In search of dispersed memories: Generative diffusion models are associative memory networks

Luca Ambrogioni
Radboud University, Donders Institute for Brain, Cognition and Behaviour
luca.ambrogioni@donders.ru.nl

Abstract

Hopfield networks are widely used in neuroscience as simplified theoretical models of biological associative memory. The original Hopfield networks store memories by encoding patterns of binary associations, which result in a synaptic learning mechanism known as Hebbian learning rule. Modern Hopfield networks can achieve exponential capacity scaling by using highly non-linear energy functions. However, the energy function of these newer models cannot be straightforwardly compressed into binary synaptic couplings and it does not directly provide new synaptic learning rules. In this work we show that generative diffusion models can be interpreted as energy-based models and that, when trained on discrete patterns, their energy function is equivalent to that of modern Hopfield networks. This equivalence allows us to interpret the supervised training of diffusion models as a synaptic learning process that encodes the associative dynamics of a modern Hopfield network in the weight structure of a deep neural network. Accordingly, in our experiments we show that the storage capacity of a continuous modern Hopfield network is identical to the capacity of a diffusion model. Our results establish a strong link between generative modeling and the theoretical neuroscience of memory, which provide a powerful computational foundation for the reconstructive theory of memory, where creative generation and memory recall can be seen as parts of a unified continuum.

1 Introduction

In psychology and neuroscience, associative memory is defined as the ability to recall associations between items such as, for example, different sensory patterns (Squire et al., 1993; Mayes et al., 2007). An episodic memory, such as the memory of your first day of primary school, is often conceptualized as a constellation of associations between the desperate sensory and conceptual elements that were "activated" when the memory was first formed (Tulving, 2002). The recall of a memory can then be seen as a form of pattern completion, where the initial (possibly random) activation of a few of the elements (e.g. the smell of a pencil) triggers the re-activation of the other associated elements, leading to the re-activations of the elements forming the original memory (Mace and Clevinger, 2019). In the human brain, associations between two items can be represented by synaptic connections between neurons. The simplest way to learn these binary associations is to strengthen the synaptic coupling between two neurons every time they are simultaneously active. This is known as Hebbian learning and it is still central to our understanding of synaptic plasticity in biological neural networks (Hebb, 2005; Sejnowski and Tesauro, 1989). Hopfield networks have been developed to formalize this form of associative memory in a simplified distributed artificial neural system (Hopfield, 1982). We will denote the activity of the D memory units with a vector $\boldsymbol{x}(t) = (x_1(t), \dots, x_D(t))$. In the original works, these neural activities where assumed to be binary variables ($x_j(t) \in \{-1, 1\}$), respectively denoting states of rest and states of firing. The dynamic of a Hopfield network is regulated by the update equation: $x_j(t + \mathrm{d}t) = \left[\text{sign}\left(W\boldsymbol{x}\right)\right]_j$, where W is a real-valued symmetric matrix of synaptic

Associative Memory & Hopfield Networks in 2023. NeurIPS 2023 workshop.

couplings (weights) with null diagonal ($W_{jj} = 0,\ \forall j$). It can be show that this update rule decreases monotonically the following energy function: $u(\boldsymbol{x}) = \boldsymbol{x}^T W \boldsymbol{x}$, and it will therefore converge to one of its local minima. These local minima can then be used to encode memories, which can be retrieved through the Hopfield dynamics if initialized in an incomplete or perturbed version of the memory state. The simplest way to encode memories in the coupling matrix is to use the Hebb's rule of association, which in its simplest form is $W_{j,k} = \sum_{n=1}^{N} y_j^{(n)} y_k^{(n)}$ where $\{\boldsymbol{y}^{(1)}, \dots, \boldsymbol{y}^n, \dots \boldsymbol{y}^n\}$ is a set of N "experienced" patterns of neural activity. A pattern is considered to be successfully stored if it is a stable fixed-point of the discrete dynamics. If the patterns are random, it can be proven that the storage capacity of Hopfield nets scales as $D/4 \log_2 D$ (Abu-Mostafa and Jacques, 1985). The storage capacity of Hopfield-like networks can be increased by including non-linear mappings $F(\cdot)$ in its energy function (Krotov, 2023; Krotov and Hopfield, 2016; Demircigil et al., 2017). The general form for the energy of a (discrete) modern Hopfield network can be written as $u(\boldsymbol{x}) = h\left(\sum_{n=1}^{N} F\left(\boldsymbol{x}^T \boldsymbol{y}^n\right)\right)$, where $h(\cdot)$ is an arbitrary differentiable and strictly monotonic function, which does not affect the location and stability of the local minima. This expression reduces to the standard Hopfield energy for $F(x) = x^2$ and $h(x) = x$. However, it is possible to achieve much higher theoretical capacity by using more complex functions. For example, a modern Hopfield network with $F(x) = e^x$ can store up to $2^{D/2}$ binary patterns (Demircigil et al., 2017). These associative networks have been recently generalized to have continuous dynamics. If the activation vector is continuous, the energy function needs to include a regularization term to enforce stability. For example, (Ramsauer et al., 2021) proposed the use of an energy function of the form

$$u(\boldsymbol{x}, \beta) = -\beta^{-1} \log\left(\sum_{n=1}^{N} e^{\beta \boldsymbol{x}^T \boldsymbol{y}^n}\right) + \|\boldsymbol{x}\|_2^2/2 \tag{1}$$

where β is a positive-valued parameter. We omitted the terms that are additive constant in $\boldsymbol{x}$ since they do not change the fixed-points and the resulting dynamics. Krotov and Hopfield (2021) showed that this dynamics can be expressed in terms of biologically plausible binary association between latent neurons, in a way that is similar to the architecture of a restricted Boltzmann machine Fischer and Igel (2012). Unfortunately, in these models, the numerical values of the patterns directly determine the synaptic straights between latent and observable neurons. This implies that the patterns need to be stored in memory instead of being converted into distributed synaptic patterns. In this sense, modern Hopfield network offer a model of memory recall but do not provide insight into learning and memory storage in the brain.

Generative diffusion models are defined in terms of a noise-injection process that gradually turns the training data into pure noise (Sohl-Dickstein et al., 2015; S. et al., 2021; Song et al., 2021). To be consistent with the notation used in continuous Hopfield model, we deviate from the diffusion modeling literature by writing this process in reversed time, with the noiseless data corresponding to a final time T. In this notation, the diffusive dynamics being determined by the following backward recursive equation: $\boldsymbol{x}(t - \mathrm{d}t) = \boldsymbol{x}(t) + \sigma\sqrt{\mathrm{d}t}\boldsymbol{\epsilon}_t$, where σ determines the standard deviation of the noise injected at time t and $\boldsymbol{\epsilon}_t$ follows a standard normal distribution. This is known as the variance exploding equation in the generative diffusion modeling literature (S. et al., 2021). For $\mathrm{d}t \to 0$, it possible to show that, given a final density $p_T(x)$, the noise-injection process can be inverted by the forward equation

$$\boldsymbol{x}(t + \mathrm{d}t) = \boldsymbol{x}(t) + \sigma^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}(t))\mathrm{d}t + \sigma\sqrt{\mathrm{d}t}\boldsymbol{\epsilon}_t \ . \tag{2}$$

where $p_T(\boldsymbol{x}(t))$ is the marginal distribution of the noise-injection process at time t. This process can then be used for generative modeling if we choose $p_T(x)$ to be given by a target data source $\phi(\boldsymbol{y})$. This allow us to define a stochastic process that turns an initial noise state into synthetic data. Since the noise-injection process is a simple Brownian motion, we can write down the conditional density exactly as $p(x(t) \mid x(t_0)) = \mathcal{N}(x(t); x(t_0), (t - t_0)\sigma^2)$, this allows us to write the marginal density as follows $p_t(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{y}}\left[\frac{1}{\sqrt{2\pi(T-t)\sigma^2}} e^{-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|_2^2}{2(T-t)\sigma^2}}\right]$. This formula involves an average over the distribution of the data, which is usually not available in a generative modeling task. However, assuming that we can sample from a dataset $\boldsymbol{y}^{(1)}, \dots, \boldsymbol{y}^{(M)}\}$, we can train a feedforward neural network $\boldsymbol{s}(\boldsymbol{x}(t), t; W)$, parameterized by the synaptic weights W, to approximate the score function $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}(t))$. The network can be trained by minimizing a denoising loss (Song et al., 2021).

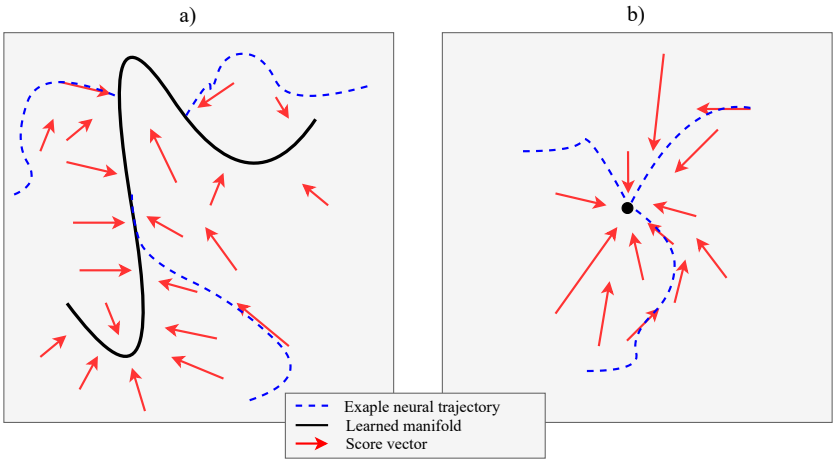

Figure 1: Visualization of the dynamics of a) a model trained as a generative model (semantic memory) and b) a model trained to retain an episodic memory

2 The equivalence between diffusion models and modern Hopfield networks

We can now show that, when used for storing discrete patterns, the dynamics of generative diffusion models minimizes the energy function of continuous modern Hopfield networks. The first step is to formulate the deterministic dynamics of the model as the negative gradient of an energy function: $\boldsymbol{x}(t + \mathrm{d}t) = \boldsymbol{x}(t) - \nabla_{\boldsymbol{x}} u(\boldsymbol{x}, t)\mathrm{d}t + \sigma\sqrt{\mathrm{d}t}\boldsymbol{\epsilon}_t$. As noted in (Raya and Ambrogioni, 2023), the energy function is $u(\boldsymbol{x}, t) = -\sigma^2 \log p_t(\boldsymbol{x}) = -\sigma^2 \log \mathbb{E}_{\boldsymbol{y}}\left[e^{-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|_2^2}{2(T-t)\sigma^2}}\right] + c$, where c does not depend on $\boldsymbol{x}$ and can therefore be omitted without affecting the dynamics. In order to establish a link between the diffusion model and Hopfield networks, we can now assume that the data source is a finite collection of N patterns that we wish to store as memories. This lead to the energy

$$u(\boldsymbol{x}, t) = -\sigma^2 \log\left(\frac{1}{N}\sum_{n=1}^{N} e^{-\frac{\|\boldsymbol{x}-\boldsymbol{y}^n\|_2^2}{2(T-t)\sigma^2}}\right) . \tag{3}$$

If we now assume that the patterns are normalized ($\|\boldsymbol{y}\|_2^2 = 1$), by expanding the square in the exponent and omitting constant additive terms, we obtain

$$u(\boldsymbol{x}, t)/\sigma^2 = -\log\left(\sum_{n=1}^{N} e^{\frac{\boldsymbol{x}^T\boldsymbol{y}^n}{2(T-t)\sigma^2}}\right) + \frac{\|\boldsymbol{x}\|_2^2}{(T-t)^2} . \tag{4}$$

Finally, if we define $\beta(t)^{-1} = (T - t)\sigma^2$ and we multiply both sides by $\beta(t)^{-1}$, we obtain

$$\beta(t)^{-1}u(\boldsymbol{x}, t)/\sigma^2 = -\beta(t)^{-1}\log\left(\sum_{n=1}^{N} e^{\beta(t)\boldsymbol{x}^T\boldsymbol{y}^n}\right) + \frac{\|\boldsymbol{x}\|_2^2}{2} , \tag{5}$$

which for a fixed t is identical to the continuous Hopfield network energy in Eq. 1, and it therefore has the same fixed-point structure at the limit $\beta \to \infty$. Note that the scaling factor $\beta(t)/\sigma^2$ does not change the fixed-points and their stability as it is a positive constant of $\boldsymbol{x}$. The main difference between the two approaches is that, in diffusion models, $\beta(t)$ tend to this divergent limit as part of the denoising dynamics, while the denoising iterations of modern Hopfield networks keep β fixed. However, as shown in our experiments, this does not result in meaningful differences as far as β is large enough. While we derived this result assumes normalized patterns, this is not actually necessary since, for $\beta(t) \to \infty$, we have that $\beta(t)^{-1}u(\boldsymbol{x}, t)/\sigma^2 \sim -\frac{1}{2}\left(\boldsymbol{x}^T\boldsymbol{y}^* + \|\boldsymbol{y}^*\|_2^2 + \|\boldsymbol{x}\|_2^2\right)$ where $\boldsymbol{y}^*$ is the pattern that maximizes the quadratic form $\boldsymbol{x}^T\boldsymbol{y} + \|\boldsymbol{y}\|_2^2$. As shown by this expression, at this limit the norm $\boldsymbol{y}^*$ only adds an irrelevant constant shift in the energy. The use of diffusion models as a framework for reconstructive memory is discussed in Supp.A, while the connections with neuroscience are discussed in Supp.B.

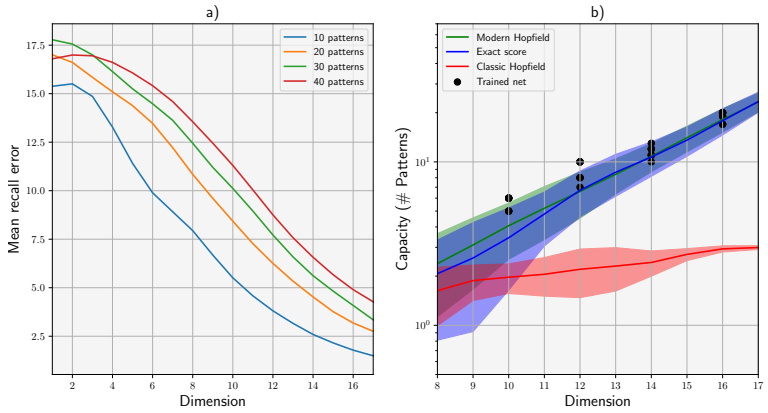

Figure 2: 1) Median error of exact diffusion model as function of the dimensionality. b) Capacity of diffusion models and Hopfield networks in log scale. The shaded area denotes the estimated 95% intervals.

3 Experiments

Denoising task	# Patterns	Diffusion Model	Classic Hopfield	True patterns
	10	0.995	0.732	0.893
	20	0.991	0.704	0.822
	30	0.991	0.715	0.81
Completion task				
	10	0.996	0.741	0.897
	20	0.991	0.707	0.838
	30	0.989	0.700	0.795

Table 1: Pearson correlation between output of modern Hopfield network and other models (plus ground truth pattern) in both denoising and completion experiments.

We tested how the storage capacity of exact diffusion models, trained diffusion models and Hopfield networks scales as a function of the dimensionality of the input patterns. As first analysis we evaluated the Pearson correlation coefficient between the output of modern Hopfield iteration and a) a diffusion model, b) a classical Hopfield network and c) the ground truth pattern. For a given dimensionality d, n binary patterns $\boldsymbol{y}$ were randomly generated and subsequently corrupted with noise using the formula $\tilde{\boldsymbol{y}} = \theta \boldsymbol{y} + \sqrt{1-\theta^2}\epsilon$, where ϵ is a standard Gaussian noise vector. We used a noise level of $\theta = 0.68$. We kept the dimensionality equal to 10 and we evaluated the correlation for 10, 20 and 30 stored patterns. The simulation was repeated 100 times in order to reliably compute the correlations. The modern Hopfield iterations were implemented as specified in (Ramsauer et al., 2021), with $\beta = 5$ and 150 updates. In order to avoid to have to re-train a neural model hundreds of times, we used the exact score formula (see Supp.C. We used a variance-preserving model as they are numerically more stable and more widely used Song et al. (2021). Table 1 shows that the estimated Pearson correlations. As expected from our analysis, the correlation between the modern Hopfield iterations and the diffusion model is extremely close to one. We also performed the same experiments in a completion task, where the patterns were partially zero-masked instead of being corrupted by white noise. The binary masks were sampled randomly from a Bernoulli distribution with $p = 0.5$. Again, the output of the exact diffusion models correlates almost perfectly with the output of the modern Hopfield iterations. Next, we estimated the error and capacity of the models. The corrupted patterns were fed to the algorithms and the results were compared with the original pattern using the Hamming error. The patterns were considered to be correctly recovered if the error was smaller than 3%. Fig. 2a shows the error of an exact diffusion model for different numbers of patterns as function of the dimensionality. For a given noise level and threshold, the capacity was defined as the maximum number of patterns that can on average be recovered. Fig. 2b shows the estimated capacity of the exact diffusion model (blue), modern Hopfield network (green), classical Hopfield network (red) and trained diffusion model (black dots). The details of the experiments are given in Supp. C.

References

Abu-Mostafa, Y. and Jacques, J. S. (1985). Information capacity of the hopfield model. *IEEE Transactions on Information Theory*, 31(4):461–464.

Barron, H. C., Auksztulewicz, R., and Friston, K. (2020). Prediction and memory: A predictive coding account. *Progress in Neurobiology*, 192:101821.

Buhry, L., Azizi, A. H., Cheng, S., et al. (2011). Reactivation, replay, and preplay: how it might all fit together. *Neural Plasticity*.

Demircigil, M., Heusel, J., Löwe, M., Upgang, S., and Vermet, F. (2017). On a model of associative memory with huge storage capacity. *Journal of Statistical Physics*, 168:288–299.

Fischer, A. and Igel, C. (2012). An introduction to restricted boltzmann machines. *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*.

Güçlü, U. and van Gerven, M. A. J. (2015). Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35(27):10005–10014.

Hebb, D. O. (2005). *The organization of behavior: A neuropsychological theory*. Psychology press.

Hemmer, P. and Steyvers, M. (2009). A bayesian account of reconstructive memory. *Topics in Cognitive Science*, 1(1):189–202.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558.

Krotov, D. (2023). A new frontier for hopfield networks. *Nature Reviews Physics*, pages 1–2.

Krotov, D. and Hopfield, J. (2021). Large associative memory problem in neurobiology and machine learning. *International Conference on Learning Representations*.

Krotov, D. and Hopfield, J. J. (2016). Dense associative memory for pattern recognition. *Advances in Neural Information Processing Systems*.

Mace, J. H. and Clevinger, A. M. (2019). The associative nature of episodic memories. *The organization and structure of autobiographical memory*, pages 183–200.

Mayes, A., Montaldi, D., and Migo, E. (2007). Associative memory and the medial temporal lobes. *Trends in cognitive sciences*, 11(3):126–135.

Millidge, B., Seth, A., and Buckley, C. L. (2021). Predictive coding: a theoretical and experimental review. *arXiv preprint arXiv:2107.12979*.

Ramsauer, H., Schäfl, B., Lehner, J., Seidl, P., Widrich, M., Adler, T., Gruber, L., Holzleitner, M., Pavlović, M., Sandve, G. K., et al. (2021). Hopfield networks is all you need. *Internetional Conference on Learning Representations*.

Rao, R. P. N. and Ballard, D. H. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2(1):79–87.

Raya, G. and Ambrogioni, L. (2023). Spontaneous symmetry breaking in generative diffusion models. *arXiv preprint arXiv:2305.19693*.

Roediger, H. L. and McDermott, K. B. (1995). Creating false memories: Remembering words not presented in lists. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 21(4):803.

S., Y., S., J., K., D. P., K., A., E., S., and P., B. (2021). Score-based generative modeling through stochastic differential equations. *International Conference on Learning Representations*.

Salvatori, T., Song, Y., Hong, Y., Sha, L., Frieder, S., Xu, Z., Bogacz, R., and Lukasiewicz, T. (2021). Associative memories via predictive coding. *Advances in Neural Information Processing Systems*, 34:3874–3886.

Sejnowski, T. J. and Tesauro, G. (1989). The hebb rule for synaptic plasticity: algorithms and implementations. In *Neural Models of Plasticity*, pages 94–103. Elsevier.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. *International Conference on Machine Learning*.

Song, J., Meng, C., and Ermon, S. (2021). Denoising diffusion implicit models. *Internetional Conference on Learning Representations*.

Spratling, M. W. (2017). A review of predictive coding algorithms. *Brain and Cognition*, 112:92–97.

Squire, L. R., Knowlton, B., and Musen, G. (1993). The structure and organization of memory. *Annual review of psychology*, 44(1):453–495.

Stachenfeld, K. L., Botvinick, M. M., and Gershman, S. J. (2017). The hippocampus as a predictive map. *Nature Neuroscience*, 20(11):1643–1653.

Tulving, E. (2002). Episodic memory: From mind to brain. *Annual review of psychology*, 53(1):1–25.

A Generative modeling and the reconstructive theory of memory

In psychology and neuroscience, an semantic memory system learns the general structure of the sensory input and discards the idiosyncratic details of individual events. This corresponds to an energy function where the sum over the patters is replaced by a distribution $\phi(\boldsymbol{y})$ over all possible patterns:

$$u_{imp}(\boldsymbol{x}, t) = -\sigma^2 \log \int e^{-\beta(t)\|f_n(\boldsymbol{x})-\boldsymbol{y}/2\|_2} \phi(\boldsymbol{y}) \mathrm{d}\boldsymbol{y} \, . \tag{6}$$

In practice, the distribution $\phi(\boldsymbol{y})$ is often defined on a manifold or some other lower dimensional structure, leading to a dynamics visualized in Fig.1 a. This is exactly the kind of behavior we expect from a generative model, initial perturbed states are gradually pushed towards a point on the manifold of possible patterns. Conversely, the attractor dynamics of Hopfield networks and of the equivalent diffusion models is visualized in Fig.!1b, and formalize the concept of a pure episodic memory. However, modern research on human memory paints a different picture. Real-life episodic memories are thought to be largely reconstructive, meaning that most of the sensory details are re-created during recall based on contextual information (Roediger and McDermott, 1995; Hemmer and Steyvers, 2009). For example, the memory of a car crash may evoke the memory of broken glass, although the windshield was not actually broken during the real event. This suggests that human episodic memory has a stored lower dimensional "representational core" that does not fully constrains the dynamics of the system. This can be formalized using a mixture of discrete and continuous distributions:

$$u_{re}(\boldsymbol{y}) = \log \left(\sum_n^N e^{-\beta(t)\|f_n(\boldsymbol{x})-\boldsymbol{\xi}^n\|_2^2/2} + \int e^{-\beta(t)\|f_n(\boldsymbol{x})-\boldsymbol{y}/2\|_2} \phi(\boldsymbol{y}) \mathrm{d}\boldsymbol{y} \right) , \tag{7}$$

where $f_n : \mathbb{R}^D \to \mathbb{R}^W$ with $W < D$ is a lower dimensional encoding of the state and $\boldsymbol{\xi}^n \in \mathbb{R}^W$ is a stored lower-dimensional pattern. Since $W < D$, this energy will only constrain W degrees of freedom to converge to the pattern while the other degrees of freedom are left free to evolve under the dynamics determined by the corpus of semantic memory.

A.1 Episodic and semantic training of diffusion models

In a generative diffusion model, the score network is trained by minimizing a denoising autoencoder loss:

$$\mathcal{L}(W) = \mathbb{E}_{\boldsymbol{y},t}\Big[\mathbb{E}_{\boldsymbol{x}(t)|\boldsymbol{y}}\Big[\|\epsilon(\boldsymbol{x}(t), \boldsymbol{y}) - \boldsymbol{s}(\boldsymbol{x}(t), t; W)\|_2^2\Big]\Big] \tag{8}$$

where $\epsilon(\boldsymbol{x}(t), \boldsymbol{y}) = \boldsymbol{x}(t) - \boldsymbol{y}$ is the total noise added to the pattern $\boldsymbol{y}$ up to time t.

Minimizing this loss results in a generative model (or equivalently to semantic memory) if $\boldsymbol{y}$ is sampled from a continuous distribution (possibly defined on a lower-dimensional manifold), while it results in Hopfield-like episodic memory encoding when each pattern $\boldsymbol{y}$ is (re-)sampled with finite probability. As stated above, both regimes can be trained simultaneously if $\boldsymbol{y}$ is sampled from

a mixture of continuous and discrete distributions. Finally, he model can be trained on episodic "representational cores" such as in 7 if $\boldsymbol{y} \sim p(\boldsymbol{\xi} \mid \boldsymbol{\xi})$, with $\boldsymbol{\xi}$ being sampled from a discrete distribution. In the Discussion section, we briefly outline how this form of training could be implemented in the brain.

B Diffusion models and the brain

In this paper, we demonstrated that a popular class of modern Hopfield networks with exponential non-linearities is mathematically equivalent to variance-exploding diffusion models at the limit of $\beta \to \infty$. In our experiments, we showed that this equivalence holds almost exactly for a finite β value and for the more stable variance-preserving models, both with exact and trained score. The equivalence depends on the fact that the diffusion models are trained on a finite number of discrete patterns, and in fact the diffusion models can generalize the modern Hopfield energy function to setting where both episodic memory and semantic memory (i.e. generative manifolds) are jointly encoded by the dynamics of the same network (see Eq.7). From the point of view of theoretical neuroscience, this may be used to model the different forms of long-term memory as a result of the same learning mechanism, in a single distributed neural system.

While generative diffusion models offer an attractive paradigm for modeling memory, imagery and even perception in the brain, more work needs to be done in order for its components to be implemented in a biologically plausible way. In particular, it is unlikely that biological neural networks implement pure noise-injection dynamics. However, the mathematics of generative diffusion models can be written in term of any stochastic differential equation (S. et al., 2021), which can be used to implement more plausible transformation such as, for example, the feedforward perceptual feature as implemented in the hierarchy of sensory cortical areas (Güçlü and van Gerven, 2015). More fundamentally, it is unclear how the reverse and forward dynamics can be simultaneously implemented in the brain networks.

The denoising loss given by Eq. 8 is somewhat similar to the self-prediction errors used in predictive coding models (Rao and Ballard, 1999; Spratling, 2017; Millidge et al., 2021). This opens the door for potentially fruitful connections with existing predictive models of memory Stachenfeld et al. (2017); Barron et al. (2020). Particularly interesting is the recent use of generative predictive coding models for associative memory storage Salvatori et al. (2021), which is also based on a generative model and has some similarities with the approach discussed here.

The main biological issue with episodic memory as conceptualized in this paper is that it seems to require the re-sampling of the same events, while in the real world each event can only happen once. This can be potentially implemented with some form of bootstrap re-sampling, which might correspond to the replays observed in the hippocampus Buhry et al. (2011).

C Details of the experiments

Binary patterns were generated independently by applying the sign function to standard Gaussian vectors. To test recovery, we corrupted the patterns using the formula $\tilde{\boldsymbol{y}} = \theta \boldsymbol{y} + \sqrt{1-\theta^2}\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon}$ is a standard Gaussian vector. In all the experiments, we used $\theta = 0.68$.

We compered the storage and recovery performance of several models. Modern Hopfield network were implemented with the iterative update given in (Ramsauer et al., 2021). We set β to be equal to 5 and we applied 150 iterations to each noisy pattern.

For reasons of stability, we used variance-preserving diffusion models defined by the following noise-injection SDE:

$$\mathrm{d}\boldsymbol{x}_t = -\gamma \boldsymbol{x}_t + \gamma \mathrm{d}W_t \ , \tag{9}$$

where W_t is a standard Wiener process. This is slighly different from the variance-exploding models discussed in the main text, but it delivers nealy identical results. The diffusion models were used as denoiser by applying the deterministic ODE dynamics given in (S. et al., 2021):

$$\frac{\mathrm{d}\boldsymbol{x}_t}{\mathrm{d}t} = \frac{1}{2}\left(\boldsymbol{x}_t + \sigma^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}(t))\right) \ , \tag{10}$$

which exactly reproduces the marginal densities of the stochastic dynamics. We integrated the dynamics using 300 stepos of Euler integration. The initial time was set to match the match level

using the following formula: $t_{\text{start}} = -\gamma^{-1} 2 \log(\theta)$, with $\gamma = 0.8$. This formula implies the use of a constant noise scheduling with variance equal to γ.

In exact score-based diffusion model, we used the exact formula for the score given the marginal density:

$$\nabla \log p_t(\boldsymbol{x}) = (\boldsymbol{x} - Y \boldsymbol{h}_t(\boldsymbol{x}))/(1 - \theta_t^2) \, , \tag{11}$$

where Y is a matrix having the patterns $\boldsymbol{y}^j$ as columns. The weight vector $\boldsymbol{h}_t$ is obtained using the softmax function:

$$h_{t,j}(\boldsymbol{x}) = \text{softmax}\left(\dots, -\left\| \boldsymbol{x} - \boldsymbol{y}^k \right\|_2^2 / (2(1 - \theta_t^2), \dots \right)_j \, , \tag{12}$$

which depends on the correlation between the state $\boldsymbol{x}$ and each of the patterns. This is very similar to the softmax formula given in the update of modern Hopfield networks, which is unsurprising given the equivalence of their energy functions.

Learned generative diffusion models used a three layers fully connected architecture with reLu non-linearities, d input and output units and $80d$ hidden units in each layer. The time index was embedded by converting it to $\theta_t = \exp(-0.5\gamma t)$ and then by concatenating this value to each layer. They were trained using the optimizer Adam with base rate 0.001 and fixed batches containing all the generated patterns.

For the Hopfield and exact models, the capacity was estimated by evaluating the average reconstruction error in a grid of dimensionalities and number of patters. Given each combination, the error was computed 140 using randomly sampled patterns. The capacity was then estimated by finding the maximum number of patterns such that, for a given dimensionality, the average Hamming error did not exceed a threshold value of 3%. Error intervals were obtained by bootstrap re-sampling of the error. In order to increase the snd, we smoother both the mean capacity and the bounds by convolving them with a Gaussian kernel with $\sigma = 1.5$.

This estimation method would have been too expensive for evaluating the capacity of learned models, since the network needs to be fully re-trained for any given set of patters.

Instead, for a given dimensionality d, we trained the network with n_d patterns, where n_d is the estimated capacity of the exact score model plus 4. After training we evaluated the error on 30 batches of unseen noise corrupted versions of the training patterns. If the error was below threshold, we returned n_d as the estimated capacity, otherwise, we reduced n_d by one and repeated the treaning until the error was below threshold. This procedure was repeated 8 times for d equal to 10, 12, 14 and 16.