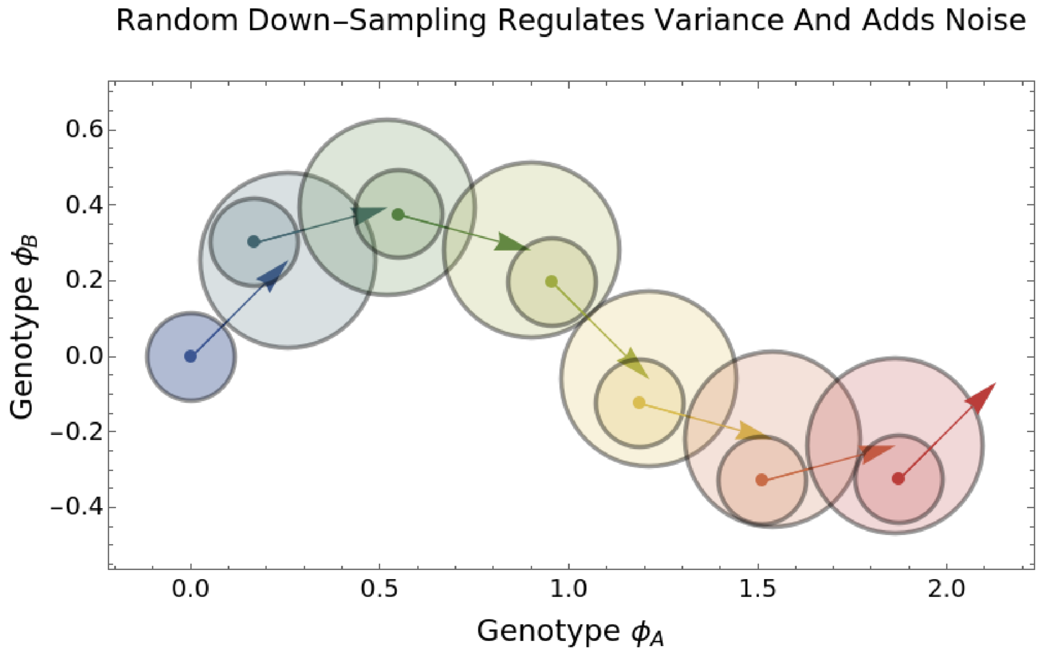

Figure 3: The trajectory of a Darwinian lineage is plotted in a 2D genotype space (ϕ_A, ϕ_B) with random down-sampling. The left-most disk at $(0, 0)$ represents the Gaussian distribution, $P_0(\phi)$, for the population's genotype at generation $g = 0$. The point at the center of this disk represents the mean genotype, ϕ_0, whereas its radius represents the population's variance, σ_0. The arrow beginning at $(0, 0)$ indicates how selection effects change the mean genotype, $\Delta\phi_g = \sigma_g^2 \nabla \log(\mathcal{F}_\lambda)(\phi_g)$. As discussed in the main text, mutations naturally increase the population's variance each generation as $\sigma_{g+1}^2 = \sigma_g^2 + \mu^2$. Hence, each generational update naturally leads to a larger population variance (indicated as a larger disk). We can, however, reduce the variance of the population that we are tracking by randomly selecting a Gaussian sub-population to continue following. As desired, the variance of this sub-population is smaller, $\sigma_{g+1}^2 = \sigma_g^2 + \mu^2 - \beta_g^2$. However, since this random sub-population may have a different mean than the larger population, a noise term, $\xi_g \sim \mathcal{N}(0, \beta_g^2 I)$, must be added to our revised update equation for ϕ_g.

3 Modeling the Fitness Landscape Behind the Baldwin Effect

The previous section has shown how Darwinian Lineage Simulations can be used to efficiently and faithfully simulate how evolutionary dynamics drive a lineage up the log-fitness landscape. For this to be scientifically useful, however, we must use a biologically well-motivated model of the fitness landscape. In this section we ask: What is the overall "shape" of the fitness landscape which underwrites the Baldwin effect? To this end, we shall posit a simple model of child mortality: every failure (e.g., at a predator identification task) results in a chance ϵ of instant death. As we shall see, this constant threat of failure-induced death causes the fitness function to depend upon the entire learning trajectory, not just its endpoint. Concretely, we find that the log-fitness landscape is exactly the MAML++ meta-objective introduced above, see Eq. (2). Whereas we are here focused upon modeling the time-pressure behind Baldwinian meta-learning, future work could include the metabolic costs of learning. Indeed, learning costs calories. Importantly, however, these metabolic effects should point in effectively the same direction as our current model, i.e., towards achieving high-performance in less time (and using less calories).

Sec. 3.1 will use our model of child mortality to compute the average fitness of the children with a particular sequence of successes/failures. Sec. 3.2 will then apply this model of child mortality to a neural network whose genotype, ϕ, determines both it initialization, $\theta_0(\phi)$, and its learning algorithm, $\mathcal{A}(\phi)$. The resulting log-fitness rating, $\log \mathcal{F}_\lambda$, is exactly the MAML++ meta-objective. Recall from Sec. 2 that it is the shape of the fitness landscape, $\nabla \log \mathcal{F}_\lambda$, which determines how the mean genotype, ϕ_g, changes from generation to generation. Sec. 3.3 will decompose this meta-gradient into two terms: one which updates ϕ in light of its effect on the network's initialization, $\theta_0(\phi)$, and one which updates ϕ in light of its effect on the learning algorithm, $\mathcal{A}(\phi)$. As we shall discuss, these evolutionary dynamics could result in fitness-relevant information being encoded into it θ_0 (a Nativist adaptation). Alternatively, these dynamics could enable a wide range of learning dynamics from rigid canalization (another Nativist adaptation) to domain-general associationism (an Empiricist adaptation). Importantly, it is the topography of the fitness landscape which will ultimately decide between these various adaptation strategies. This is our neutral testing ground.

3.1 Modeling the Fitness of a Given Success/Failure History

Let us return to our running example for this paper: a species of bird where each generation is born into a different environment with a different set of predators. Suppose that whenever a nearby predator attacks the bird's flock, those who had successfully identified its location/type are guaranteed to escape. All those who failed, however, have a chance ϵ of instant death. We shall model the childhood of such a bird as a sequence of N predator identification tasks ($n = 0, \ldots, N-1$) followed by a single opportunity for reproduction for any remaining survivors. (We shall briefly consider some alterations to this fitness model at the end of this subsection.)

Importantly, this model of fitness is compatible with a great deal of intra- and inter-generational variability (indexed by η and λ respectively) in addition to genetic variability. Indeed, some birds will simply be better at learning than others on the basis of their genotypes, ϕ. Moreover, even if two identical twins happen to share the exact same success/failure history, one of them may encounter a lethal failure where the other does not. Even assuming these twins both make it to the end of childhood, they may have different numbers of children. Finally, as noted above, the set of predators and environmental conditions which the birds have to deal with may change from generation to generation. All of this variability will now go into determining the fitness rating, $\mathcal{F}_{\lambda,\eta}(\phi)$, of each individual bird.

In what follows, λ, η, and ϕ are understood as being so detailed as to provide a deterministic description of our model organism's life (i.e., they include all hidden variables). In particular, specifying λ, η, and ϕ determines an individual's entire failure history, $F_n(\lambda, \eta, \phi) \in \{0, 1\}$,

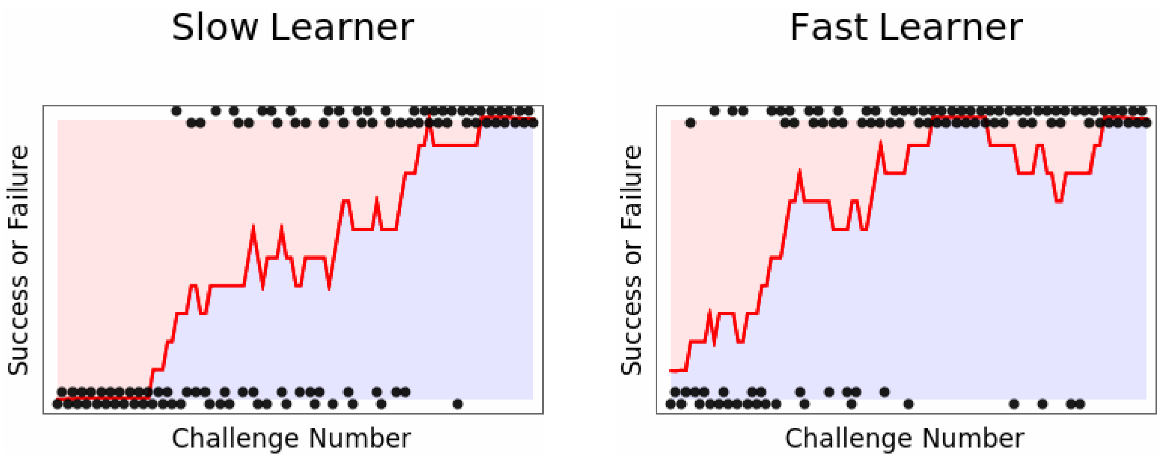

Figure 4: Two success/failure histories are plotted corresponding to a slow learner (left) and a fast learner (right). The black points at the top and bottom of each figure mark their successes and failures respectively (offset slightly for visibility). In each subfigure, the red line traces a moving average of the success rate. Crucially, the shaded region above this curve (red) corresponds to the accumulated number of failures, N_F, whereas the area below the curve (blue) corresponds to the accumulated number of successes, N_S. Since each failure carries a risk of instant death, ϵ, evolutionary dynamics seeks to reduce the total number of failures, N_F, or equivalently to maximize the area below the curve, N_S.

i.e., a binary record of whether or not they failed (or would have failed) their n^{th} task. Fig. 4 shows two such histories (for a slow learner and a fast learner) with their successes and failures being marked by the black points at the top and bottom of the figures respectively (offset slightly for visibility). Moreover, these three factors determine, $c_N(\lambda, \eta, \phi) \in \mathbb{N}$, the number of children this individual has (or would have had) after surviving all N tasks. Similarly, they determine, $\epsilon_n(\eta) \in \{0, 1\}$ a binary record of which tasks would be lethal if failed. In total, therefore, the fitness rating of an η-index individual with genotype ϕ in a λ-indexed generation is,

$$\mathcal{F}_{\lambda,\eta}(\phi) = c_N(\lambda, \eta, \phi) \prod_{n=0}^{N-1} \left(1 - \epsilon_n(\eta)\, F_n(\lambda, \eta, \phi)\right). \tag{9}$$

Notice that $\mathcal{F}_{\lambda,\eta}(\phi) \in \mathbb{N}$ is non-zero if and only if all failed tasks ($F_n = 1$) happen to be non-lethal ($\epsilon_n = 0$) and some children are had, $c_N(\lambda, \eta, \phi) > 0$.

Recall from Eq. (4) that the fitness function which appears in all of the generational update equations is not $\mathcal{F}_{\lambda,\eta}(\phi)$ but rather $\overline{\mathcal{F}_\lambda}(\phi)$, the η-averaged fitness function. We shall conduct this average over η in two stages (with the second step being completed in Sec. 3.2) by assuming that it can be divided, $\eta = (r, \sigma)$, into two *independent* factors: First, r, which determines the lethality of each task, $\epsilon_n(\eta)$, independently as well as $c_N(\lambda, \eta, \phi)$ given λ and ϕ; Second, σ, which determines all other aspects of intra-generational variability. The average over η can then be decomposed into two steps as $\mathbb{E}_\eta[\dots] = \mathbb{E}_\sigma[\mathbb{E}_{r|\sigma}[\dots]]$. Applying these independence assumptions and taking the inner-most average, one finds,

$$\overline{\mathcal{F}}_{\lambda,\sigma}(\phi) = c(\lambda,\phi) \prod_{n=0}^{N-1} \left(1 - \epsilon\, F_n(\lambda,\sigma,\phi)\right), \tag{10}$$

where $c(\lambda,\phi) := \mathbb{E}_r[c_N(\lambda, r, \phi)]$ captures reproductive capacity and where the average lethality of failure, $\epsilon := \mathbb{E}_r[\epsilon_n(r)]$, is assumed to be time-independent. Note that each failure contributes a factor of $1-\epsilon$ to the product whereas each success contributes a factor of 1. Using this fact and taking the logarithm we have,

$$\log \overline{\mathcal{F}}_{\lambda,\sigma}(\phi) = \underbrace{\log\ c(\lambda,\phi)}_{\text{Reproduction}} + \underbrace{\log(1-\epsilon)}_{\text{Danger}} \underbrace{\sum_{n=0}^{N-1} F_n(\lambda,\sigma,\phi)}_{\text{Failure Count}}. \tag{11}$$

This is the log of the mean fitness for every individual who has a failure history of $F_n(\lambda,\sigma,\phi)$. Notice that this log-fitness rating includes a term relating to reproduction as well as a survival term which is proportional to the total number of failures, $N_F(\lambda,\sigma,\phi) := \sum_{n=0}^{N-1} F_n(\lambda,\sigma,\phi)$ multiplied by a negative factor, $\log(1-\epsilon) < 0$, which quantifies how dangerous each failure is. Intuitively, to maximize fitness the number of failures should be minimized.

Notice that (according to this model, at least) early failures and late failures are equally harmful to one's expected fitness: both lessen the odds of surviving childhood (and hence of reproduction) by a factor of $(1-\epsilon)$. This does not mean, however, that late-learning individuals are as fit as early-learning ones. Indeed, as Fig. 4 shows, a slow learner (left) will have many more failures than a fast learner (i.e., more black dots at the bottom of the figure). Correspondingly, the slow learner has a much higher chance of dying before their chance at reproduction. In quantifying the relative fitness of these two learning profiles, it is helpful to consider the moving average of their success rates. This is shown in Fig. 4 as a red curve. The area below the moving average curve (blue) is proportional to the total number of successes, N_S, whereas the area above the curve (red) is proportional to the total number of failures, N_F. Comparing these shaded areas immediately shows that the fast learner has a much smaller chance of death than the slow learner even though they end up having the exact same level of competence.

This relationship between log-fitness and the area above/below the learning curve bears a striking resemblance to the MAML++ meta-objective introduced in Eq. (2). In particular, the child's average success rate is here playing the role of the objective function, $\mathcal{L}$, which tracks the performance of the network over time. The MAML++ meta-objective, $\mathcal{M}$, is exactly the area under this performance curve. This connection will be developed further in Sec. 3.2.

Before this, however, let us briefly discuss how an alteration to our model of fitness changes the picture. Suppose that there are now opportunities for reproduction after every task. Assuming that ϵ is small one then finds,

$$\log \overline{\mathcal{F}}_{\lambda,\sigma}(\phi) = \log\, c(\lambda,\phi) - \epsilon \sum_{n=0}^{N-1} \left(1 - \frac{n}{N}\right)\, F_n(\lambda,\sigma,\phi) + \mathcal{O}(\epsilon^2). \tag{12}$$

This change in the opportunities for reproduction leads to a reweighting of the significance of each failure, $F_n(\lambda,\sigma,\phi)$. In brief, early failures are now more harmful to one's fitness than late failures are. The reason for this can be seen by interpreting the reweighting factor, $1 - n/N$, as the fraction of one's life (and reproductive opportunities) which lies ahead. According to this model, early failures are more harmful to one's expected fitness because they affect more opportunities for reproduction than a late failure would. This alternative model of fitness is sensitive to the distribution of failures over time, whereas our original model only cared about the total number of failures. As above, however, one can still understand this model of fitness in terms of a (weighted) area above/below the learning curve and thereby relate it to a (weighted) MAML++ meta-objective.

One can, of course, continue to play with different models of fitness by (e.g.) introducing a time-dependent fertility rate or risk-free learning period in infancy. The general point, however, remains; The constant threat of failure-induced death causes the fitness function to depend upon the entire learning trajectory, not just its endpoint. This corresponds to the difference between the MAML and MAML++ meta-objectives (Finn et al.; 2017; Antoniou et al.; 2019).

3.2 How To Evolve Your MAML

This subsection will explicitly connect all of the biological tools which we have been developing to the common practices and methods of machine learning. Continuing with our bird example, let us model the overall connectivity of the bird's brain (or at least its vision module) as a neural network with some fixed non-evolving architecture, e.g., FNN or CNN. While it is possible to have the architecture evolve (e.g., via a Neural Architecture Search, NAS) we are proceeding without this complication primarily for ease of exposition. In applications of the EML framework which use a fixed (i.e., non-evolving) architecture, it should be acknowledged that the choice between Empiricist and Nativist adaptations is being made against this fixed backdrop.

Let us model the particular connectivity of the bird's brain (vision module) at the time of each attack as its parameters, θ_n, e.g., its weights and biases. Each time a predator attacks the flock, the birds all face what amounts to a classification task: On the basis of some sensory data, x_n (e.g., vision and hearing), each bird must identify the correct label, y_n, indicating the location/type of the nearest predator. For such classification tasks, neural networks combine θ_n and x_n to output a probability vector, $\vec{p}_n(\theta_n, x_n)$, over the space of possible labels. Let us suppose that each bird randomly selects its prediction, z_n, from this probability distribution. (Technically speaking, this choice is determined by the bird's σ-index which covers all aspects of intra-generational variability, see Sec. 3.1.) A failure occurs, $F_n = 1$, if and only if this prediction is wrong, $z_n \neq y_n$. As discussed above, each failure results in a chance, ϵ, of instant death. Moreover, making it to the end of childhood (i.e., surviving all N tasks, $n = 0, \ldots, N-1$) offers a single chance of reproduction. In what follows, however, we will assume that the reproductive capacity, $c(\lambda, \phi)$, is constant so as to isolate survival effects. To reduce clutter, we will in fact assume that $c(\lambda, \phi) = 1$ so that $\log c(\lambda, \phi) = 0$ drops out of the equations entirely.

As each bird encounters (and ostensibly survives) some sequence of predators, its network's parameters, θ_n, will change in accordance with some learning algorithm, $\mathcal{A} : \theta_n \mapsto \theta_{n+1}$. For instance, these parameters could be updated after each attack so as to better align the prediction vector, $\vec{p}_n$, with the correct label, y_n.[15] In this way, each bird might learn from its experiences and, indeed, evolution will favor those who do. But what learning algorithm, $\mathcal{A}$, does the bird use? And where did its learning process begin from, θ_0? These are both fixed by the bird's genes! Namely, the bird's continuous genotype, ϕ, will specify them both $\theta_0(\phi)$ and $\mathcal{A}(\phi)$. Importantly, the population of birds will have a distribution of different genotypes (recall Sec. 2) and hence a distribution of different initializations and learning algorithms. Consequently, the birds in this flock will follow a wide variety of different learning trajectories, make different predictions, and ultimately have different survival/reproductive outcomes. This provides the variation upon which selection pressures can then act. In this way, the flock of birds will "learn how to learn".

To complete our model we must discuss the inter-generational variability, λ, of the fitness environment. As we noted above, each generation of birds will encounter a different set of predators than the previous generation because they are born into a different environment. For instance, whereas one generation might need to identifying brown snakes against a dry red dirt background, the next generation could be tasked with identifying alligators against a murky water background. Let us further suppose that the population of birds which we are tracking are always in a flock and hence will all face the same sequence of predator identification tasks. Hence, the sequence of classification tasks is assumed to have no intra-generational variability (η) but a lot of inter-generational variability (λ). This is a form of classroom learning wherein the whole generation is put through the same curriculum, but subsequent generations are put through a very different curriculum.

With these modeling assumptions in place, we can now return to Eq. (10) to complete the

[15] Beyond this simple example, evolutionary dynamics might shape $\mathcal{A}$ into something much more complicated. In what follows, however, we will demand that one aspect of this simple update strategy is maintained. Notice that the way in which θ_n is updated is independent of which particular prediction, z_n, was made; Instead, the update was computed by comparing $\vec{p}_n$ to y_n. The key fact here is that the learning algorithm, $\mathcal{A}$, does not make use of any σ-stochasticity.

average over $\eta = (r, \sigma)$ as follows:[16]

$$\mathcal{F}_\lambda(\phi) = \prod_{n=0}^{N-1} \left(1 - \epsilon\, f_n(\lambda, \phi)\right), \tag{13}$$

where $f_n(\lambda, \phi) = \mathbb{E}_\sigma\left[F_n(\phi, \sigma, \lambda)\right]$ is the probability of the bird making an incorrect prediction during the n^{th} task. It is worth reviewing, briefly, how λ and ϕ together determine this probability. Recall that the bird's genotype, ϕ, determines its initialization, $\theta_0(\phi)$, and its learning algorithm, $\mathcal{A}(\phi)$. The stochastic index, λ, for its generation (perhaps together with ϕ) determines the stream of classification tasks, (x_n, y_n). Thus, the bird's individual genotype, ϕ, together with its generational situation, λ, determine its particular sequence of brain states, θ_n, and correspondingly its prediction vectors, $\vec{p}_n(\theta_n, x_n)$. The bird's chance of failure, $f_n(\lambda, \phi)$, is just the measure of probability which $\vec{p}_n(\theta_n, x_n)$ has placed outside of the correct prediction, y_n.

We are now in a position to derive a concrete model for the stochastic log-fitness function which drives the evolutionary dynamics discussed in Sec. 2. Taking the logarithm of the above formula we have,

$$\log \mathcal{F}_\lambda(\phi) = \sum_{n=0}^{N-1} \mathcal{L}_\lambda(\theta_n) \qquad \text{where } \mathcal{L}_\lambda(\theta_n) = \log\left(1 - \epsilon\, f_n(\lambda, \phi)\right). \tag{14}$$

Indeed, as promised $\log \mathcal{F}_\lambda(\phi)$ is exactly the MAML++ meta-objective function introduced by Antoniou et al.'s (2019) paper "How to Train your MAML". Notice that $\mathcal{L}_\lambda(\theta_n)$ is the log probability of surviving the n^{th} task such that $\log \mathcal{F}_\lambda(\phi)$ is straightforwardly the log-probability of surviving childhood.

3.3 Differentiating Adaptation Strategies via the Evolutionary Meta-Gradient

To review: We have just provided a biologically well-motivated model of the fitness landscape which underwrites the Baldwin effect. Per the results of Sec. 2, a Darwinian lineage which evolves in such an environment will stochastically climb the fitness landscape. Concretely, this means selecting for individuals which minimize the area above (or, equivalently maximize the area below) the performance curve, $\mathcal{L}(\theta_n)$. By this metric, the lineage will "learn how to learn". But which kinds of adaptations will be selected for: Nativist or Empiricist? To help answer this, we will now analytically decompose the meta-gradient, $\nabla \log \mathcal{F}_\lambda(\phi)$, into two terms: one which updates ϕ in light of its effect on the initialization, $\theta_0(\phi)$, and one which updates ϕ in light of

[16] Note that we are allowed to distribute the $\mathbb{E}_\sigma$ through the product because $F_n(\phi, \sigma, \lambda)$ and $F_m(\phi, \sigma, \lambda)$ are independent variables. This follows directly from our modeling assumptions which limit σ's role to only picking each z_n from each $\vec{p}_n$. Namely, σ does not affect θ_0 or $\mathcal{A}$ or the sequence of classification tasks, (x_n, y_n), such that it does not affect the sequences of θ_n or the sequence of prediction vectors, $\vec{p}_n$. Importantly, however, σ does still affect which z_n is drawn from $\vec{p}_n$ and subsequently failure, death, and children.

its effect on the learning algorithm, $\mathcal{A}$. This decomposition reveals that evolution does not simply choose a side in the Nativism-Empiricism debate; rather, it dynamically modifies which adaptation strategy it employs based upon the local topography of the fitness landscape.

Recall from above that the bird's individual genotype, ϕ, together with its generational situation, λ, determine its particular sequence of brain states, θ_n. This in turn determines its probability of failure at each task, $f_n(\lambda, \phi)$, and thereby a log-fitness rating, $\log \mathcal{F}_\lambda(\phi)$, for the genotype, ϕ, for this generation's set of predator identification tasks, λ. By carefully unpacking all of these dependencies one can apply the chain rule to take the derivative of $\log \mathcal{F}_\lambda(\phi)$ with respect to ϕ as follows.

$$\nabla \log \mathcal{F}_\lambda(\phi) = \underbrace{\sum_{n=0}^{N-1} \left(\frac{\partial \theta_0}{\partial \phi}\right)^\intercal \left(\prod_{k=0}^{n-1} \frac{\partial \theta_{k+1}}{\partial \theta_k}\right)^\intercal \frac{\partial \mathcal{L}}{\partial \theta}\bigg|_n}_{\text{Update Based on Initialization}} + \underbrace{\sum_{n=0}^{N-1} \sum_{l=0}^{n-1} \left(\frac{\partial \mathcal{A}}{\partial \phi}\bigg|_l\right)^\intercal \left(\prod_{k=l+1}^{n-1} \frac{\partial \theta_{k+1}}{\partial \theta_k}\right)^\intercal \frac{\partial \mathcal{L}}{\partial \theta}\bigg|_n}_{\text{Update Based on Learning Algorithm}} \tag{15}$$

The first sum captures how making changes to the genotype, ϕ, will affect the initialization state, $\theta_0(\phi)$, and thereby the overall log-fitness, $\log \mathcal{F}_\lambda(\phi)$. As such, every term in this sum includes $\partial\theta_0/\partial\phi$ as a factor. Let us understand the n^{th} term in this sum by reading it from right to left, beginning with the gradient $\partial\mathcal{L}/\partial\theta|_n$. This can be understood as indicating the direction which θ_n ought to be moved to maximally increase $\mathcal{L}(\theta_n)$. This gradient is then multiplied (on the left) by a series of Jacobians, $\partial\theta_{k+1}/\partial\theta_k$, which translate this gradient signal at θ_n into a recommendation regarding how to move θ_{n-1}, and then θ_{n-2}, etc. Through all of these Jacobians, the gradient signal at θ_n is ultimately "back-propagated" to a corresponding recommendation for how to move θ_0. The last Jacobian, $\partial\theta_0/\partial\phi$ then translates this into a recommendation for how to change the genotype ϕ. Importantly, however, our goal is is not just to improve the network's performance at one time point, $\mathcal{L}(\theta_n)$, but rather across the entire learning trajectory. By accumulating across n, we find the total recommendation for how to change ϕ on the basis of how to optimize the initialization, $\theta_0(\phi)$.

By contrast, the second sum captures how making changes to the genotype, ϕ, will affect the learning algorithm, $\mathcal{A}(\phi)$, and thereby the overall log-fitness, $\log \mathcal{F}_\lambda(\phi)$. For instance, recall our example of a learning algorithm from Eq. (1) which included a pre-conditioning matrix, P, and some variable learning rates, η_n. These (and many other) aspects of the learning algorithm may be dependent upon the genotype ϕ. The $\partial\mathcal{A}/\partial\phi$ factor which appears in every term of the second sum captures all of the ways in which $\mathcal{A}(\phi)$ depends upon ϕ. As before, we can understand the n^{th} term in this sum by reading it from right to left, beginning with the gradient $\partial\mathcal{L}/\partial\theta|_n$. Once again, this recommendation for how to move θ_n is back-propagated through a sequence of Jacobians $\partial\theta_{k+1}/\partial\theta_k$. Unlike above, however, this sequence of Jacobians terminates with a recommendation for how to change θ_{l+1}. Viewing θ_{l+1} as the result of applying $\mathcal{A}(\phi)$ to θ_l, one final Jacobian, $\partial\mathcal{A}_\phi/\partial\phi|_l$, then provides a recommendation for how to change

ϕ (on the basis of how $\mathcal{A}$'s action at θ_l would affect $\mathcal{L}(\theta_n)$). Accumulated across all l and n, a total change to ϕ is thus recommended on the basis of how to optimize the learning algorithm $\mathcal{A}(\phi)$.

This decomposition of the meta-gradient clarifies the sense in which the topography of the log-fitness landscape (i.e., the shape of $\log \mathcal{F}_\lambda(\phi)$) determines which adaptations are adopted as the lineage learns how to learn. It is in this sense that evolutionary meta-learning is a *neutral testing ground* to adjudicate between Nativist and Empiricist hypotheses. Indeed, both Nativist and Empiricist adaptation strategies are possible within this framework. Consider our running example for this paper: the evolution of a vision module within a species of bird where each generation must learn to identify a different set of predators than the previous generation. Even with this highly variable fitness-environment, one might expect for θ_0 to evolve some ubiquitously useful structures such as edge detection, a foreground/background distinction, and perhaps an in-built sky/land/water-threat distinction. Such a pre-structured θ_0 could be preferred either for reasons of proximity (it is already near to a high-performance state) or for reasons of canalization (it may sit at the beginning of a pre-determined learning pathway).[17]

Whether θ_0 arises due to proximity or canalization, the baby bird will likely not be able to immediately identify its generation's predators given the variety of different predator-sets available. Some ontogenic development (repeated applications of $\mathcal{A}$ in response to data) would then be needed to learn how to properly combine the innate capabilities which are encoded into θ_0. Importantly, these θ_0-adaptations could be complemented by Nativist adaptations to the learning algorithm, $\mathcal{A}$. For instance, if the pre-conditioning matrix, P, in Eq. (1) is diagonal then it effectively assigns a plasticity score to each neuronal connection. These plasticities may evolve to prevent the unlearning of whatever fitness-enhancing information is encoded within θ_0. Alternatively, it could evolve to direct learning along some pre-determined pathways (canalization). Similarly, the time-dependent learning rates, η_n, could evolve to control the timing of specific developmental stages.

But how can Empiricist adaptations arise within the Evolutionary Meta-Learning framework? These would be modifications of the learning algorithm, $\mathcal{A}$, which are *domain-general*, e.g., not about locking in certain innate structures or canalizing certain developmental pathways. Witnessing such domain-general adaptations (by definition) requires a complex fitness environment, i.e., one comprised of multiple domains.[18] According to Godfrey-Smith's (1996) Environmental Complexity Thesis (ECT), "the function of cognition is to enable agents to deal with environmental complexity."[19] Alternatively, Potts's (1998) Variability Selection Hypothe-

[17]The concept of canalization was first introduced in Waddington (1942) and is fully elaborated in Waddington (1957).

[18]Ironically, Empiricist adaptations (as well as Nativist ones) become impossible if the domain is taken to be *fully general*. This is a consequence of the "No Free Lunch" theorem by Wolpert and Macready (1997): No learner can outperform another when performance is averaged across *all possible learning tasks*. Thus, the "general purpose" mechanisms of Empiricism cannot be completely general but only general relative to a subset of possible learning tasks (which includes, at least, those passed by our evolutionary ancestors).

[19]A more recent study by Dridi and Lehmann (2015) supports this claim.

sis (VSH) argues that human intelligence did not evolve to solve a specific ecological problem (like hunting or social maneuvering), but rather to solve the problem of environmental fluctuation itself (e.g., drastic climate shifts in the Pleistocene).[20] Concretely, $\mathcal{A}$ might evolve so as to bolster the traditional learning methods of Humean Associationism, e.g., cause and effect, contiguity, and resemblance.

These are, of course, speculations about which kinds of adaptations might be favored in which circumstances. Ultimately, however, it should be the topography of the fitness landscape which decides between these adaptation strategies, not a brute stipulation about our model of cognition.

4 Conclusion: Historical Framing and Significance

This work began with a sharp critique of using off-the-shelf AI systems as scientific models of human cognition. We argued that in order to move from "submarines" (systems optimized for engineering utility) to "robot fish" (systems optimized for biological fidelity), scientists must redeploy the mechanisms of machine learning so that they align with the dynamics of evolution. To this end, we have introduced the Evolutionary Meta-Learning (EML) framework as a means to efficiently and faithfully simulate the evolutionary pressures that shaped human cognition using artificial neural networks.

Sec. 2 has provided the evolutionary engine for this framework: A Darwinian Lineage Simulation (DLS) is formally equivalent to a noisy Stochastic Gradient Ascent (SGA) up the log-fitness landscape. Our approach ensures biological fidelity by deriving the meta-learning rates and noise scales from purely biological considerations. Specifically, we have found that the meta-learning rate must be the population variance (a consequence of Fisher's Theorem). The particular relationship between step-size and noise level which we have found, $\xi_g \sim \sqrt{\mu^2 - (\sigma_{g+1}^2 - \sigma_g^2)}$, is not an ad hoc "evolutionary temperature" but rather a necessity in order for an ensemble of lineages to faithfully recover the genotype dynamics of the full population. The equivalence between DLS and SGA allows us to reinterpret the standard machinery of deep learning not as a loose metaphor for evolution, but as a precise simulation of it.

Sec. 3 subsequently proposed a method of modeling the fitness landscape. In particular, by modeling the Baldwin effect as a selection pressure to minimize failure-induced mortality during childhood, we have demonstrated that the resulting fitness landscape corresponds to the MAML++ meta-objective function proposed by Antoniou et al. (2019). Thus, modern techniques of meta-learning algorithms are not merely an engineering trick, they can be redeployed in biologically-faithful simulations of how a Darwinian lineage "learns how to learn" in harsh, variable environments where achieving early competence is necessary for long-term survival.

The results of this paper have the potential to radically reshape the long-standing debate be-

[20]See Ellefsen (2014) for a more recent study on how environmental variability affects the evolution of learning.

tween Nativism and Empiricism. Historically, this conflict has oscillated between the Nativism of Plato and Descartes, who posited innate ideas, and the Empiricism of Locke and Hume, who championed *tabula rasa* epistemologies. In more recent centuries, this dialectic continued through Kant's categories, Jung's theory of archetypes, and most significantly for cognitive science, Chomsky's arguments for a Universal Grammar.

A critical shift in this history has been the secularization of the "origin story" for our cognitive faculties. Whereas earlier Nativists often postulated theological origins for innate structures (viewing them as the handiwork of God), later thinkers have posited evolutionary origins. By contrast, early Empiricists generally tended to avoid such theological baggage by adopting a much more individualistic account of learning.[21] Importantly, though, modern Empiricism must make an evolutionary turn just as the modern Nativists have: They must provide evolutionary and neuro-mechanistic accounts for the associationistic mechanisms that allow for general-purpose learning.

With both sides having accepted this neuro-evolutionary framing, this age-old philosophical debate can (at least, in principle) resolve itself into a scientific research program: Which kinds of fitness environments favor Nativist adaptation strategies (e.g., instincts and canalization) versus Empiricist ones (e.g., general-purpose learning)? Under which kind of conditions did each of our various cognitive faculties evolve? In either case, how are the resulting innate structures and/or general-purpose learning mechanisms realized in terms of neurons and synaptic connections?

Much scientific work has been done in this direction, but progress has been slow and laborious; Experimental neuroscience is very difficult and soft matter doesn't fossilize. While much remains unknown, the commonly accepted neuro-evolutionary framing of the remaining debates indicates a foundational consensus: *The human mind is the result of an embodied neural system having undergone a Darwinian process of evolutionary meta-learning.* That is, we have "learned how to learn" through a process of Darwinian evolution. At every point in our evolutionary history, it is ultimately the fitness landscape which has decided between Nativist and Empiricist adaptations. Thus, evolutionary meta-learning is (and, in fact, long has been) the proper scientific testing ground for Nativist and Empiricist theories of epistemology. In this light, the function of this paper is to *return* our attention to this foundational agreement, but now equipped with a new capacity to efficiently and faithfully simulate evolutionary meta-learning *in silico*.

But why is this a "return", when did we lose sight of this foundational consensus? In the 20th century, the debate between Nativism and Empiricism was operationalized into two competing AI research programs: Symbolic AI (championing innate grammar and logic) and Connectionism (championing distributed learning). Given that evolutionary scales of compute/data have only became available in the 2020s (see Sec. 1), neither of these research programs could pursue evolutionary meta-learning and hence were forced adopt an ontogenic approach (focus-

[21] See Samet et al. (2024) for an overview of this history.

ing only on a single lifetime).[22] Symbolicism hypothesized that cognition is fundamentally the manipulation of meaningful symbols in accordance with the logical/grammatical rules of thought.[23] This approach is closely related to the Computational Theory of Mind. Connectionism, by contrast, proceeded along a sub-symbolic route by using neural networks with fixed architectures and random initializations. Either way, adopting a non-evolutionary approach requires one to hand-code all of the structures and learning mechanisms that would have been produced by evolution. This includes the Symbolicist's logic and grammar rules (Hand-Coded Nativism) *as well as* the Connectionist's optimizer (Stipulated Empiricism). Both of these non-evolutionary operationalizations are scientifically limiting: they both sever the possibility of evolutionary explanations and weaken the possibility of a fruitful AI-human analogy (e.g., via convergent evolution).

As these two AI research programs developed (and as our capacity to leverage compute/data increased) the dynamics behind Sutton's Bitter Lesson led to Connectionism outperforming Symbolicism. This often-misinterpreted "Engineering Victory" for the Connectionist program gives the impression that the Empiricists are winning and that the Nativists have been left behind by recent advances in AI. Importantly, however, this historical development does not tip the scales of any scientific or philosophical debate. It simply follows from the fact that "Stipulated Empiricism" involves less hand-coding than "Hand-Coded Nativism" allowing it to scale better with increases in compute/data. The Nativists were left behind not because innate structures are non-existent or unimportant, but because they lacked a method to produce them that scales with compute/data. In this light, the function of this paper is to establish *Scalable Nativism* by allowing us to efficiently and faithfully simulate the evolution of innate structures in silico. Moreover, this paper also enables and recommends *Evolved Empiricism* by allowing us to simulate the evolution of a general-purpose learning mechanism in silico.

Hence, this paper recommends *returning* to the foundational consensus of evolutionary meta-learning with our newfound capacity for simulating evolution *in silico*. Indeed, whereas neural networks were originally a site of great conflict in the Nativism-Empiricism debate (see our opening question in Sec. 1) they can now be redeployed as neutral testing ground upon which this debate can be empirically adjudicated. Evolutionary scales of data are now available and scientists should study the effects of applying this scale of phylogenic development to AI systems. In this way, the EML framework offers a brand new way for us to make scientific progress in the long-standing debates between Nativists and Empiricists. Indeed, Sec. 3.3 has shown that we can decompose the evolutionary meta-gradient into two terms each of which can be analyzed as a combination of Nativist and Empiricist adaptations. Future in silico experiments in complex, multi-modal environments will allow us to probe exactly which features of the environment trigger which adaptive strategy. We can also investigate how these adaptations

[22] This is not to say that no evolutionary approaches were pursued in this time-period, but any such attempts would have to have been severely simplified to be computationally feasible.

[23] See the Physical Symbol System Hypothesis of Newell and Simon (1976).

are realized in the connectivity of (artificial) neurons.

In closing, let us consider how this work might reshape our self-concept. We have long seen ourselves reflected in the current-best technology of the day. In the 18th century, the mind was clockwork; in the 19th, a steam engine; in the 20th, a digital computer (The Computational Theory of Mind); and now, it is an artificial neural network. This historical tendency of projecting ourselves onto the latest tech is often folly because (as we noted in Sec. 1) good scientific models do not simply fall off of the engineering tree! The spirit of this work, however, has been to take our current-best technology (neural networks) and carefully redeploy it so as to purposefully bolster how analogous it is to the human mind. The resulting AI systems are (artificial) neural networks that have undergone a Darwinian process of evolutionary meta-learning. Similarly, the human mind is a (biological, embodied) neural network that has undergone a Darwinian process of evolutionary meta-learning. Importantly, we have known this fact about ourselves long prior to the AI revolution. As such, the AI-human analogies proposed in this paper are not merely a psychological projection. Rather than build AI systems according to "how we think we think" (Sutton; 2019), we can instead evolve them according to how we *know* we evolved.

References

Ackley, D. and Littman, M. (1992). Interactions between Learning and Evolution, *in* C. G. Langton, C. Taylor, C. D. Farmer and R. S. (eds), *Artificial Life II, SFI Studies in the Sciences of Complexity*, Vol. X, Addison-Wesley, Reading, MA, USA, pp. 487–509.

Antoniou, A., Edwards, H. and Storkey, A. (2019). How to Train your MAML, *International Conference on Learning Representations*.

Baldwin, J. M. (1896). A New Factor in Evolution, *The American Naturalist* **30**(354): 441–451.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I. and Amodei, D. (2020). Language Models are Few-Shot Learners.

Charpentier, L., Choshen, L., Cotterell, R., Gul, M. O., Hu, M., Liu, J., Jumelet, J., Linzen, T., Mueller, A., Ross, C., Shah, R. S., Warstadt, A., Wilcox, E. and Williams, A. (2026). BabyLM Challenge, `https://babylm.github.io/index.html`. [Accessed 23-02-2026].

Chomsky, N. (1998). Nature, Use, and Acquisition of Language, Brill, Leiden, The Netherlands, chapter 2, pp. 33 – 54.

Clune, J. (2020). AI-GAs: AI-Generating Algorithms, an Alternate Paradigm for Producing General Artificial Intelligence.

Deacon, T. W. (1997). *The Symbolic Species: The Co-Evolution of Language and the Human Brain*, Penguin, London.

Dijkstra, E. W. (1984). The Threats to Computing Science, Circulated privately.

Dridi, S. and Lehmann, L. (2015). Environmental Complexity Favors the Evolution of Learning, *Behavioral Ecology* **27**(3): 842–850.

Ellefsen, K. O. (2014). The Evolution of Learning Under Environmental Variability, pp. 649–656.

Fernando, C., Sygnowski, J., Osindero, S., Wang, J., Schaul, T., Teplyashin, D., Sprechmann, P., Pritzel, A. and Rusu, A. (2018). Meta-learning by the Baldwin Effect, *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 1313–1320.

Finn, C., Abbeel, P. and Levine, S. (2017). Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks, *Proceedings of the 34th International Conference on Machine Learning*, PMLR, pp. 1126–1135.

Fisher, R. A. (1930). *The Genetical Theory of Natural Selection*, Clarendon Press.

Godfrey-Smith, P. (1996). *Complexity and the Function of Mind in Nature*, Cambridge Studies in Philosophy and Biology, 1st ed. edn, Cambridge University Press, Cambridge.

Hinton, G. E. and Nowlan, S. J. (1987). How Learning can Guide Evolution, *Complex Systems* **1**(3): 495–502.

Moravec, H. P. (1988). *Mind Children: the Future of Robot and Human Intelligence*, Harvard University Press, Cambridge, Mass;.

Morgan, C. L. (1896). On Modification and Variation, *Diacronia* (10): (1–6).

Mu, N. (2025). The myth of data inefficiency in large language models. https://www.normanmu.com/2025/02/14/data-inefficiency-llms.html.

Newell, A. and Simon, H. A. (1976). Computer Science as Empirical Inquiry: Symbols and Search, *Commun. ACM* **19**(3): 113–126.

Osborn, H. F. (1897). The Limits of Organic Selection, *Diacronia* (4): (1–4).

Potts, R. (1998). Variability Selection in Hominid Evolution, *Evolutionary Anthropology: Issues, News, and Reviews* **7**(3): 81–96.

Samet, J., Bronstein, D., Brown, D., Helmig, C. and Normore, C. (2024). The Historical Controversies Surrounding Innateness, *in* E. N. Zalta and U. Nodelman (eds), *The Stanford Encyclopedia of Philosophy*, Summer 2024 edn, Metaphysics Research Lab, Stanford University.

Simpson, G. G. (1953). The Baldwin Effect, *Evolution* **7**(2): 110–117.

Sutton, R. (2019). The Bitter Lesson, Incomplete Ideas (Blog).

Waddington, C. H. (1942). Canalization of Development and the Inheritance of Acquired Characters, *Nature (London)* **150**(3811): 563–565.

Waddington, C. H. (1957). *The Strategy of the Genes: A Discussion of Some Aspects of Theoretical Biology*, George Allen & Unwin, London.

Wolpert, D. and Macready, W. (1997). No Free Lunch Theorems for Optimization, *IEEE Transactions on Evolutionary Computation* **1**(1): 67–82.

Wong, J. C., Ooi, C. C., Gupta, A., Chiu, P.-H., Low, J. S. Z., Dao, M. H. and Ong, Y.-S. (2026). Evolutionary Optimization of Physics-Informed Neural Networks: Advancing Generalizability by the Baldwin Effect.