bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Complex Computation from Developmental Priors

Dániel L. Barabási*

Biophysics Program, Harvard University, Cambridge, MA, USA

Taliesin Beynon

Robotics and Articial Intelligence Laboratory,

University of Witwatersrand, Johannesburg, South Africa

Abstract

Artificial Inteligence (AI) research has provided key insights into the mechanics of learning complex tasks. However, AI models have long overlooked innateness: how strong pressures for survival lead to the encoding of complex behaviors in the nascent wiring of a brain. Although innate neural solutions have inspired AI approaches from layered architectures to ConvNets, the underlying neuroevolutionary search for novel heuristics has not been succesfully systematized. In this manuscript, we examine how neurodevelopmental principles can inform the discovery of computational heuristics. We begin by considering the weight matrix of a neural network to be emergent from well-studied rules of neuronal compatibility. Rather than updating the network's weights directly, we improve task fitness by updating the neurons' wiring rules, thereby mirroring evolutionary selection on brain development. We find that the resulting framework can not only achieve high performance on standard machine learning tasks, but does so with a fraction of the full network's parameters. When we condition neuronal identity on biologically-plausible spatial constraints, we discover weight structures that resemble visual filters while further compressing parameters count. In summary, by introducing realistic developmental considerations into machine learning frameworks, we not only capture the emergence of innate behaviors, but also define a discovery process for structures that promote complex computations.

* barabasi@fas.harvard.edu

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

INTRODUCTION

The diversity and specificity of animal behaviors, as well as their neural correlates, has received attention from diverse areas of study [1]. Recently, machine learning has provided key insights into the mechanics of solving complex behaviors [2, 3]. However, AI frameworks do not capture the emergence of innate behaviors, as conventional models require extensive update rules and training examples to achieve desired fitness on a task [4, 5]. Nevertheless, a number of complex tasks seem to be hard-coded into the development of the nervous system, such as mice responding to looming stimuli [6], hatched turtles heading out to sea [7], or humans recognizing face-like objects in the womb [8]. In cases where evolutionary pressures for survival outweighs learning, wiring embeddings evolve in order to encode crucial behaviors into the nascent connectome [4]. The brain's innate solutions have long inspired AI techniques [9], from convolutional neural networks to reinforcement learning, yet evolutionary innovation has not been successfully recapitulated for the systematic discovery of powerful architectures.

In order to reproduce the selection process behind innate behaviors, we must first confront the mystery of the "genomic bottleneck": development's uncanny ability to unpack a genome in order to produce specific, task-relevant neural circuits [4, 10]. Although past work has introduced biological considerations for the selection of task knowledge [11, 12], recent studies have mathematically formalized the neuronal partner-selection processes that enable the hard-coding of connectivity [13, 14]. The resulting Genetic Connectome Model considers how interactions between expressed proteins seed synapse formation, and was validated by predicting the wiring rules that code for the *C. elegans* gap junction connectome [14]. Intriguingly, such genetic wiring rules promote structured connectivity, including feed-forward and scale-free networks [15].

In this manuscript we explore realistic developmental mechanisms' potential for guiding AI exploration. We begin by considering neural network weights to be emergent from the developmental processes formalized in the Genetic Connectome Model [13, 14], thus seeking to learn biologically-validated neuronal wiring rules. By mapping the genetic material passed on in evolution to individual fitness on a task, we provide a mechanistic model for the evolution of innate behaviors. Finally, we examine how incorporating realistic models of cell identity could allow for visual filters to be hard-coded in development.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

GENETIC NEUROEVOLUTION FORMALISM

The processing capabilities of neural systems arise from a mixture of neuronal partner selection, learning, and random noise. Of these, neuronal partner selection, consisting of axonal targeting and synaptic compatibility, provide the basis for hardcoded circuits and innate behaviors. The cellular identity of neurons, as represented by their genetic profile, plays a crucial role in their preferred projections and synaptic partners. This mapping from genes to connectivity is captured by the Genetic Connectome Model [13, 14], which defined the wiring of the brain ($\mathbf{W}$) as a function of neuronal identity ($\mathbf{X}$), and interactions between genetic factors ($\mathbf{O}$). Specifically, the connectivity of N neurons is described by the adjacency matrix $\mathbf{W}$ of size NxN, with $W_{ij} = 1$ if a connection is present between neurons, and 0 otherwise. Individual neurons are identified by their expression of G genes, defining a cell-identity matrix $\mathbf{X}$ of size NxG, where $X_{ia} = 1$ if neuron i expresses gene a (Figure 1a, blue to orange links). Interactions between expressed genes determine the wiring of the connectome, represented as the GxG matrix $\mathbf{O}$ (Figure 1a, orange to orange links). Thus, the developmentally produced connectome can be formulated as

$$W = \mathcal{H}(XOX^T),$$

where $\mathcal{H}$ represents the heaviside function that produces a binary connectivity matrix. Previous work used binary values for $\mathbf{W}$, $\mathbf{X}$ and $\mathbf{O}$ to provide interpretable results for connectivity, genetic interactions and expression patterns [13, 14]. In reality, genes ($\mathbf{X}$) have continuous expression, interactions ($\mathbf{O}$) may be probabilistic, and connections ($\mathbf{W}$) vary in size and number. This prompts a relaxation of $\mathbf{W}$, $\mathbf{X}$, and $\mathbf{O}$ to $\mathbb{R}$, allowing for the three matrices to be continuous and differentiable.

Based on this extended framework, we propose a Genetic neuroEvolution Model (GEM) that utilizes the generative process of $W = XOX^T$ to move flexibly between a wiring diagram and its encoding in neural identities (Figure 1b). We start by taking an architecture known to be effective on a task, then define the weights of the network using $W = X_iOX_o^T$ (Figure 1b, downward blue arrows). Here, $\mathbf{W}$ corresponds to the weights of a layer, X_i stands for the genetic expression of the input neurons, and X_o represents the output neurons' gene expression. We begin with untrained $\mathbf{X}$s and $\mathbf{O}$ chosen from a fitting random distribution.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

With each training batch, we backpropagate the loss *through* the weights (Figure 1b, red leftward arrow), updating the **X**s and **O** using gradients calculated by PyTorch's autograd (Figure 1b, red upward arrows). In theory, this results in a new developmental ruleset in the **X**s and **O**, which produces a more fit **W** in the next task initialization. Thus we adapt a standard AI approach to model the evolution of cell identities and genetic interactions, generating 'individuals' that have prewired behavioral programs at 'birth.'

In order to validate the hypotheses of GEM we turn to the MNIST hand-drawn digit classification task (Figure 1c), which has well-studied accuracies for a number of architectures [16]. We aim to use GEM to learn "developmentally initialized" network weights that have high task performance, without further direct weight updates ("learning") after initialization. To minimize the number of free variables (i.e. size and number of hidden layers), we start by encoding a single-layer linear classifier (Figure 1d), consisting of $28 * 28 + 10 = 794$ nodes (neurons) with 7,840 learnable weights (synapses). A single layer neural network initialized with kaiming uniform performs at an accuracy of 10% without any training ("naive"), however, with direct training of weights can achieve 93% accuracy ("learned", dashed line in Figure 1c). We aim to encode the weights of this network using GEM, with the number of genes G as the only free variable in the system. This results in a parameter scaling of $P = N_x * G + G * G + G * N_y = G(N_x + N_y + G)$, where N_x and N_y are the number of input and output neurons, respectively. We find that GEM can achieve nearly 25% accuracy "at birth" when neurons "express" only 1 gene (corresponding to under 10% of the parameters of the full single-layer classifier). The accuracy increases to above 80% for $G = 5$, corresponding to 50% of the parameters of a linear classifier. The GEM model achieves full 93% accuracy with 9 genes, or 90% of the baseline parameters. Thus, we can not only match the task performance of the classically trained network with fewer parameters, but we also achieve high accuracy with half of the original parameters.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

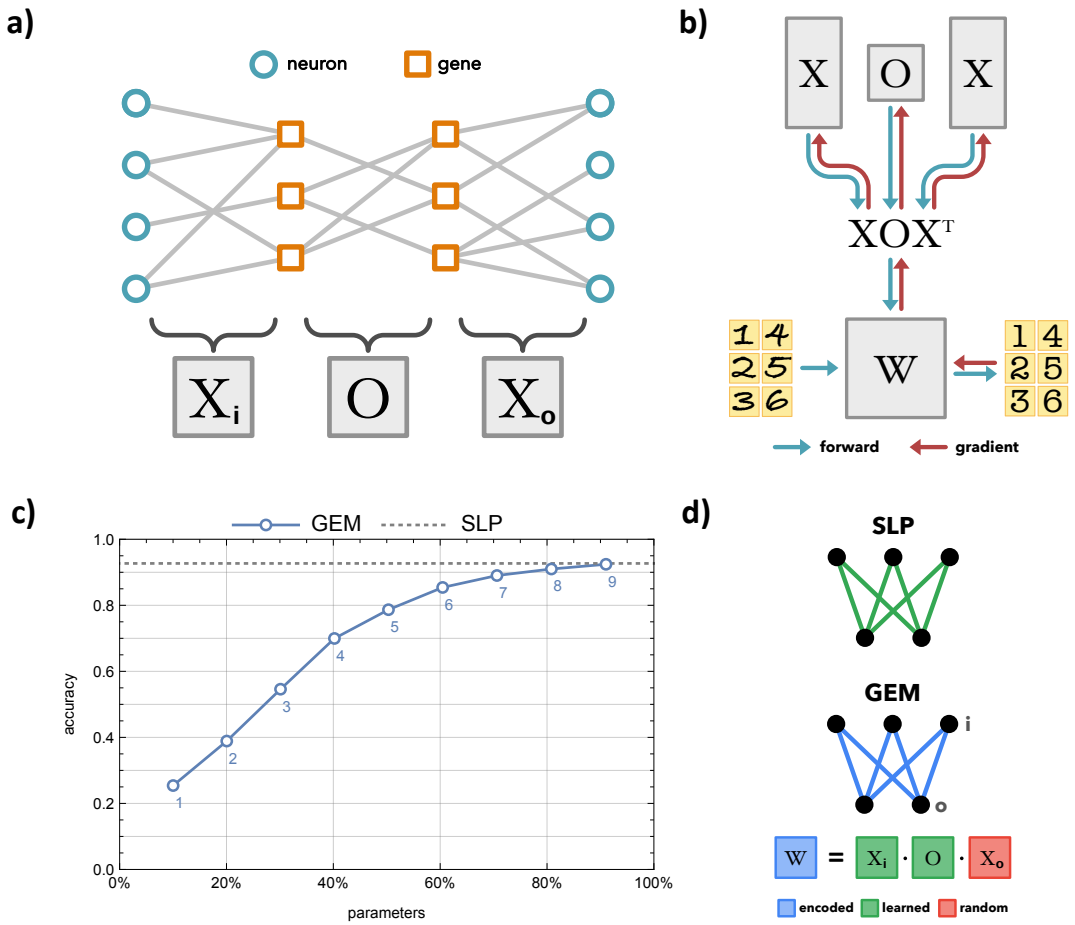

FIG. 1: **The Genetic neuroEvolution Model.(a)** Visualization of the Genetic Connectome Model [13, 14]. Matrices X_i and X_o represent the gene expression of input and output neurons, respectively. The **O** matrix corresponds the genetic interactions that underlie neuronal partner selection. **(b)** Traditionally, AI techniques define an architecture (**W**), which can receive data as inputs (handwritten digits, left of **W**), and produce predictions (digital numbers, right of **W**). The weights (**W**) of the architecture can be updated (leftward red arrow) based on the distance of the predictions from known values, thereby producing a more accurate system with training. In the Genetic neuroEvolution Model (GEM), the architecture (**W**) is produced from a small set of wiring rules defined by **X**s and **O** (downward blue arrows). At each training step, the data is passed through to make predictions (rightward blue arrows). However, rather than altering the architecture weights, gradients are computed to update **X**s and **O** (upward red arrows). At the next training step the revised wiring rules generate a revised **W** (downward blue arrows). **(c)** Performance of GEM on the MNIST task. The accuracy of a single-layer linear classifier (784 by 10 nodes) is shown, either with learned weights (dashed line) or weights encoded by GEM's wiring rules (blue line, with number of genes labeled below each marker). Parameters are expressed as a percentage of a learned SLP's weights. **(d)** Visualization of a learned SLP and an SLP encoded by GEM in Figure 1c. We do not learn X_o, as we find it adds parameters without increasing task performance.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

EMERGENCE OF CELL IDENTITY

Current understandings of neurodevelopment indicate that reproducible connectivity arises from interactions between cell identity and neuronal compatibility rules [17]. Thus far, we have used a biologically-validated model of neuronal compatiblity to derive an AI model for the emergence of innate behaviors. Next, we examine how more realistic considerations for cell identity can impact neuroevolutionary search. Although constraining the system could destabilize learning, we aim to show that complex computations can be primed by developmental processes.

To first approximation, a neuron's cell type (1) defines its connectivity and function, and (2) emerges from well-known factors, such as cell lineage and spatial gradients [18–20]. To make a machine learning analogy, the developmental patterning of cell types corresponds to a reproducible processing hierarchy, which we will assume to be a layered structure (Figure 2a). We can think of these as 'deterministic': if the organism develops normally, these cell identities emerge consistently and define the broad strokes of neurons' projections and connections. However, neurons have also been shown to exhibit a random component to their wiring. For instance, stochastic alternative splicing of cell adhesion molecules allows for a significant amount of genetic variability to emerge [21, 22]. Such processes further partition a single coarse-grained cell type, leading to divergent sub-identities, and thus divergent input and output profiles within a neural population.

Yet, the emergence of cellular variability is not completely random: the alternative splicing of implicated genes, and thus the resulting cell identity, has crucial biases in space and time [23, 24]. We can represent the expression of genes, or their alternative splices, as a mixture of gaussians, $G(\vec{x}|\mu_i, \sigma_i) \propto exp\{-\frac{1}{2}(\vec{x} - \vec{\mu}_i)^T \Sigma_i^{-1} (\vec{x} - \vec{\mu}_i)\}$, where G describes the expression level of a gene over a relevant parameterization of space and time ($\vec{x}$). Thus, the spectrum of 'fine-grained' cell identities at a given location in the brain is dependent on the distribution of locally expressed genes (Figure 2b). What emerges is a definition of cell types that can be linked not only to a well-studied statistical learning paradigm, but is rooted in specific, highly profiled, genetic mechanisms.

At this point, we have arrived at developmentally motivated models of both cell identity and connectivity rules. As we meld these two components, we find that even simple heuristics applied to a small number of genes with spatial biases can produce complex circuits.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Returning to the four genes in Figure 2, we can see that if a neuron only accepts connections from cells expressing blue genes, it will wire to cells in the top left corner of the space (Figure 2c, top left). A different neuron may select for the top center region by accepting connections from neurons expressing *both* blue and red genes (Figure 2c, top middle). As we extend these varying levels of selectivity, we find that we can partition the space in Figure 2c to form a convolutional tiling. Finally, if we incorporate 'or' operators, we can produce more specific filters on a given region (Figure 2d). Thus, spatially biased genes and simple wiring rules can begin to reproduce the inductive biases of the convolutional neural networks that underpin modern image recognition solutions.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

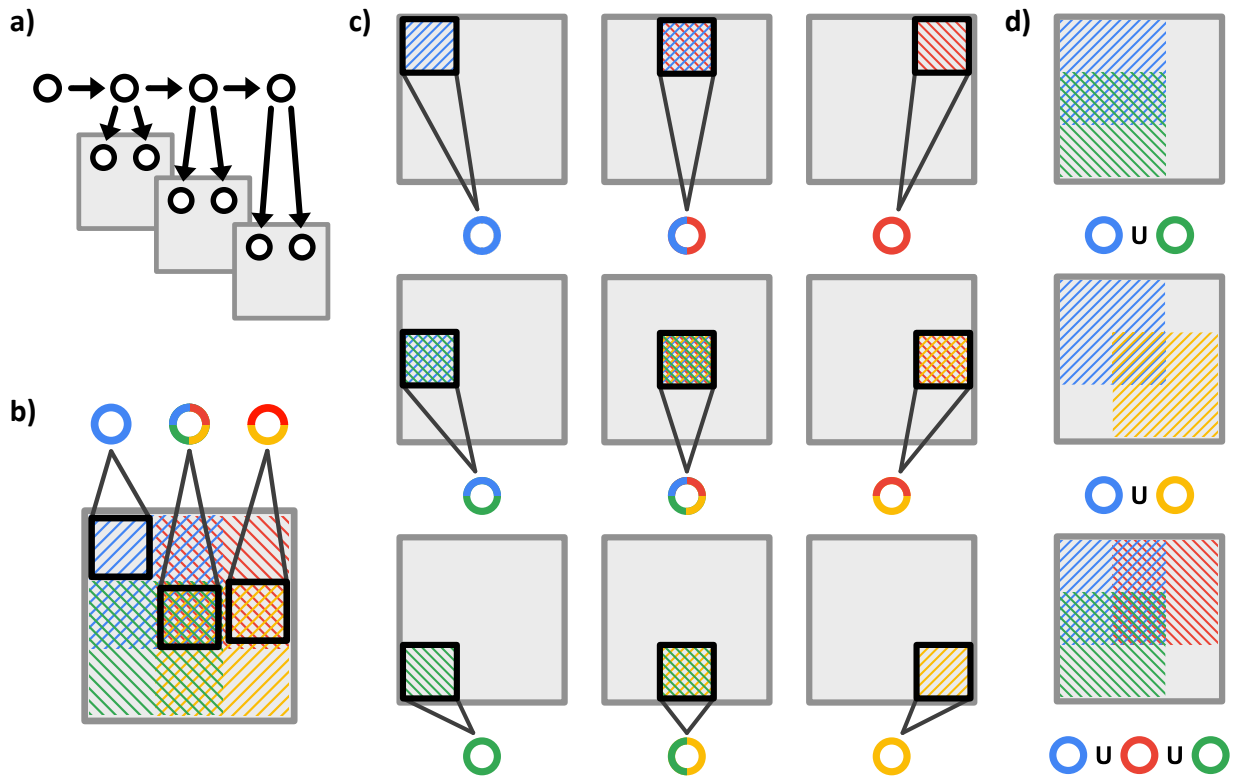

FIG. 2: **Developmental Priming of Computation.** **(a)** Coarse-grained cell types, such as those arising from lineage, can define a neuron's position in a processing hierarchy. **(b)** Spatial gene expression can produce diversity within the coarse-grained cell types. Here we have four genes (blue, red, green and yellow), each of which has a spatial bias to its expression (shading). When sampling from the top left, we find neurons that express the blue gene (left circle). However, if we sample at locations where the gene expression distributions overlap, we can find neurons with multiple genes expressed, such as red and yellow (right circle), or even all four genes (middle circle). **(c)** Four genes with spatial biases are sufficient for a spatial tiling to be developmentally primed. Neurons can have specific wiring rules, such as connecting to neurons that only expresses blue genes (top left). Other neurons may be more specific, incorporating 'and' operators, for instance only receiving connections from neurons expressing both blue and red genes (top, center). **(d)** Incorporating 'or' operators allows for the coding of spatial filters. Neurons that allow connections from blue or green genes can select for the left side of the region (top). A diagonal filter can be achieved by selecting for blue or yellow genes (middle). Finally, an upper triangle can be selected by being sensitive to blue, red, or green genes, but not yellow (bottom).

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

SPATIAL PRIORS ON NEUROEVOLUTION

In order to test the hypotheses of Figure 2, we ask whether competitive task performance can be evolved from a GEM approach where expression (e.g. $\mathbf{X}$ matrix) is fully determined by genetic gradients. We define a spatial GEM by placing N neurons on a grid (28x28 for an MNIST input layer), and determine a neuron's gene expression profile ($\mathbf{X}$ matrix) by its distance from G 2-D gaussians with given σ and μ (Figure 3a). As training progresses the locations of neurons remain fixed, however we update the gene distributions' σ and μ, thereby learning a biologically constrained $\mathbf{X}$ (Figure 3b). When we use the spatial GEM (S-GEM, Figure 3c right) to encode the SLP from Figure 1c, we achieve over 80% accuracy with less than 2% of the total parameters (Figure 3c, left). S-GEM converges towards 93% accuracy with 10% of total parameters, a point at which the non-spatial GEM only performs at 25% accuracy. To better contextualize the performance of S-GEM, we turn to a random basis encoding (RB, Figure 3c left), which has been proposed as measure for the number of free parameters needed to solve a task to a given accuracy [25]. We find that the S-GEM not only matches the parameter-accuracy tradeoff of the random basis model, but outperforms it at low parameter counts.

In order to confirm the robustness of the S-GEM encoding, we extend our validation to multilayer perceptrons (Figure 3d, right). Doing so requires refining our biological assumptions. Given that gene interaction are determined by universal physical rules, and therefore are not specific to a set of neurons, we choose to learn a single $\mathbf{O}$ matrix that is shared by all encoded layers. Further, we assume that neurons have a single cell identity that determines their input and output connectivity profiles, thus we learn a single $\mathbf{X}$ matrix for each hidden unit, using it to define both its pre- and post-synaptic weights. Although neurons can express different genes in axons and dendrites, we found that learning separate input and output $\mathbf{X}$ matrices for each hidden unit only increases the number of parameters, without increasing accuracy. The total parameters of this spatial GEM model scales as $P = G * G + L * 3G = G(G + 3L)$, as we have a single $\mathbf{O}$ matrix with $G * G$ parameters, and *each* encoded layer (L) has a learned μ_x, μ_y and σ for *each* gene. It is interesting to note that the parameter count no longer scales with the number of neurons in the architecture, as cell identities are emergent from the local gene expression gradients. In fact, although we currently restrict neurons to a grid, future work could explore an S-GEM encoding where the

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

locations of neurons are learned, or sampled from a learned distribution, thereby naturally accommodating the addition or subtractions of neurons from a layer.

Given this definition of a multi-layer spatial GEM (ML-S-GEM, Figure 3d right) we aim to encode an MLP of size 784x784x10, where the input and hidden neurons are placed on 28x28 grids. We find that the parameter-accuracy tradeoff continues to match a random basis encoding, achieving above 90% accuracy with less than 0.1% of the original parameters. As a point of comparison, a less biological "genomic bottleneck" model achieved 79% accuracy with 1038-fold parameter compression on this architecture [10], while the ML-S-GEM achieves above 80% accuracy with roughly 0.025% of the original MLP parameters, amounting to a 4000-fold compression. Even for this larger network, ML-S-GEM trains in seconds to minutes, corresponding to only a slight increase in computational burden compared to a standard MLP. In summary, we find that introducing biologically motivated constraints on cell identity not only retains the performance of our model of innate behavior, but provides additional compression of the hardcoded circuit.

Having confirmed the representation and compression capabilities of S-GEM, we turn to the final prediction of Figure 2: introducing biological biases to gene expression should produce recognizable structures in the networks' weights. Indeed, when we examine the weights of S-GEM (Figure 4a), we find a few numerals directly drawn out by the weight preferences (0, 1, 3 and 8). In order to explore the robustness of this phenomenon, we applied the spatial GEM to a denoising task: we took the standard MNIST digits (Figure 4b), set 10% of values to 0, and asked an autoencoder with 100 hidden units to reconstruct the original image (Figure 4f, and SI I.A) [26]. We find that compressing the autoencoder with the spatial GEM not only solves the task at the same accuracy as a linear network or standard GEM approach, but also discovers weights akin to a feature map (Figure 4c,e), especially when compared to the sparse features of the standard approaches (Figures S1 and S2). Thus, we find that S-GEM can indeed utilize a simple developmental process to evolve primordial feature detectors, a hallmark of complex computation.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

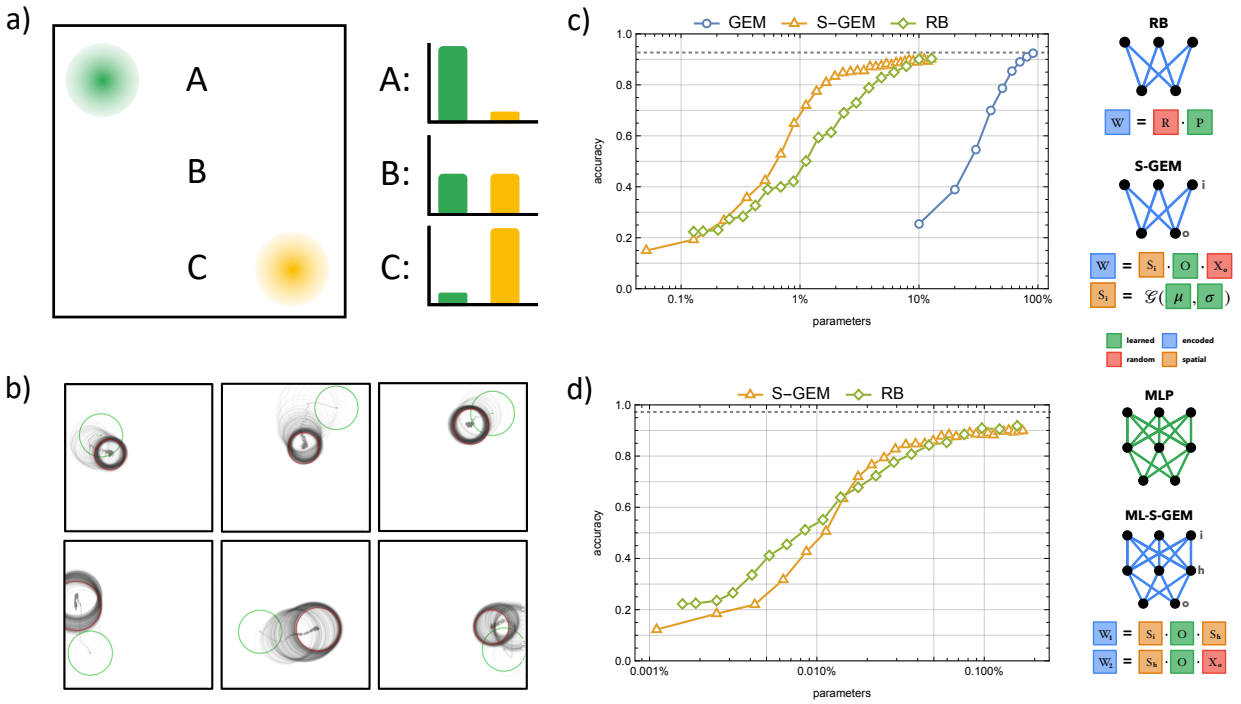

FIG. 3: **Spatial GEM. (a)** In the spatial GEM, cell identities are determined by neurons' distance from 2-D gene expression gaussians. Given two spatial gene distributions, green and yellow, we have: neuron A with high expression of green and low expression of yellow, neuron B with medium expression of both genes, and neuron C with low expression of green but high expression of yellow. **(b)** As learning progresses, the location (μ) and dispersion (σ) of the gene expression distributions change from start (green) to finish (red). **(c)** Left: Parameter-accuracy tradeoff for encoding a single layer linear classifier that solves MNIST, where parameters are expressed as a % of an uncompressed SLP. The spatial GEM (S-GEM) model significantly outperforms the GEM framework from Figure 1c, as well as a random basis (RB) encoding. Right: Visualization of the used RB and S-GEM weight encodings. **(d)** Left: Parameter-accuracy tradeoff for encoding an MLP that solves MNIST, where parameters are expressed as a % of an uncompressed MLP. The multi-layer S-GEM perform comparatively to a random basis encoding. Right: Visualization of the MLP and ML-S-GEM encoding used.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

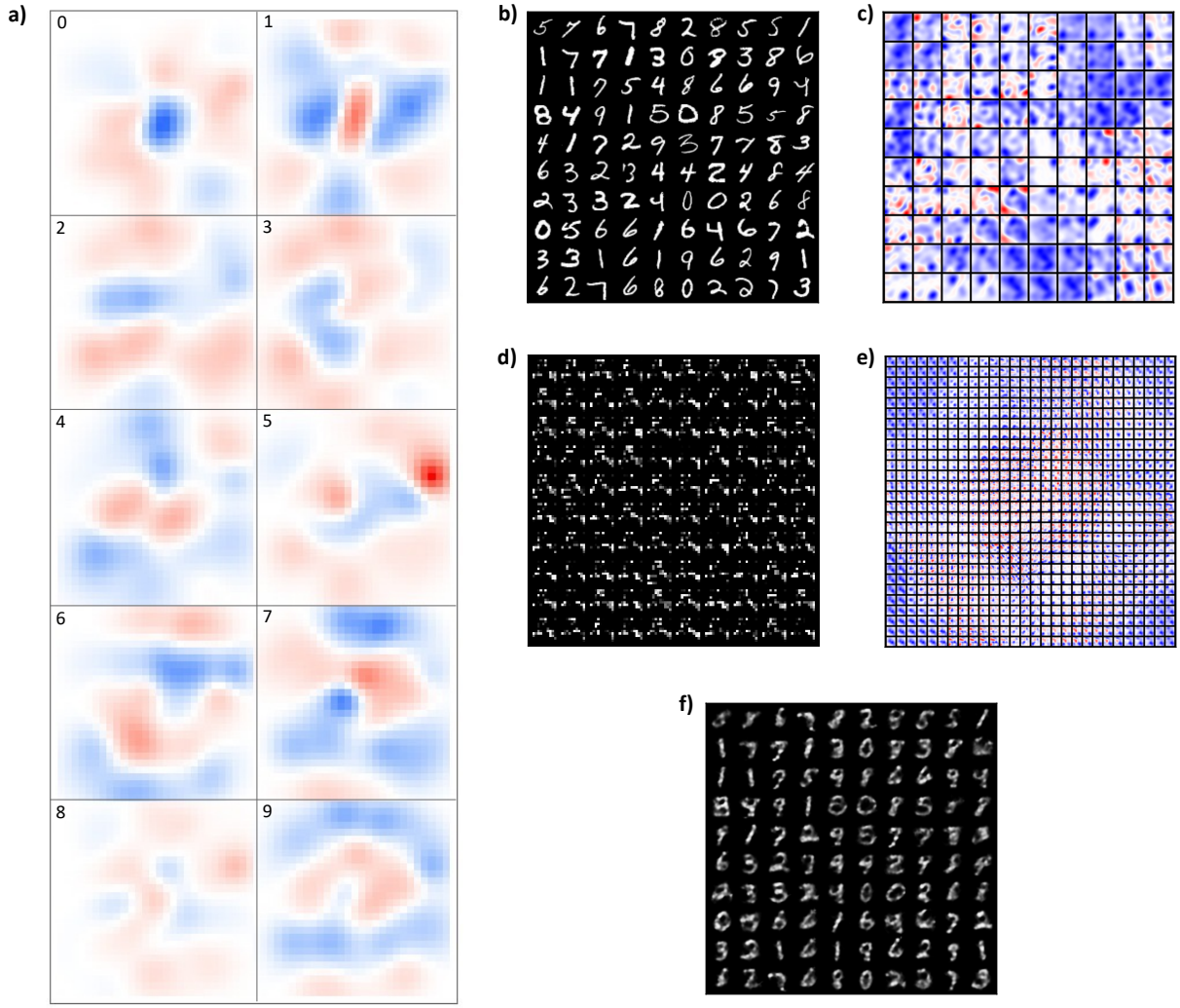

FIG. 4: **Filters in Spatial GEM.** **(a)** Spatial GEM learns weight matrices ranging from primordial filters (numerals 2, 4, 5, 6, 7, and 9) to faithful recreations of represented numbers (0, 1, 3, and 8). Each tile shows the weight matrix of an output neuron, with red representing positive weights, and blue corresponding to negative weights. **(b)** 10x10 grid of standard MNIST digits. **(c)** Weight matrices of the 10x10 hidden layer for the MNIST denoising task, with each grid corresponding to a separate neuron. **(d)** Representation of MNIST digits from **(b)** in the hidden layer. **(e)** Weight matrices of the 28x28 output neurons. **(f)** Sample output of the denoising autoencoder, given digits from **(b)**.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

DISCUSSION

In this paper we explored a biologically-validated embedding for the evolution and encoding of innate behaviors. We began from a model of developmental wiring rules, and proposed the Genetic neuroEvolution Model (GEM) in order to "evolve" a neural network that has high task performance "at birth." Next, we examined how a biologically realistic model of cell identity can promote structured connectivity. When applied to MNIST tasks, we found that incorporating spatially defined genes into the GEM led to a more compact encoding, all while allowing for topographic maps and primordial kernels to be developmentally primed. By understanding, then building on neurodevelopmental embeddings, we expect GEM to provide unique insights to both neuroscience and machine learning.

In defining the spatial GEM we aimed to utilize genetic phenomena profiled in multiple systems and organisms. For instance, stochastic splicing of Pcdh has been implicated in mouse OSN [27], purkinje [27], and cortical microcircuit wiring [28], as well as the somatosensory mapping of the barrel cortex [29]. In *Drosophila* Dscam alternative splicing is crucial for mechanosensory neurons [22]. Perturbation experiments highlight the level of specificity achievable by this genetic mechanism. One method of reducing the available splicing diversity of Dscams in *Drosophila* affected only mechanosensory neurons, which achieved the 'broad strokes' of connectivity (initial axon guidance), but lost the specificity of the "last stretch": where to branch, how many synapses to make, and which neurons to connect to [22]. In mice the Pcdh stochastic expression repertoire was found to be crucial for short-term memory and sensory integration [30].

Given our focus on simple, but well documented, developmental phenomena, further work could benefit from examining genetic and activity-dependent finetunings of neural circuits. For instance, although the topographic mapping from retina to SC is initially established through the graded expression of receptors and ligands [31, 32], the process is aided by interaxon competition [33] and is refined by spontaneous activity before eye opening [31, 32]. Spontaneous activity is also crucial for the alignment of the topographic maps formed from retina to SC and from SC to V1 [34]. Yet, spontaneous activity may play only a guiding role in circuit formation; experiments in ferrets have shown orientation selectivity prior to eye opening, a result which persist even under dark-rearing [35] or silencing of spontaneous activity [36]. Further, ex-vivo preparations, which lack spontaneous activity,

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

still develop proper subcellular projection patterns [37]. These studies suggest that the genetic mechanisms incorporated into the spatial GEM model already play a significant role in the required specificity of hardwired computational circuits.

Although our work on GEM focused on evolutionary hardcoded behaviors, also if interest is the developmentally priming of learning, where the nascent connectome promotes the acquisition of a complex task. For instance, humans are not born with the ability to speak, nor are infants' brains predisposed to learn only their parents' native tongue. Rather, during a critical period children can rapidly acquire languages they are exposed to. This suggests that development has poised the wiring of language areas to rapidly analyze and distill sound patterns, with local wiring recontouring to encode language in the process. Such observations relate to the meta-learning topic of few-shot classification [38, 39], which future work on the GEM could leverage to study the trade-off between learned and hardwired circuits.

Acknowledgements

We wish to thank Gabriel Kreiman, Adam Marblestone, Dániel Czégel, Albert-László Barabási, and the IBRO-Simons Computational Neuroscience Imbizo for fruitful discussions. D.L.B. was supported by NIH NIGMS T32 GM008313 and a Mind Brain and Behavior Graduate Award.

Author Contributions

Both authors have contributed to the design of the project and the writing of the manuscript. Software developed by T.B.

Data and Code Availability

We provide a PyTorch package for the Genetic neuroEvolution Model (GEM) in the public repository with DOI: (Forthcoming). The analyses of the study utilized a number of publicly available software packages and datasets which could not be reproduced in the repository, but can be found through in-text citations.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Conflict of Interest

The authors declare no conflict of interest.

---

[1] S. Musall, A. E. Urai, D. Sussillo, and A. K. Churchland, Harnessing behavioral diversity to understand neural computations for cognition, Current opinion in neurobiology **58**, 229 (2019).

[2] B. A. Richards, T. P. Lillicrap, P. Beaudoin, Y. Bengio, R. Bogacz, A. Christensen, C. Clopath, R. P. Costa, A. de Berker, S. Ganguli, *et al.*, A deep learning framework for neuroscience, Nature neuroscience **22**, 1761 (2019).

[3] S. Srinivasan, R. J. Greenspan, C. F. Stevens, and D. Grover, Deep (er) learning, Journal of Neuroscience **38**, 7365 (2018).

[4] A. M. Zador, A critique of pure learning and what artificial neural networks can learn from animal brains, Nature communications **10**, 1 (2019).

[5] T. P. Lillicrap and K. P. Kording, What does it mean to understand a neural network?, arXiv preprint arXiv:1907.06374 (2019).

[6] M. Yilmaz and M. Meister, Rapid innate defensive responses of mice to looming visual stimuli, Current Biology 10.1016/j.cub.2013.08.015 (2013).

[7] A. F. Carr and L. H. Ogren, The ecology and migrations of sea turtles. 4, the green turtle in the caribbean sea., Bulletin of the AMNH (1960).

[8] V. M. Reid, K. Dunn, R. J. Young, J. Amu, T. Donovan, and N. Reissland, The Human Fetus Preferentially Engages with Face-like Visual Stimuli, Current Biology 10.1016/j.cub.2017.05.044 (2017).

[9] D. Hassabis, D. Kumaran, C. Summerfield, and M. Botvinick, Neuroscience-inspired artificial intelligence, Neuron **95**, 245 (2017).

[10] A. Koulakov, S. Shuvaev, and A. Zador, Encoding innate ability through a genomic bottleneck, bioRxiv (2021).

[11] D. Ha, A. Dai, and Q. V. Le, Hypernetworks, arXiv preprint arXiv:1609.09106 (2016).

[12] K. O. Stanley, D. B. D'Ambrosio, and J. Gauci, A hypercube-based encoding for evolving large-scale neural networks, Artificial life **15**, 185 (2009).

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

[13] D. L. Barabási and A.-L. Barabási, A genetic model of the connectome, Neuron **105**, 435 (2020).

[14] I. A. Kovács, D. L. Barabási, and A.-L. Barabási, Uncovering the genetic blueprint of the c. elegans nervous system, Proceedings of the National Academy of Sciences **117**, 33570 (2020).

[15] D. L. Barabasi and D. Czegel, Constructing graphs from genetic encodings, bioRxiv (2020).

[16] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, *et al.*, Gradient-based learning applied to document recognition, Proceedings of the IEEE **86**, 2278 (1998).

[17] T. C. Südhof, Towards an understanding of synapse formation, Neuron **100**, 276 (2018).

[18] S. N. Sansom and F. J. Livesey, Gradients in the brain: the control of the development of form and function in the cerebral cortex, Cold Spring Harbor Perspectives in Biology **1**, a002519 (2009).

[19] G. Fishell and A. Kepecs, Interneuron types as attractors and controllers, Annual review of neuroscience **43** (2019).

[20] G. Ciceri, N. Dehorter, I. Sols, Z. J. Huang, M. Maravall, and O. Marín, Lineage-specific laminar organization of cortical gabaergic interneurons, Nature neuroscience **16**, 1199 (2013).

[21] B. Treutlein, O. Gokce, S. R. Quake, and T. C. Südhof, Cartography of neurexin alternative splicing mapped by single-molecule long-read mrna sequencing, Proceedings of the National Academy of Sciences **111**, E1291 (2014).

[22] B. E. Chen, M. Kondo, A. Garnier, F. L. Watson, R. Püettmann-Holgado, D. R. Lamar, and D. Schmucker, The molecular diversity of dscam is functionally required for neuronal wiring specificity in drosophila, Cell **125**, 607 (2006).

[23] S. L. Zipursky and J. R. Sanes, Chemoaffinity revisited: dscams, protocadherins, and neural circuit assembly, Cell **143**, 343 (2010).

[24] G. Neves, J. Zucker, M. Daly, and A. Chess, Stochastic yet biased expression of multiple dscam splice variants by individual cells, Nature genetics **36**, 240 (2004).

[25] C. Li, H. Farkhoor, R. Liu, and J. Yosinski, Measuring the intrinsic dimension of objective landscapes, arXiv preprint arXiv:1804.08838 (2018).

[26] C. Fernando, D. Banarse, M. Reynolds, F. Besse, D. Pfau, M. Jaderberg, M. Lanctot, and D. Wierstra, Convolution by evolution: Differentiable pattern producing networks, in *Proceedings of the Genetic and Evolutionary Computation Conference 2016* (2016) pp. 109–116.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

[27] D. Canzio and T. Maniatis, The generation of a protocadherin cell-surface recognition code for neural circuit assembly, Current Opinion in Neurobiology **59**, 213 (2019).

[28] X. Lv, S.-Q. Ren, X.-J. Zhang, Z. Shen, T. Ghosh, A. Xianyu, P. Gao, Z. Li, S. Lin, Y. Yu, *et al.*, Tbr2 coordinates neurogenesis expansion and precise microcircuit organization via protocadherin 19 in the mammalian cortex, Nature communications **10**, 1 (2019).

[29] T. Hirayama, E. Tarusawa, Y. Yoshimura, N. Galjart, and T. Yagi, Ctcf is required for neural development and stochastic expression of clustered pcdh genes in neurons, Cell reports **2**, 345 (2012).

[30] T. Yamagishi, K. Yoshitake, D. Kamatani, K. Watanabe, H. Tsukano, R. Hishida, K. Takahashi, S. Takahashi, A. Horii, T. Yagi, *et al.*, Molecular diversity of clustered protocadherin-α required for sensory integration and short-term memory in mice, Scientific reports **8**, 1 (2018).

[31] D. Tsigankov and A. A. Koulakov, Sperry versus hebb: Topographic mapping in isl2/epha3 mutant mice, BMC neuroscience **11**, 1 (2010).

[32] A. D. Huberman, M. B. Feller, and B. Chapman, Mechanisms underlying development of visual maps and receptive fields, Annu. Rev. Neurosci. **31**, 479 (2008).

[33] J. W. Triplett, C. Pfeiffenberger, J. Yamada, B. K. Stafford, N. T. Sweeney, A. M. Litke, A. Sher, A. A. Koulakov, and D. A. Feldheim, Competition is a driving force in topographic mapping, Proceedings of the National Academy of Sciences **108**, 19060 (2011).

[34] J. W. Triplett, M. T. Owens, J. Yamada, G. Lemke, J. Cang, M. P. Stryker, and D. A. Feldheim, Retinal input instructs alignment of visual topographic maps, Cell **139**, 175 (2009).

[35] L. E. White, D. M. Coppola, and D. Fitzpatrick, The contribution of sensory experience to the maturation of orientation selectivity in ferret visual cortex, Nature **411**, 1049 (2001).

[36] B. Chapman and M. P. Stryker, Development of orientation selectivity in ferret visual cortex and effects of deprivation, Journal of Neuroscience **13**, 5251 (1993).

[37] G. Di Cristo, C. Wu, B. Chattopadhyaya, F. Ango, G. Knott, E. Welker, K. Svoboda, and Z. J. Huang, Subcellular domain-restricted gabaergic innervation in primary visual cortex in the absence of sensory and thalamic inputs, Nature neuroscience **7**, 1184 (2004).

[38] C. Finn, P. Abbeel, and S. Levine, Model-agnostic meta-learning for fast adaptation of deep networks, in *Proceedings of the 34th International Conference on Machine Learning-Volume 70* (JMLR. org, 2017) pp. 1126–1135.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

[39] A. Nichol, J. Achiam, and J. Schulman, On first-order meta-learning algorithms, arXiv preprint arXiv:1803.02999 (2018).

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

I. SI

A. MNIST Autoencoder Task

To test the possibility of hard-coding filters by modeling gene expressions as gaussians, we utilize an MNIST denoising autoencoder task [26]. To begin, we set 10% of pixels in each MNIST digits to 0, and ask a network to reconstruct the original digit. We use a standard architecture [26] of 1 hidden layer with 100 neurons, and input and output layers of size $28 * 28 = 784$ neurons (the size of an MNIST image). A sigmoid layer is included after the hidden layer and the output layer. For the spatial GEM approach, we place all the neurons on a grid (10x10 for hidden, 28x28 for input and output), and then pick G gaussians, with fixed σ but random μ, on the grid. This leads to the NxG-sized $\mathbf{X}$ matrix being fixed as the value of the gaussian g_i at that neuron's location. We then train the $\mathbf{O}$ matrix, as well as any bias vectors. We compare this to a linear network and a standard GEM approach, where the $\mathbf{X}$ matrix can be learned. Training occurs on 5000 iterations of batches of size 64. We use a BCE loss, which compares the output image to the initial, unperturbed, MNIST digit.

We find that all three approaches can obtain a BCE of 0.001. However, the linear network requires 157,684 parameters, while the GEM can achieve competitive fitness with 10 genes, corresponding to 17,674 parameters. Finally, the spatial GEM can solve the task with 50 random gaussians, corresponding to only 5,884 trainable parameters.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

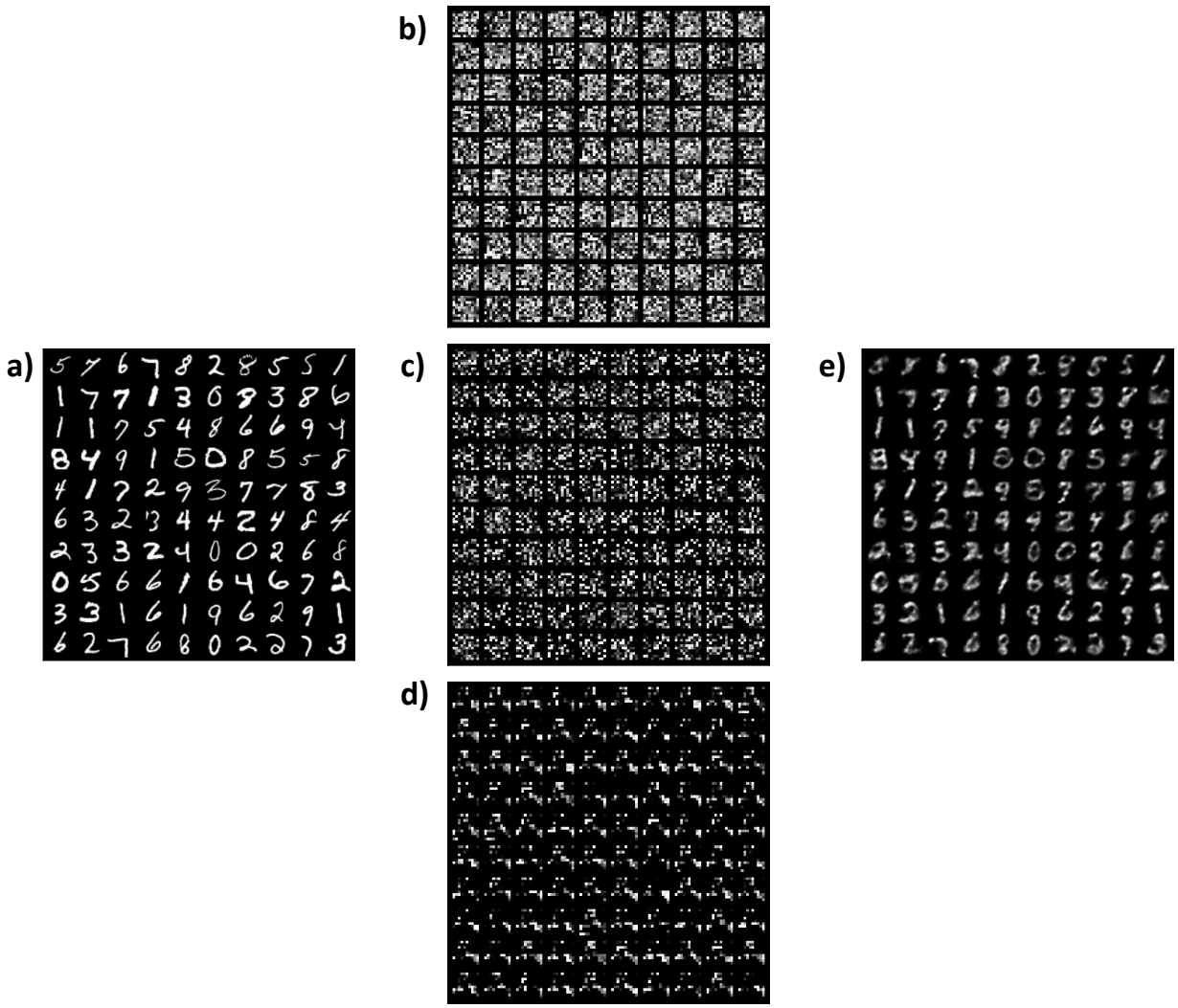

FIG. S1: **Visual Representation of MNIST Denoising Autoencooder Task.** **(a)** Visualization of standard MNIST digits used as input. **(b-d)** Representation of MNIST digits at hidden layer from **(b)** a linear network, **(c)** a standard GEM network, and **(d)** a spatial GEM network. The linear and GEM approaches have noisy, seemingly random, hidden representations, while the spatial GEM approach's representation recapitulates the spatial structure of the weights (Figure S2). **(e)** Sample output of the task.

bioRxiv preprint doi: https://doi.org/10.1101/2021.03.29.437584; this version posted March 30, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

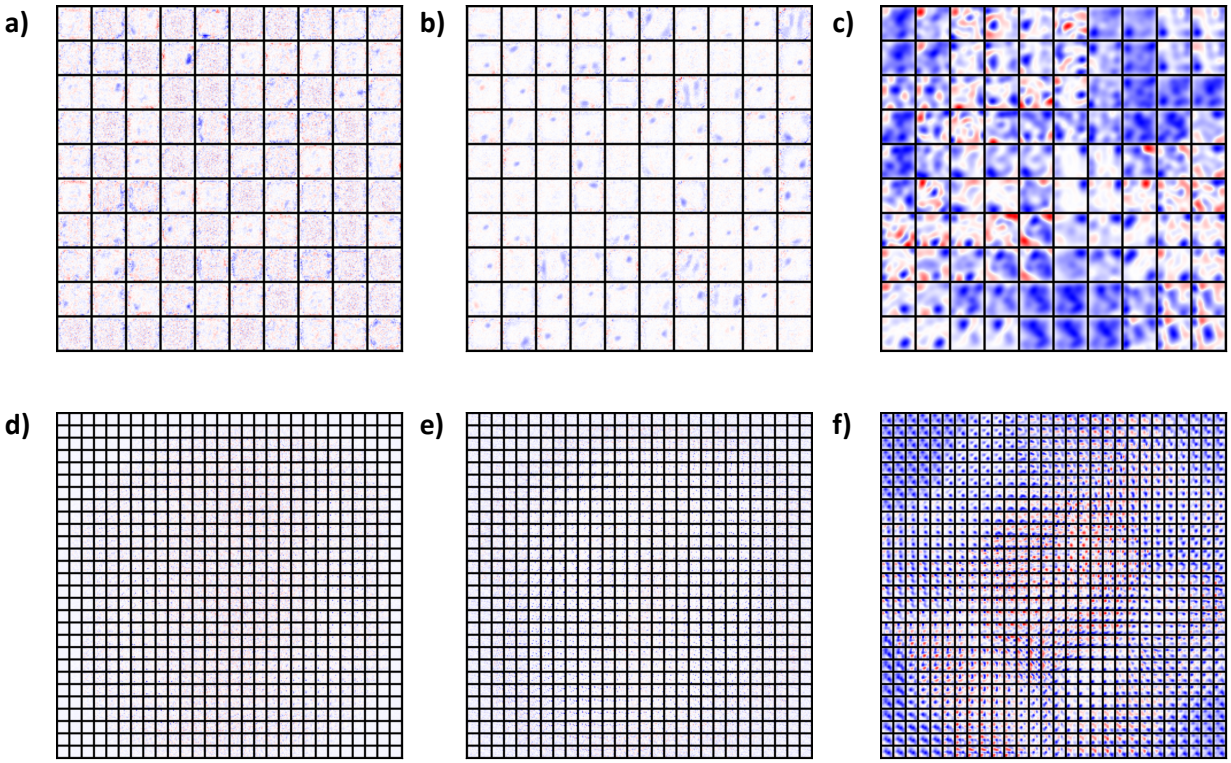

FIG. S2: **Visualization of MNIST denoising autoencoder weights.(a-c)** Hidden layer weights for **(a)** a standard linear network, **(b)** a standard GEM network, and **(c)** a spatial GEM network. Red corresponds to positive weights, white is near-zero, and blue is negative. While the spatial GEM network produces weight patterns akin to feature detectors, both the linear and standard GEM approaches learn weights that display less local similarity, both within and between neurons. **(d-f)** Output layer weights of the **(d)** linear, **(e)** GEM, and **(f)** spatial GEM networks. Again, the spatial GEM network displays apparent topographical structure, both within and between neurons, while the other two approaches learned heterogeneous weights.