Random Feature Hopfield Networks generalize retrieval to previously unseen examples

Matteo Negri[12] * **Clarissa Lauditi**[34] **Gabriele Perugini**[4] **Carlo Lucibello**[45]

Enrico M. Malatesta[45]

[1] University of Rome 'La Sapienza'
Department of Physics

[2] CNR-Nanotec
Rome Unit

[3]Department of Applied Science and Technology
Politecnico di Torino

[4] Department of Computing Sciences
Bocconi University

[5] Institute for Data Science and Analytics
Bocconi University

Abstract

It has been recently shown that, when an Hopfield Network stores examples generated as superposition of random features, new attractors appear in the model corresponding to such features. In this work we expand that result to superpositions of a finite number of features and we show numerically that the network remains capable of learning the features. Furthermore, we reveal that the network also develops attractors corresponding to previously unseen examples generated with the same set of features. We support this result with a simple signal-to-noise argument and we conjecture a phase diagram.

1 Introduction

The Hopfield Model (HM) [1] is a paradigmatic model of associative memory with relevance in physics, biology, and computer science. Starting from corrupted signals, stored binary memories are retrieved as fixed points of a dynamical system which is also an energy minimization process.

Recently, generalizations of the HM have gained attention thanks to the addition of several desirable properties that nonetheless preserve the energy minimization and the associative mapping paradigms. In particular, Modern Hopfield Networks overcome the linear (in the system size) capacity limit of the HM and are able to store a polynomial [2, 3, 4] or even exponential [5, 6, 7] number of memories. Continuous variables and differentiable update rules allow to plug in trainable components in machine learning applications [6, 8]. A Lagrangian formalism can be used to describe a large family of such models [9], also accommodating popular deep learning components such as the attention mechanism [10] and layer normalization [11].

*Corresponding author, matteo.negri@uniroma1.it

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

In connection with recent advances in theoretical machine learning [12, 13, 14], it is shown in Ref. [15] that even the standard HM, which has no trainable parameters, when given memories generated by a latent manifold, $\boldsymbol{\xi}_\nu = \sigma(F\boldsymbol{c}_\nu)$, is able to "learn" essential features of the data generating process: when provided with enough samples, the (unobserved) columns of F become attractors of the dynamics.

In this work, we further characterize the Random Feature Hopfield Model (RFHM) of Ref. [15]. In particular, we show that the model is also able to "generalize", that is to store unobserved samples of the data manifold. Spurious minima, normally detrimental for the HM, become beneficial in this context.

2 Model description

Hopfield Model. Given N binary neurons $s_i = \pm 1$, and P binary memories $\{\boldsymbol{\xi}_\nu\}_{\nu=1}^P$ that we want to store, the HM defines the sequential update rule

$$s_i^{(t+1)} = \text{sign}\left(\sum_{j\neq i}^{N} J_{ij} s_j^{(t)}\right), \qquad J_{ij} = \frac{1}{N}\sum_{\nu=1}^{P} \xi_{\nu i}\xi_{\nu j}. \tag{1}$$

It can be shown [1] that as long as the memories are few (at most $O(N)$) and far apart enough, they approximately correspond to fixed points of the dynamics and can be retrieved from a perturbed configuration. The model admits an energy function $\mathcal{H}(\boldsymbol{s}) = -\frac{1}{2}\sum_{i\neq j} J_{ij}s_i s_j$. Using statistical physics techniques [16, 17], it has been shown that the model is able to store up to $P \approx 0.138N$ memories for large N.

Random Feature Hopfield Model. The RFHM [15] considers a data structure given by a random projection of a D-dimensional latent space [18, 13, 14, 19]:

$$\boldsymbol{\xi}_\nu = \text{sign}\left(F\boldsymbol{c}_\nu\right). \tag{2}$$

The matrix $F \in \{-1, +1\}^{N\times D}$ has i.i.d. uniform components. We call *features* its columns $\boldsymbol{f}_k$. The latents vectors $\boldsymbol{c}_\nu \in \mathbb{R}^D$ are called *coefficients* instead. We take each $\boldsymbol{c}_\nu$ to have exactly L non-zero entries, in random locations and uniformly sampled in ± 1. We will discuss two cases: the sparse case, $L = O(1)$ as $D \to \infty$, and the fully dense case, $L = D$. In the dense case Ref. [15] shows that when a large number of examples is given to the RFHM, beyond its storage capacity, it enters a phase where features instead become attractors. The model enters the learning phase if $\alpha = P/N$ is larger than a critical value that depends on $\alpha_D = D/N$. We refer to this as the *learning transition*.

Generalization in the RFHM. Given that the examples are correlated, it is useful to call *train example*, $\boldsymbol{\xi}^{\text{train}}$, any of the P examples that are used in the Hebb rule defining the couplings matrix J in (1). We also call *test example*, $\boldsymbol{\xi}^{\text{test}}$, any linear combination (followed by the sign activation) of the features that is not used in J. We can now say that an HM *generalizes* if test examples are fixed points of the update rule (1). In the same spirit of the learning phase in [15], we study if the network enters a *generalization phase* in some region of α, α_D and L space.

3 Numerical results

Methods. The numerical results of this work are all measures of *magnetizations*, obtained in the following way. First, we initialize the model to the configuration $\boldsymbol{s}^{(0)}$ whose stability we want to check (we consider $\boldsymbol{s}^{(0)} = \boldsymbol{\xi}^{\text{train}}$, $\boldsymbol{s}^{(0)} = \boldsymbol{\xi}^{\text{test}}$, and $\boldsymbol{s}^{(0)} = \boldsymbol{f}$, with $\boldsymbol{f}$ any of the columns of F). Then, we run the update rule (1) until we reach a fixed point $\tilde{\boldsymbol{s}}$. Finally, we compute the magnetization as the normalized scalar product between the fixed point and the initial condition: $m^{\text{train}} = \frac{1}{N}\sum_{i=1}^N \tilde{s}_i\xi_i^{\text{train}}$, $m^{\text{test}} = \frac{1}{N}\sum_{i=1}^N \tilde{s}_i\xi_i^{\text{test}}$, and $\mu = \frac{1}{N}\sum_{i=1}^N \tilde{s}_i f_i$ respectively. If, for given α and α_D, we find that the magnetization is close to 1, we say that there is a fixed point of eq. (1) that corresponds to $\boldsymbol{\xi}^{\text{train}}$, $\boldsymbol{\xi}^{\text{test}}$ or $\mathbf{f}$ respectively.

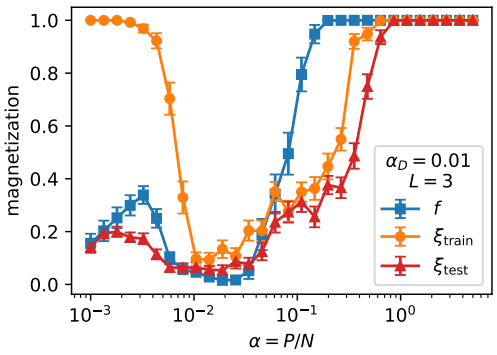

Figure 1: **Train and test examples become fixed points after the features have been learned.** Magnetization as a function of α, for fixed α_D. The blue line is the magnetization μ of hidden features, which grows to 1 if α is high enough (*learning phase*). The orange line is the magnetization m^{train} of the train examples, which is $m^{\text{train}} \simeq 1$ for low α and drops when α increases, as expected from an associative memory (*storage phase*). Surprisingly, m^{train} grows to 1 again for high values of α. Near this transition, also test examples have $m^{\text{test}} = 1$, as shown by the red line (*generalization phase*). $N = 32000$; averages of 40 samples.

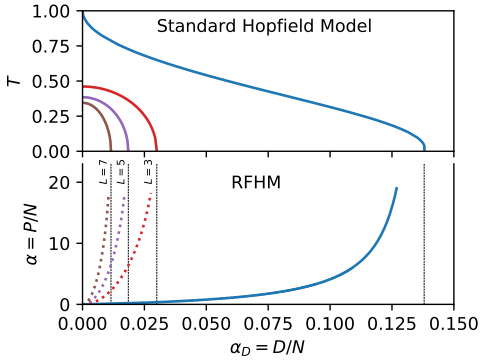

Figure 2: **Conjectured phase diagram.** Comparison between the phase diagram of a standard HM from [16] (*top*, temperature T vs α) and the phase diagram of the dense RFHM (*bottom*, α vs α_D). *Top panel:* the blue line is the retrieval line, below which the examples can be stored and retrieved. The (red, violet and brown lines are the retrieval lines of mixtures, respectively of 3, 5 and 7 examples. *Bottom panel:* the blue line is the learning transition from [15], above which the features can be stored and retrieved. The (red, violet and brown dotted lines are the *conjectured* retrieval lines of mixtures, respectively of 3, 5 and 7 features.

Learning phase. The first result that we present is that the model shows a learning transition even when it is trained with a sparse combination of features (see Fig. 1, blue line), extending the result of [15]. Surprisingly, it appears that the position of the learning transition depends weakly (if at all) on the number L of features per example (see Fig. 3a).

Generalization phase. Surprisingly, after the features have been learned, the model enters a phase in which the train examples are stable again (see Fig. 1, orange line). We note that it is physically implausible that a phase disappears for low α and reappears when α is large. To understand what is happening, we check the magnetization of test examples: we find that they become fixed points together with the train examples. We call this the *generalization phase*, as it resembles the behavior of inference models that perform well on previously unseen examples. The fact that both train and test examples are fixed points for high α, while only train examples are fixed points for low α, indicates that the model should be using different mechanisms to achieve these results.

Denser combinations In Fig. 3a we show learning (dashed lines) and generalization transitions (solid lines) for different values of L and α_D. We see that increasing either L or α_D has the effect of moving the generalization transition to higher values of α. Surprisingly, instead, we see that the learning transition depends weakly (if at all) on L. Additionally, in Fig. 3b we study the maximum number of features D_{gen} compatible with a generalization transition. We see that is $D_{\text{gen}} = O(N)$ in the sparse case $L = O(1)$ and $D_{\text{gen}} = O(\sqrt{N})$ in the dense case $L = D$. This scaling is compatible with what we find with a signal-to-noise analysis discussed in section 4.

4 Discussion

Conjectures on the phase diagram We propose the hypothesis that the mechanism that the network uses to recognise previously unseen examples is by "spurious" mixtures of features, in the same sense in which the classic Hopfield Network develops mixtures of train examples for low α [16]. Let us consider a RFHM when $\alpha \to \infty$. In this regime, it was shown in [15] that the RFHM becomes equivalent to a classic model with the features in place of the examples. Therefore we can expect

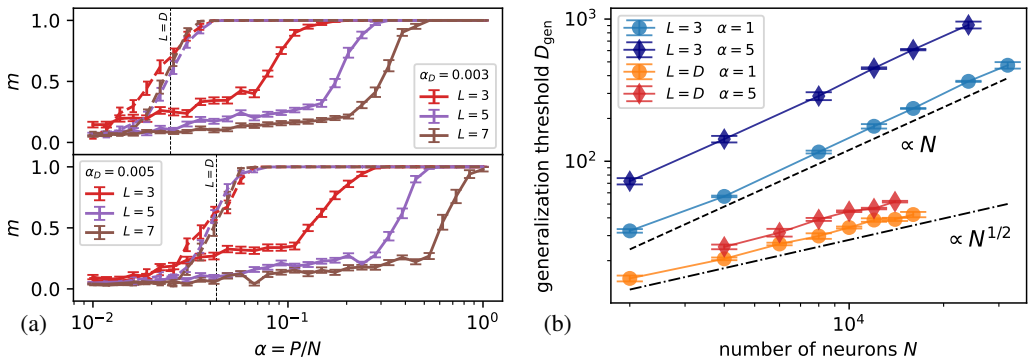

Figure 3: **Combinations of more features require more train examples.** a) Feature magnetization μ (*dashed*) and test examples magnetization m^{test} (*solid*) as a function of α for different α_D (subplots). Different colors represent increasing features per example L. Dashed vertical lines are the analytical predictions from [15] in the dense case $L = D$. ($N = 32000$; averages of 40 samples.) b) Scaling with N of the maximum number of features D_{gen} for which we observe a generalization transition. Specifically, we plot the maximum $D(N)$ at which 10 samples have $m^{\text{test}} > 0.9$, and we average over 4 to 10 groups of 10 samples, depending on N.

a transition at $\alpha_D \simeq 0.03$ where mixtures of $L = 3$ features become fixed points. Mixtures with increasing values of L are expected to become stable at lower values of α, as discussed in [16]. We conjecture that each of these transitions is the starting point of the generalization lines in the plane α_D vs α (see Fig. 2): we expect these lines to follow the shape of the learning transition, namely that when α is finite, the generalization transition moves towards lower values of α_D. This would be consistent with the result that combinations with larger L require more examples, and also with the fact that combinations with $L = D$ are stable only when $\alpha_D = 0$, which was already noted in [16].

Signal-to-noise analysis We provide an intuitive signal-to-noise argument to support the phase diagram shown in Fig. 3 and the conjecture we discussed above. For simplicity we ignore non-linearities. Let's define a mixture of features as $\chi_{\nu i} = \sum_{k''} c_{\nu k''} f_{k'' i}$, where at this level the coefficients can be either dense or sparse. We can see that the local field on a mixture can be written as a signal term, proportional to the mixture itself, plus a "crosstalk" term that includes the noise coming from the other features:

$$\sum_{j=1}^{N} J_{ij}\chi_{\nu j} = \sum_{k''} c_{\nu k''} \left[\frac{1}{N} \sum_{j} \sum_{\mu} \left(\frac{1}{\sqrt{L}} \sum_{k} c_{\mu k} f_{ki} \right) \left(\frac{1}{\sqrt{L}} \sum_{k'} c_{\mu k'} f_{k' i} \right) f_{k'' j} \right] \tag{3}$$

$$= \frac{P}{D} \left[\chi_{\nu i} + O\left(\sqrt{\frac{LD}{N}} \right) + O\left(\sqrt{\frac{L^3}{DP}} \right) + O\left(\sqrt{\frac{L^3}{NP}} \right) \right] \tag{4}$$

Now we can see that, if $L = O(1)$, the second and third noise terms go to zero when $N, D \to \infty$; then, to keep the first noise term finite, we must set $D = O(N)$. This means that the retrieval of (sparse) mixtures happens at $\alpha > 0$. On the contrary, it is impossible to retrieve (dense) mixtures at finite α when $L = D = O(N)$, since the signal would be overwhelmed by noise. It becomes possible instead when $L = D = O(N^{1/2})$, meaning $\alpha = 0$: in this regime, the first noise term is finite again (as well as the second and third ones). This scaling is confirmed by numerical results (see Fig. 3b).

5 Conclusions and perspectives

We showed that the learning phase studied in [15] still exists when the superposition of features is sparse. We also showed that the model produces attractors in correspondence of examples that were not used in the hebbian rule. We conjecture that this surprising behaviour is intimately related to the presence of mixtures of memories in Hopfield Networks.

The ability of a Hopfield Network to store linear combinations of examples has been known since [16, 17], but it gains a new framework in the light of [15]: if the features hidden in the examples become attractors, the network can combine them to produce new attractors that will recognize all the possible examples generated with the same features.

Note that one downside of the learning phase described in [15] is that, while the model is able to retrieve features if initialized close enough to one of them, there is no known way to find features if no information is known, as the model converges to a spurious state if initialized at random. The generalization phase seems to circumvent this problem: the model seems to mix the correct features without any prior information about them.

This work showed only a scheme of generalization that might be relevant for more complicated models, but before approaching more realistic situations we think that an analytical prediction of the conjectured phase diagram should be possible. Another important step will be to test this scheme on real datasets (drastic modifications to the Hebb rule are probably needed in this case).

Acknowledgements

MN acknowledges the support of PNRR MUR project PE0000013-FAIR.

References

[1] 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.

[2] E Gardner. Multiconnected neural network models. *Journal of Physics A: Mathematical and General*, 20(11):3453, 1987.

[3] Dmitry Krotov and John J Hopfield. Dense associative memory for pattern recognition. *Advances in neural information processing systems*, 29, 2016.

[4] Elena Agliari, Linda Albanese, Francesco Alemanno, Andrea Alessandrelli, Adriano Barra, Fosca Giannotti, Daniele Lotito, and Dino Pedreschi. Dense hebbian neural networks: a replica symmetric picture of supervised learning. *Physica A: Statistical Mechanics and its Applications*, 626:129076, 2023.

[5] Mete Demircigil, Judith Heusel, Matthias Löwe, Sven Upgang, and Franck Vermet. On a model of associative memory with huge storage capacity. *Journal of Statistical Physics*, 168:288–299, 2017.

[6] Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Thomas Adler, Lukas Gruber, Markus Holzleitner, Milena Pavlović, Geir Kjetil Sandve, et al. Hopfield networks is all you need. *arXiv preprint arXiv:2008.02217*, 2020.

[7] Carlo Lucibello and Marc Mézard. The exponential capacity of dense associative memories. *arXiv preprint arXiv:2304.14964*, 2023.

[8] Dmitry Krotov. A new frontier for hopfield networks. *Nature Reviews Physics*, pages 1–2, 2023.

[9] Dmitry Krotov and John J. Hopfield. Large associative memory problem in neurobiology and machine learning. In *International Conference on Learning Representations*, 2021.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[11] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[12] Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and the double descent curve. *Communications on Pure and Applied Mathematics*, 75(4):667–766, 2022.

[13] Sebastian Goldt, Marc Mézard, Florent Krzakala, and Lenka Zdeborová. Modeling the influence of data structure on learning in neural networks: The hidden manifold model. *Physical Review X*, 10(4):041044, 2020.

[14] Federica Gerace, Bruno Loureiro, Florent Krzakala, Marc Mézard, and Lenka Zdeborová. Generalisation error in learning with random features and the hidden manifold model. In *International Conference on Machine Learning*, pages 3452–3462. PMLR, 2020.

[15] Matteo Negri, Clarissa Lauditi, Gabriele Perugini, Carlo Lucibello, and Enrico Malatesta. Storage and learning phase transitions in the random-features hopfield model. *arXiv preprint arXiv:2303.16880*, 2023.

[16] Daniel J Amit, Hanoch Gutfreund, and Haim Sompolinsky. Statistical mechanics of neural networks near saturation. *Annals of physics*, 173(1):30–67, 1987.

[17] Daniel J Amit, Hanoch Gutfreund, and Haim Sompolinsky. Information storage in neural networks with low levels of activity. *Physical Review A*, 35(5):2293, 1987.

[18] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, pages 1177–1184, 2007.

[19] Carlo Baldassi, Clarissa Lauditi, Enrico M. Malatesta, Rosalba Pacelli, Gabriele Perugini, and Riccardo Zecchina. Learning through atypical phase transitions in overparameterized neural networks. *Phys. Rev. E*, 106:014116, Jul 2022.

A Signal-to-noise analysis

A.1 Retrieval of a feature with $L = \mathcal{O}(D)$ coefficients

In the fully dense case we study the signal-to-noise ratio of a single feature retrieval, being the local field

$$\begin{aligned} h_{ki} &= \sum_j J_{ij} f_{kj} \\ &= \frac{1}{N} \sum_j \sum_\mu \xi_{\mu i} \xi_{\mu j} f_{kj} \\ &= \frac{1}{ND} \sum_{k''} \sum_{k'} f_{k'i} \left[\sum_\mu c_{\mu k'} c_{\mu k''} \right] \left[\sum_j f_{k''j} f_{kj} \right] \\ &= \frac{1}{D} \sum_{k''} \sum_{k'} f_{k'i} \left(P \delta_{k'k''} + (1 - \delta_{k'k''}) O(P^{1/2}) \right) \left(\delta_{k''k} + (1 - \delta_{k''k}) O(N^{-1/2}) \right) \end{aligned} \tag{5}$$

which is separated into the true signal (contributing to the retrieval) for the diagonal term, while the rest is noise. By making the orders of different contributions explicit, we have

$$h_{ki} = \frac{P}{D} \left[f_{ki} + O\left(\sqrt{\frac{D}{N}} \right) + O\left(\sqrt{\frac{D}{P}} \right) + O\left(\sqrt{\frac{D}{N}} \sqrt{\frac{D}{P}} \right) \right] \tag{6}$$

where we collected the order of the signal. This is compatible with the learning phase described in [15] for finite values of $\alpha = \frac{P}{N}$ and $\alpha_D = \frac{D}{N}$: in fact, taking $D = \mathcal{O}(N)$ and $P = \mathcal{O}(N)$ makes all the noise terms finite in the limit $N \to \infty$. Fixing the scaling of the lower order noises also guarantees that the higher orders one are fine in the termodynamic limit.

A.2 Retrieval of a feature with $L = \mathcal{O}(1)$ coefficients

For the sparse case instead,

$$\begin{aligned} h_{k''i} &= \sum_{j=1}^{N} J_{ij} f_{k''j} \\ &= \frac{1}{N} \sum_{j} \sum_{\mu} \left(\frac{1}{\sqrt{L}} \sum_{k} c_{\mu k} f_{ki} \right) \left(\frac{1}{\sqrt{L}} \sum_{k'} c_{\mu k'} f_{k'i} \right) f_{k''j} \\ &= \frac{1}{L} \sum_{kk'} \left(\delta_{kk'} P \frac{L}{D} + (1 - \delta_{kk'}) O\left(\sqrt{P}\right) \frac{L^2}{D^2} \right) \left(\delta_{k'k''} + (1 - \delta_{k'k''}) O\left(\frac{1}{\sqrt{N}} \right) \right) f_{ki}. \end{aligned} \tag{7}$$

where now the patterns in N-dimension are properly scaled with $\sqrt{L}$. Solving for the four cases and by rescaling with the signal (feature) order we have

$$h_{k''i} = \frac{P}{D} \left[f_{k''i} + O\left(\sqrt{\frac{D}{N}} \right) + \frac{L}{D} O\left(\sqrt{\frac{D}{P}} \right) + \frac{L}{D} O\left(\sqrt{\frac{D^2}{PN}} \right) \right] \tag{8}$$

which provides a consistent insight that the learning transition also exists in the sparse case at finite α and that, as long as $L = \mathcal{O}(1)$ and $P, D, N \gg 1$ the transition seems to weakly depend on L (look consistently at Fig. 3a), since the noise terms depending on L can be ignored.

A.3 Retrieval of a mixture of features with $L = \mathcal{O}(1)$ coefficients

For the sake of clarity, we report the steps for eq. (3) in the main text. Being the mixture of features defined as $\chi_{\nu i} = \sum_{k''} c_{\nu k''} f_{k''i}$, we have

$$\begin{aligned} h_{\nu i} &= \sum_{j=1}^{N} J_{ij} \chi_{\mu j} \\ &= \sum_{k''} c_{\mu k''} \sum_{j=1}^{N} J_{ij} f_{k''i} \\ &= \sum_{k''} c_{\mu k''} \left[\frac{1}{N} \sum_{j} \sum_{\mu} \left(\frac{1}{\sqrt{L}} \sum_{k} c_{\mu k} f_{ki} \right) \left(\frac{1}{\sqrt{L}} \sum_{k'} c_{\mu k'} f_{k'i} \right) f_{k''j} \right] \\ &= \sum_{k''} c_{\mu k''} \frac{P}{D} \left[f_{k''i} + O\left(\sqrt{\frac{D}{N}} \right) + \frac{L}{D} O\left(\sqrt{\frac{D}{P}} \right) + \frac{L}{D} O\left(\sqrt{\frac{D^2}{PN}} \right) \right] \\ &= \frac{P}{D} \left[\sum_{k''} c_{\mu k''} f_{k''i} + O\left(\sqrt{\frac{LD}{N}} \right) + O\left(\sqrt{\frac{L^3}{DP}} \right) + O\left(\sqrt{\frac{L^3}{NP}} \right) \right]. \end{aligned} \tag{9}$$