
Dense Associative Memory with Epanechnikov Energy

Benjamin Hoover
IBM Research
Georgia Tech

Zhaoyang Shi
Harvard

Krishnakumar Balasubramanian
UC Davis

Dmitry Krotov
IBM Research

Parikshit Ram
IBM Research

Abstract

We propose a novel energy function for Dense Associative Memory (DenseAM) networks, the log-sum-ReLU (LSR), inspired by optimal kernel density estimation. Unlike the common log-sum-exponential (LSE) function, LSR is based on the Epanechnikov kernel and enables exact memory retrieval with exponential capacity without requiring exponential separation functions. Moreover, it introduces abundant additional *emergent* local minima while preserving perfect pattern recovery — a characteristic previously unseen in DenseAM literature. Empirical results show that LSR energy has significantly more local minima (memories) that have comparable log-likelihood to LSE-based models. Analysis of LSR’s emergent memories on image datasets reveals a degree of creativity and novelty, hinting at this method’s potential for both large-scale memory storage and generative tasks.

1 Associative Memories and Energy Landscapes

Energy-based Associative Memory networks or AMs are models parameterized with M “memories” in d dimensions, $\Xi = \{\xi_\mu \in \mathbb{R}^d, \mu \in [M]\}$. A popular class of models from this family can be described by an energy function defined on the state vector $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$:

$$E_\beta(\mathbf{x}; \Xi) = -Q \left[\sum_{\mu=1}^M F(\beta S(g(\mathbf{x}), \xi_\mu)) \right], \quad (1)$$

where $g : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is a vector operation (such as binarization, (layer) normalization), $S : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ is a similarity function (e.g., dot-product, negative Euclidean distance), $\beta > 0$ denotes the inverse temperature, $F : \mathbb{R} \rightarrow \mathbb{R}$ is a rapidly growing separation function (power, exponential) and Q is a monotonic scaling function (logarithm, linear) [1, 2, 3]. With g as the sign-function, $\xi_\mu \in \{-1, +1\}^d$, $S(\mathbf{x}, \mathbf{x}') = \langle \mathbf{x}, \mathbf{x}' \rangle$ and F as the quadratic function, and Q as a linear function, we recover the classical Hopfield model [4]. The output of an associative memory (AM) corresponds to one of the local minima of its energy function. A memory ξ_μ is said to be retrieved if $\mathbf{x} \approx \xi_\mu$ corresponds to such a local minimum, with exact retrieval occurring when $\mathbf{x} = \xi_\mu$. The memory capacity of the AM is defined as the maximum number M^* of correctly retrieved memories. For classical AMs, the capacity scales as $M^* = O(d)$. With the introduction of power-law separation functions — that is, $F(x) = x^p$ for $p > 2$ — modern *Dense Associative Memories* (DenseAMs) achieve a significantly higher capacity of $M^* = O(d^p)$ [1, 5].

The use of an exponential separation function combined with a logarithmic scaling function — $F(x) = \exp(x)$ and $Q(x) = \log x$ — leads to the widely studied log-sum-exp (LSE) energy function [6, 7, 8], yielding exponential memory capacity $M^* \sim \exp(d)$ [9]. Hierarchical organization of memories has also been explored in Krotov [10] and Hoover et al. [11]. Given that the gradient

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

LSR **preserves** memories while creating **novel** ones.

LSE can do only one or the other.

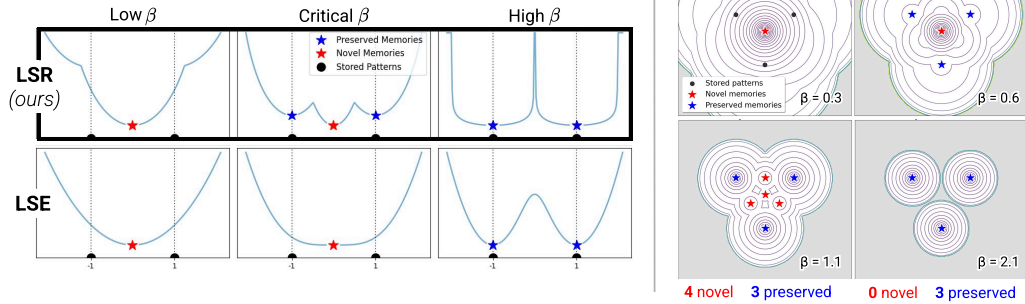


Figure 1: LSR energy can create more memories than there are stored patterns under critical regimes of β . Left: 1D LSR vs LSE energy landscape. Note that LSE is never capable of having more local minima than the number of stored patterns. Right: 2D LSR energy landscape, where increasing β creates novel local minima where basins intersect. Unsupported regions are shaded gray.

of the LSE energy corresponds to a softmax over all stored patterns, recent works have proposed sparsified variants to improve scalability. In particular, Hu et al. [12] and dos Santos et al. [13] consider sparsified softmax-based gradients, effectively projecting the full gradient onto a reduced support.¹ Alternatively, Wu et al. [14] propose learning new representations for the memories to increase capacity, while continuing to use the LSE energy in the transformed representation space.

In this work, we consider the following motivating question: *can we achieve simultaneous perfect memorization and generalization in associative memory models?* While the exponential separation function enables DenseAMs to achieve high memory capacity, capacity alone is not the only desideratum. In standard supervised learning, it was long believed that exactly interpolating or memorizing the training data — achieving zero training loss — would harm generalization. However, recent advances, particularly in deep learning, have challenged this belief: models that perfectly fit the training data can still generalize well. Although this phenomenon gained prominence with deep networks, it has earlier roots in kernel methods and boosting [15, 16, 17].

An analogous question arises in the context of associative memory (AM) models. Traditionally, AMs focus on storing a fixed set of patterns. But from a broader machine learning perspective, the goal extends beyond memorization to include the generation of new, meaningful patterns. Prior work using LSE-type energy functions has shown that generating such novel patterns typically requires sacrificing perfect recall of the original patterns. This trade-off highlights a core tension between memorization and generalization. To address this, we explore alternative separation functions that can preserve exact memorization while enabling the emergence of new patterns — pushing toward models that truly unify memory and generalization.

Our approach is also motivated by the well-established connection between the energy and probability density function. An energy function $E : \mathbb{R}^d \rightarrow \mathbb{R}$ induces a probability density function $p : \mathbb{R}^d \rightarrow \mathbb{R}_{\geq 0}$ with $p(\mathbf{x}) = \exp[-E(\mathbf{x})] / \int_{\mathbf{z}} \exp[-E(\mathbf{z})] d\mathbf{z}$. Conversely, given a density p , we have an energy $E(\mathbf{x}) \propto -\log p(\mathbf{x})$, the negative log-likelihood. Minimizing the energy corresponds to maximizing the log-likelihood (with respect to the corresponding density). Based on this connection, with $Q(\cdot) = \log(\cdot)$, the $\exp[-E_{\beta}(\mathbf{x}; \Xi)] = \sum_{\mu} F(\beta S(\mathbf{x}, \xi_{\mu}))$ in eq. (1) (assuming g is identity) is the corresponding (unnormalized) density at $\mathbf{x}$. Assuming that the memories $\xi_{\mu} \sim p$ are sampled from an unknown ground truth density p , the $\exp[-E_{\beta}(\mathbf{x}; \Xi)]$ is an unnormalized **kernel density estimate** or KDE of p at $\mathbf{x}$ with the *kernel* F and bandwidth $1/\beta$ [18]. Thus, the LSE energy with $F(x) = \exp(x)$ and $S(\mathbf{x}, \mathbf{x}') = -1/2 \|\mathbf{x} - \mathbf{x}'\|^2$ corresponds to the KDE of p with the Gaussian kernel.

KDE is well studied in nonparametric statistics [18, 19], and various forms of kernels have been explored. The quality of the estimates are well characterized in terms of properties on the kernels; we will elaborate on this in the sequel. While the Gaussian kernel is extremely popular for KDE (much like LSE in AM literature), there are various other kernels which have better estimation abilities than the Gaussian kernel. Among the commonly used kernels, the Epanechnikov kernel

¹Sparsified softmax-based gradients can be interpreted as specific projections of the original gradient.

has the most favorable estimation quality (see section 2). In our notation, this corresponds to a kernel $F(x) = \max(1 + x, 0) = \text{ReLU}(1 + x)$, a shifted ReLU operation (again with $S(\mathbf{x}, \mathbf{x}') = -1/2\|\mathbf{x} - \mathbf{x}'\|^2$). This results in a novel energy function that we name log-sum-ReLU or LSR (see eq. (3)). *Surprisingly, we show that this energy function is capable of both exactly memorizing all M original patterns (with $M \sim \exp(d)$) and simultaneously generating new, meaningful emergent memories* (see Definition 2). This defies the conventional tradeoff seen in prior AM models — where improving generalization (ability to create emergent local minima) typically requires compromising exact memorization — and reveals that precise memory storage and creative pattern generation are not inherently at odds with one another. To summarize, we make the following contributions in this work:

- **Novel ReLU-based energy function with exponential memory capacity.** We propose a LSR energy function for DenseAM using the popular ReLU activation, built upon the connection between energy functions and densities. In Theorems 1 and 2 respectively, we demonstrate exact retrieval and exponential memory capacity of LSR energy, without the use of $\exp(\cdot)$ separation function.
- **Simultaneous memorization and emergence.** We show that this LSR energy has a unique property of *simultaneously* being able to *exactly* retrieve all original memories (training data points) while also creating many *emergent* memories (additional local minima). The total number of memories of LSR can exceed the number of stored patterns, a property absent with LSE (see Proposition 1). When applied to images, LSR can generate novel and seemingly creative memories that are not present in the training dataset.

2 Kernel Density Estimation and the Choice of Kernels

We now provide a brief overview of Kernel Density Estimation (KDE) considering the univariate setting for simplicity; similar conclusions also hold in higher dimensions. Given a sample $\Xi = \{\xi_\mu \in \mathbb{R}, \mu \in [M]\}$ drawn from an unknown density f , the KDE is defined as $\hat{f}_h(\xi) = (Mh)^{-1} \sum_{\mu=1}^M K\left(\frac{\xi - \xi_\mu}{h}\right)$, where $K(\cdot)$ is the kernel function and $h > 0$ is the bandwidth parameter. The kernel function is assumed to satisfy: (i) symmetry (i.e., $K(-x) = K(x)$, for all $x \in \mathbb{R}$), (ii) positivity (i.e., $K(x) \geq 0$, for all $x \in \mathbb{R}$) and (iii) normalization (i.e., $\int_{\mathbb{R}} K(x) dx = 1$). Note that for the purpose of KDE, the scale of the kernel function is not unique. That is, for a given $K(\cdot)$, we can define $\tilde{K}(\cdot) = b^{-1}K(\cdot/b)$, for some $b > 0$. Then, one obtains the same KDE by rescaling the choice of h . Hence, the shape of the kernel function plays a more important role in determining the choice of the kernel. We now introduce two parameters associated with the kernel, $\mu_K := \int_{\mathbb{R}} x^2 K(x) dx$ and $\sigma_K := \int_{\mathbb{R}} K^2(x) dx$ that correspond to the *scale* and *regularity* of the kernel. We will discuss below how the *generalization error* of KDE depends on the aforementioned parameters.

The *generalization error* of $\hat{f}_h(\xi)$ is measured by the Mean Integrated Squared Error (MISE), given by $\text{MISE}(h) = \mathbb{E} \left[\int_{\mathbb{R}} (\hat{f}_h(\xi) - f(\xi))^2 d\xi \right]$. Assuming that the ground-truth density $f(\xi)$ is twice continuously differentiable, a second-order Taylor expansion gives the leading terms of the $\text{MISE}(h)$, which decomposes into squared bias and variance terms: $\text{MISE}(h) \approx \frac{\mu_K^2}{4} h^4 \int_{\mathbb{R}} |f''(\xi)|^2 d\xi + \frac{\sigma_K}{Mh}$; see Wand and Jones [18, Section 2.5] for details. This result shows that reducing h decreases bias but increases variance, while increasing h smooths the estimate but introduces bias, highlighting the bias-variance trade-off. The optimal mean-square is obtained by minimizing $\text{MISE}(h)$ with respect to h . We thus obtain the optimal choice of h and the optimal generalization accuracy as

$$h_* := \left(\frac{\sigma_K}{M\mu_K^2} \frac{4}{\int_{\mathbb{R}} |f''(\xi)|^2 d\xi} \right)^{1/5} \quad \text{and} \quad \text{MISE}(h_*) \approx \frac{5}{4} \left(\frac{\sqrt{\mu_K} \sigma_K \int_{\mathbb{R}} |f''(\xi)|^2 d\xi}{M} \right)^{4/5}, \quad (2)$$

respectively. From this, we see that the choice of the kernel K in the KDE, controls the generalization error via the term $\sqrt{\mu_K} \sigma_K$.

Thus, a natural question is to find the choice of kernel $K(\cdot)$ that results in the minimum $\text{MISE}(h_*)$. As discussed above, the scale of the kernel function is non-unique. Hence, the problem boils down to minimizing σ_K (which is regularity parameter of the kernel, determining the shape), subjected to $\mu_K = 1$ (without loss of generality), over the class of normalized, symmetric, and positive kernels.

This problem is well-studied (see, for example, [20], [21], Wand and Jones [18, Section 2.7]), and, as it turns out, the Epanechnikov kernel $K_{\text{epan}}(x) = \max\{1 - x^2, 0\} = \text{ReLU}(1 - x^2)$ achieves the optimal *generalization error*. The quantity, $\text{Eff}(\mathbf{K}) := \sigma_K / \sigma_{K_{\text{epan}}}$ is hence referred to as the efficiency of any kernel with respect to the Epanechnikov kernel. A description of choices for kernel functions and their efficiency relative to the Epanechnikov kernel is provided in Appendix C.

The number of modes of the KDE has also been examined in the literature, mostly when the target is unimodal. Assuming that the target is unimodal, a direct consequence of Mammen [22, Theorem 1] on the number of modes of the KDE when $d = 1$ (also see Geshkovski et al. [23, Theorem 1.1]) is that the number of modes of KDE with a Gaussian kernel with bandwidth h is $\tilde{\Theta}(1/\sqrt{h})$; see Geshkovski et al. [23, Section 1.2] for extensions to dimension $d > 1$.

3 A New Energy Function with Emergent Memory Capabilities

So far, we have seen the relationship between the LSE energy and the KDE, i.e., $\exp[-E_\beta(\mathbf{x}; \Xi)]$ is an unnormalized kernel density estimate with the Gaussian kernel and the bandwidth $1/\beta$, and the optimality of using the Epanechnikov kernel in KDE. Given these observations, we will explore the use of the corresponding shifted-ReLU separation function $\text{ReLU}(1 + x)$ in the energy function instead of the widely used exponentiation. Before we state the precise energy functions, we compare and contrast the shapes of these separation functions $F(\beta x)$ in fig. 2 for varying values of the inverse temperature β . Note that, as the β increases, both these separation functions decay faster. However, as expected, the shifted-ReLU separation linearly decays and then zeroes out.

Recall that LSE ENERGY is given by $E_\beta^{\text{LSE}}(\mathbf{x}; \Xi) = -\frac{1}{\beta} \log \sum_{\mu=1}^M \exp(-\frac{\beta}{2} \|\mathbf{x} - \xi_\mu\|^2)$. Based on the discussion on separation functions, our proposed LSR ENERGY (which we also refer to as Epanechnikov energy) is given by

$$E_\beta^{\text{LSR}}(\mathbf{x}; \Xi) = -\frac{1}{\beta} \log \left(\epsilon + \sum_{\mu=1}^M \text{ReLU} \left(1 - \frac{\beta}{2} \|\mathbf{x} - \xi_\mu\|^2 \right) \right), \quad (3)$$

where $\|\cdot\|$ describes the Euclidean norm and β is an inverse temperature.

The factor $\epsilon \geq 0$ in the LSR energy is a small non-negative constant, where an $\epsilon > 0$ ensures that every point in the space has finite (albeit extremely large $O(\log(1/\epsilon))$) energy for all values of β . Indeed, with $\epsilon = 0$, defining $S_\mu \triangleq \{\mathbf{x} \in \mathcal{X} : \|\mathbf{x} - \xi_\mu\| \leq \sqrt{2/\beta}\}$, it is easy to see that $\forall \mathbf{x} \in \mathcal{X} \setminus \cup_{\mu=1}^M S_\mu$, $E_\beta^{\text{LSR}}(\mathbf{x}) = \infty$. This is a result of the finite-ness of the ReLU separation function. Regions of infinite energy implies zero probability density, which matches the finite support of the density estimate with the Epanechnikov kernel. Based on the introduced LSR energy, we next highlight the following favourable properties; see appendix E for the proofs and technical details.

Theorem 1. Let $r = \min_{\mu, \nu \in [M], \mu \neq \nu} \|\xi_\mu - \xi_\nu\|$ be the minimum Euclidean distance between any two memories. Let $S_\mu(\Delta) = \{\mathbf{x} \in \mathcal{X} : \|\mathbf{x} - \xi_\mu\| \leq \Delta\}$ be a basin around the μ^{th} memory for some basin radius $\Delta \in (0, r)$. Then, with $\beta = 2/(r - \Delta)^2$, for any $\mu \in [M]$ and any input $\mathbf{x} \in S_\mu(\Delta)$, the output of the DenseAM via LSR energy gradient descent is exactly ξ_μ , implying that all memories $\xi_\mu, \mu \in [M]$ are retrievable. Furthermore, if the learning rate of the energy gradient descent is set appropriately, then for any $\mu \in [M]$ and any $\mathbf{x} \in S_\mu(\Delta)$, the memory is exactly retrieved with a single energy gradient descent step (single step retrieval).

The above result states that, given a set of memories, and an appropriately selected β , there is a distinct basin of attraction $S_\mu(\Delta)$ around each memory ξ_μ , and any input $\mathbf{x}$ from within that basin exactly retrieves the memory as the output of the DenseAM.

Remark 1. For a finite but appropriately large β , the LSR energy gradient $\nabla E_\beta^{\text{LSR}}(\xi_\mu; \Xi)$ at any memory ξ_μ is exactly zero, implying exact retrieval of the memory. The LSE energy gradient $\nabla E_\beta^{\text{LSE}}(\xi_\mu; \Xi)$ is only approximately zero, and the retrieved point is approximately equal to an

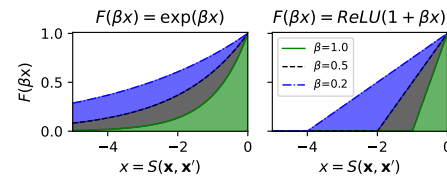


Figure 2: Visualizing the separation functions $F(\beta x) = \exp(\beta x)$ (LSE) and $F(\beta x) = \text{ReLU}(1 + \beta x)$ (LSR) with $x = S(\mathbf{x}, \mathbf{x}')$ for varying values of β . We focus on $S(\mathbf{x}, \mathbf{x}') = -1/2 \|\mathbf{x} - \mathbf{x}'\|^2$.

original memory [7, Theorem 3]. However, if $\beta = \infty$ then the LSE energy gradient is exactly zero at the memory.

The striking phenomenon that we observe with LSR energy is that the DenseAM can simultaneously create local energy minima around the original memories as well as additional local minima around points that are not part of the set of original memories; see fig. 1. We formalize this concept below through the notion of *global emergence*.

Definition 1 (Novel local minima). Consider a DenseAM parameterized with M memories $\Xi = \{\xi_1, \dots, \xi_M\}$, $\xi_\mu \in \mathcal{X}$, and equipped with an energy function $E_\beta(\mathbf{x}; \Xi)$ at any state $\mathbf{x} \in \mathcal{X}$ for a specific inverse temperature $\beta > 0$. For some $\varepsilon > 0$, we define $\mathcal{M}_\varepsilon$ as the (possibly empty) set of **novel local minima** $\tilde{\xi} \in \mathcal{X}$ such that $\forall \tilde{\xi} \in \mathcal{M}_\varepsilon$,

- (a) $\tilde{\xi}$ is a local energy minimum with $\nabla E_\beta(\tilde{\xi}; \Xi) = 0$ and $\nabla^2 E_\beta(\tilde{\xi}; \Xi) \succ 0$,
- (b) $\tilde{\xi}$ is novel with respect to the original memories, that is, $\min_{\mu \in [M]} \|\tilde{\xi} - \xi_\mu\| \geq \varepsilon$.

Definition 2 (Global emergence). For the DenseAM in Definition 1 and for some $\varepsilon > 0$, let $\mathcal{M}_\varepsilon$ be the (possibly empty) set of novel local minima. For some $\beta \in (0, \infty)$, we claim that this system exhibits ε -**global emergence** if (i) each original memory ξ_μ , $\mu \in [M]$ is a local energy minimum with $\nabla E_\beta(\xi_\mu; \Xi) = 0$ and $\nabla^2 E_\beta(\xi_\mu; \Xi) \succ 0$ (positive definite), and (ii) the set $\mathcal{M}_\varepsilon$ is non-empty. We term $\mathcal{M}_\varepsilon$ as the set of ε -**globally emergent memories**.

The notion of ε -global emergence specifically refers to those new patterns that arise *after* all original memories have been exactly stored. Definition 2 characterizes emergence as a property of the global energy function at a specific inverse temperature, requiring simultaneous exact recovery of all original memories and the presence of at least one novel local minimum (parameterized with ε). It is instructive to start by understanding the above definition for DenseAMs equipped with LSE energy. According to Ramsauer et al. [7], any point $\mathbf{x}^*$ such that $\nabla E_\beta^{\text{LSE}}(\mathbf{x}^*; \Xi) = 0$ is defined via the softmax corresponding to the transformer attention as follows: $\mathbf{x}^* = \sum_{\mu=1}^M \text{softmax}(\beta(\mathbf{x}^*)^\top \xi_\mu) \xi_\mu$, and the softmax can be highly peaked if all $\{\xi_\mu\}_{\mu=1}^M$ are well separated and $\mathbf{x}^*$ is near a stored pattern ξ_μ . If no stored pattern ξ_μ is well separated from the others, then $\mathbf{x}^*$ is close to a global fixed point, which is the arithmetic mean of all the stored patterns. Based on this, we can make the following observations:

- Case I: *All LSE memories are novel.* With a large enough but finite β , there is a minimum close to each of the original memories. However, each of these local minima will be considered novel local minima as these are distinct from the original memories, thus condition (ii) in Definition 2 will be satisfied. However, then the condition (i) in Definition 2 would not be satisfied.
- Case II: *No LSE memories are novel.* If we do consider the case $\beta = \infty$, then the original memories would exactly be the local energy minima, and condition (i) will be satisfied. But then the set of novel local minima $\mathcal{M}_\varepsilon$ for a strictly positive ε would be empty, violating condition (ii).
- Case III: *Novel LSE memories form only when basins merge.* For a moderate β , LSE can form novel local minima by merging the basins of attractions of the original memories, thereby giving us a non-empty $\mathcal{M}_\varepsilon$ and satisfying condition (ii) in Definition 2. However, condition (i) will be violated as the memories whose basins are merged would no longer be local minima.

Thus, we can make this more formal in the following:

Proposition 1. Assume $\{\xi_\mu\}_{\mu=1}^M$ are i.i.d. from any density fully supported on $\mathcal{X}$. Note that they are linearly independent with probability 1, as otherwise they lie in a lower dimensional space. Then, for any $\beta > 0$, the LSE energy, $E_\beta^{\text{LSE}}(\cdot)$, does not satisfy the ε -global emergence in Definition 2.

One can argue that global emergence as in Definition 2 is too restrictive; we also want to characterize an individual local minimum as “emergent”, or not. Thus we present a relaxed *local* notion of emergence in the following, noting that LSE does not satisfy this weaker form of emergence either:

Definition 3 (Locally emergent memory). Consider the DenseAM in Definition 1 with a non-empty set of novel local minima $\mathcal{M}_\varepsilon$ for a $\varepsilon > 0$. For any $\tilde{\xi} \in \mathcal{M}_\varepsilon$, let $S(\tilde{\xi}) \subseteq \Xi$ be the minimal non-empty subset of Ξ such that, for each $\xi_\mu \in S(\tilde{\xi})$, $\tilde{\xi}$ is no longer a local minimum of the energy $E_\beta(\cdot, \Xi \setminus \{\xi_\mu\})$ that excludes the memory. Then we define $\tilde{\xi} \in \mathcal{M}_\varepsilon$ as a ε -**locally emergent memory** if there is some original memory $\xi_\mu \in S(\tilde{\xi})$ which still is a local minimum of the energy

$E_\beta(\cdot; \Xi)$. If every original memory $\xi_\mu \in \mathcal{S}(\tilde{\xi})$ is a local minimum of $E_\beta(\cdot; \Xi)$, we call $\tilde{\xi} \in \mathcal{M}_\varepsilon$ a ε -local strongly emergent memory.

While Definition 2 discusses emergence at a global energy level, Definition 3 characterizes emergence locally for each of the novel local minima. This distinction is important as we see emergence as a general property of the energy function that can be driven by a subset of the memories. Definition 2 requires all original memories to be retrievable, while in Definition 3 we allow for emergence due to the interaction of stored patterns in a system even if not all original memories are retrievable, so long as a critical subset of the original memories are. Global emergence implies the existence of at least one local strongly emergent memory; every globally emergent memory is a local strongly emergent one. However, the existence of a local strongly emergent memory does not imply global emergence. As with global emergence, we can see that a DenseAM with LSE energy does not also have locally emergent memories. First note that, for a finite β , all local minima are novel local minima, and the minimal set $\mathcal{S}(\tilde{\xi})$ for a novel local minimum $\tilde{\xi}$ is the whole set Ξ given the infinite support of the exponential function, with none of them being a local minimum. For $\beta = \infty$, the set of novel local minima $\mathcal{M}_\varepsilon$ is empty. So in both cases, the required conditions are not satisfied and there are no locally emergent memories with LSE energy.

Note that these novel local minima are different from the well-studied spurious memories or parasitic memories [24, 25]. In classical AMs, spurious memories start appearing when the AM is packed with memories beyond its memory capacity. In contrast, the appearance of emergence (novel local minima) does not seem to be related to whether the DenseAM is over or under capacity — as we show in fig. 1, a locally emergent memory can appear even with just 2 stored patterns of any dimension. Spin-glass states [26] do not occur in either the LSE or LSR energy due to our use of Euclidean similarity over the dot product in both energies.

The next result provides explicit characterization of the form of novel memories in LSR energy.

Proposition 2. Consider the LSR energy in eq. (3). For any $\mathbf{x} \in \text{interior}(\mathcal{X})$, letting $B(\mathbf{x}) \triangleq \{\mu \in [M] : \|\mathbf{x} - \xi_\mu\| \leq \sqrt{2/\beta}\}$, there is a local minima of the LSR energy which is given by $\frac{1}{|B(\mathbf{x})|} \sum_{\mu \in B(\mathbf{x})} \xi_\mu$.

Note that when $|B(\mathbf{x})| = 1$, the local minima in Proposition 2 is exactly the stored memory $\{\xi_\mu\}_{\mu=1}^M$. With $|B(\mathbf{x})| > 1$, it is not equal to any of the original memories $\{\xi_\mu, \mu \in [M]\}$ (with probability 1). The region $\{\mathbf{x} \in \mathcal{X} : |B(\mathbf{x})| > 1\} \subset \mathcal{X}$ is precisely characterized as $(\cup_{\mu \in [M]} S_\mu) \setminus (\cup_{\mu \in [M]} S_\mu(\Delta))$ where S_μ is the region of finite energy around the μ^{th} memory and $S_\mu(\Delta)$ (defined in Theorem 1) is the distinct attracting basin for the μ^{th} memory. The following theorem shows that this LSR based DenseAM is capable of simultaneously retrieving all (up to exponentially many) memories while also creating many novel local minima, and quantifies this phenomenon precisely.

Theorem 2. Consider a DenseAM parameterized with M memories $\Xi = \{\xi_1, \dots, \xi_M\}$ sampled uniformly from $\mathcal{X}$ with $\text{vol}(\mathcal{X}) = V < \infty$ and the LSR energy $E_\beta^{\text{LSR}}(\cdot; \Xi)$ defined in eq. (3). For each novel local minimum $\mathbf{x}^* = \frac{1}{|B(\mathbf{x}^*)|} \sum_{\mu \in B(\mathbf{x}^*)} \xi_\mu$ given in Proposition 2, define $D_{\max}(\mathbf{x}^*) := \max_\mu \|\mathbf{x}^* - \xi_\mu\|$,

$$\delta_{\min}(\mathbf{x}^*) := \min_{\mu \in B(\mathbf{x}^*)} \left(\frac{2}{\beta} - \|\mathbf{x}^* - \xi_\mu\|^2 \right), \quad \gamma_{\min}(\mathbf{x}^*) = \min_{\nu \notin B(\mathbf{x}^*)} \left(\|\mathbf{x}^* - \xi_\nu\|^2 - \frac{2}{\beta} \right).$$

Then, for all $\beta > 0$ such that $\max_{\mathbf{x}^*} \delta_{\min}(\mathbf{x}^*) > 0$ and $\min_{\mathbf{x}^*} \gamma_{\min}(\mathbf{x}^*) > 0$, there exists an $\varepsilon := \min_{\mathbf{x}^*} \left(\sqrt{D_{\max}^2(\mathbf{x}^*) + \min\{\delta_{\min}(\mathbf{x}^*), \gamma_{\min}(\mathbf{x}^*)\}} - D_{\max}(\mathbf{x}^*) \right) > 0$ such that $E_\beta^{\text{LSR}}(\cdot)$ satisfies the ε -global emergence condition (Definition 2) with high probability, as:

- With probability at least $\delta \in (0, 1)$, and $M = \Theta(\sqrt{1-\delta} \exp(\alpha d))$ for a positive α , all memories are retrievable as per Theorem 1 with the value of the minimum pairwise distance $r = \min_{\mu, \nu \in [M], \mu \neq \nu} \|\xi_\mu - \xi_\nu\| \geq (V_d/V)^{-1/d} e^{-2\alpha}$ and per-memory basin radius $\Delta \in (0, (V_d/V)^{-1/d} e^{-2\alpha})$ with a $\beta \leq 2/((V_d/V)^{-1/d} e^{-2\alpha} - \Delta)^2$, where V_d is the volume of the unit ball in $\mathbb{R}^d$.
- For each novel local minimum $\mathbf{x}^*$, there exists a radius

$$r^* := \sqrt{D_{\max}^2(\mathbf{x}^*) + \min\{\delta_{\min}(\mathbf{x}^*), \gamma_{\min}(\mathbf{x}^*)\}} - D_{\max}(\mathbf{x}^*) > 0, \quad (4)$$

such that $S_{\mathbf{x}^*}(r^*) = \{\mathbf{x} \in \mathcal{X} : \|\mathbf{x} - \mathbf{x}^*\| \leq r^*\}$ forms a basin around the novel memory $\mathbf{x}^*$, and for any $\mathbf{x} \in S_{\mathbf{x}^*}(r^*)$, the output of the DenseAM via energy gradient descent is exactly $\mathbf{x}^*$, implying the novel memories are retrievable.

Furthermore, with probability at least $1 - M^{-2}$, the number of ε -globally emergent memories is

$$O\left(\exp\left(M \frac{V_d}{V} \left(\frac{2}{\beta}\right)^{\frac{d}{2}} \log\left(\frac{eV}{V_d} \left(\frac{\beta}{2}\right)^{\frac{d}{2}}\right)\right)\right). \quad (5)$$

In particular, for fixed $\beta > 0$ and d , the bound grows with M whenever $\beta > 0$ satisfies $|B(\mathbf{x})| \in (1, M]$ due to Proposition 2. Here, for any $\mathbf{x} \in \mathcal{X}$, a large β leads to a small $|B(\mathbf{x})|$, and $|B(\mathbf{x})| > 1$ is the required condition for a novel local minima; a small β leads to a large $|B(\mathbf{x})|$, and $|B(\mathbf{x})| \leq M$ stands for the case $B(\mathbf{x})$ at most covers the entire domain $\mathcal{X}$.

The above result demonstrates that with the LSR energy, it is possible to exactly memorize all original patterns (with high probability) and still generate new patterns — what we term emergent memories (Definition 2). This behavior is surprising in the same way interpolating models in deep learning generalize unexpectedly well: both challenge the classical bias-variance intuition [15, 16]. While LSE-based models also produce novel memories, they typically do so at the expense of perfect recall of the original patterns. In Proposition 1 we show that LSE based DenseAMs do not have the global emergence property. This distinction highlights a key contribution of our work: *new memory creation need not come at the cost of perfect memorization*.

Next, we provide an exact order (i.e., upper and lower bounds) of the number of emergent memories under a grid design assumption.

Proposition 3. If $\{\xi_\mu\}_{\mu=1}^M$ form a grid over $\mathcal{X}$ of equal size with $\text{Vol}(\mathcal{X}) = V < \infty$, the number of emergent memories is of order $\Theta\left((M^{1/d} - \lambda^{1/d} + 1)^d\right)$, where $\lambda = \Theta\left(MV^{-1}(8/\beta)^{\frac{d}{2}}\right)$ and for $\beta > 0$ such that $1 < \lambda \leq M$.

Note that we showed an explicit form of the emergent memories $\mathbf{x}^* = \frac{1}{|B(\mathbf{x}^*)|} \sum_{\mu \in B(\mathbf{x}^*)} \xi_\mu$, where $B(\mathbf{x}^*) = \{\mu : \|\mathbf{x}^* - \xi_\mu\| < \sqrt{2/\beta}\} \subset \{1, \dots, M\}$. equation 5 in Theorem 2 is stated under the uniform sampling regime, and Proposition 3 is stated under a fixed grid setting. In general, the number of emergent memories varies according to the specific geometry of the stored patterns $\{\xi_\mu\}_{\mu=1}^M$, i.e., whether such a subset of $\{1, \dots, M\}$ can be realized by a ball $\{\|\mathbf{x} - \xi_\mu\| < \sqrt{2/\beta}\}$. This can grow much faster than a linear order of M , and is naively bounded by 2^M .

4 Experiments

4.1 Quantifying the scaling of emergent memories

How many local minima do we see in practice as we: (a) vary the number of stored patterns, (b) change the dimensionality of those patterns, and (c) vary the inverse temperature β ? We observe that, at critical values of β , we can create *orders of magnitude* more emergent memories than stored patterns. These results are shown in fig. 3 (left).

To quantify the number of local minima induced by the LSR energy, we uniformly sample M patterns from the d -dimensional unit hypercube to serve as memories Ξ . We enumerate all possible local minima of the LSR energy by computing the centroid $\bar{\xi}_K := |\mathcal{K}|^{-1} \sum_{\mu \in \mathcal{K}} \xi_\mu$ for every possible subset of stored patterns $\mathcal{K} \subseteq [M]$ (there are 2^M possible subsets if we allow for singleton sets). For each subset, we first check that its centroid is supported (i.e., that $E_\beta^{\text{LSR}}(\bar{\xi}_K; \Xi) < \infty$ at $\epsilon = 0$), and then declare that $\bar{\xi}_K$ is a local minimum of the LSR energy if $\|\nabla E_\beta^{\text{LSR}}(\bar{\xi}_K; \Xi)\| < \delta$ for small $\delta > 0$. β values are varied across the “interesting” regime between fully overlapping support regions (a single local minimum in the unit hypercube) to fully disjoint support regions around each memory. See experimental details in appendix D.2.

Certain values of β yield particularly interesting behavior. For example, we observe that LSR can create orders of magnitude more emergent memories under ranges of β where: (i) a majority (>60%) of stored patterns are also recoverable, and (ii) around 20 percent of the unit hypercube is still

LSR Energy creates **novel memories** while preserving **stored patterns**

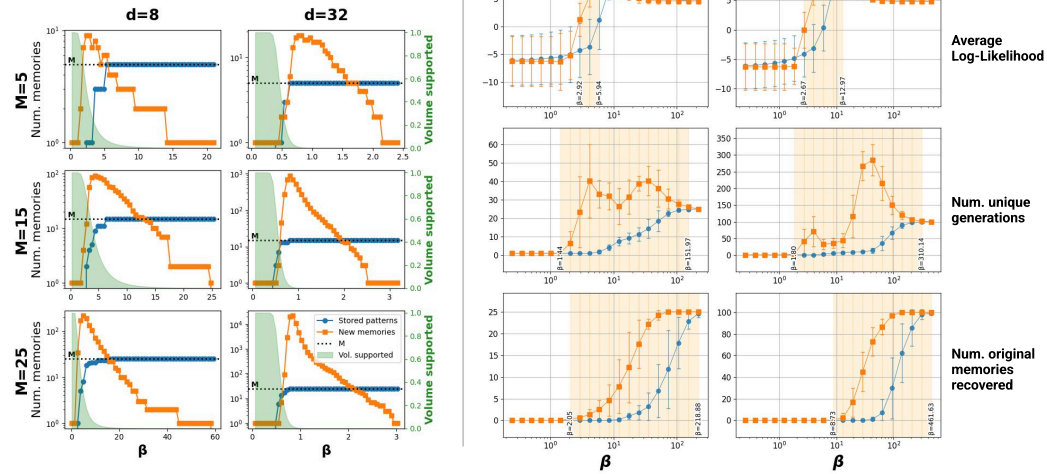


Figure 3: (Left) Analyzing local minima in LSR energy reveals a number of **novel memories** several orders of magnitude larger than M , the number of stored patterns, at critical values of β (note that the y-axes are logscale). These emergent memories occur even while still preserving the **stored patterns** as memories. Smaller values of β have a larger **region of support** on the unit hypercube. (Right) Given samples from some known true density function (in this case, a $k = 10$ mixture of 8-dim Gaussians with means drawn uniformly from the unit hypercube and $\sigma = 0.1$), memories from LSR energy have a log-likelihood comparable to, and occasionally slightly higher than, LSE under the true density function. Note that LSR achieves comparable log-likelihood while having more unique samples than LSE, even when both are seeded with the same $N = 500$ queries. Regions of β where LSR outperforms LSE on a metric are specified by the orange regions. Error bars indicate the standard error across 5 different seeds for sampling stored patterns and initial queries.

supported. Note that in each experiment there are choices of β such that the LSR energy does not exhibit *global emergence* (Definition 2, i.e., at low β where novel memories are forming but not all stored patterns are yet retrievable). However, in these regions *local emergence* (Definition 3) of the novel memories still holds (see fig. 1 for intuition).

4.2 Generative quality of emergent memories

LSR memories are certainly more diverse than those of LSE, but do they represent more “meaningful” samples from a true, underlying density function $p(\mathbf{x})$ (as measured by their log-likelihood)? The experimental setup is as follows: Let $p(\mathbf{x})$ be a mixture of k Gaussians whose means $\mu_i \sim \mathcal{U}([0, 1]^d)$ for $i \in \llbracket k \rrbracket$ are uniformly sampled from the d -dimensional unit hypercube with scalar ($\sigma = 0.1$) covariances such that $p(\mathbf{x}) = \frac{1}{k} \sum_{i=1}^k \mathcal{N}(\mathbf{x} \mid \mu_i, \sigma^2 \mathbf{I}_d)$. We sample M points $\{\xi_1, \dots, \xi_M\}, \xi_\mu \sim p(\mathbf{x})$ to serve as the stored patterns Ξ used to parameterize both the LSE and LSR energies from eq. (3). Define a thin *support boundary* induced by pattern ξ_μ to be $\text{supp}[\xi_\mu; \delta] = \{\mathbf{x} : 2\beta^{-1} - \delta \leq \|\mathbf{x} - \xi_\mu\|^2 < 2\beta^{-1}\}$ for some small $\delta > 0$. Then, for initial points $\mathbf{x}_n^{(0)}$, $n \in \llbracket N \rrbracket$ sampled from the support boundary around each stored pattern,² LSE memories can be found using gradient descent

$$\mathbf{x}_n^{(t)} = \mathbf{x}_n^{(t-1)} - \alpha \nabla E_\beta^{\text{LSE}}(\mathbf{x}_n^{(t-1)}; \Xi), \quad (6)$$

until convergence to a *memory* $\mathbf{x}_n^*$. We use algorithm 1 to efficiently find the LSR memory corresponding to each initial point. Thus we have N “samples” (memories) from both LSE and LSR on which we compare three metrics of interest in fig. 3 (right):

1. **Average Log-Likelihood.** Do LSR memories $\mathbf{x}_{\text{LSR}}^*$ have higher $\log p(\mathbf{x}_{\text{LSR}}^*)$ than LSE memories?
2. **Number of Unique Samples.** Does high $\log p(\mathbf{x}_{\text{LSR}}^*)$ occur alongside many emergent memories?

²We use the same initial points to seed the dynamics of both E^{LSR} and E^{LSE} . See appendix D.3 for details.

All LSR memories

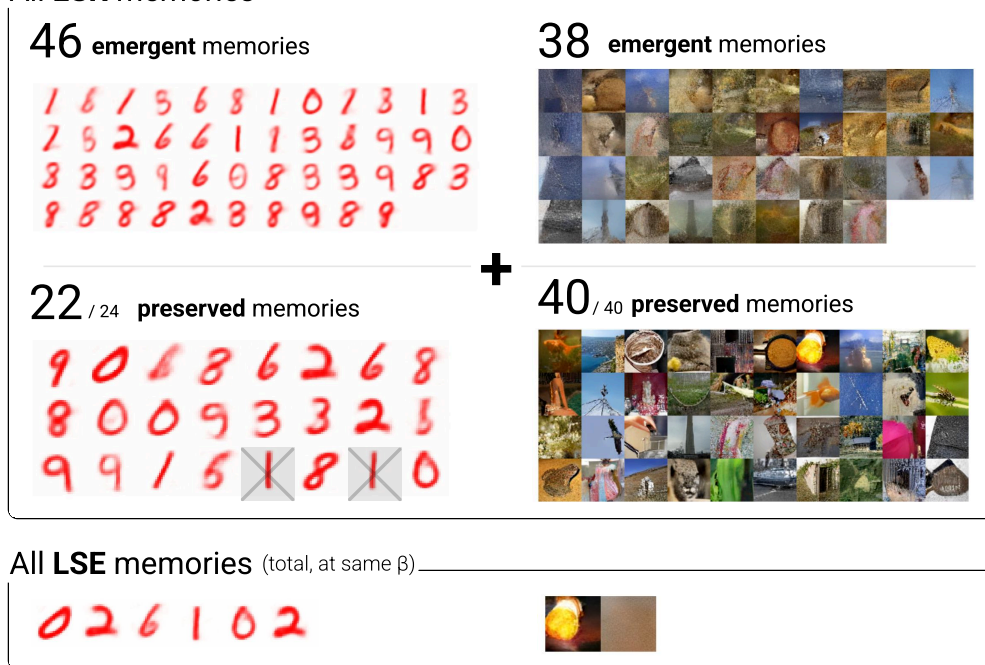


Figure 4: LSR’s emergent memories appear as novel, creative generations when the energy is applied to a semantically meaningful latent space. **(Left)** 24 randomly-selected MNIST images are encoded into 10-dim VAE latents and stored into an LSR- and LSE-energy using a carefully chosen β (see algorithm 2). Gray boxes indicate which stored patterns were not preserved at the chosen β . **(Right)** 40 TinyImagenet [27] images are encoded into 256-dim latents using a pretrained VAE [28] and stored into an LSR- and LSE-energy using a carefully chosen β . Note that in this TinyImagenet example the LSR energy is, by definition, *globally emergent* since all stored patterns are recoverable, while the MNIST example is not. See experiment details in appendix D.4.

3. **Number of Original Memories Recoverable.** Does high $\log p(\mathbf{x}_{\text{LSR}}^*)$ occur when alongside high numbers of preserved memories? How does this trend compare with LSE memory performance?

The results tell a consistent story. Despite LSE energy being a more natural choice to model the underlying Mixture of Gaussians’ density $p(\mathbf{x})$ (LSE has a Gaussian kernel that makes it ideal for the modeling task), LSR can match LSE in log-likelihood while simultaneously generating more diverse samples and preserving the stored patterns. See appendix D.3 for more experimental results and extended discussion.

4.3 Emergent memories in latent space

What do emergent memories look like when LSR is applied to real-world datasets? To study this behavior, we use a VAE to encode MNIST and TinyImagenet [27] images into latent vectors that serve as the stored patterns for LSR and LSE energies (see fig. 4). Using a carefully chosen β , we compute *all* memories (both preserved and novel) for each energy. The emergent memories of LSR in principle are simply the centroids of small subsets of the stored patterns, yet when decoded they appear as plausible and creative generations.

With the same β value, LSR generates an order of magnitude more total memories than LSE. While this choice of β is somewhat arbitrary and could be tuned separately for each energy, LSE would only ever be able to retrieve up to M memories (where M is the total number of stored patterns). See full experiment details in appendix D.4.

Emergent memories are mechanistically simple: they are simply the centroids of small subsets of the stored patterns. The semantic novelty of the emergent memories in Figure 4 occur because the latent space is structured to be semantically meaningful, where averaging two or more stored patterns

produces seemingly novel semantics. We conduct a similar experiment in appendix D.5 where we ablate the VAE to show what emergent memories look like in pixel space.

5 Discussion

Emergent memories are a powerful tool for creating novel samples, but the “meaningfulness” of these samples is a nuanced question that depends heavily on the specific application domain and task requirements. For example, in Figure 3 (right) we show that the LSR energy approximates an unknown p.d.f. better than LSE’s energy while simultaneously generating diverse samples. This represents a desirable behavior of emergence, since high log-likelihood samples from the LSE energy are quite homogeneous, causing 500 initial queries to converge to the same ~ 10 memories. In this density estimation context, the emergent memories serve a clear functional purpose: they capture meaningful interpolations within the data distribution that improve generalization.

However, consider the novel memories from Tiny Imagenet in Figure 4. Though visually plausible, many of the emergent generations appear blurry and would be considered “undesirable” from the perspective of a high-fidelity image generation model. We discuss the limitations of emergent memories further in appendix A.

We note that there are potential parallels between emergent memories and “hallucinations” as observed in LLMs. We discuss the philosophical similarities between emergent memories and hallucinations in appendix B.1. It is also interesting to note that the LSR Energy presented in this work can have a feasible biological implementation using bipartite neurons. See further discussion in appendix B.2.

Finally, we reiterate that there are many choices for alternative kernels, and not just the Epanechnikov kernel. We discover that many kernels with compact support are capable of producing emergent memories and even manifolds, and the energy landscapes they produce can look quite different from each other. To this end, we show the “basin merging” behavior across different kernels in 1D in Figure 6 and we include an extended discussion on these other kernels in appendix C.

6 Conclusion

Our work introduces the LSR energy function which achieves the surprising combination of exact memorization of exponentially many patterns and the emergence of new, meaningful memories, thereby providing a powerful alternative to traditional AM formulations. The properties of the LSR energy define a novel class of *emergent memories* in Dense Associative Memory systems — a phenomenon not observed in any previous AM formulations. Unlike conventional models (e.g., LSE-based energies) where generalization (formation of local minima different from training data) typically comes at the cost of perfect memorization, LSR demonstrates that these objectives can coexist harmoniously in a single energy function. We additionally demonstrated that the diverse memories created by LSR achieve log-likelihood comparable to LSE when sampling from a true density function, while generating an order of magnitude more unique memories. Finally, we showed that when applied to latent representations of real-world image datasets, LSR’s emergent memories represent plausible and creative generations.

References

- [1] Dmitry Krotov and John J Hopfield. Dense associative memory for pattern recognition. *Advances in Neural Information Processing Systems*, 29, 2016.
- [2] Dmitry Krotov. A new frontier for hopfield networks. *Nature Reviews Physics*, 5(7):366–367, 2023.
- [3] Benjamin Hoover, Duen Horng Chau, Hendrik Strobelt, Parikshit Ram, and Dmitry Krotov. Dense associative memory through the lens of random features. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [4] John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.
- [5] Dmitry Krotov and John Hopfield. Dense associative memory is robust to adversarial inputs. *Neural computation*, 30(12):3151–3167, 2018.
- [6] Mete Demircigil, Judith Heusel, Matthias Löwe, Sven Uppgang, and Franck Vermet. On a model of associative memory with huge storage capacity. *Journal of Statistical Physics*, 168: 288–299, 2017.
- [7] Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=tL89RnzIiCd>.
- [8] Dmitry Krotov and John J Hopfield. Large associative memory problem in neurobiology and machine learning. In *International Conference on Learning Representations*, 2021.
- [9] Carlo Lucibello and Marc Mézard. Exponential capacity of dense associative memories. *Physical Review Letters*, 132(7):077301, 2024.
- [10] Dmitry Krotov. Hierarchical associative memory. *arXiv preprint arXiv:2107.06446*, 2021.
- [11] Benjamin Hoover, Duen Horng Chau, Hendrik Strobelt, and Dmitry Krotov. A universal abstraction for hierarchical hopfield networks. In *The Symbiosis of Deep Learning and Differential Equations II*, 2022.
- [12] Jerry Yao-Chieh Hu, Donglin Yang, Dennis Wu, Chenwei Xu, Bo-Yu Chen, and Han Liu. On sparse modern hopfield model. *Advances in Neural Information Processing Systems*, 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/57bc0a850255e2041341bf74c7e2b9fa-Paper-Conference.pdf.
- [13] Saul José Rodrigues dos Santos, Vlad Niculae, Daniel C McNamee, and Andre Martins. Sparse and structured hopfield networks. In *Forty-first International Conference on Machine Learning*, 2024. URL <https://openreview.net/forum?id=OdPlFWExX1>.
- [14] Dennis Wu, Jerry Yao-Chieh Hu, Teng-Yun Hsiao, and Han Liu. Uniform memory retrieval with larger capacity for modern hopfield models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 53471–53514. PMLR, 21–27 Jul 2024. URL <https://proceedings.mlr.press/v235/wu24i.html>.
- [15] Mikhail Belkin, Alexander Rakhlin, and Alexandre B Tsybakov. Does data interpolation contradict statistical optimality? In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1611–1619. PMLR, 2019. URL <http://proceedings.mlr.press/v89/belkin19a/belkin19a.pdf>.
- [16] Mikhail Belkin. Fit without fear: remarkable mathematical phenomena of deep learning through the prism of interpolation. *Acta Numerica*, 30:203–248, 2021.
- [17] Peter L Bartlett, Andrea Montanari, and Alexander Rakhlin. Deep learning: a statistical viewpoint. *Acta numerica*, 30:87–201, 2021.

- [18] Matt P Wand and M Chris Jones. *Kernel smoothing*. CRC press, 1994.
- [19] Luc Devroye and Gábor Lugosi. *Combinatorial methods in density estimation*. Springer Science & Business Media, 2001.
- [20] Vassiliy A Epanechnikov. Non-parametric estimation of a multivariate probability density. *Theory of Probability & Its Applications*, 14(1):153–158, 1969.
- [21] Hans-Georg Müller. Smooth optimum kernel estimators of densities, regression curves and modes. *The Annals of Statistics*, pages 766–774, 1984.
- [22] Enno Mammen. On qualitative smoothness of kernel density estimates. *Statistics: a journal of theoretical applied statistics*, 26(3):253–267, 1995.
- [23] Borjan Geshkovski, Philippe Rigollet, and Yihang Sun. On the number of modes of gaussian kernel density estimators. *arXiv preprint arXiv:2412.09080*, 2024.
- [24] Shun-Ichi Amari and Kenjiro Maginu. Statistical neurodynamics of associative memory. *Neural Networks*, 1(1):63–73, 1988.
- [25] Anthony V Robins and Simon JR McCallum. A robust method for distinguishing between learned and spurious attractors. *Neural Networks*, 17(3):313–326, 2004.
- [26] Adriano Barra, Giuseppe Genovese, Peter Sollich, and Daniele Tantari. Phase diagram of restricted boltzmann machines and generalized hopfield networks with arbitrary priors. *Physical Review E*, 97(2):022310, 2018.
- [27] Ya Le and Xuan S. Yang. Tiny imagenet visual recognition challenge. 2015. URL <https://api.semanticscholar.org/CorpusID:16664790>.
- [28] Ollin Boer Bohan. Tiny autoencoder for stable diffusion. <https://github.com/madebyollin/taesd>, 2023.
- [29] A Paszke. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.
- [30] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL <http://github.com/jax-ml/jax>.
- [31] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. URL <https://arxiv.org/abs/1312.6114>.
- [32] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. URL <https://openreview.net/forum?id=Sy2fzU9gl>.
- [33] Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press, 2019.

A Limitations

To enable both emergence and exact memorization, the LSR energy studied in this work comes with certain limitations. Unlike the LSE energy whose gradient remains nonzero regardless of the query's distance from the stored patterns, the gradient of LSR energy vanishes exactly outside the support of the stored pattern. This makes gradient-based retrieval ineffective when the query lies far from any memorized pattern (though one can easily introduce a query-dependent temperature parameter $\beta(\mathbf{x})$ that dynamically adjusts to ensure the query lies within at least one basin of attraction). Alternatively, one can create a “hybrid” energy function that combines the LSE and LSR energies, taking advantage of LSE's non-zero gradient everywhere outside LSR's support around the stored patterns. Such temperature-tuned DenseAMs and hybrid energies are of independent interest and we leave their systematic study to future work.

Additionally, we want to emphasize that LSR can only create emergent memories precisely at the centroid of overlapping basins between stored patterns. This makes it possible to quickly and exactly retrieve memories (see Algorithm 3), but it also means that the “creativity” of this model is limited to a predictable subset of the convex hull of the stored patterns. The apparent novelty and creativity of Figure 4 is strongly aided by the semantic structure contained the VAEs' latent space.

B Extended Discussion

B.1 On the relationship between emergent memories and hallucination

In everyday use of LLMs, we call it a *hallucination* when the model confidently produces an incorrect fact, especially when we know the model is capable of producing the correct fact in other settings. To formalize this idea into the context of emergent memories, we must imagine a setting where the model has an explicit energy function and a known set of stored “facts.”

This is the setting of Associative Memory as studied in this work, where the outputs of our model are always energy minima (memories). A “hallucination” can occur when two or more stored facts interfere, leading the system to settle into an emergent minimum that looks meaningful (i.e., it lies near the “factual manifold” that generated our stored patterns) but does not correspond to any true, stored fact. Emergent memories are precisely these “interference minima” when they coexist with the original stored patterns. By definition, they are not stored facts, yet they can resemble meaningful combinations of them — making them the memory-system analog of hallucinations.

B.2 On a biological implementation of emergent memories

Dense AMs permit a standard mapping onto “biological” networks by introducing auxiliary hidden neurons following the method of [8]. Thus, the feature neurons in both LSE and LSR models can be augmented with auxiliary hidden neurons, resulting in a model with only pair-wise neuron interactions and bipartite connectivity between feature and hidden layers. This augmented model is more biological compared to the model considered in the main paper. The proposed model defined by the energy eq. (3) and the corresponding update rule eq. (7) can be obtained by integrating out these auxiliary hidden neurons, which means LSR energy will still have the same emergent behavior. The augmented model will still have some un-biological aspects because of the weight symmetry between forward and backward projections, which is necessary to ensure that the network has a global energy function.

C On Kernels

We show different kernels that are typically used for KDE and their efficiency relative to the Epanechnikov kernel in fig. 5. See the explanation on optimal kernel density estimation in section 2 for more details.

Though our main submission focuses on the Epanechnikov kernel, we find that the phenomenon of emergence (Definitions 2 and 3) is not limited to the Epanechnikov kernel. We state our findings for other popular kernels below and graphically in Figure 6. In summary, all compact support kernels are capable of producing emergent memories *except* for the TriWeight kernel (meanwhile, the uniform kernel is impractical for associative memory). Details below:

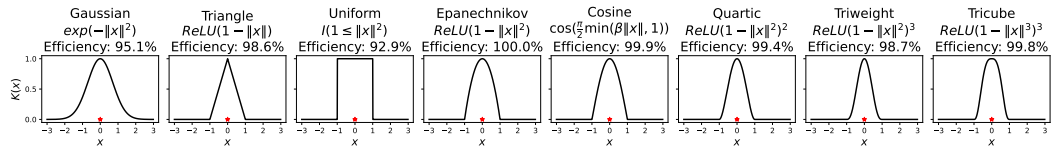


Figure 5: Different kernels used in KDE with their expression and KDE efficiency relative to the Epanechnikov kernel (*higher is better*, see text for details). The center of each kernel is marked with a red $\star$. To highlight the shape of the kernel, we have removed any scaling in the kernel expression. Note that all above kernels except Gaussian have finite support. The Epanechnikov kernel has the highest efficiency (100%). While the Gaussian kernel is extremely popular, and it is more efficient (95.1%) than the Uniform kernel (92.9%), there are various other kernels with better efficiency.

1. **Triangle kernel** — The triangle kernel exhibits very interesting emergence behavior. A perfectly *flat* energy manifold is formed where two or more basins overlap. If the energy of this manifold is lower than the energy of the stored patterns, the stored patterns “merge” and are no longer retrievable. If the energy of the manifold is higher, both original patterns are preserved and we observe a local *emergent manifold* of memory.
2. **Uniform kernel** — A uniform kernel produces an energy landscape that is impractical for associative memory because the gradient is zero everywhere except at the discontinuities. Emergent minima exist according to Definition 3 and have exclusively lower energy than the stored patterns.
3. **Triweight kernel** — We were not able to find emergence at any temperature with the triweight kernel, despite its compact support. Indeed, looking at the $\beta = 0.19$ row of Figure 6, we see that the transition of two basins merging results in a single, almost flat minimum. This phenomenon of compact support without emergence (or perhaps, emergence that is limited to an extremely narrow range of β that we were not able to find) requires further investigation.
4. **Quartic kernel** — The quartic kernel produces smoother energy landscapes than the Epanechnikov does, and emergence appears within a very narrow range of β . Unlike the Epanechnikov kernel, overlapping basins do not guarantee emergent minima, and the emergent minimum is unlikely to have lower energy than the stored patterns.
5. **Tricube kernel** — The tricube kernel behaves like the Quartic kernel, but emergent memories are likely to have lower energy than the stored patterns. The Tricube kernel has an interesting property where local minima flatten right before basins merge. The resulting energy landscape is smoother than that of the Epanechnikov kernel.
6. **Cosine kernel** — The cosine kernel looks remarkably like the Epanechnikov kernel, and its emergence properties are almost identical to that of LSR. However, it appears that Cosine-emergent memories have slightly higher energy than their Epanechnikov counterpart.

D Experimental Details

D.1 Reproducibility & Technical Resources

The codebase is published in a [GitHub repository](#) and contains necessary instructions to setup the environment and to recreate all results, down to the same seed used for training and sampling. Experiments use both PyTorch [29] and JAX [30]. Each experiment was run on a system with access to 8xA40 GPUs each with 40GB of memory. The log-likelihood experimental sweep in appendix D.3 took ~ 12 hours using an optimized scheduler across 45 available CPU cores (memory turned out to be the bottle neck and the CPUs had access to much more RAM than the GPU). Searching for all novel minima at different β and M in appendix D.2 took ~ 48 hours using an optimized scheduler across all 8 GPUs. Training the VAE in appendix D.4 took < 30 min on a single GPU; enumerating all local minima using the efficient algorithm 2 and algorithm 3 took < 15 min.

D.2 Details: Quantifying Novel Minima

In this experiment we tested across a geometrically spaced range of $\beta \in [2d^{-1}, 2r_{\min}^{-2}]$, where $r_{\min} := \min_{\mu \neq \nu} \|\xi_{\mu} - \xi_{\nu}\|$ is the minimum pairwise distance between any two stored patterns in the current subset $\mathcal{K} \subseteq \llbracket M \rrbracket$. At the largest β , the support regions of the stored patterns are disjoint and the only memories are the M stored patterns themselves; this configuration has a very small support region (shown as the **shaded green** curve in fig. 3, which is computed by monte carlo sampling $1e6$

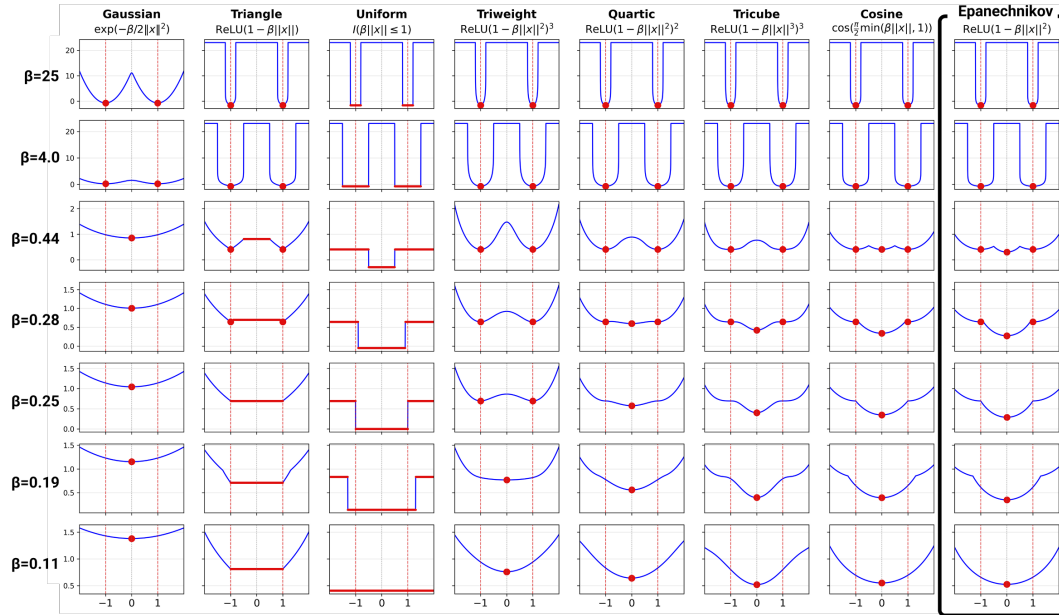


Figure 6: Comparing emergence across different choices of kernel in the DenseAM energy function. Emergent memories are highlighted in red, where manifolds are shown as a flat line and single points as larger dots. Interestingly, all compact kernels exhibit some form of emergence *except* the TriWeight kernel.

points on the unit hypercube and computing the fraction of energies that are finite at $\epsilon = 0$) as a fraction of the unit hypercube. At the smallest tested β , only a single energy minimum is induced at the centroid of all stored patterns with a region of support covering the whole unit hypercube. At the largest tested β , all original memories are recoverable and there are no spurious memories.

D.3 Details: Generative quality of memories

D.3.1 Additional experiments

We ran the same experiment in section 4.2 under varying dimensions $d = [8, 16]$ and number of mixtures $k = [5, 10]$, averaging the results of each run across 5 different random seeds. The results for each of these experiments is shown below in figs. 7 and 8, where fig. 8 (left) is the same as reported in fig. 3 (right) in the main paper.

D.3.2 Aligning β 's across random seeds

We used the same setup for choosing an interesting range of β as we did in appendix D.2. However, over random seeds the value for $r_{\min}$ can vary since it is dependent on the random choice of stored patterns. This makes it difficult to plot error bars over an individual β across seeds. To fix this, we use the fact that each experiment has the same number of geometrically spaced β 's that start from the same β_{low} and compute statistics averaged across the β 's that share an index. The x -axis then represents the β value for each index averaged over seeds.

D.3.3 Determining the uniqueness of minima

To sample the LSE minima, 500 initial points are uniformly sampled from the support boundary around each memory and gradient descent eq. (6) is performed for 13000 steps at a cosine-decayed learning rate α from $0.01 \rightarrow 0.0001$. However, even after this descent process, there are variations in the retrieved memories due to discrete step size α and floating point precision requiring us to be careful when deciding if two samples are distinct. Generally, memory retrieval is said to converge when $\|\nabla_{\mathbf{x}} E(\mathbf{x})\| < \epsilon$ for some small $\epsilon > 0$, or when the number of iterations T exceeds some threshold at small α . Because we know the β of the LSE energy, by the properties of the Gaussian kernel we know that two basins merge when their means are within two standard deviations of each

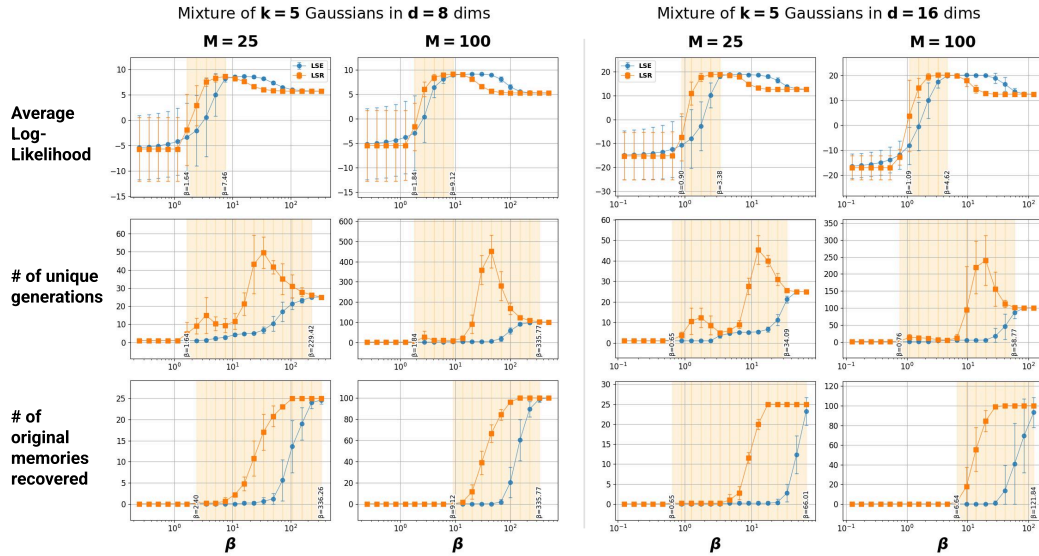


Figure 7: Comparing $d = 8$ and $d = 16$ for $k = 5$ mixture of gaussians at number of stored patterns $M = 10$ and $M = 100$. Error bars are computed by averaging the results of 5 different random seeds. Regions of β where LSR outperforms LSE on a particular metric (on average) are shaded orange.

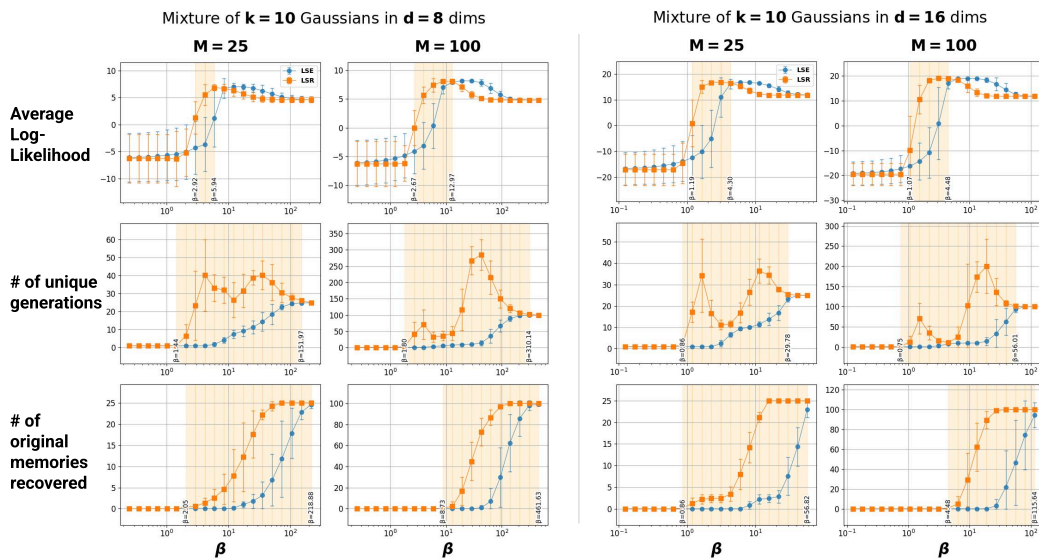


Figure 8: Comparing $d = 8$ and $d = 16$ for $k = 10$ mixture of gaussians at number of stored patterns $M = 10$ and $M = 100$. Error bars are computed by averaging the results of 5 different random seeds. Regions of β where LSR outperforms LSE on a particular metric (on average) are shaded orange.

other. Thus, we can say that two distinct samples are generated by the same memory if they are within $\frac{2}{\sqrt{\beta}}$ of each other.

When counting the uniqueness of the samples from the LSR energy, we perform the following trick to exactly compute the fixed points of the dynamics. We first compute our “best guess” for the fixed point by performing standard gradient descent according to eq. (6) for T steps, at which point $\mathbf{z} := \mathbf{x}^{(T)}$ is close (but not exactly equal) to the fixed point $\mathbf{x}^*$. We then pass $\mathbf{z}$ to algorithm 1 to compute the fixed point exactly. With a good initial guess $\mathbf{z}$, algorithm 1 converges after a single iteration.

Algorithm 1: Fixed Point Computation for the LSR Memory Retrieval

Input: Initial guess $\mathbf{z}$, stored patterns $\{\xi_\mu\}_{\mu=1}^M$, inverse temperature β

Output: Fixed point $\mathbf{z}^*$

Initialize previous point $\mathbf{z}_{\text{prev}} \leftarrow \mathbf{z} + \infty$

while $\mathbf{z}_{\text{prev}} \neq \mathbf{z}$ **do**

$\mathbf{z}_{\text{prev}} \leftarrow \mathbf{z}$

 Compute supports near $\mathbf{z}$ $S(\mathbf{z}) \leftarrow \{\xi_\mu : \|\mathbf{z} - \xi_\mu\| \leq \sqrt{\frac{2}{\beta}}\}$

 Update to mean of support centroids $\mathbf{z} \leftarrow \frac{1}{|S(\mathbf{z})|} \sum_{\xi_\mu \in S(\mathbf{z})} \xi_\mu$

end

return $\mathbf{z}$

Finally, we choose to sample points near the support boundary of each stored pattern because this maximizes the probability that we will end up in a spurious minimum. The size of spurious basins in high dimension can be very small, and the probability of landing in them decreases rapidly with increasing β (see the region of support plot in fig. 3).

D.4 Details: Qualitative reconstructions

The experiments in fig. 4 are conducted in two steps. First, we **design an energy** for each dataset. Then, we **discover all memories** for each system.

Designing the energy The DenseAM energies studied in this work are described by a matrix of stored patterns Ξ and an inverse temperature hparam β .

Choosing Ξ . For MNIST, Ξ is obtained as follows: 24 random images from the MNIST training set are normalized to be $[0, 1]$, rasterized into a 784-dim vector, and projected into a 10-dim latent space of a β -VAE trained according to the methods laid out below. These latents become our stored patterns $\Xi_{\text{MNIST}} \in [0, 1]^{24 \times 768}$ that we use to parameterize both the LSR energy and the LSE energy. The procedure for TinyImagenet [27] is similar. We randomly select 40 images from the dataset, each of shape $(C, H, W) = (3, 64, 64)$. These samples are passed through a small pretrained VAE called TAESD [28] to produce latents that are of shape $(4, 8, 8)$ which are then rasterized into vectors of shape (256,). Thus, our stored pattern matrix for TinyImagenet is $\Xi_{\text{TinyImagenet}} \in \mathbb{R}^{40 \times 256}$.

Choosing β Tuning β for the LSR energy is challenging. When β is too small, each stored pattern interacts with all other stored patterns to induce one minimum at the centroid and several minima far from the data distribution; when β is too large, each stored pattern will interact with no other stored patterns and will induce only a single minimum at itself. Neither of these regimes are interesting. For large ranges of β between these limits, the combinatorial search space of possible memories is computationally prohibitive. For this reason, we use a binary search algorithm to choose a β that, on average, causes each stored pattern to interact with approximately 4 other stored patterns (in the 10-dim MNIST case) and 2 other stored patterns in the 256-dim TinyImagenet case. This β encourages the LSR energy to exhibit emergence where it is computationally feasible to enumerate all memories, and we use it for both the LSR and LSE energy experiments. In pseudocode, the search algorithm is given by algorithm 2

Training the β -VAE for MNIST. MNIST images are encoded by a β -VAE [31, 32] with a latent dimension of 10. The VAE takes as input the 784-dim rasterized MNIST images. The VAE’s encoder and decoder are two layer MLPs configured with LeakyReLU and BatchNorm activations, with a hidden dimension size of 512. Training proceeded for 50 epochs using a learning rate of 1e-3, the Adam optimizer, and a minibatch size of 128. The β of the β -VAE (distinct from the inverse temperature β used by the LSR energy) is set to 4.

Discovering all memories Once we have chosen a β that results in a feasible number of basin interactions K (where the combinatorial search space is tractable), all memories for LSR are discovered by filtering the set of “memory candidates” — the set of centroids formed by at overlapping basins around each stored patterns — to those whose energy gradient is zero. This method is explicitly described in algorithm 3, which iterates through each stored patterns ξ_μ and only searches for emergent memories formed by near-enough stored patterns.

Algorithm 2: Binary Search over β to achieve desired memory interactions

Input: Desired avg. interactions per pattern K , stored patterns $\{\xi_\mu\}_{\mu=1}^M$, max iterations $n_{\max}$ **Output:** Optimal β^* achieving number of basin interactions K **Compute**

Distance matrix $D_{\mu\nu} \leftarrow \{\|\xi_\mu - \xi_\nu\|\}$
Binary bounds $(r_{\min}, r_{\max}) \leftarrow (0.5 \min(D), 4 \max(D))$
Initial basin radius $r \leftarrow \text{mean}(r_{\min}, r_{\max})$

end $n_{\text{iter}} \leftarrow 0$ **repeat**

Compute avg. number of interacting basins per memory

 $K' \leftarrow \text{mean}_\mu \left(\sum_{\nu=1}^M \mathbb{I}[D_{\mu\nu} \leq 2r] \right)$

Update binary search conditions

 $r_{\min} \leftarrow r$ if $K' < K$ $r_{\max} \leftarrow r$ if $K' > K$ $r \leftarrow \text{mean}(r_{\min}, r_{\max})$ $n_{\text{iter}} \leftarrow n_{\text{iter}} + 1$ **until** $K' = K$ or $n_{\text{iter}} \geq n_{\max}$ $\beta^* \leftarrow 2/r^2$ **return** β^*

Algorithm 3: Discover local minima of the LSR energy at a specific β .

Input: Stored patterns $\{\xi_\mu\}_{\mu=1}^M$, inverse temperature β , gradient norm threshold δ near 0**Output:** Set of LSR memories $\mathcal{X}^*$.**Compute**Distance matrix $D_{\mu\nu} \leftarrow \{\|\xi_\mu - \xi_\nu\|\}$ Basin radius $r \leftarrow \sqrt{2/\beta}$ **end**Initialize set of local minima $\mathcal{X}^* \leftarrow \emptyset$ **for** $\mu \in [M]$ **do**Compute set of interacting neighbors $\mathbf{X}_\mu \leftarrow \{\xi_\nu : D_{\mu\nu} \leq 2r\}$ Compute the set of all non-empty subsets $\mathcal{C} \leftarrow \{S \subseteq \mathbf{X}_\mu : \text{size}(S) > 0\}$ **for** $S \in \mathcal{C}$ **do**Compute centroid of neighbors $\bar{\mathbf{x}}_S \leftarrow \text{mean}(S)$ Compute $\bar{\mathbf{x}}_S$ neighbors $T_S \leftarrow \{\xi_\nu : \|\xi_\nu - \bar{\mathbf{x}}_S\| \leq r\}$ **if** $\|\nabla E_{\text{LSR}}(\bar{\mathbf{x}}_S)\| < \delta$ **&** $E_{\text{LSR}}(\bar{\mathbf{x}}_S) < \infty$ **then**
Update set of local minima $\mathcal{X}^* \leftarrow \mathcal{X}^* \cup \{\bar{\mathbf{x}}_S\}$ **end****end****end****return** $\mathcal{X}^*$

All LSE memories are discovered via gradient descent. We initialize queries $\mathbf{x}_\mu = \xi_\mu$ and perform gradient descent according to eq. (6) until convergence (for this experiment, we iterated for 20k steps at small step-size $\alpha = 0.002$). The retrieved fixed points are the complete set of LSE memories.

D.5 Additional experiment: Pixel-space emergence

When the stored patterns live in a semantically structured latent space, as is done in fig. 4, the energy landscape inherits the structure such that centroids of subsets of stored patterns appear semantically novel. However, the latent space visually obscures the mechanistic simplicity of emergent memories. Thus, we repeat the experiment in pixel space to reinforce how emergent memories work.

We store 8 randomly selected MNIST images as rasterized pixels (normalized between 0 and 1) into the LSR energy. The resulting emergent minima and stored patterns are shown in Figure 9, where the β is chosen to balance the number of emergent minima and the retrievability of the stored patterns (*global emergence* Definition 2).

D.6 Additional experiment: Scaling number of stored patterns

The LSR energy function is simple and its properties hold across any scale of stored patterns. To show this, we store all 60,000 MNIST training images into the LSR energy and select a β for which $\sim 50\%$ of the stored patterns are still retrievable. We select a “seed” image at random and randomly select 15 images from all images whose basins interact with the seed image at the chosen β . This example is shown in Figure 10.

E Proofs

E.1 Proof of Theorem 1

For any $\mathbf{x} \in \mathcal{X}$, let $B(\mathbf{x}) = \{\mu \in \llbracket M \rrbracket : \|\mathbf{x} - \xi_\mu\|^2 \leq 2/\beta\}$. Then the gradient of the LSR energy in eq. (3) is given by

$$\nabla_{\mathbf{x}} E^{\text{LSR}}(\mathbf{x}) = \sum_{\mu=1}^M \frac{(\mathbf{x} - \xi_\mu) \mathbb{1}[\|\mathbf{x} - \xi_\mu\|^2 \leq \frac{2}{\beta}]}{\epsilon + \left[\sum_{\nu=1}^M \text{ReLU} \left(1 - \frac{\beta}{2} \|\mathbf{x} - \xi_\nu\|^2 \right) \right]} \quad (7)$$

$$= \frac{\sum_{\mu \in B(\mathbf{x})} (\mathbf{x} - \xi_\mu)}{\epsilon + \left[\sum_{\nu \in B(\mathbf{x})} \text{ReLU} \left(1 - \frac{\beta}{2} \|\mathbf{x} - \xi_\nu\|^2 \right) \right]} \quad (8)$$

With $\beta = 2/(r - \Delta)^2$, $B(\mathbf{x}) = \{\mu \in \llbracket M \rrbracket : \|\mathbf{x} - \xi_\mu\| \leq (r - \Delta)\}$. For any $\mathbf{x} \in S_\mu(\Delta)$, $B(\mathbf{x}) = \{\mu\}$. Thus the LSR energy gradient simplifies to

$$\forall \mathbf{x} \in S_\mu(\Delta), \quad \nabla_{\mathbf{x}} E^{\text{LSR}}(\mathbf{x}) = \frac{(\mathbf{x} - \xi_\mu)}{\epsilon + \text{ReLU} \left(1 - \frac{\beta}{2} \|\mathbf{x} - \xi_\mu\|^2 \right)}, \quad (9)$$

which is exactly zero at $\mathbf{x} = \xi_\mu$, thus giving us the retrieval of the μ^{th} memory via energy gradient descent.

Furthermore, again for $\mathbf{x} \in S_\mu(\Delta)$ with a energy gradient descent learning rate set to $\eta \leftarrow \epsilon + \text{ReLU} \left(1 - \beta/2 \|\mathbf{x} - \xi_\mu\|^2 \right)$, the update is exactly $\eta \nabla_{\mathbf{x}} E^{\text{LSR}}(\mathbf{x}) = (\mathbf{x} - \xi_\mu)$. Thus a single step gradient descent update to $\mathbf{x}$ with $\mathbf{x} - \eta \nabla_{\mathbf{x}} E^{\text{LSR}}(\mathbf{x}) = \mathbf{x} - (\mathbf{x} - \xi_\mu) = \xi_\mu$ results in the retrieval of the μ^{th} memory.

E.2 Proof of Proposition 2

Proof. (Proposition 2.)

For any $\mathbf{x} \in \mathcal{X}$, given the definition of $B(\mathbf{x})$, recall that the gradient of the LSR energy is given in eq. (7). This gradient is zero when

$$\mathbf{x} = \frac{1}{|B(\mathbf{x})|} \sum_{\mu \in B(\mathbf{x})} \xi_\mu,$$

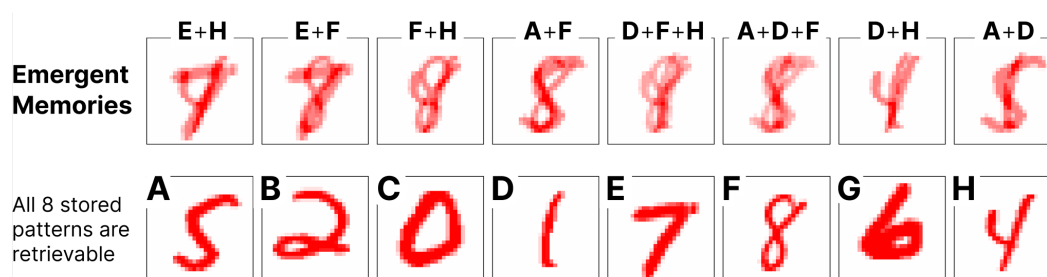


Figure 9: **Emergent memories** are centroids of subsets of stored patterns, shown clearly when 8 stored images are visualized in pixel space alongside their induced emergent memories. Stored patterns (bottom, indexed A-H) merge to form emergent memories (top, labeled by the stored patterns that merged to form the emergent memory). β is chosen such that the number of emergent memories is approximately the same as the number of stored patterns.

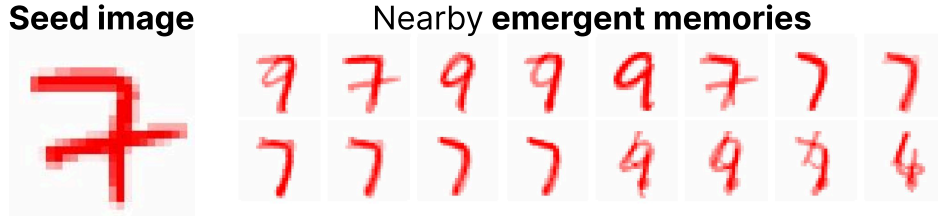


Figure 10: Sampling emergent memories near a **seed image** when all 60k MNIST training images are stored into the LSR energy. Left: Random seed image, which is a preserved memory. Right: 16 randomly sampled emergent memories formed by the seed image's interactions with other stored patterns (at $\beta = 0.11$). Because the seed image interacts with the basins of $\sim 7.3k$ other stored patterns, these emergent memories represent a *tiny sample* of the total emergent memories near the seed image.

the geometric mean of the memories corresponding to the set $B(\mathbf{x})$. Moreover, by standard algebraic computations, we have

$$\nabla_{\mathbf{x}}^2 E_{LSR}(\mathbf{x}^*) = \frac{|B(\mathbf{x}^*)|}{\epsilon + \left[\sum_{\nu=1}^M \text{ReLU} \left(1 - \frac{\beta}{2} \|\mathbf{x}^* - \xi_{\nu}\|^2 \right) \right]} I_d \succ 0.$$

□

E.3 Proof of Theorem 2

Note that once we prove properties (a) and (b), the ε -global emergence in Definition 2 follows immediately.

We first focus on (a). Note that for any fixed $\mathbf{x} \in \mathcal{X}$ and $\varepsilon > 0$, $\text{vol}(B(\mathbf{x}, \varepsilon) \cap \mathcal{X})$ is at most $\text{vol}(B(\mathbf{x}, \varepsilon)) = V_d \varepsilon^d$, where V_d is the volume of the unit ball in $\mathbb{R}^d$. Thus, we obtain: for any pair (ξ_{μ}, ξ_{ν}) with $\mu \neq \nu$, we have:

$$\Pr \left[\|\xi_{\mu} - \xi_{\nu}\| < \varepsilon \right] \leq \int_{\mathcal{X}} \frac{\text{vol}(B(x, \varepsilon) \cap \mathcal{X})}{\text{vol}(\mathcal{X})} dx \leq V_d V^{-1} \varepsilon^d.$$

Then, we have:

$$\Pr \left[\min_{1 \leq \mu \neq \nu \leq M} \|\xi_{\mu} - \xi_{\nu}\| < \varepsilon \right] \leq M^2 V_d V^{-1} \varepsilon^d. \quad (10)$$

Now, we let $\varepsilon = (V_d/V)^{-1/d} e^{-2\alpha}$ for a positive α . Thus, with probability at least $1 - M^2 e^{-2\alpha d}$, the minimum pairwise Euclidean distance $r = \min_{1 \leq i \neq j \leq M} \|\xi_{\mu} - \xi_{\nu}\| \geq (V_d/V)^{-1/d} e^{-2\alpha}$. Then, with

$$0 < \beta = \frac{2}{(r - \Delta)^2} \leq \frac{2}{((V_d/V)^{-1/d} e^{-2\alpha} - \Delta)^2},$$

we are able to retrieve any memory ξ_{μ} with an $\mathbf{x} \in S_{\mu}(\Delta)$, for $\Delta \in (0, (V_d/V)^{-1/d} e^{-2\alpha})$. Moreover, the size of M is given by the success probability:

$$\delta = 1 - M^2 e^{-2\alpha d}.$$

This gives

$$M = \Theta \left(\sqrt{1 - \delta} e^{\alpha d} \right).$$

Next, we focus on (b). Recall the gradient is given by

$$\nabla_{\mathbf{x}} E_{LSR}(\mathbf{x}) = \frac{\sum_{\mu \in B(\mathbf{x})} (\mathbf{x} - \xi_{\mu})}{\epsilon + \left[\sum_{\mu \in B(\mathbf{x})} \text{ReLU} \left(1 - \frac{\beta}{2} \|\mathbf{x} - \xi_{\mu}\|^2 \right) \right]},$$

where it is determined by the set

$$B(\mathbf{x}) = \left\{ \mu : \|\mathbf{x} - \boldsymbol{\xi}_\mu\| < \sqrt{\frac{2}{\beta}} \right\}.$$

Let $S_{\mathbf{x}^*}(\Delta) = \{\mathbf{x} : \|\mathbf{x} - \mathbf{x}^*\| \leq \Delta\}$ be a basin around an emergent memory $\mathbf{x}^*$. Note that by Cauchy–Schwarz inequality, the change within the basin of the squared distance to any $\boldsymbol{\xi}_\mu$ is at most

$$|\|\mathbf{x} - \boldsymbol{\xi}_\mu\|^2 - \|\mathbf{x}^* - \boldsymbol{\xi}_\mu\|^2| \leq 2\|\mathbf{x}^* - \boldsymbol{\xi}_\mu\|\Delta + \Delta^2.$$

We now for activated memories $\mu \in B(\mathbf{x}^*)$ let

$$\delta_{\min}(\mathbf{x}^*) = \min_{\mu \in B(\mathbf{x}^*)} \left(\frac{2}{\beta} - \|\mathbf{x}^* - \boldsymbol{\xi}_\mu\|^2 \right) > 0.$$

This is the margin to the boundary of activation. Moreover, for inactivated memories $\nu \notin B(\mathbf{x}^*)$, let

$$\gamma_{\min}(\mathbf{x}^*) = \min_{\nu \notin B(\mathbf{x}^*)} \left(\|\mathbf{x}^* - \boldsymbol{\xi}_\nu\|^2 - \frac{2}{\beta} \right) > 0.$$

This is the margin to being activated. Given $\{\boldsymbol{\xi}_\mu\}_{\mu=1}^M$ has a density, the probability of either of them being exact zero is zero.

Finally, we determine the radius Δ . To ensure that for any $\mathbf{x} \in S_{\mathbf{x}^*}(\Delta)$, $\nabla_{\mathbf{x}} E_{\text{LSR}}(\mathbf{x}) = \nabla_{\mathbf{x}} E_{\text{LSR}}(\mathbf{x}^*)$, it suffices to make sure $B(\mathbf{x}) = B(\mathbf{x}^*)$ for any $\mathbf{x} \in S_{\mathbf{x}^*}(\Delta)$. To this end, we can pick Δ such that

$$2D_{\max}(\mathbf{x}^*)\Delta + \Delta^2 < \min\{\delta_{\min}(\mathbf{x}^*), \gamma_{\min}(\mathbf{x}^*)\}, \text{ where } D_{\max}(\mathbf{x}^*) = \max_{\mu} \|\mathbf{x}^* - \boldsymbol{\xi}_\mu\|.$$

This gives

$$\Delta < \sqrt{D_{\max}^2(\mathbf{x}^*) + \min\{\delta_{\min}(\mathbf{x}^*), \gamma_{\min}(\mathbf{x}^*)\}} - D_{\max}(\mathbf{x}^*) =: r^*.$$

To prove the final claim, without loss of generality, we consider the case when $\mathcal{X} = [0, 1]^d$ with $V = 1$. Since $\{\boldsymbol{\xi}_\mu\}_{\mu=1}^M$ are i.i.d. uniform, $\mathbb{1}\left[\boldsymbol{\xi}_\mu \in \left\{\|\mathbf{x} - \boldsymbol{\xi}_\mu\| \leq \sqrt{\frac{2}{\beta}}\right\}\right]$ are i.i.d. indicators over $\mu = 1, \dots, M$, by the concentration of sub-Gaussian random variables [33, Proposition 2.5], we have: with probability at least $1 - M^{-1}$,

$$\begin{aligned} \lambda &= |B(\mathbf{x})| \\ &\in \left[MV_d \left(\frac{2}{\beta} \right)^{\frac{d}{2}} - 5\sqrt{MV_d \left(\frac{2}{\beta} \right)^{\frac{d}{2}} \log M}, MV_d \left(\frac{2}{\beta} \right)^{\frac{d}{2}} + 5\sqrt{MV_d \left(\frac{2}{\beta} \right)^{\frac{d}{2}} \log M} \right]. \end{aligned} \quad (11)$$

This implies $\lambda = \Theta\left(MV_d \left(\frac{2}{\beta}\right)^{\frac{d}{2}}\right)$. Note that according to Proposition 2, for each $|B(\mathbf{x})| \in (1, M]$, there is a corresponding new emergent memory as the average of all stored patterns over $\mu \in B(\mathbf{x})$. Then, the number of new emergent memories is bounded as

$$Q_{\beta, \text{LSR}} \leq \binom{M}{\lambda}, \text{ where } \lambda = \Theta\left(MV_d \left(\frac{2}{\beta}\right)^{\frac{d}{2}}\right).$$

By Stirling's formula, we have:

$$Q_{\beta, \text{LSR}} \leq \left(\frac{eM}{\lambda} \right)^\lambda.$$

Thus, it holds that

$$\log Q_{\beta, \text{LSR}} \leq \lambda \log \left(\frac{eM}{\lambda} \right) = O\left(MV_d \left(\frac{2}{\beta} \right)^{\frac{d}{2}} \log \left(\frac{e}{V_d} \left(\frac{\beta}{2} \right)^{\frac{d}{2}} \right) \right).$$

This implies

$$Q_{\beta, \text{LSR}} = O\left(\exp \left(MV_d \left(\frac{2}{\beta} \right)^{\frac{d}{2}} \log \left(\frac{e}{V_d} \left(\frac{\beta}{2} \right)^{\frac{d}{2}} \right) \right) \right).$$

E.4 Proof of Proposition 3

Without loss of generality, we consider the case when $\mathcal{X} = [0, 1]^d$ with $V = 1$. Now, we divide $[0, 1]^d$ into an equally spaced grid of size M , i.e., we divide each dimension $[0, 1]$ into $[mM^{-1/d}, (m+1)M^{-1/d}]$ for $m = 0, \dots, M-1$. Under our grid setting, $\{\xi_\mu\}_{\mu=1}^M$ lie in such a grid of equal size.

Now, note that

$$B(\mathbf{x}) = \left\{ \mu : \|\mathbf{x} - \xi_\mu\| < \sqrt{\frac{2}{\beta}} \right\}.$$

Suppose the ball in definition of $B(\mathbf{x})$ contains λ of those equally spaced stored patterns such that $\lambda = |B(\mathbf{x})| = \lambda \in (1, M]$. We want to put a hypercube in the grid such that it contains exact λ out of M equally spaced stored patterns in the grid. Here, for simplicity, it is up to a constant depending on d if we think of the ball as a hypercube in $\mathbb{R}^d$ containing the points in the grid in this case. Hence, in each dimension $1 \leq i \leq d$, we put an interval containing of order $\lambda^{1/d}$ points in the grid inside the entire interval containing of order $M^{1/d}$ points in the grid. It thus gives the number of choice for each dimension $1 \leq i \leq d$ as:

$$M^{1/d} - \lambda^{1/d} - 1$$

Hence, counting all choices over d dimensions, we obtain the number of such $B(\mathbf{x})$ as

$$\left(M^{1/d} - \lambda^{1/d} - 1 \right)^d.$$

Moreover, due to the equal spacing of the grid, we have:

$$\lambda = \Theta \left(M \left(\frac{8}{\beta} \right)^{\frac{d}{2}} \right). \quad (12)$$

Hence, combining all bounds above, we obtain: the number of emergent memories under uniform sampling is of order

$$\Theta \left(\left(M^{1/d} - \lambda^{1/d} + 1 \right)^d \right),$$

where λ satisfies equation 12 and $1 < \lambda \leq M$.

E.5 Proof of Proposition 1

Recall the LSE energy:

$$E^{\text{LSE}}(\mathbf{x}) = -\frac{1}{\beta} \log \sum_{\mu=1}^M e^{-\frac{\beta}{2} \|\mathbf{x} - \xi_\mu\|^2}.$$

Thus, the gradient is

$$\nabla_{\mathbf{x}} E^{\text{LSE}}(\mathbf{x}) = \mathbf{x} - \sum_{\mu=1}^M p_\mu(\mathbf{x}) \xi_\mu, \quad p_\mu(\mathbf{x}) := \frac{e^{-\frac{\beta}{2} \|\mathbf{x} - \xi_\mu\|^2}}{\sum_{\mu=1}^M e^{-\frac{\beta}{2} \|\mathbf{x} - \xi_\mu\|^2}}.$$

Suppose ξ_ν for some ν is the zero of this gradient, it yields that

$$(1 - p_\nu(\xi_\nu)) \xi_\nu = \sum_{\mu \neq \nu} p_\mu(\xi_\mu) \xi_\mu.$$

This is a contradiction of the assumption that $\{\xi_\mu\}_{\mu=1}^M$ are linearly independent.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: The abstract and introduction make claims that are clearly backed by theorems and experiments in the paper.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: Limitations are discussed in the limitations section of the supplementary.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Each theorem provides the full set of assumptions and a complete (and correct) proof in the Proofs section of the Appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.

- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: The Reproducibility section of the Appendix provides information about the hardware and a description of the code used to create the results reported in the main paper. For important algorithms, we include succinct and clear pseudocode in the Appendix. For review, the anonymized code is submitted alongside the full supplementary as a zip file, while the code is made publicly available on [GitHub](#) for the final version.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [\[Yes\]](#)

Justification: We link to a [repository](#) that contains code to completely reproduce the results reported in the main paper.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).

- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [\[Yes\]](#)

Justification: The attached code specifies all the training and test details necessary to reproduce the results. Important hyperparameter configurations for the experiment are provided in the paper and appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [\[Yes\]](#)

Justification: Experiments where the results are strongly affected by random seeds are repeated over multiple seeds. The results are averaged and reported alongside errorbars.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [\[Yes\]](#)

Justification: The Reproducibility section of the Appendix provides information about the hardware and resources.

Guidelines:

- The answer NA means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines?>

Answer: [Yes]

Justification: In publishing this paper we have adhered in full to the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: This work is primarily theoretical and considers only toy examples (MNIST, synthetic data, TinyImagenet). It will have no societal impact.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: There is no risk of misuse of the contributions of this paper.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.
12. **Licenses for existing assets**
 Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?
 Answer: [Yes]
 Justification: All code and data is properly credited and the licenses properly respected.
 Guidelines:
- The answer NA means that the paper does not use existing assets.
 - The authors should cite the original paper that produced the code package or dataset.
 - The authors should state which version of the asset is used and, if possible, include a URL.
 - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
 - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
 - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
 - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
 - If this information is not available online, the authors are encouraged to reach out to the asset's creators.
13. **New assets**
 Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?
 Answer: [NA]
 Justification: No new assets are introduced in the paper.
 Guidelines:
- The answer NA means that the paper does not release new assets.
 - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
 - The paper should discuss whether and how consent was obtained from people whose asset is used.
 - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.
14. **Crowdsourcing and research with human subjects**
 Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?
 Answer: [NA]
 Justification: The paper does not involve crowdsourcing nor research with human subjects.
 Guidelines:
- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
 - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
 - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.
15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**
 Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?
 Answer: [NA]
 Justification: The paper does not involve research with human subjects.
 Guidelines:
- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The paper does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.