SAPIENZA
UNIVERSITÀ DI ROMA

The random features p-spin Hopfield model

Facoltà di Scienze Matematiche, Fisiche e Naturali

Corso di Laurea Magistrale in Magistrale in Fisica

Candidate

Francesco D'Amico

Thesis Advisors

Prof. Chiara Cammarota

Prof. Matteo Negri

Academic Year 2022/2023

The random features p-spin Hopfield model
Master's thesis. Sapienza – University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Author's email: francescoluigidamico@gmail.com

Alla scuola romana di Fisica: a chi lo era, a chi lo è, a chi lo sarà.

Agli scienziati, ai pensatori e agli artisti, che hanno deciso il mondo in cui viviamo, e che hanno creato un nuovo mondo adatto a me e ai miei simili, e agli insegnanti, che lo tramandano.

Alla democrazia liberale, e a chi ha combattuto per conquistarla.

Alle miliardi di persone che non hanno avuto le mie stesse possibilità: che il mondo un giorno si capovolga.

Ai miei genitori, che con l'impegno hanno fatto nascere la vita vera.

Alla mia famiglia, che me ne ha fatta avere una.

Ai miei amici, che sono la mia casa.

A Paolo e ad Elisa, perché esistiamo solo nella condivisione.

Summary

Machine learning is the field of computer science that studies models that are capable of learning autonomously to solve a task directly from examples without providing an explicit coding of the solution. In recent years it has revolutionised many aspects of industry, research and everyday life. Since the beginnings, important contributions to the machine learning field have come from statistical physics. The first primitive models of artificial neural networks of the last century such as the Perceptron, a model of an artificial neuron, or the original Hopfield model, a system that retrieves stored memories starting from corrupted version, have been understood and popularized also thanks to the works of statistical physicists such as John Hopfield [Hop82] or Elizabeth Gardner [Gar87]. Although this models have been an important starting point in the construction of modern machine learning field, nowadays the state-of-the-art machine learning models are placed at a completely different scale of complexity and capabilities. However, it has been shown that recent Hopfield generalizations [KH16] [LM23] [Neg+23] can overcome some of the limitations of the original Hopfield network that led to its abandoning as a central model in the computer science field, and have consequently revived the interest of the statistical physics community. In the long-term effort to construct a *theory of machine learning*, such modern generalizations of Hamiltonian-based models that can be studied with the methods of statistical physics are a possible framework to model and understand modern machine learning. A clear example of the deep relation between modern Hamiltonian-based models and Transformers [Vas+17] - the most capable machine learning architecture today available - has been demonstrated recently [Ram+20]. For these reasons, the core of this work has been to investigate such new Hopfield neural networks with more powerful capabilities with respect to the original formulation, and to perform analytical and numerical investigations using the methods and the conceptual framework of the statistical physics of disordered systems. Two generalizations are considered. The first, the p-spin or polynomial energy model [KH16], to increase Hopfield capabilities increasing the order of the interaction between spins. The second, the random features model [Neg+23], to take into account that real-world data are correlated. In the first part the work is focused on the local attraction properties of the memories. The increase in the ability to memorize examples of a dataset of these systems is studied analytically with the signal to noise ratio method (SNR) and validated numerically. With respect to the original SNR application to uncorrelated Hopfield model [Gar87], this method is applied to the new random features case using a novel scheme to handle and carry out computations efficiently. Furthermore, it is obtained that increasing the power of the polynomial energy the number of memories that can be stored increases from linear to superlinear in the size of the system N, also for the random features case. In the second part of the work the dynamical phenomena taking place when the systems start from a random initialization are studied through numerical methods. First, it is obtained that there are certain operating regimes in which retrieval of memories can be obtained even without starting from a partial corrupted version of them, and secondly it is shown numerically how the ability to retrieve memory from a random initialization changes depending on model parameters.

Acknowledgments

I would like to thank Prof. Chiara Cammarota and Prof. Matteo Negri for the long commitment with which they supported me in the thesis development and taught me how to deal with a scientific research. Furthermore, thanks to their everyday work I have discovered an interesting path I want to take.

I would like to thank also my colleagues because they have been fundamental in getting me through university and one of the reasons why I pursued this path.
From books we know, but from others we learn.

Contents

Chapter 1

Introduction

1.1 The context of research

1.1.1 Machine learning

Machine learning has been a subject of interest for several decades among researchers, academicians, and industry experts. It consists in algorithms and methods that enable computers to learn how to solve a task directly from data through the training of a machine learning model without the need for a specific coding of the solution. Until today, in most of the cases the models with best performances have been the so called feed-forward artificial neural networks.
Recent years have seen an exponential surge in the development and application of machine learning techniques. This surge is primarily attributed to two main factors happened around the year 2010:

1. The increasing availability of large and diverse datasets, due to the rise of social networks, the ubiquitous usage of internet, the diffusion of clouds of data, phenomena usually referred to as "big data".

2. The advancements in computational power, in particular the sharp increase in production and availability of GPU processors, fundamental to achieve the parallelization of the operations needed to train the machine learning models.

So, after decades of experimentation, these engineering improvements have resulted in a jump in machine learning models performance, opening the possibility of application in real life contexts. Due to the renewed interest in the field, in the last ten years we have seen an explosion in machine learning architectures, algorithms complexity and performances.
These factors have collectively propelled machine learning from theoretical research to practical application, making it an essential component in a wide range of fields including healthcare, finance, transportation, and entertainment.

In general, in machine learning we have a model, a task to be solved, and a dataset of examples X.

- A model is an algorithm that takes an input X, then combines the input with its internal weights W determined by a certain architecture, and finally gives

an output $\hat{Y}$. The architecture is fixed - there are many architectures available, each of which is suited to a certain class of problems - meanwhile the internal weights can be changed. At first W are randomly initialized, and the model produces random $\hat{Y}$ values given some inputs X.

- The task consists in assigning for each input X an output $\hat{Y}$ that is as similar as possible to a given Y, that we define as the solution of the task.

To accomplish the task the only possibility for the model is to change its internal weights W in an appropriate way; this procedure is called the *training of the model*. During the *training phase* a dataset of examples is presented, and the system is updated through a *training algorithm* that changes the internal parameters W of the model following a certain rule. Usually this rule consists in a minimization of a certain cost function, called usually *loss function* $\mathcal{L}$. There are two conceptually different classes of training:

1. If, for every input example X, the correct solution of the task Y is also given, the training is called *supervised*. Usually, in this case, the *loss function* is defined as a measure of a certain discrepancy between Y and the output of the model $\hat{Y}$. So, for every task, it is defined an appropriate mathematical distance between Y and $\hat{Y}$, for example the MSE (mean square displacement) in the case Y and $\hat{Y}$ are vectors in a euclidean space. We note that in this case we need to already know the Y values, for example because they have been computed in another way.

2. If we don't have the correct solution Y for each X of the training set, the training is called *unsupervised*, and the loss function is defined in another way that doesn't require the Y values.

During the training, the examples X are continuously presented to the model, which every time updates it's weights W, until a certain stop criterion is reached, concluding the training procedure. At this point, the weights W are frozen, and the model is ready to be used.

The fundamental idea is that we are interested in the performance of the model for new data X that are not inside the training dataset, because it is useless to solve the task when we already know the solutions. Here comes the real point of neural networks: the architectures are built in such a way that by training them on a given dataset, they can generate good predictions on new data never seen before. In practice, to simulate new data never seen before, we leave out of the training set a fraction of the examples, and we call it the *test set*. Then, the real performances of the model are computed after the training using the test set.
In this context, three concepts are defined:

- **Storage** is the capability of the model to code the training set inside it's internal weights, in such a way that given a training input X, the model produces the correct $\hat{Y}$.

- **Learning** is the ability of the model to construct an internal representation of the training data that is useful to perform *generalization*. This concept is more abstract with respect to the other two, but as we will see it will acquire a more specific meaning in the context of intrinsic dimensionality of a dataset, that will be shortly presented.
- **Generalization** is the key ability of a machine learning model. It is defined as the capability to provide good performances with the test set, which means using the information of the training dataset to solve the task for examples never seen before.

Usually machine learning is applied to difficult, high-dimensional and highly non convex problems, because for problems in which we can code directly the solution it is not justified the complicated and costly training of a model. It is possible that a dataset has an intrinsic natural representation made of latent variables that can be obtained through some sort of transformations from the original data, and in this new latent space the original difficult problem is mapped in a simple way [**FER17**]. For example, in a classification task, it is possible that inside this internal representation inputs X are mapped in clusters of points that shares the same Y value and are linearly separable. In this context, a possibility is that artificial neural networks (ANN) show good generalizations properties because they are able to capture this intrinsic latent representation of those high-dimensional data provided, and in this case we say that it is capable of learning instead of just storing. In Sec. 1.5 it is explained in which way the Hopfield model generalizations analyzed in this work are linked to this idea of latent spaces, and what is the meaning of *storage, learning and generalizaton* in those models.

1.1.2 Statistical mechanics for machine learning and the context of the present work

Statistical mechanics is a branch of physics that uses statistics to explain the collective behavior of systems with a large number of variables. As we will see in this section, it has played a fundamental role in structuring the foundations of machine learning field in the early stage, specifically with the Perceptron and the Hopfield models, solved in the framework of the physics of disordered systems.

The *Perceptron* is a model of neurons invented in 1943 by Warren McCulloch and Walter Pitts [**MW43**]. Some years later, in 1958, Frank Rosenblatt used this model to perform binary classification with *supervised learning* [Ros58]. A Perceptron takes as input a vector $\mathbf{x}$ and gives as output $f(\mathbf{x})$ a binary value:

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} + b > 0, \\ 0 & \text{otherwise} \end{cases} \tag{1.1}$$

where $\mathbf{w}$ and b are respectively the *synaptic weights* and the *threshold*. In particular, these weights have to be learned using a learning algorithm and a dataset, and after the training they codify the Perceptron solution to a given binary classification problem. Many learning algorithm have been invented, mainly using an iterative

updating of the weights, presenting every time an example from the dataset, computing the predicted classification and adjusting slightly the weights. In particular, every weight has to be changed of an amount proportional to a chosen *learning rate* γ in the direction of improving the classification. For example, if an element of the dataset $\mathbf{x}$ has a class $y = 1$, but $\mathbf{w} \cdot \mathbf{x} + b < 0$ and so it is misclassified, the adjustment of every weight has to be done so that with the new weights $\mathbf{x}', b'$ the system will make a better prediction, that means in this case $\mathbf{w}' \cdot \mathbf{x} + b' > \mathbf{w} \cdot \mathbf{x} + b$.
After Rosenblatt's work the Perceptron sparked interest as a promising model to perform machine learning, but then it was understood that only linearly separated classification problems could be solved. It means that in the vector space of $\mathbf{x}$, Perceptron can learn the classification task only if the vectors belonging to the two classes can be separated by a plane, making the Perceptron too trivial to work with real data. Then the model was deeply studied with great success using statistical mechanics tools; in particular with the replica method it is possible to obtain analytical predictions of the result of many different classes of learning methods.
Anyway, although the Perceptron is a too simple model of a single neuron that can be trained to classify inputs, it has been a stepping stone for the development of more advanced neural networks and for the field of Machine Learning in general. After many years it was understood that stacking two or more layers in what is now called *Multilayer Perceptron* or *Feed-forward neural network* the model have greater processing power and can deal with general classification problems, not being restricted to only linearly separated ones.

The *Hopfield model* is a form of recurrent artificial neural network popularized by physicist John Hopfield, capable of storing patterns and retrieve them starting from distorted or partial versions. It is called a model of *associative memory* in the field of pattern recognition, an important branch of machine learning. This model is the main object of study of this entire work and will be explained in detail in the next sections.
It was studied initially by John Hopfield [Hop82], then the phase diagram of the model was obtained with statistical mechanics [AGS85]. In particular, given an amount of stored examples P proportional to the number of neurons N, namely $P = \alpha N$, it was shown that even in absence of noise there is a critical value α^c such that for $\alpha > \alpha^c$ the system is not capable anymore to retrieve the memories.
Then, as the original Perceptron, the Hopfield model was too limited to be used for real-world applications, and the interest for the Hopfield model inside the artificial intelligence field faded, even if it was continued to be used in the modeling of biological neural networks, such as in [GC99].

In the last years, new generalizations of the original Hopfield network [KH16] [LM23] have shown increased capabilities that makes them more similar to modern high performances state-of-the-art machine learning models [Ram+20], or enables them to handle structured memories [Neg+23]. We will call those *Hamiltonian-based models*, because they are defined through an Hamiltonian. Even though they have not the same capabilities, the interest in the analysis of those models is based on the idea that they could share some general properties with state-of-the-art machine learning models. Due to the possibility to study Hamiltonian-based models with

statistical mechanics, and thanks to the decades of work already done by physicists in this field, those neural networks represent a promising framework from which to study more general and capable classes of machine learning models.
An example in this direction is represented by the recent intuition that Transformers - the most advanced machine learning architecture at the moment - are based on a mechanism, called *attention*, that is deeply related to Hopfield models [Ram+20].

The present work fits within this context of studying modern Hopfield generalizations using typical statistical physics concepts and theoretical methods. The idea is to understand the phenomena behind their functioning, and what changes in their internal behaviour as parameters and architectures changes. The work is interesting in itself in the theory of disordered systems because Hopfield belongs to spin glasses models, but the phenomena investigated in this work were also chosen on the basis of how relevant they are in the context of machine learning.
In the end, historically, Physics has provided a successful conceptual framework to analyze neural networks models and to understand the complex phenomena arising in neurons. Even though nowadays machine learning works with way more complex architectures, the Hamiltonian-based models and their analysis have been a fundamental stepping stone and now they are viewed as a possible starting point for the construction of a *theory of machine learning*.

1.1.3 The need for a theory of machine learning

During the last decade we have assisted to a machine learning revolution, in which progress was made mainly due to engineering factors, intuitions, trials and errors, stacking of discoveries. The rapid advances achieved in model performances and complexity opened a gap between the models capabilities and the theoretical comprehension of the emergent phenomena inside them. Models with billions of parameters (even more then the data size) are capable of not overfit, something opposite to the central variance-bias tradeoff of statistics.
On one side it is of theoretical interest to have a theory of machine learning, because we have a system that is working and that is not understood deeply. Thus, it would be of scientific interest to construct a theory of machine learning, linking the functioning of modern ANN to the theory of complex systems. Moreover, there is the possibility that at the heart of the functioning of these architecture there are new phenomena, and machine learning models can act as a stimulus in the research field of complex systems.
On the other side, for applications, a better understanding of the functioning of ANN's could be useful to improve the architectures themselves. To increase performances, models are constantly increasing their dimension, in a phenomenon called *parameter proliferation*. The most powerful ones need huge amounts of electricity to be trained, turning this technology against ecological issues. It's possible that understanding better the way they work could open the possibility to reduce their dimension and to make them more efficient. Even if the models inherently need giant dimensions to work properly due to some sort of general principles not yet definitively understood, theoretical knowledge could bring to eliminate unnecessary architectures features or let developers to focus on the most important features of

the models.
These ideas, although they are mainly speculations at the moment, represent the underlying reasons for the effort of focusing on a theoretical approach to machine learning.

One of the recent breakthroughs has been the development and implementation of Transformer architectures in machine learning. Transformers, introduced in the seminal paper [Vas+17], have revolutionized the landscape of deep learning, particularly in the realm of natural language processing (NLP) [**Khu+23**].
Transformers are based on the concept of *attention*, enabling the model to dynamically focus on different portions of the input data by assigning different weights or 'attention scores'. This mechanism allows Transformers to handle long-term dependencies effectively, making them particularly suitable for very complex tasks, such as text and speech processing, machine translation or natural-language generation.
Recently, in [Ram+20], the attention mechanism at the heart of transformers has been demonstrated to be closely related to the modern Hopfield model, the main object of the present work. This fact suggests that the fundamental models studied in statistical mechanics before the current deep learning revolution could still be relevant in order to achieve a theoretical understanding of deep architectures, particularly in the case of new appropriate models generalizations.

1.2 The original Hopfield model

1.2.1 From biological to artificial neural networks

Biological neural networks are made of neurons, each of which receives signals from many other neurons and transmits signals through a single axon. In the process of building a simple model of a neural network inspired by biological ones, a fundamental fact known from biology is that between neurons there is a continuous exchange of all-or-nothing signals. Without going too deep into biology, the all-or-nothing law refers to the observed fact that along an axon an excited neuron can produce its maximal response, or no response at all, and not a continuous signal. This law justify a discrete variables model of a neural network, assigning each of the N neurons making up the network the values $\sigma_i = \{-1, +1\}$.
A discrete-time model of this process could be written as

$$\sigma_i(t + \Delta t) = \text{sign}(\sum_{j(\neq i)} J_{ij}\sigma_j(t) - \theta_i) \tag{1.2}$$

where J_{ij} encodes the connections between neurons, and $\sum_{j(\neq i)}$ is defined as the sum over all j apart from i. We consider a fully-connected network in which all neurons are connected, and we think about the learning of the model as a process in which the couplings J_{ij} between neurons are tuned by external stimuli.
So, at time $t + \Delta t$, the binary neuron σ_i takes the input from every other σ_j at time t, modulated through synaptic weights J_{ij}, and then if this signal is bigger than a certain threshold θ_i the neuron fires through his axon. In this model, the information to do any possible function that a brain is required to do has to be

written in J_{ij} and in θ_i. However, for the sake of simplicity, for the rest of the work we assume $\theta_i = 0$ for every neuron.
From now on, we focus on the particular function of memory. Images, sounds or in general physical experiences are described in terms of firing patterns in the brain. Then, we imagine that a memory consists in the storage of a firing pattern $\vec{\xi} \in \{-1, +1\}^N$. We model the memory functioning as a process in which, starting from a corrupted version of one memory stored, the network has to be capable of retrieving the (almost) complete information. That is, in our framework, if we consider a certain state of a system $\vec{\xi}$ as a memory, we want to define a choice of the couplings J_{ij} such that the system converges to this firing pattern $\vec{\xi}$ starting from a configuration at a given Hamming distance from it.
Starting from experimental evidence of 40's by Donald Hebb, it was hypothesized that "cells that fire together, wire together". It means that, in the process of retrieving a memory $\vec{\xi}$, when it happens that the neuron σ_j fires just before σ_i, the weight J_{ij} gets more positive. This idea was hypothesized to be at the foundation of neural plasticity, that is the ability of the brain to change the synaptic weights in the process of learning a new memory. The most interesting feature of this model of neural plasticity is that it is local: two neurons change their reciprocal weights using only the local property of firing correlation.
If the model has only one single memory $\vec{\xi}$, the Hebb's idea is implemented in the Hopfield model defining

$$J_{ij} \coloneqq \xi_i \xi_j \tag{1.3}$$

To understand the relation of this rule with the Hebb's idea, let's take spins $\sigma_i = \{0, 1\}$ (also $J_{ij} = \{0, 1\}$), interpreted as firing or not firing a signal to other neurons. If the neural network is exposed to an input $\vec{\xi}$, if $\xi_i = 1$ and $\xi_j = 1$, σ_i and σ_j are firing together. In this case using Eq. 1.3 the coupling $J_{ij} = 1$, and so there is an attractive coupling between them: two neurons firing together, now are wired together. In every other case, the two neurons are not firing together, and J_{ij} reflects the absence of a reciprocal wiring. From this reasoning, it is straightforward to change variables to the usual definition $\sigma_i = \pm 1$. In this case it is possible to interpret $J_{ij} = +1$ when ξ_i, ξ_j had the same sign as the neural plasticity wiring neurons that both fire together or don't fire together, and unwiring neurons that do the opposite behaviour (one firing, the other not).
Later, once the process of storing that memory has ended and then J_{ij} have been defined in according to the Hebb's rule, the model works as an associative memory, at least for the retrieval of a pattern if only one single spin is $\sigma_i \neq \xi_i^\mu$. This is true because with this prescription when all $\sigma_j(t) = \xi_j$ apart from σ_i, using the Eq. 1.2 that neuron will be set to

$$\sigma_i(t + \Delta t) = \text{sign}\left(\sum_{j(\neq i)} J_{ij}\xi_j\right) = \text{sign}\left(\sum_{j(\neq i)} \xi_i\xi_j\xi_j\right) = \text{sign}(N\xi_i) = \xi_i$$

Then, with the Hebb's prescription in Eq. 1.3, if the system is in the proximity of a memory, it will converge to it.
Now, if we have a number P of memories $\vec{\xi^\mu} \in \{-1, +1\}^N$ to be remembered, we

can generalise Hebb's idea defining

$$J_{ij} = \frac{1}{N}\sum_{\mu=1}^{P} \xi_i^\mu \xi_j^\mu \tag{1.4}$$

This is useful to store memories because we can see from the dynamics in Eq. 1.2 that, if the system is already in the memory $\vec{\xi^\mu}$

$$\sigma_i = \text{sign}\left(\sum_{j(\neq i)} J_{ij}\xi_j^\mu\right) = \text{sign}\left(\frac{1}{N}\sum_{j(\neq i)}\sum_{\nu}\xi_i^\nu\xi_j^\nu\xi_j^\mu\right)$$

Using the approximate orthogonality between random patterns for the number of neurons $N \gg 1$

$$\frac{1}{N}\sum_{j(\neq i)} \xi_j^\mu \xi_j^\nu = \delta_{\mu\nu} + \mathcal{O}\left(\frac{1}{\sqrt{N}}\right)$$

we get

$$\sigma_i \approx \text{sign}\left(\sum_{\nu}\xi_i^\nu\delta_{\mu\nu}\right) = sign(\xi_i^\mu)$$

and so if the system is over a memory, it is stable. However we have neglected the role of the $\mathcal{O}(\frac{1}{\sqrt{N}})$ term, and as we will see in the next section this approximation works only until a certain P value, after which the system is not stable anymore when initialized over memories.
In the next section we will see that the dynamical rule in Eq. 1.2 with the choice of J_{ij} in Eq. 1.4 coincides exactly to the original formulation of the Hopfield model: in particular, it is exactly the Metropolis dynamics at zero temperature.

1.2.2 The Hopfield model

In the Hopfield model the Hamiltonian is defined as

$$H = -\frac{1}{N}\sum_{\mu}^{P}\sum_{(i,j)}^{N}\xi_i^\mu\xi_j^\mu\sigma_i\sigma_j \tag{1.5}$$

where $\vec{\xi^\mu}$, with $\mu = 1, ..., P$, are P random generated patterns, and N is the number of discrete variables $\sigma_i \in \{-1, +1\}$. With the notation $\sum_{(i,j)}$ it is meant to sum over all the couples (i, j) taken only once, without the diagonal terms, that is when $i = j$. For example the term $(1, 2)$ has to be taken, but then $(2, 1)$ has to be excluded. The reason is that these two terms are equal and then it is a waste of computational power to consider them both and then to divide all by two. Defining the couplings matrix J_{ij} as

$$J_{ij} := \frac{1}{N}\sum_{\mu}^{P}\xi_i^\mu\xi_j^\mu \tag{1.6}$$

it is possible to write

$$H = -\sum_{(i,j)}^{N} J_{ij}\sigma_i\sigma_j \tag{1.7}$$

In statistical mechanics, given an Hamiltonian, a probability measure is defined for each possible state of the system:

$$P(\vec{\sigma}) = f(H[\vec{\sigma}], \{A\}) \tag{1.8}$$

where $\{A\}$ is a set of parameters of the system. In the fixed-temperature case the relevant parameter is $\beta \coloneqq 1/T$, being T the temperature, and the measure is the Gibbs one:

$$P(\vec{\sigma}) \propto \exp(-\beta H[\vec{\sigma}]) \tag{1.9}$$

In the Hopfield case the temperature is intended as the level of noise in the system, namely how much probable is that during a dynamic a move that increases the energy is accepted. The *Metropolis dynamics* is defined updating one randomly chosen σ_i at a time with the prescription

$$\sigma_i(t + \Delta t) = \begin{cases} \sigma_i(t) & \text{with probability } \min(e^{\beta \Delta E}, 1) \\ -\sigma_i(t) & \text{otherwise} \end{cases} \tag{1.10}$$

where $\Delta E \coloneqq E(-\sigma_i(t)) - E(\sigma_i(t))$ is the difference of energy of the system that would occur flipping a spin.
The zero-temperature dynamics of this algorithm is

$$\sigma_i(t + \Delta t) = \begin{cases} \sigma_i(t) & \text{if } \Delta E > 0 \\ -\sigma_i(t) & \text{otherwise} \end{cases} \tag{1.11}$$

Using the variable

$$\delta \coloneqq \begin{cases} \Delta E & \text{if } \sigma_i(t) = +1 \\ -\Delta E & \text{otherwise} \end{cases} \tag{1.12}$$

for the Hopfield model the prescription in Eq. 1.11 becomes

$$\sigma_i(t + \Delta t) = \begin{cases} +1 & \text{if } \delta > 0 \\ -1 & \text{otherwise} \end{cases} \tag{1.13}$$

Thus, in the case of the zero-temperature Hopfield models investigated throughout the work, this algorithm become

$$\sigma_i(t + \Delta t) = \text{sign}\left(\sum_{j(\neq i)} J_{ij} \sigma_j(t) \right) \tag{1.14}$$

updating asynchronously a random chosen spin σ_i. With the notation $\sum_{j(\neq i)}$ it is intended the sum over all j except the case $j = i$. The fundamental observation is that this equation is exactly the dynamical Eq. 1.2 of a neural network with the simplifications presented in the previous section. Therefore, the Hopfield model is the statistical mechanics analogous of the Hebb's neural network.
Finally, summing up, the dynamics 1.14 it's interesting from a neural networks perspective because it can be interpreted as a binary neuron σ_i, like a biological one, receiving the signals of all the other ones modulated from synapses weights J_{ij} constructed through the Hebb's rule.

1.2.3 The spin glasses and the relation with SK model

Relation of Hopfield with the spin glass theory

The Hopfield model with the Hamiltonian defined in Eq. 1.5 belongs to the category of disordered systems because the couplings J_{ij} as defined in Eq. 1.6 between variables are randomly cooperative (positive) or anti-cooperative (negative).
As opposed to ordered systems such as ferromagnets in which it is possible to satisfy all bonds at the same time, the presence of positive and negative couplings could lead to frustration, a phenomenon in which to satisfy some bonds it is necessary to unsatisfy others. From an energy point of view, frustration creates the possibility for variables to make not trivial rearrangements of bonds that will decrease in the end the total energy, but in this process it could requested to break other already present bonds.
The presence of this possibilities makes the energetic landscape highly complex due to the presence of many local minima, among which it's highly non-trivial to find ground states, or even possible lower energy states starting from a certain configuration.
In Hopfield model most of these minima are not directly correlated to the patterns $\vec{\xi}^{\mu}$ of the system, and then they are called *uninformative* or *spurious* minima, to distinguish them from the minima that contain the patterns to be retrieved. This phenomenon in which are present many minima and the energetic landscape is highly non-convex will be referred to as generically the *roughness* of the landscape.
It is possible to formalize the definition of roughness in a mathematically rigorous way, but in this work this phenomenon will be analyzed and referred to in a more qualitative than quantitative way. In particular, given an area of the configurations space, we describe the landscape as more *rough* both when there are more spurious minima and when they increase in depth.

Spin glasses, Edwards-Anderson and Sherrington-Kirkpatrick models

From an historical point of view, the interest for disordered systems was born with particular metallic alloys in which ferromagnetic impurities were positioned in random places. Experimentally these alloys didn't show the phenomenon of magnetization for any temperature, but responded to certain kinds of magnetic perturbations.
One interesting example was observed immersing a sample of such alloys in an oscillating magnetic field $h(t)$: below a certain critical temperature T_c an out-of-phase magnetic susceptibility χ'' was present.
One important model of the behaviour of such metallic alloys was the **Edwards-Anderson model** [EA75]. Given N Ising variables $\sigma_i = \pm 1$, the Hamiltonian is defined as

$$H = -\sum_{\langle ij \rangle} J_{ij}\sigma_i\sigma_j \tag{1.15}$$

where $\langle ij \rangle$ means that the sum has to be computed over every neighboring couple of spins σ_i and σ_j in the lattice of variables, and J_{ij} is the coupling matrix. This matrix have both ferromagnetic and anti-ferromagnetic values, extracted from a probability distribution, with mean J_0 and a variance $J^2 = O(1)$.

As for the alloys from which this model is inspired, calling $m_i \coloneqq \langle \sigma_i \rangle$, the magnetization

$$M \coloneqq \frac{1}{N} \sum_i m_i \tag{1.16}$$

is near zero for every temperature, and so it cannot be used as the order parameter of the system such as for ordered magnetic models. Instead, the order parameter was defined as

$$q_{EA} \coloneqq \frac{1}{N} \sum_i m_i^2 \tag{1.17}$$

and there is a critical temperature dividing the cold phase $q_{\mathrm{EA}} > 0$ from the hot phase with $q_{EA} = 0$.
The main difficulty of this model is that the variables are defined on a lattice, making computations difficult to carry out. For this reason the **Sherrington-Kirkpatric model** was studied, the infinite range version of the Edwards-Anderson spin glass. So, the difference is that in the former the interaction network is fully-connected, instead in the latter the spins are in a lattice and interact only with their neighbors. This difference changes completely the difficulty of treating the models analytically, because in a fully-connected model it's possible to use mean-field methods, such as the Thouless-Anderson-Palmer (TAP) equations [TAP77]. In particular, for this model it is possible to make computations with the replica method, as done firstly by G. Parisi [Par80], and to obtain an analytical thermodynamical equilibrium solution.
The Hamiltonian of this model is defined as

$$H = -\sum_{(i,j)} J_{ij} \sigma_i \sigma_j \tag{1.18}$$

where couplings are random extracted binary or standard Gaussian variables, for example from the distribution

$$P(J_{ij}) = \sqrt{\frac{N}{2\pi J^2}} \exp\left\{ -\frac{N}{2J^2} \left(J_{ij} - \frac{J_0}{N} \right)^2 \right\} \tag{1.19}$$

and therefore

$$J_{ij} = O\left(\frac{1}{\sqrt{N}} \right) \tag{1.20}$$

making the Hamiltonian correctly defined as an extensive quantity.

From Hopfield to SK model

It's interesting to observe that the Hamiltonian of the Hopfield model in the form 1.7 and the Sherrington-Kirkpatric model defined in Eq. 1.18 have the same form. The difference is that in the latter the couplings matrix J_{ij} is a random Gaussian matrix, so it has not an internal structure like the Hopfield model's one.
In particular, there is a case in which the two definitions coincides, that is

$$\begin{cases} P = \alpha_P N \\ \alpha_P \to \infty \end{cases} \tag{1.21}$$

In this limit the Hopfield coupling matrix for the Central Limit Theorem (CLT) tends to a random Gaussian matrix, just like in the SK model, and

$$J_{ij} = \frac{1}{N}\sum_{\mu}^{P} \xi_i^\mu \xi_j^\mu \longrightarrow \frac{O(\sqrt{P})}{N} = O\left(\frac{1}{\sqrt{N}}\right) \tag{1.22}$$

thus it has the same scaling with N as for the J_{ij} of the SK model.

1.2.4 Statistical mechanics of the Hopfield model

The Hopfield network is an interesting model due to its associative memory behaviour. Given an appropriate choice of the parameters (P, N, T) that makes the system be in the so called *retrieval phase*, if the system is initialized near enough a certain memory $\vec{\xi}^\mu$ (that will be also referred to as *pattern*), it will do a dynamic according to Eq. 1.14 such that the final state will be close enough to this pattern $\vec{\xi}^\mu$.
The similarity between a system configuration $\vec{\sigma}$ and a given pattern $\vec{\xi}^\mu$ is quantified via

$$m^\mu := \frac{1}{N}\sum_j \xi_j^\mu \sigma_j \tag{1.23}$$

The maximum value of them is defined as

$$m := \max_\mu(|m^\mu|) \tag{1.24}$$

and corresponds to the order parameter of the system, where the absolute value is used because the Hamiltonian is symmetric to global spin inversion. This quantity $m \in [0, 1]$, where the value 1 indicates that the system is exactly on a memory, instead a value near to zero means that the system is almost perpendicular to all memories.

Thermodynamics

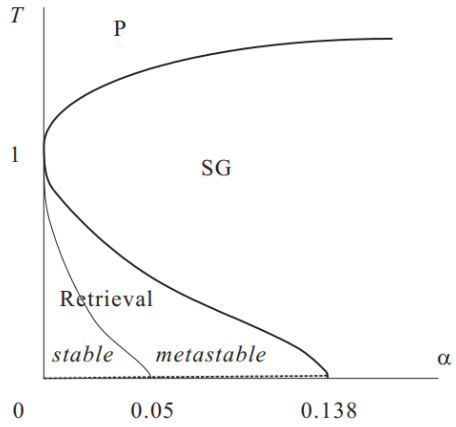


Figure 1.1. The phase diagram of the system studied with replica method in [AGS85]. Figure from [Nis01].

The thermodynamics of the original Hopfield model has been studied and understood extensively in 80's. The main result (obtained in [AGS85]) using the replica method is that the phase diagram is subdivided between a paramagnetic, a spin glass and a retrieval phase, as described in Fig. 1.1.
In general, as described in Eq. 1.8 and in Fig. 1.1, the system can be studied at any temperature T. However, from now on, in this work will be considered only the $T = 0$ case.
In particular, as we can see from the figure, at $T = 0$, if $\alpha := \frac{P}{N}$ is bigger then the critical value $\alpha_c \approx 0.138$, the system is in a spin glass phase and it's not possible to store memories, being them unstable states. Instead, below α_c the memories are stored but they are metastable states. Finally, below $\alpha \approx 0.05$ not only the memories are stored, but they are also the equilibrium states of the system.

Dynamics

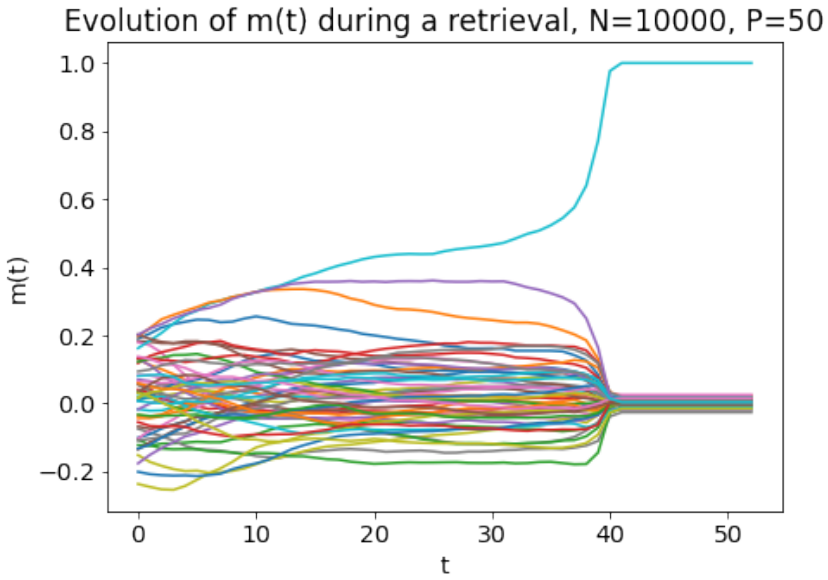


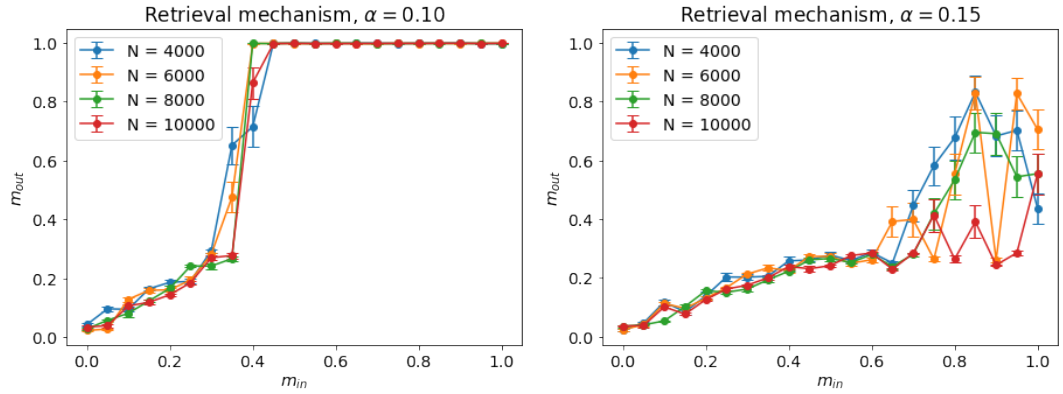
Figure 1.2. An example of the evolution of the observables $m^\mu(t)$ starting from a random state, in a regime where dynamics usually converge to a memory also starting from a random configuration. After some Δt the system is attracted by a single memory, and the dynamics converges rapidly leaving all other m^ν near to zero. Being the dynamics and the variables discrete, if it is reached a state that has *point stability* (as defined in a while), the dynamics stop and the reached point is the final state of the system.

The analysis of the thermodynamics of a statistical mechanics system is a fundamental step in the comprehension of its properties. However, in disordered models such as a Hopfield neural network, often complex dynamical properties are present and can influence considerably the real behaviour of the system. Given that, beyond the thermodynamics, it is of central importance to study the dynamics of the models.
Depending on the parameters of the model P, N, T, and on its initial state $\vec{\sigma}(0)$ the system shows different behaviours. As already stressed the interesting behaviour of the network is its associative property, and so the interesting regime is the retrieval

phase in Fig. 1.1.
In the retrieval phase, namely if $\alpha < \alpha^c$, if the initial state $\vec{\sigma}(0)$ is close enough to a memory $\vec{\xi}^\mu$, the system will evolve until it will stop close or above the state $\vec{\sigma} \equiv \vec{\xi}^\mu$. However, if the system starts from a random configuration, and then with high probability nearly perpendicular to all patterns, it's not true in general that it will evolve to reach a certain memory. As described in Fig. 3.1 and in Chapter 3 in general, only for some choices of parameters the dynamics are typically attracted by a memory, in a phase we call *retrieval from random phase*. An example of a dynamic in this phase is presented in Fig. 1.2. Note that in the example $\alpha = 50/10000 \ll 1$. As it will be shown in Chapter 3, the regime in which the retrieval from random happens is with a sublinear P with respect to N, that in the $N \to \infty$ limit means $\alpha \to 0$.

Attractors and basins of attraction

The "basin of attraction" of a certain pattern ξ^μ is defined as the ensamble of states such that starting from them the system will converge to $m^\mu \approx \pm 1$. Although it is a simple notion to define, it can be quite difficult to be analyzed properly, because in principle we should know every starting state $\vec{\sigma}(0)$ that converges in that memory. A simple way to have an idea of the typical dimension of the basins given certain parameters is presented in Fig. 1.3: we simulate the final magnetization from a memory starting from random points at a specific magnetization m_{in}, and the range of m_{in} values from which it is possible to magnetize is a measure of the typical amplitude of basins of attraction. The resulting plot is called the *retrieval map*.



(a) **(b)**

Figure 1.3. Retrieval maps for two different α values, in which it is measured the final magnetization m^μ starting from an initial magnetization m^μ_{in}. The points in $m_{in} = 0$ corresponds to random initialization, the case presented in Fig. 1.2. (a) Basin of attraction at $\alpha = 0.10$, below α^c (b) Same, but with $\alpha = 0.15 > \alpha^c$. It's possible to observe that in the retrieval phase the basins are non-zero, meaning that starting from a certain finite distance from a memory typically it will be recalled. Instead, in the spin glass phase usually the memories are unstable, meaning that it's not possible to have a memory as an exit state, even if it coincides with the initial configuration.

During the present work, three conceptually different meaning of stability will be referred to:

- The *point stability* is the ability of a memory to be stable, that means that starting from it the system won't move. However, as it will be discussed later, it is possible that the system will do some moves, so in this definition it is tolerated that the system moves if it remains very close to the starting pattern. This kind of stability is the object of the study of the storage phase of the models in Chapter 2. In the rest of the work, the line in the (α_P, α_D) of the phase diagram that demarks the storage phase will be referred to as the *spinodal line* of point stability.
- A pattern is *locally stable* when it has a non-vanishing basin of attraction. The local stability is a sufficient condition for the point stability; the opposite isn't true. If almost all memories have local stability, the system is a model of an *associative memory*.
- A pattern is a *global attractor* if starting from random with high probability the starting state is inside the basin of a pattern, and so it will converge to $m \approx 1$. In particular the analysis in Chapter 3 will be focused on the parameters choices in which the configurations space is almost entirely occupied by basins of attractions of patterns.

Note that retrieval maps contain the information about the three different types of stabilities:

- Point stability occur when $m_{out}(m_{in} = 1) \approx 1$.
- Local stability is when $m_{out}(m_{in}) \approx 1$ for a finite range of $m_{in} < 1$.
- Global attractiveness when $m_{out}(m_{in} = 0) \approx 1$.

1.3 The modern Hopfield

The original Hopfield model has been a long studied neural network model and the starting point for a conceptual framework of the usage of statistical mechanics into the field of artificial neural networks. However, in this form presented until now the model has not the characteristics to be used or compared to modern artificial neural networks, being too limited in it's performances, as described in Sec. 1.1.2. In particular the presence of a critical value of capacity α^c if the number of patterns P is $O(N)$, makes the amount of memories that can be stored too limited for any real-world application.
In recent years new generalizations have proven to be capable of storing much more memories, such as a polynomial function of N [KH16] or even a exponential number [LM23] of them, changing completely their performances. In the first case [KH16], the Hamiltonian of the system is defined as

$$H = -\sum_{\mu}^{P} F\left(\sum_{i} \xi_i^{\mu} \sigma_i\right) \tag{1.25}$$

in which we choose

$$F(x) = \begin{cases} x^r & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{1.26}$$

This prescription is necessary in the odd r case, because without that the memories would not be attractors, being $x \to -\infty$ instead of $x = 0$ the minimization of the energy. In the case of even r this problem it's not present, then it's possible to define only

$$F(x) \coloneqq x^r \tag{1.27}$$

During this work it will be studied only the $r = 2$ and $r = 4$ cases to overcome the presence of this difference and to simplify the analytical computations, not exploring the intermediate case $r = 3$. The goal it's to observe what differs increasing the r value from 2 to a bigger value, leaving a more complete treatment of the odd cases to further works.
We observe that if $r = 2$ in Eq. 2.1, it is again the standard Hopfield but without the prescription $i \neq j$ in the energy. It's possible to show that if $r = 2$ the two definitions of the energy are equal apart from the diagonal terms, which sum up to an irrelevant constant. Instead if $r > 2$ there are non-constant terms and so it is necessary to be careful on the precise form used.
For a general r we have

$$E = -\sum_{\mu}^{P} \left(\sum_{i}^{N} \xi_i^\mu \sigma_i \right)^r = -\sum_{\mu} \sum_{i_1,\dots,i_r} \xi_{i_1}^\mu \cdots \xi_{i_r}^\mu \sigma_{i_1} \cdots \sigma_{i_r} = -\sum_{i_1,\dots,i_r} J_{i_1,\dots,i_r} \sigma_{i_1} \cdots \sigma_{i_r} \tag{1.28}$$

having defined the tensor of the couplings as $J_{i_1,\dots,i_r} \coloneqq \sum_{\mu}^{P} \xi_{i_1}^\mu \cdots \xi_{i_r}^\mu$.
If we have the couplings matrix J, it is simple to define the dynamical rule to update a single spin based on the value of the local field

$$\sigma_i(t+1) = \text{sign}\left(\sum_{i_2,\dots,i_r} J_{i,i_2,\dots,i_p} \sigma_{i_2}(t) \cdots \sigma_{i_r}(t) \right) \tag{1.29}$$

This definition scales poorly with the growing of the exponent r, because the J couplings tensor has to be created from the patterns $\vec{\xi^\mu}$ and has a dimension $\mathcal{Z}^{(N \times \cdots \times N(\text{r times}))}$. For example in the $r = 4$ case analyzed in this work to compute the tensor there are PN^4 operations to be done, making the tensor generation an operation completely impossible to accomplish with a normal computer cluster, even for very low sizes.
However, it is possible to define the update rule without computing the tensor J at all, using the modern formulation with the polynomial energy. In fact, for the Metropolis algorithm in this case, to decide the next move it is sufficient to know the value

$$\begin{aligned} \delta(t) =& E(\sigma_i(t) = +1) - E(\sigma_i(t) = -1) = \\ & \sum_{\mu=1}^{P} F\left(\xi_i^\mu + \sum_{j(\neq i)} \xi_j^\mu \sigma_j(t) \right) - F\left(-\xi_i^\mu + \sum_{j(\neq i)} \xi_j^\mu \sigma_j(t) \right) \end{aligned}$$

and then from Eq. 1.12

$$\sigma_i(t+1) = \text{sign}\left[\delta(t)\right] = \text{sign}\left[\sum_{\mu=1}^{P} F\left(\xi_i^\mu + \sum_{j(\neq i)} \xi_j^\mu \sigma_j(t)\right) - F\left(-\xi_i^\mu + \sum_{j(\neq i)} \xi_j^\mu \sigma_j(t)\right)\right] \tag{1.30}$$

With a trivial algorithm, one single update of the system that consists in updating once all N spins $\sigma_i(t+1)$ picked in a random order, has a cost of PN^2 operations, without difference for any r value. However, computing before the start of the dynamics the values

$$V^\mu(0) \coloneqq \sum_j \xi_j^\mu \sigma_j(0)$$

the single spin update algorithm becomes

$$\sigma_i(t+1) = \text{sign}\left[\sum_{\mu=1}^{P} F\left(\xi_i^\mu + V^\mu(t) - \xi_i^\mu \sigma_i(t)\right) - F\left(-\xi_i^\mu + V^\mu(t) - \xi_i^\mu \sigma_i(t)\right)\right]$$

$$V^\mu(t+1) = \begin{cases} V^\mu(t) & \text{if } \sigma_i(t+1) = \sigma_i(t) \\ V^\mu(t) + 2\xi_i^\mu \sigma_i(t+1) & \text{otherwise} \end{cases} \tag{1.31}$$

Both the moves have a cost of P operations, and then to update N spin with this algorithm the total cost is of PN operations, an acceptable cost to simulate big enough N sizes (for example $N \approx 10^4$ if $P \propto N$) on a typical computer cluster.
In this work we will study $r = 2$ and $r = 4$ networks, that are not exactly the same of the 2-spins and the 4-spins Hopfield models, that as we have seen have an excessive computational cost. However, in the next two sections we will compute the relation between the two models in the two cases. We will see that for $r = 2$ the difference is irrelevant from a point of view of dynamical behaviour, instead in the $r = 4$ there is a non-constant term which makes the two systems in principle different. However, the same results were obtained using the two different formulations in all phenomena analyzed. For this reason, in all the results presented we will refer using the r value both to r-spins models and polynomially-defined Hamiltonian with the exponent r. Moreover, through the relations between the two formulations, it is possible to use the energy-based algorithm to simulate also the 2-spins and the 4-spins models.

1.3.1 Relation between 2-spin model and $r = 2$ polynomial Hopfield

Defining $\mathcal{H}_{2,r}$ the Hamiltonian of the $r = 2$ energy-based system and $\mathcal{H}_{2,H}$ the one of 2-spins Hopfield, the only difference is that the second one has the prescriction $j \neq i$ in the sum. To find the relation between them

$$\begin{aligned} \mathcal{H}_{2,r} &= -\sum_\mu \left(\sum_j \xi_j^\mu \sigma_j\right)^2 = -\sum_\mu \sum_{i,j} \xi_i^\mu \xi_j^\mu \sigma_i \sigma_j = -2\sum_\mu \sum_{(i,j)} \xi_i^\mu \xi_j^\mu \sigma_i \sigma_j \\ &\quad - \sum_\mu \sum_i 1 = 2\mathcal{H}_{2,H} - NP \end{aligned} \tag{1.32}$$

It means that the 2-spin Hopfield model can be obtained from $r = 2$ polynomial Hopfield with the relation

$$\mathcal{H}_{2,H} = \frac{\mathcal{H}_{2,r} + NP}{2} \tag{1.33}$$

In particular, apart from additive and multiplicative constants $\mathcal{H}_{2,H} \propto \mathcal{H}_{2,r}$, meaning that a dynamical algorithm field-based or energy-based will always have the same probability of flipping each spin (1 or 0 in the zero-temperature scenario).

1.3.2 Relation between 4-spin model and $r = 4$ polynomial Hopfield

Defining $\mathcal{H}_{4,r}$ the Hamiltonian of the $r = 4$ energy-based system and $\mathcal{H}_{4,H}$ the 4-spins Hopfield model

$$\mathcal{H}_{4,r} = -\sum_{\mu}\left(\sum_{i}\xi_i^\mu\sigma_i\right)^4 = -\sum_{\mu}\sum_{i_1 i_2 i_3 i_4}\xi_{i_1}^\mu\xi_{i_2}^\mu\xi_{i_3}^\mu\xi_{i_4}^\mu\sigma_{i_1}\sigma_{i_2}\sigma_{i_3}\sigma_{i_4}$$

Calling $A \coloneqq \xi_{i_1}^\mu\xi_{i_2}^\mu\xi_{i_3}^\mu\xi_{i_4}^\mu\sigma_{i_1}\sigma_{i_2}\sigma_{i_3}\sigma_{i_4}$ it is possible to write

$$\mathcal{H}_{4,r} = -4!\sum_{\mu}\sum_{i_1<i_2<i_3<i_4} A - 6\sum_{\mu}\sum_{i_1=i_2(\neq i_3\neq i_4),i_3(\neq i_4),i_4} A - 3\sum_{\mu}\sum_{i_1=i_2(\neq i_3),i_3=i_4} A$$

$$- 4\sum_{\mu}\sum_{i_1=i_2=i_3(\neq i_4),i_4} A - \sum_{\mu}\sum_{i_1=i_2=i_3=i_4} A =$$

$$4!\mathcal{H}_{4,H} - 6\cdot 2N\sum_{\mu}\sum_{(i_3,i_4)}\xi_{i_3}^\mu\xi_{i_4}^\mu\sigma_{i_3}\sigma_{i_4} - N^2P - 4\cdot 2\sum_{\mu}\sum_{(i_3,i_4)}\xi_{i_3}^\mu\xi_{i_4}^\mu\sigma_{i_3}\sigma_{i_4} - NP$$

So, in the limit $N \gg 1$

$$\mathcal{H}_{4,H} = \frac{1}{4!}(\mathcal{H}_{4,r} + 12N\mathcal{H}_{2,H} + N^2P) \tag{1.34}$$

It means that the 4-spins Hamiltonian equals the polynomially defined $r = 4$ model, minus a correction equals to a 2-spin Hamiltonian. So, the quantitative results for the 2-spin model are not the same as for the $r = 2$ Hopfield. Anyway, through this relation, we can always use a energy-based algorithm to compute the 4-spin Hopfield model such as the one described in Eq. 1.31, with the computational advantage of the energy-based algorithm.

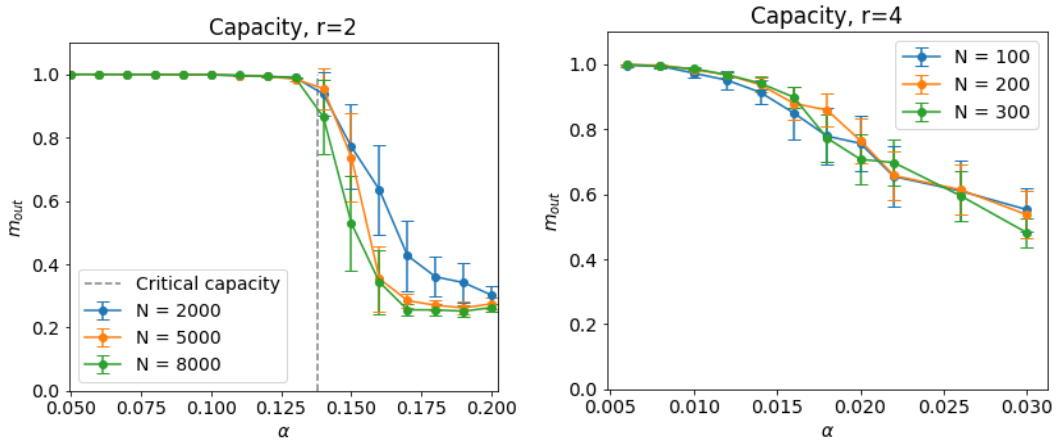
1.3.3 Modern Hopfield increases dramatically the storage capacity

The main difference, when r>2, is that the maximum capacity of the network becomes superlinear in N, at variance with the linear scaling with N of capacity for the $r = 2$ case. If we call P^{max} the critical capacity, that is the maximum number

of patterns that can be *stored* - namely that have *point stability*, as defined before - it grows like [KH16]:

$$P^{max} = \alpha^{max} N^{r-1} \tag{1.35}$$

where α^{max} is a coefficient that depends on the exponent r of the energy function $F(x)$. In particular, this result will be obtained again in Eq. 2.11.



(a) **(b)**

Figure 1.4. Comparison of the capacity in the generalized Hopfield model between $r = 2$ and r=4. The system is prepared upon a memory, it evolves freely, then the final magnetization is measured. A number of random patterns equal to $P = \alpha N^{r-1}$ are generated for each value of α. The storing capacity increases dramatically: in the standard Hopfield case (a), the critical capacity provided by replica computation (dotted line) means that with $N = 8000$ it is possible to store $P \approx 1100$ memories. In the modern Hopfield $r = 4$ case (b), due to superlinear behaviour, to have the possibility to simulate models that reach the critical capacity, it is necessary to decrease sharply the size of them. For example, with $N = 300$, we can see that at $\alpha = 0.01$ the patterns are locally stable, even if they are a huge number: $P = \alpha N^3 \approx 270000$.

1.4 The random features generalization

The second generalization that will be investigated in this work is related to patterns $\vec{\xi^{\mu}}$ generation [Méz17] [Neg+23]. The Hamiltonian is the same as in Eq. 1.5:

$$H = -\frac{1}{N} \sum_{\mu}^{P} \sum_{(i,j)}^{N} \xi_i^{\mu} \xi_j^{\mu} \sigma_i \sigma_j \tag{1.36}$$

In the standard model the patterns are chosen in a uncorrelated way, thus every value ξ_i^{μ} is chosen uniformely in $\{-1, +1\}$. Instead, as in this generalization, patterns are said to be correlated in a *combinatorial* way if, taken D *factors* (also referred to as *features* or *subpatterns*) $\vec{f^k} \in \{-1, +1\}^N$, with k=1,...,D, extracted in a random uniform way, patterns are generated with the rule

$$\xi_i^{\mu} = \text{sign}\left(\frac{1}{\sqrt{D}} \sum_{k}^{D} c_k^{\mu} f_i^k \right) \tag{1.37}$$

where c_k^μ are i.i.d. variables, for example standard Gaussian or uniformly distributed. The correlation between patterns is chosen as in Eq. 1.37 because we want to model the presence of an internal intrinsic representation of data, following the *hidden manifold model* [Gol+20]. We are creating a dataset of patterns $\vec{\xi}^\mu$ that are a projection in the configurations space $\{-1,+1\}^N$ from an *hidden manifold* of variables $\vec{f}^k$.
In this model two different kinds of α are defined:

1. α_P is defined from the number P of patterns: for example if it's considered a linear relation between P and N, $\alpha_P \coloneqq P/N$.

2. α_D is defined from the number D of factors, for example in the linear scenario $\alpha_D \coloneqq D/N$.

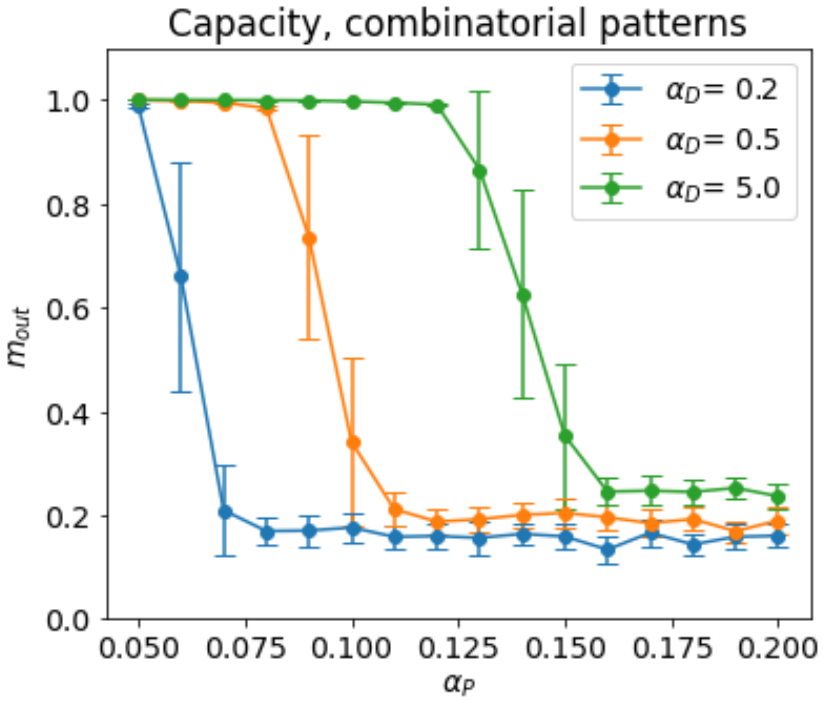


Figure 1.5. Capacity analysis with r=2, but with data correlated through D subpatterns, and N=10000. When α_D grows up, patterns $\vec{\xi}^\mu$ becomes less and less correlated due to the combinatorial construction in Eq. 1.37, then when $\alpha_D \longrightarrow \infty$ we have to obtain again the standard Hopfield capacity plot in Fig. 1.4 (a).

1.5 Storage, learning and generalization in the Hopfield model

In this paragraph it will be explained in which way the random-features Hopfield model is linked to the *storage, learning, generalization* concepts of modern machine learning, described in Sec. 1.1.1.
As we will see in the next section, with the signal-to-noise ratio (SNR) method it's

possible to show that in the Hopfield model the correlation between the energetic landscape and the stored patterns $\vec{\xi}^{\mu}$ results in the presence of minima directly upon them, or slightly distant. When P, the number of examples P, is bigger then a certain threshold, the patterns (and their neighborhoods) become unstable, and so a $T = 0$ dynamics will drive the state of the system away from them. We call *capacity* this threshold, and we call this task *storage*. From a computer science point of view, as described in the Sec. 1.1.1, the *exposure* of the model to the patterns is called *the training phase*, despite in this case it's not a dynamical process as in usual machine learning, being the memories inserted statically inside the coupling matrix J_{ij}. However, the model is exposed to this set of examples and then they can be viewed as a the *training set*.

In a random-features generated dataset the P patterns $\vec{\xi}^{\mu}$ are built from D factors $\vec{f}^{k}$. As we will see in the next sections, with some choice of parameters also the factors become point stable states of the dynamics (or even locally stable, so with a finite attraction basin). The ability to store and recall vectors that are not directly the $\vec{\xi}^{\mu}$, but some building blocks of them, it's called the *learning* in analogy with this definition in the context of machine learning, as described in 1.1.1. The interest in this phenomenon is clear in the context of the hidden manifold model, discussed in [Gol+20].

The last definition, the *generalization*, as already discussed, is about the possibility to use the learned feature to "recognize" new objects built from them. It's the most important ability of a model, and it's possible to state that in general the generalization is the main objective of a model construction and training.
We can imagine the data fed to a model as a cloud of points in a high dimensional space. In this context, from a point of view of the hidden manifold model, learning is the ability to reconstruct a useful representation of the hidden manifold inside the model using the examples of the training. Then, after the training, the interesting feature of a model is its capability to perform well on new and unseen example. In the context of Hopfield models, the generalization could be imagined as the ability to retrieve vectors $\vec{g}^{s}$ constructed starting from features $\vec{f}^{k}$, in the same way as patterns:

$$\vec{g}^{s} = \text{sign}\left(\frac{1}{\sqrt{D}}\sum_{k}^{D} c_k^s f_i^k\right) \tag{1.38}$$

but where now c_k^s are new randomly generated coefficients.
Despite the relevance of generalization capabilities, in this work we will focus on the basics of the models, with the aim of understanding the differences in the behaviour of the models depending on the J_{ij} definitions and the choice of parameters. An analysis of the generalization of the Hopfield models can be done relying on the methods and the knowledge obtained in the present work as discussed in the conclusions, and can be viewed as the natural continuation of this work.

Chapter 2

The SNR method and the study of capacity with random features

In this Chapter it will be applied the method of the signal-to-noise ratio (SNR) in order to have an analytical treatment of the capacity of the models studied throughout this work. The SNR method was used by John Hopfield itself [Hop82], and applied to the p-spin model by Elizabeth Gardner in the 80's [Gar87]. This method is used to obtain for which set of parameters patterns and factors show point stability, and it's characterized by simplicity and the possibility to easily interpret its results.
The point stability in the Hopfield model it's an important phenomena because it represents the first step to have the retrieval mechanism working, as described in Sec. 1.2.4. However, it's a necessary but not sufficient condition: if the interesting vectors have point stability but no basin of attraction, so they are not dynamically reached, the neural network will retrieve memories only if the initial configuration is already a memory, making the network useless.

2.1 The method

In the polynomial Hopfield network [KH16], with the Hamiltonian

$$H = -\sum_{\mu}^{P} F\left(\sum_{i} \xi_i^{\mu}\sigma_i\right) \tag{2.1}$$

using the fact that, if r is even, $F(x) = x^r$, if the system has a configuration $\vec{\sigma} \equiv \vec{\xi^{\mu}}$, we have that the energy of this initial state:

$$E_{in} = -\sum_{\nu=1}^{P} F\left(\sum_{j=1}^{N} \xi_j^{\nu}\xi_j^{\mu}\right) = -\sum_{\nu=1}^{P}\left(\xi_i^{\nu}\xi_i^{\mu} + \sum_{j(\neq i)}^{N} \xi_j^{\nu}\xi_j^{\mu}\right)^r$$

If we flip only a single spin, so if $\sigma_i = -\xi_i^{\mu}$:

$$E_{fin} = -\sum_{\nu=1}^{P}\left(-\xi_i^{\nu}\xi_i^{\mu} + \sum_{j(\neq i)}^{N} \xi_j^{\nu}\xi_j^{\mu}\right)^r$$

Then the difference of the energy between the configuration aligned with a pattern $\vec{\xi^{\mu}}$ and the one with one single spin flipped is

$$\Delta E = \sum_{\nu=1}^{P} \left(\xi_i^{\nu} \xi_i^{\mu} + \sum_{j(\neq i)}^{N} \xi_j^{\nu} \xi_j^{\mu} \right)^r - \sum_{\nu=1}^{P} \left(-\xi_i^{\nu} \xi_i^{\mu} + \sum_{j(\neq i)}^{N} \xi_j^{\nu} \xi_j^{\mu} \right)^r \tag{2.2}$$

If $\Delta E < 0$ it means that the dynamical Metropolis algorithm will flip the spin. To treat the randomness of the generated patterns, we compute the mean value and the standard deviation of ΔE. When $\nu \equiv \mu$, we have the signal

$$\langle \Delta E \rangle = N^r - (N-2)^r \approx 2rN^{r-1} \tag{2.3}$$

When $\nu \neq \mu$, we have the noise

$$\Delta E_{noise} = 2r(P-1)\xi_i^{\nu}\xi_i^{\mu} \left(\sum_{j(\neq i)}^{N} \xi_j^{\nu} \xi_j^{\mu} \right)^{r-1}$$

This quantity is a gaussian random variable due to Central Limit Theorem, with zero mean and a variance (in the limit of large N)

$$\Sigma^2 \approx 4r^2 \, (2r-3)!! \, (P-1) N^{r-1} \tag{2.4}$$

The configuration is unstable when $< \Delta E > + \Delta E_{noise} < 0$, so the probability that the single spin i is unstable in the configuration $\vec{\sigma} \equiv \vec{\xi^{\mu}}$ is

$$P[\Delta E < 0] = \int_{-\infty}^{0} \frac{dx}{\sqrt{2\pi\Sigma^2}} e^{-\frac{(x - <\Delta E>)^2}{2\Sigma^2}} = \int_{<\Delta E>}^{\infty} \frac{dx}{\sqrt{2\pi\Sigma^2}} e^{-\frac{x^2}{2\Sigma^2}} = \tag{2.5}$$

$$\int_{\frac{\langle \Delta E \rangle}{\Sigma}}^{\infty} Dx = \mathcal{H}\left(\frac{< \Delta E >}{\Sigma} \right)$$

where we have defined the integral function $\mathcal{H}(x) \coloneqq \int_x^{\infty} Dx$ and the Gaussian differential $Dx \coloneqq e^{-\frac{x^2}{2}} dx$. Now we have an expression of the probability that a single spin is unstable when the system is initialized upon a stored pattern $\vec{\xi^{\mu}}$.
In principle, if a single spin is unstable, not necessarily the system will move much from the starting pattern $\vec{\xi^{\mu}}$. Therefore we talk about *perfect storage* when all the spins are stable, instead we refer simply to *storage* when the system will remain highly correlated to the starting memory $\vec{\xi^{\mu}}$, despite some unstable spins. However, in both cases we have point stability, and so simple storage is enough, because of the idea that an associative memory is a system that takes a corrupted input and retrieves the correct input, and if the latter has a certain amount of error remaining we are still satisfied with the model.

Now, given a certain probability C that a single spin is unstable, we have from Eq. 2.5

$$\mathcal{H}\left(\frac{\langle\Delta E\rangle}{\Sigma}\right) = C \tag{2.6}$$

and substituting the Eq. 2.3 and 2.4 in the last equation we obtain

$$\mathcal{H}\left(\frac{2rN^{r-1}}{\sqrt{4r^2\,(2r-3)!!\,(P-1)N^{r-1}}}\right) = C \tag{2.7}$$

This equation can be solved numerically, or it is possible to analyze the solution expanding for $x \gg 1$ the function

$$\mathcal{H}(x) \approx \frac{1}{\sqrt{2\pi}x} e^{-\frac{x^2}{2}} \tag{2.8}$$

obtaining the approximate expression

$$C \approx \sqrt{\frac{(2r-3)!!P}{2\pi N^{r-1}}} e^{-\frac{N^{r-1}}{2P(2r-3)!!}} \tag{2.9}$$

From this equation we can obtain the maximum scaling in N of the number of stored patterns P to have a finite value of C: from Eq. 2.9, the relation between P and N has to be

$$P = \alpha N^{r-1} \tag{2.10}$$

At a critical value of C will correspond an amount of unstable spins such that typically the system initialized upon a memory will undergo a spin cascade that will drive away the system, making the retrieval impossible. For that critical C value it will correspond an α^c, with a corresponding maximum number of stored memories

$$P^{max} = \alpha^c N^{r-1} \tag{2.11}$$

(as previously anticipated in 1.35).
The SNR method is not capable of linking C with the probability that some unstable spins will generate a spin cascade, driving away the system from the initial memory $\vec{\xi^{\mu}}$. The critical value of C, and also α^c, can be obtained either analytically studying the thermodynamics, for example with the replica method, or numerically, simulating many systems and measuring from which α^{max} the memories are not locally stable anymore.
The principal limitation of the SNR method is that it can only predict point stability, but in doing so it has many advantages:

1. It's the simplest compared to others such as replica computation.

2. It catches even more easily the scaling with N of the storage capacity P^{max}, that is the main interesting behavior of modern Hopfield, due to its superlinear capacity.

3. Even though it's not possible to compute the critical value of C, we are not completely ignorant about it. We know from Eq. 2.7 that for the original Hopfield model ($r = 2$)

$$\mathcal{H}\left(\sqrt{\frac{N}{P}}\right) = C \Leftrightarrow \mathcal{H}\left(\sqrt{1/0.138)}\right) = C \tag{2.12}$$

due to the fact that in this case from replica computation we know that $\alpha^{max} \approx 0.138$. Solving numerically the equation 2.12, in standard Hopfield we have

$$C \approx 0.0036 \tag{2.13}$$

In principle C could depend in any way from the model, but as we will see in the models analyzed, this value of C is not so far from the numerical results in all different cases.

4. As we will see in the $r = 4$ patterns capacity, it's possible to analyze the terms a posteriori and to understand which of these have the main contributions in the signal and in the noise.

2.2 Random features $r = 2$ model

2.2.1 SNR analysis of point stability of factors

We will describe in this section a scheme to compute efficiently an analytical approximation of the spinodal line for the storage phase with the SNR method.
In the Random features model, the patterns are chosen as $\xi_i^\mu = \text{sign}(\frac{1}{\sqrt{D}} \sum_{k=1}^{D} c_k^\mu f_i^k)$. Then, if the system is in the configuration $\vec{\sigma} \equiv \vec{f^1}$, the difference in the energy caused by a single spin flip would be

$$\Delta E = \sum_\nu \left(\xi_i^\nu f_i^1 + \sum_{j(\neq i)} \xi_j^\nu f_j^1\right)^r - \left(-\xi_i^\nu f_i^1 + \sum_{j(\neq i)} \xi_j^\nu f_j^1\right)^r \tag{2.14}$$

If we approximate $\text{sign}(x) \approx x$,

$$\begin{aligned}\Delta E &= \sum_\nu \left(\sum_k c_k^\nu f_i^k f_i^1 + \sum_{j(\neq i)} \sum_{k'} c_{k'}^\nu f_j^{k'} f_j^1\right)^r - \left(-\sum_k c_k^\nu f_i^k f_i^1 + \sum_{j(\neq i)} \sum_{k'} c_{k'}^\nu f_j^{k'} f_j^1\right)^r \\ &\approx \sum_\nu 2r \sum_k c_k^\nu f_i^k f_i^1 \left(\sum_{j(\neq i)} \sum_{k'} c_{k'}^\nu f_j^{k'} f_j^1\right)^{r-1}\end{aligned} \tag{2.15}$$

Focusing to the $r = 2$ case

$$\Delta E \approx 4 \sum_\nu \sum_k c_k^\nu f_i^k f_i^1 \sum_{j(\neq i)} \sum_{k'} c_{k'}^\nu f_j^{k'} f_j^1 = 4 \sum_\nu \sum_{k,k'} \sum_{j(\neq i)} C_{kk'}^\nu F_i^{k1} F_j^{k'1} \tag{2.16}$$

where we have defined the tensors $C^{\nu}_{kk'} \coloneqq c^{\nu}_{k} c^{\nu}_{k'}$ and $F^{k1}_{i} \coloneqq f^{k}_{i} f^{1}_{i}$.
The next step is to divide every tensor in a diagonal and an anti-diagonal part:

$$C^{\nu}_{kk'} = D_{kk'} + A(C)^{\nu}_{kk'} \tag{2.17}$$

where the diagonal $D_{kk'}$ has lost the index ν because if $k = k'$, as a side effect $c^{\nu}_{k} c^{\nu}_{k'} = 1$. The same is also for indexes k, k', but these are left to remember that this happens only when $k = k'$, and this information will be relevant when all the sums will be computed.
With this notation,

$$\begin{aligned} F^{k1}_{i} C^{\nu}_{kk'} F^{k'1}_{j} &= (D_{kk'} + A(C)^{\nu}_{kk'})(D^{k1} + A(F)^{k1}_{i})(D^{k'1} + A(F)^{k'1}_{j}) \\ &= D^{k1,k'1}_{kk'} + D^{k1} A(C)^{\nu}_{kk'} A(F)^{k'1}_{j} + \\ &+ D_{kk'} A(F)^{k1,k'1}_{i,j} + D^{k'1} A(F)^{k1}_{i} A(C)^{\nu}_{kk'} + A(C)^{\nu}_{kk'} A(F)^{k1,k'1}_{i,j} \end{aligned} \tag{2.18}$$

In equation 2.18, in principle 8 terms should appear as a result. However, the cases with 2 diagonals and an anti-diagonal are zero because if two diagonal conditions are true, then the third condition follows from the first two: if $k = k', k = 1$, it implies that also $k' = 1$, and then $A(F)^{k'1}_{j}$ makes it impossible to meet all requirements.
Now, we have to compute the sums $\sum_{\nu} \sum_{k,k'} \sum_{j \neq i}$ over the 5 terms of 2.18. In particular, for every term the result has to be a combination of the parameters of the model N, P, D. To compute efficiently, it is possible to follow this 4-steps procedure:

1. Every couple of index in a diagonal term will cause one of the two to be changed for the other, erasing the sum over that index.

2. In every anti-diagonal term, if two or more couples of the same class of index (like k and k' or ν and μ in the next section) are present, it has to be treated manually because other simplifications not yet considered can occur. Anyway, in the 2.18 example this doesn't happen, meanwhile this step will be used in 2.23.

3. Every remaining anti-diagonal tensor will generate a factor $\sqrt{N}$, $\sqrt{P}$, $\sqrt{D}$ for each of its summed indexes j, ν, k or k' respectively. This happens as a consequence of the central limit theorem, because a sum of N, P or D random terms is a random Gaussian variable of order $\sqrt{N}$, $\sqrt{P}$, $\sqrt{D}$.

4. Every remaining index not yet summed and present within a diagonal tensor (we remind, only one term for every couple), or not present at all in a given term, generates a factor N, P or D depending on its nature (respectively, j, ν, k and k').

So, following these rules, and apart from irrelevant overall multiplicative constants

$$\Delta E = NP + \mathcal{N}(\sqrt{NPD}) + P\mathcal{N}(\sqrt{ND}) + N\mathcal{N}(\sqrt{PD}) + \mathcal{N}(\sqrt{NPD^2})$$

where with $\mathcal{N}(\sqrt{t})$ we refer to a random gaussian value of zero mean and standard deviation t.
Now we separate ΔE between the signal and the noise, and we leave only the dominant terms when $N \gg 1$:

$$\langle \Delta E \rangle = NP$$

$$\Sigma^2 = NP^2D + N^2PD + NPD^2 = NPD(P + N + D)$$

As we have seen in 2.6, given a probability of single spin instability C, the equation is

$$\mathcal{H}\left(\frac{\langle \Delta E \rangle}{\Sigma}\right) = C \Leftrightarrow \mathcal{H}\left(\frac{NP}{\sqrt{NPD(P+N+D)}}\right) = C \tag{2.19}$$

We can observe that right scaling for the parameters of the phase diagram to make the Eq. 2.19 not depending on N is

$$\begin{cases} \alpha_P \coloneqq P/N \\ \alpha_D \coloneqq D/N \end{cases} \tag{2.20}$$

which is compatible with numerical results in Fig. 2.1.

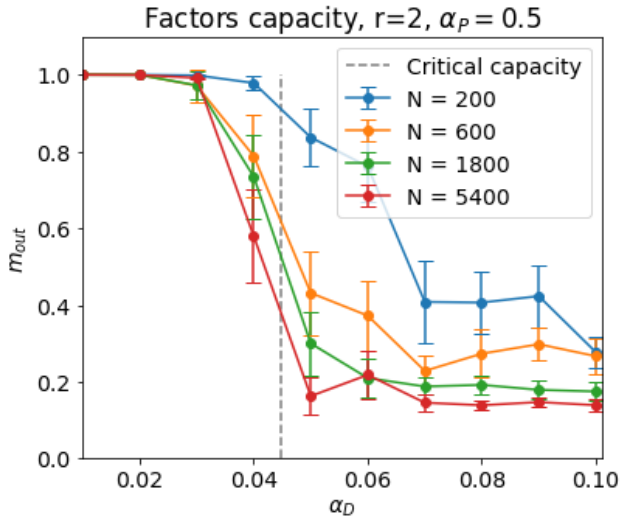
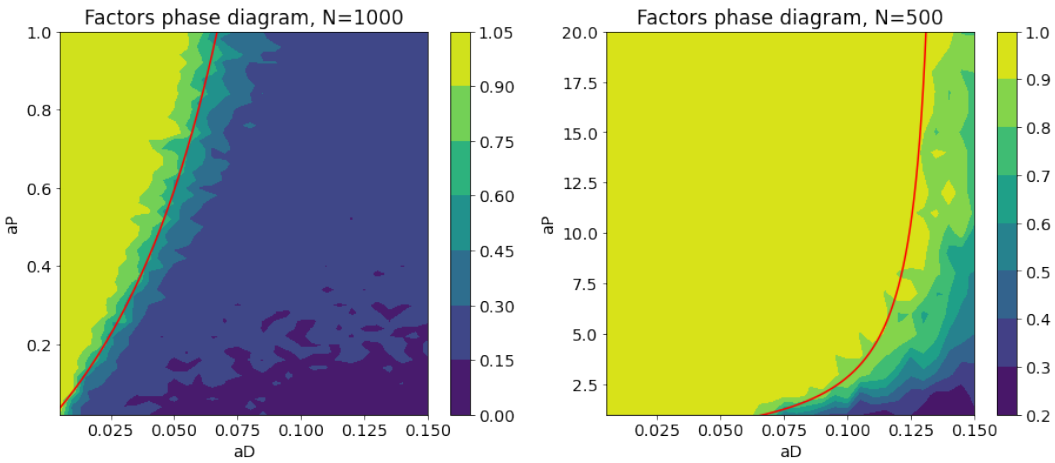


Figure 2.1. Capacity curves at different N values, with the linear scaling predicted with the SNR method. We can observe that using this scaling the curves overlap, confirming that it is a correct scaling to make the systems critical capacity not depending from N. Moreover, the curves are compatible with the critical capacity computed using Eq. 2.21 with the value $\alpha_P = 0.5$.

The equation for the single spin instability with probability C is

$$\mathcal{H}\left(\sqrt{\frac{a_P}{\alpha_D(\alpha_P + 1 + \alpha_D)}}\right) = C \tag{2.21}$$

The solution of this equation in (α_P, α_D) is an implicitly-defined curve, that is the spinodal line for factors point stability. In Fig. 2.2 it is plotted the solution of this equation, obtained with a recursive numerical method, compared to the numerical results. The analytical and the numerical predictions are in a good agreement, even though the C value used is the one of the original Hopfield model, which is the 2-spin uncorrelated patterns network.
The result is that in the upper-left part of the phase diagram the factors have point stability, and inside that area there is the retrieval phase of factors. In particular, as α_P increases, the number of factors $D = \alpha_D N$ that can be stored increases too. But, as it is suggest from the big α_P case ((b) in the figure), the maximum value of α_D can grow up only until a finite value α_D reached ideally when $\alpha_P \to \infty$, and this value seems to be similar to the $\alpha^c \approx 0.138$ of the original Hopfield model.



(a) **(b)**

Figure 2.2. Numerical simulations (in background) of many models for many choices of parameters (α_P, α_D), with superimposed the red line that is the numerical solution of Eq. 2.21, that is the analytical prediction of the spinodal line separating models with point stable factors from model without this capability. For numerical simulations, every image is the smoothing of $\approx 30 \times 30$ equally spaced points in a square grid. Every point consists in 3 different realizations of the random factors and patterns for the correspondent (α_P, α_D) parameters. For each realization, 10 simulations are carried out, in which the initial point is a different random factor f^k of the same model. Finally the colour of the point corresponds to the mean value of the final factor magnetization μ^k over all 30 simulations. In particular, the red curve corresponds to C=0.0036, that is approximately the correct value for the standard Hopfield model. Both in the small (a) and big (b) α_P cases, the red curve coincides almost perfectly to numerical simulations, even though the C value is the original Hopfield one.

Box 1: The procedure to apply SNR method to the random features Hopfield models

1. Define tensors grouping couples of variables.
2. Divide every tensor in a diagonal and an anti-diagonal part.
3. Do all the multiplications between tensors, erasing logically inconsistent terms (for example a term in which $k = k'$, $k = 1$, and $k' \neq 1$).
4. Every diagonal tensor cancels a sum over one of the summed indices in every couple, making it equal to the other of its couple.
5. Inside anti-diagonal terms, when other simplifications can occur they have to be manually treated.
6. The sum over an index of an anti-diagonal tensor is evaluated with Central Limit Theorem, giving factors $\mathcal{N}(\sqrt{N})$, $\mathcal{N}(\sqrt{P})$ or $\mathcal{N}(\sqrt{D})$ depending on the nature of the index.
7. Every index missing or present in a diagonal term and not yet summed is the counting of the same quantity repeated: if j is missing, a factor N; if ν is missing, a factor P; if one of k, k' is missing, a factor D.
8. Divide between signals and noises, define P and D with α_P and α_D parameters using a scaling that cancels N terms in the ratio between signals and noises, and finally choosing a value of the parameter C, the numerical solution of the equation 2.6 is the spinodal line.

2.2.2 SNR analysis of point stability of patterns

Placing the system upon a memory $\vec{\sigma} \equiv \vec{\xi^{\mu}}$ the difference in energy flipping a spin is

$$\Delta E = \sum_{\nu} 2r\xi_i^{\nu}\xi_i^{\mu}\left(\sum_{j(\neq i)}\xi_j^{\nu}\xi_j^{\mu}\right)^{r-1}$$

With r=2, using the combinatorial patterns definition, we neglect non-linearity to make the calculations simple, and then we do the approximation sign(x)≈x:

$$\Delta E = 4\sum_{\nu}\sum_{l} c_l^{\nu} f_i^{l} \sum_{k} c_k^{\mu} f_i^{k} \sum_{j(\neq i)}\sum_{l'} c_{l'}^{\nu} f_j^{l'} \sum_{k'} c_{k'}^{\mu} f_j^{k'}$$

Using the same notation as in the previous section, apart from irrelevant overall multiplicative constants,

$$\Delta E = \sum_{j(\neq i)}\sum_{\nu}\sum_{k,k'}\sum_{l,l'} F_i^{kl} C_{kk'}^{\mu} C_{ll'}^{\nu} F_j^{k'l'} \tag{2.22}$$

Now splitting the argument between diagonals and anti-diagonals

$$F_i^{kl} C_{kk'}^{\mu} C_{ll'}^{\nu} F_j^{k'l'} = (D^{kl} + A(F)_i^{kl})(D_{kk'} + A(C)_{kk'}^{\mu})(D_{ll'} + A(C)_{ll'}^{\nu})(D_{k'l'} + A(F)_j^{k'l'})$$

The result of this product is made of 16 terms. However, the 4 terms in which we take three diagonals and an anti-diagonal are equal to zero due to logical consistency. For example, $D^{kl} D_{kk'} D_{ll'} A(F)_j^{k'l'} = 0$ because the first three terms gives us $k = l = k' = l'$ and that's incompatible with the prescription of the A that $k' \neq l'$. The remaining 12 terms of Eq. 2.22 are

$$\begin{aligned}
&D_{kl,k'l'}^{kk',ll'} + D_{kk'}^{kl} A(C)_{ll'}^{\nu} A(F)_j^{k'l'} + D_{ll'}^{kl} A(C)_{kk'}^{\mu} A(F)_j^{k'l'} + D_{k'l'}^{kl} A(C)_{kk',ll'}^{\mu,\nu} + \\
&D^{kl} A(C)_{kk',ll'}^{\mu,\nu} A(F)_j^{k'l'} + D_{kk',ll'} A(F)_{i,j}^{kl,k'l'} + D_{kk',k'l'} A(F)_i^{kl} A(C)_{ll'}^{\nu} + \\
&D_{kk'} A(F)_{i,j}^{kl,k'l'} A(C)_{ll'}^{\nu} + D_{ll',k'l'} A(F)_i^{kl} A(C)_{kk'}^{\mu} + D_{ll'} A(F)_{i,j}^{kl,k'l'} A(F)_j^{k'l'} + \\
&D_{k'l'} A(C)_{kk',ll'}^{\mu,\nu} A(F)_i^{kl} + A(C)_{kk',ll'}^{\mu,\nu} A(F)_{i,j}^{kl,k'l'}
\end{aligned} \tag{2.23}$$

Computing the six sums $\sum_j \sum_\nu \sum_{k,k'} \sum_{l,l'}$, all the terms follows the same simple summation rules explained in the previous section.
However, the 4 terms including the tensor $A(C)_{kk',ll'}^{\mu,\nu}$ have to be carefully analyzed because the not summed index μ can be equal to ν, and in this case an index can disappear, changing the result over the six sums (the same cannot happen with $A(F)_{i,j}^{kl,k'l'}$, because $i \neq j$. Of this 4 terms, only the signal ND^2 is big enough to be meaningful in the thermodynamic limit.
So, overall, the signal is

$$\langle \Delta E \rangle = NPD + ND^2$$

and the noise

$$\Sigma^2 = NPD^2(NP + PD + ND + D^2)$$

We observe that, as for factors when $r = 2$, again a linear scaling definition for the parameters α_D and α_P is the correct choice to have signal and noise to be of the same order with N:

$$\begin{cases} P = \alpha_P N \\ D = \alpha_D N \end{cases} \tag{2.24}$$

This prediction is consistent with the numerical capacity curves presented in Fig. 2.3.
The equation for the spinodal line of the retrieval of patterns is

$$\mathcal{H}\left(\sqrt{\frac{(\alpha_P \alpha_D + \alpha_D^2)^2}{\alpha_P \alpha_D^2 (\alpha_P + \alpha_P \alpha_D + \alpha_D + \alpha_D^2)}} \right) = C \tag{2.25}$$

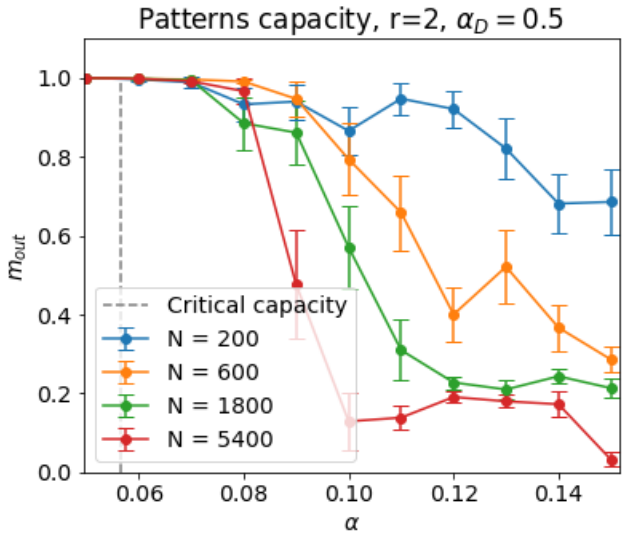
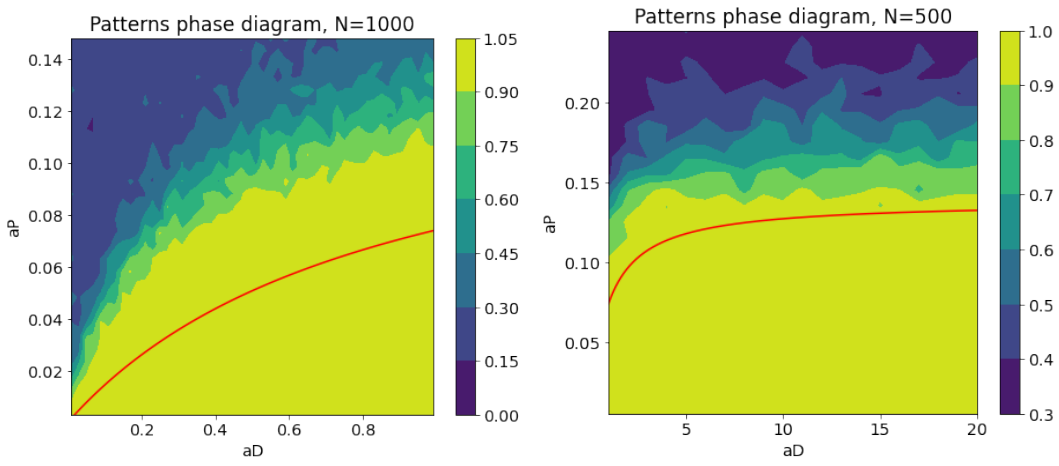


Figure 2.3. Capacity curves at different N values, using the linear scaling predicted with the SNR method. In this scaling the curves overlap and are compatible with the critical capacity computed using Eq. 2.25 with the value $\alpha_D = 0.5$. As in the phase diagrams of Fig. 2.4 the SNR prediction underestimate the numerical critical capacity.



(a) **(b)**

Figure 2.4. Numerical simulations (in background) of many models for many choices of parameters (α_P, α_D), with superimposed the red line that is the numerical solution of Eq. 2.25, that is the analytical prediction of the spinodal line separating models with point stable patterns from model without this capability. The numerical simulations are performed and represented as described in Fig. 2.2. The numerical and analytical results are in a good agreement, and the overall behaviour of the factors point stability phase is similar to the one of the $r = 2$ case. In particular, the retrieval phase of factors in this model is a subset of the yellow area.

In Fig. 2.4 it is presented the comparison between the solution of this equation that is the pattern point stability spinodal line, and the numerical results. Again

the two predictions are compatible, even if this time they are in a lesser agreement. To have patterns point stability, the area of the phase diagram is the lower-right part, where there are many factors and less patterns, the opposite of the factor case. So, the retrieval phase for patterns is inside this area. Also in this case as factors increases (and so α_D) more patterns can be stored, and again as suggested by the big α_D graph this can continue up to an α_P^c beyond which patterns are not point stable anymore, ideally reached when $\alpha_D \to \infty$. Also in this case that critical value seems to be not so far from $\alpha^c \approx 0.138$ of the original Hopfield model.

2.3 Random features $r = 4$ model

2.3.1 SNR analysis of point stability of factors

Following the SNR procedure with the scheme presented in Box 2.2.1, we start from

$$\Delta E = \sum_{\nu} \sum_{[i_2,i_3,<i_4](\neq i)} \sum_{l,l_2,l_3,l_4} C^{\nu}_{l_1,l_2,l_3,l_4} F_i^{l\,1} F_{i_2}^{l_2 1} F_{i_3}^{l_3 1} F_{i_4}^{l_4 1} \tag{2.26}$$

where $\sum_{[i_2,i_3,i_4](\neq i)}$ is the sum over all i_2, i_3, i_4 different from i.
We work out the C term anf the F terms separately, splitting them as usual between diagonal tensors and anti-diagonals

$$\begin{aligned} C^{\nu}_{l,l_2,l_3,l_4} =& D_{l_1,l_2,l_3,l_4} + 3D_{l,l_2,l_3} A(C)^{\nu}_{l_4} + D_{l_2,l_3,l_4} A(C)^{\nu}_{l} + 3D_{l\,l_2,l_3 l_4} + 3D_{l\,l_2} A(C)^{\nu}_{l_3 l_4} + \\ &+ 3D_{l_3 l_4} A(C)^{\nu}_{l\,l_2} + A(C)^{\nu}_{l\,l_2 l_3 l_4} \end{aligned} \tag{2.27}$$

We repeat the operation for

$$\begin{aligned} F_i^{l\,1} F_{i_2}^{l_2 1} F_{i_3}^{l_3 1} F_{i_4}^{l_4 1} =& D_{1l\,l_2 l_3 l_4} + 3D_{1l\,l_2 l_3} A(F)^{l_4 1}_{i_4} + D_{1 l_2 l_3 l_4} A(F)^{l\,1}_{i} + 3D_{1l\,l_2} A(F)^{l_3 1, l_4 1}_{i_3, i_4} + \\ &+ 3D_{1 l_3 l_4} A(F)^{l\,1, l_2 1}_{i, i_2} + 3D_{1 l_4} A(F)^{l\,1, l_2 1, l_3 1}_{i, i_2, i_3} + D_{1l} A(F)^{l_2 1, l_3 1, l_4 1}_{i_2, i_3, i_4} + \\ &+ A(F)^{l\,1, l_2 1, l_3 1, l_4 1}_{i, i_2, i_3, i_4} \end{aligned} \tag{2.28}$$

Multiplying the two terms in Eq. 2.27 and 2.28, many combinations make logically inconsistent and then zero-contributing terms. Computing the 8 sums of Eq. 2.26 over the remaining terms and using the rules described in 2.2.1, the only relevant terms in the thermodynamic limit $N \to \infty$ are

$$\begin{aligned} \Delta E =& PN^3 + N^3 \mathcal{N}(\sqrt{PD}) + 3PN^2 \mathcal{N}(\sqrt{ND}) + 3N^2 \mathcal{N}(\sqrt{NPD^2}) + \\ &+ 6N \mathcal{N}(\sqrt{N^2 PD^3}) + \mathcal{N}(\sqrt{N^3 PD^4}) \end{aligned}$$

Then

$$\begin{cases} < \Delta E >= PN^3 \\ \Sigma^2 = N^6 PD + 9N^5 P^2 D + 9N^5 PD^2 + 36N^4 PD^3 + N^3 PD^4 \end{cases}$$

Then the equation for the spinodal line is

$$\mathcal{H}\left(\sqrt{\frac{PN^3}{D(N^3+9N^2P+9N^2D+36ND^2+D^3}}\right)=C \tag{2.29}$$

Again, we have to choose a linear scaling

$$\begin{cases} P=\alpha_P N \\ D=\alpha_D N \end{cases} \tag{2.30}$$

as in the previous cases to have the argument of the equation 2.29 not depending on the size N, and this is consistent with numerical results in Fig. 2.5.

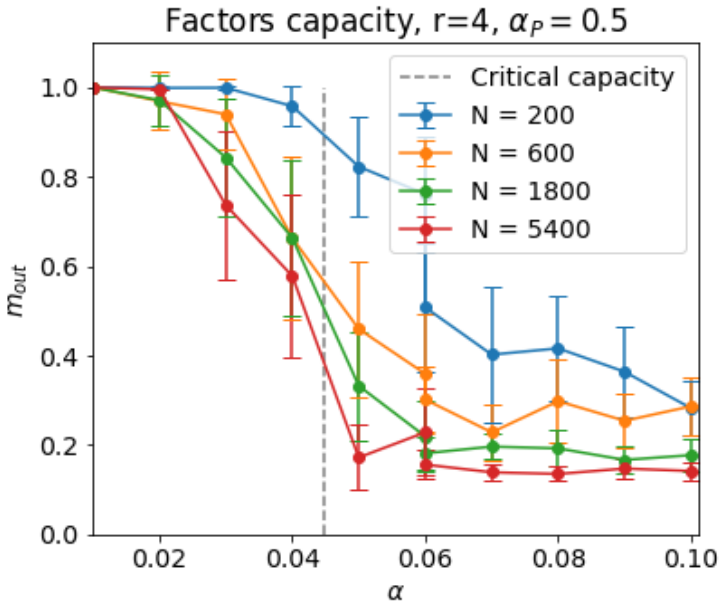


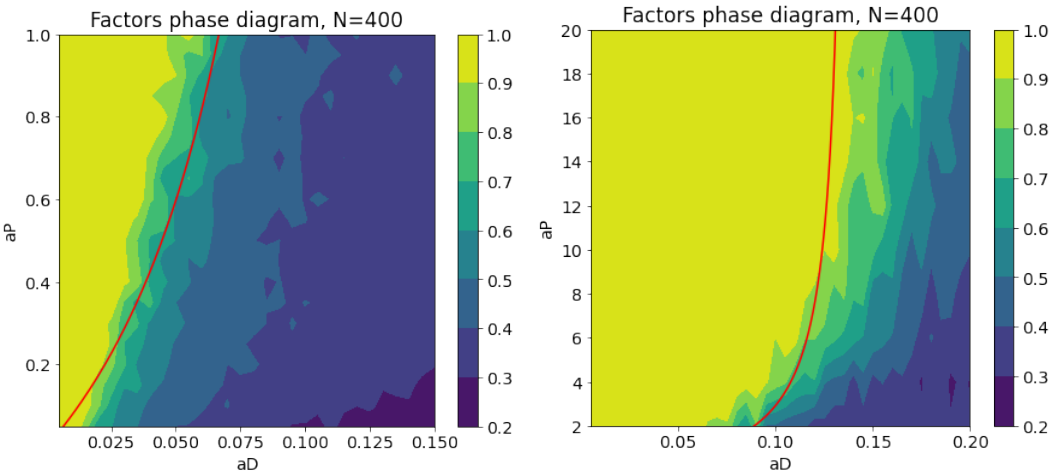
Figure 2.5. Capacity curves for different N values, with the linear scaling predicted with the SNR method. We can observe that using this scaling the curves overlap, and the predicted critical capacity computed using Eq. 2.21 with the value $\alpha_P = 0.5$ is compatible with numerical results.

With this scaling we have

$$\mathcal{H}\left(\sqrt{\frac{\alpha_P}{\alpha_D(1+9\alpha_P+9\alpha_D+36\alpha_D^2+\alpha_D^3)}}\right)=C \tag{2.31}$$

As in the previous cases, the solution of this equation is an implicitly defined curve which coincides with the prediction of the SNR method of the spinodal line which includes within it the factors point stability phase, and it is superimposed to numerical results in Fig. 2.6.
The result is similar to the previous case of factor point stability with r=2, being the factor retrieval phase in an area in the upper left of the phase diagram, and

also the critical value α_D^c when $\alpha_P \to \infty$ seems again to be similar to the original Hopfield model $\alpha^c \approx 0.138$.



(a) **(b)**

Figure 2.6. Numerical simulations (in background) of many models for many choices of parameters (α_P, α_D), with superimposed the red line that is the numerical solution of Eq. 2.31, that is the analytical prediction of the spinodal line separating models with point stable factors from model without this capability. The numerical simulations are performed and represented as described in Fig. 2.2. (a) Low α_P values. (b) High α_P values. We observe that in both regimes the analytical predictions made with SNR method are in good agreement with the numerical results.

2.3.2 SNR analysis of point stability of patterns

We use again the same algorithm to make easier the application of the SNR method. If the system is placed upon a pattern $\vec{\xi^\mu}$:

$$\Delta E = \sum_\nu \xi_i^\nu \xi_i^\mu \left(\sum_{j(\neq i)} \xi_j^\nu \xi_j^\mu \right)^{r-1} = 4 \sum_\nu \left(\xi_i^\nu \xi_i^\mu \sum_{[i_2,i_3,i_4](\neq i)} \xi_{i_2}^\nu \xi_{i_2}^\mu \xi_{i_3}^\nu \xi_{i_3}^\mu \xi_{i_4}^\nu \xi_{i_4}^\mu \right)$$

Where $\sum_{[i_2,i_3,i_4](\neq i)}$ is the sum over all i_2, i_3, i_4 different from i. Now, using the definition of patterns in the combinatorial Hopfield model 1.37 and the approximation $\text{sign}(x) \approx x$

$$\Delta E = \sum_{[i_2,i_3,i_4](\neq i)} \sum_\nu \sum_{l,l_2,l_3,l_4} \sum_{k,k_2,k_3,k_4} c_l^\nu c_{l_2}^\nu c_{l_3}^\nu c_{l_4}^\nu c_k^\mu c_{k_2}^\mu c_{k_3}^\mu c_{k_4}^\mu f_i^l f_{i_2}^{l_2} f_{i_3}^{l_3} f_{i_4}^{l_4} f_i^k f_{i_2}^{k_2} f_{i_3}^{k_3} f_{i_4}^{k_4}$$

Calling

$$\begin{cases} C_{l,l_2,l_3,l_4}^\nu := c_l^\nu c_{l_2}^\nu c_{l_3}^\nu c_{l_4}^\nu \\ C_{k,k_2,k_3,k_4}^\mu := c_k^\mu c_{k_2}^\mu c_{k_3}^\mu c_{k_4}^\mu \end{cases}$$

and the usual terms $F_i^{l\,k} := f_i^l f_i^k$ and the same for the other three couples of f terms, the sum become

$$\Delta E = \sum_{[i_2,i_3,i_4](\neq i)} \sum_{\nu} \sum_{l,l_2,l_3,l_4} \sum_{k,k_2,k_3,k_4} C^{\nu}_{l,l_2,l_3,l_4} C^{\mu}_{k,k_2,k_3,k_4} F_i^{l\,k} F_{i_2}^{l_2 k_2} F_{i_3}^{l_3 k_3} F_{i_4}^{l_4 k_4} \tag{2.32}$$

Then, as previously described, every term is divided in a diagonal and an antidiagonal part, giving 256 contributions, most of which in principle can undergo further simplifications and have to be checked one by one for their logical consistency. A reduction of terms can be obtained observing that some of them are a priori subdominant. Using this observation, the remaining terms, for every tensor, are

- In the first C,

$$C^{\nu}_{l,l_2,l_3,l_4} \approx 3D_{l\,l_2,l_3l_4} + 3D_{l\,l_2} A(C)^{\nu}_{l_3l_4} + 3D_{l_3l_4} A(C)^{\nu}_{l\,l_2} + A(C)^{\nu}_{l\,l_2l_3l_4}$$

- In the second one

$$C^{\mu}_{k,k_2,k_3,k_4} \approx 3D_{k\,k_2,k_3k_4} + 3D_{k\,k_2} A(C)^{\mu}_{k_3k_4} + 3D_{k_3k_4} A(C)^{\mu}_{k\,k_2} + A(C)^{\mu}_{k\,k_2k_3k_4}$$

- For the product of factors

$$F_i^{l\,k} F_i^{l_2k_2} F_i^{l_3k_3} F_i^{l_4k_4} \approx D^{l\,k,l_2k_2,l_3k_3,l_4k_4} + D^{l_2k_2,l_3k_3,l_4k_4} A(F)_i^{l\,k} +$$

$$+ 3D^{l_3k_3,l_4k_4} A(F)^{l\,k,l_4k_4}_{i,i_4} + 3D^{l_4k_4} A(F)^{l\,k,l_2k_2,l_3k_3}_{i,i_2,i_3} A(F)^{l\,k,l_2k_2,l_3k_3,l_4k_4}_{i,i_2,i_3,i_4}$$

Thanks to this simplification, the number of terms decreases from 256 to 80, but again much more than in the previous models analyzed. Combining all terms in 2.32 will lead to the function $F(\alpha_P, \alpha_D)$ for this model, and the spinodal line will be the solution of the equation

$$\mathcal{H}(F(\alpha_P, \alpha_D)) = C$$

However, without computing all 80 terms and obtaining the spinodal line, it is possible to obtain other interesting informations about the system, in particular the correct scaling to have an equation of spinodal lines that doesn't depend on N. First, it is possible to obtain easily two signal terms:

- Multiplying all diagonal terms when $\nu \neq \mu$,

$$3D_{l\,l_2,l_3l_4} D_{k\,k_2,k_3k_4} D^{l\,k,l_2k_2,l_3k_3,l_4k_4} \to 3N^3PD^2$$

- When $\nu = \mu$, from the term

$$A(C)^{\nu}_{l\,l_2l_3l_4} A(C)^{\mu}_{k\,k_2k_3k_4} D^{l\,k,l_2k_2,l_3k_3,l_4k_4} \to N^3D^4$$

because between the two anti-diagonal tensors a cross-simplification takes place.

It's possible to show easily that they are "maximal" terms, namely no other signal terms have bigger scalings in all of N, P, D quantities. This is true because from 2.32, the sums could in principle create a term of a maximal order in all variables of N^3PD^8. However, if $\nu \neq \mu$ (necessary condition to have the P factor), cross simplifications between $A(C)$-like tensors cannot happen, making the resulting term a noise. Then, the first term of both the C tensors C^{ν}_{l,l_2,l_3,l_4} and C^{μ}_{k,k_2,k_3,k_4} have to be taken, erasing 4 powers of D. For this reason, for the factors tensor the diagonal term have to be taken in order to maintain the N^3 scaling (the only other term that give rise to this scaling have only a single A term, impossible to simplify by itself), canceling another D^2 factor. Then, no signal can have a bigger D power (and N^3P) than the term N^3PD^2.
For the N^3D^4 signal, it comes from the case $\nu = \mu$, and it is possible to repeat an analogous argumentation as for the previous signal, and then for similar reasons it cannot have a bigger D scaling.
Now, a noise term can be found from

$$3D_{l\,l_2,l_3l_4}D_{k_3k_4}A(C)^{\mu}_{k\,k_2}D^{l_2k_2,l_3k_3,l_4k_4}A(F)^{l\,k}_{i} \to 3N^3PD\mathcal{N}(\sqrt{D^2}) = \mathcal{N}(\sqrt{9N^6P^2D^4})$$

It is precisely the term $3N^6PD^2$ squared, and we are sure that it is of the maximal scaling, because if existed a noise with one of its exponent bigger, the total noise would be always bigger than the signal, making the point stability impossible, in contrast with numerical results.
Given only this three terms, we know that there will be an equation of the form

$$\mathcal{H}\left(\sqrt{\frac{(3N^3PD^2 + N^3D^4 + \text{terms})^2}{9N^6P^2D^4 + \text{terms}}}\right) = C \tag{2.33}$$

Using the usual linear scaling 2.20 the signal would be bigger than the noise of a factor N^2, making the equation depend on the size of the system N.
For the ratio

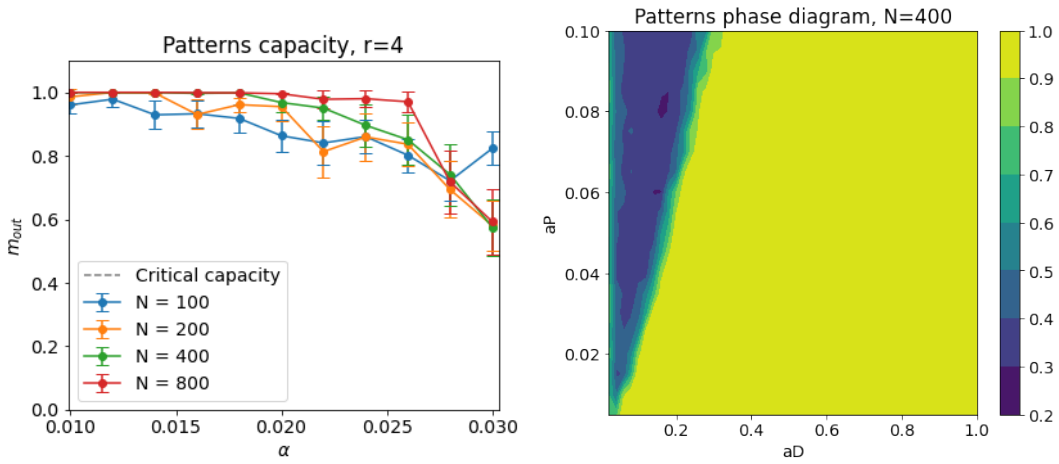
$$\frac{(N^3D^4)^2}{N^6P^2D^4} \tag{2.34}$$

to have a result not depending on N it is necessary that the scaling with N of P is double the scaling of D. A possible choice in this direction to make numerator and denominator of the same order in N is a quadratic scaling

$$\begin{cases} P = \alpha_P N^2 \\ D = \alpha_D N \end{cases} \tag{2.35}$$

making the retrieval of patterns with $r = 4$ a superlinear phenomenon also in the random features model. To state that it is the only possible scaling that makes the spinodal line not depending on N it is necessary to obtain the complete SNR method result. Other exponents have been tried numerically, but no different scaling was found. It is interesting that the point stability happens with an exponent $\beta = 2$, that it less than the cubic scaling of uncorrelated polynomial Hopfield with r=4, $\beta = 3$, discussed in Fig. 1.4.
So, the important result obtained is that one effect of having correlation between patterns (in a combinatorial way, as for the random features model) is that still it

makes it possible to have point stability of patterns, but it reduces the scaling of the number of memories that it is possible to store. The numerical results linked to the discussion above are presented in Fig. 2.7.



(a) (b)

Figure 2.7. In this case only numerical result are presented, being the analytical prediction of the spinodal line too arduous to be performed without further simplifications. (a) Capacity curves for different sizes N overlap using the quadratic scaling provided in Eq. 2.35, $\alpha_D = 0.1$. (b) Numerical simulations (in background) of many models for many choices of parameters (α_P, α_D), performed and represented as described in Fig. 2.2.

Chapter 3

Analysis of dynamics and energetic landscape of the models

In this Chapter we present a purely numerical exploratory analysis of some characteristics of the dynamics of the models studied throughout this work, in particular the possibility to have retrieval of memories starting from a random initial configuration. The reason behind this numerical analysis is that studying both phenomena of storage and retrieval from random we have a more complete vision on the behaviour of the models considered, obtaining the different regimes in which different behaviours take place.
In the case of convergence from a random initialization, so if the system typically converges to memories even though no information is given about them (random initialization), basins of attraction of memories are so big that they occupy almost all the configurations space. We expect that being in this regime in which memories are *global attractors* is also a sufficient condition for local stability, and therefore the system works also as an associative memory. However we have not proved that if the systems converges from random to some patterns then all $\vec{\xi}^{\mu}$ are locally stable, because non-trivial phenomena can happen. For example it is possible that only one or a few of them occupy all the configurations space, leaving all other memories with only point stability or no stability at all. So, to assess rigorously that there is a real memory retrieval mechanism in the models studied, a more detailed study of basins of attraction is needed. Anyway, both in the work done and in the literature, there are no suggestions or examples that is present this phenomenon in which a few memories occupy almost all configurations space in the retrieval from random phase. Instead, the usual behavior seen in Hopfield models is that if the system is in the retrieval phase, almost all memories have a non zero basin of attraction.
There is a plethora of phenomena occurring and quantities to be studied during dynamics, but we will focus only on the two quantities defined until now: the energy of the system and the magnetization of patterns or factors.
In the previous Chapter, with the analysis of capacity of different generalized Hopfield models, we have demonstrated that some of them have superlinear storing capabilities, also in the correlated patterns scenario. The aim of this section is to

show that increasing the the exponent r of the Hamiltonian from 2 to 4 results in an increase of the scaling with N of the number of memories not only stored, but also dynamically reachable from a random initialization. Remarkably, the scaling grows from underlinear to superlinear. As we will see, this happens both in the uncorrelated memories and in the random features case.

3.1 The scaling relations for retrieval from random

In this section for the two models investigated, namely $r = 2$ and $r = 4$ random features Hopfield, both for patterns and factors it is produced a magnetization vs capacity graph like the one described in Fig. 1.4. The difference is that now we focus on the *global attraction* property, and then unlike in that case in which the system was initialized over a memory, now the starting state $\vec{\sigma}_0$ is a random point in the configurations space, and the magnetization to a patterns is defined as *retrieval from random*. The number of patterns that can be stored without disrupting the ability of the model to perform retrieval from random is defined as the *capacity from random*. For this analysis the scaling exponents are defined as

$$P = \alpha_{P,\beta_P} N^{\beta_P} \tag{3.1}$$

$$D = \alpha_{D,\beta_D} N^{\beta_D} \tag{3.2}$$

and the main objective of this section is to find for each model the (β_P, β_D) exponents, being the parameters that regulates how many memories can be stored without compromising the ability of the system to perform retrieval from random.
As already discussed, to have *global attraction* there must be first the *point stability*, then it is expected that the *retrieval from random phase* is contained in the retrieval phase. For this reason we expect the scaling exponents β_P and β_D to be in each model equal or lower with respect to the values found in Chapter 2 for point stability.
The numerical results presented in Fig. 3.1, 3.2 and 3.3, are constructed simulating dynamics starting from a random initialization, and evaluating the mean final magnetization m_{out} of the system (or μ_{out} for the feature convergence) for different choices of the parameter α_{P,β_P} for patterns retrieval (α_{D,β_D} for factors retrieval).

Uncorrelated patterns Hopfield, $r = 2$ and $r = 4$

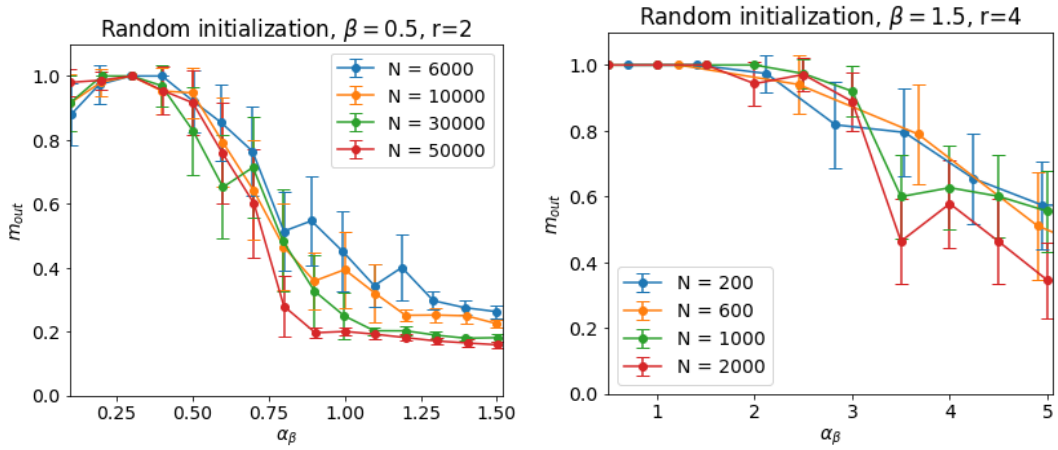
In Fig. 3.1 it is presented the numerical result. For each N, many systems for different P values have been initialized in a random configuration and the magnetization of the final state m_{out} has been computed. Then, for different values of the exponent β_P, namely for different relations of the form $P = \alpha_{P,\beta_P} N^{\beta_P}$, it was plotted for different N values the curve m_{out} vs α_{P,β_P}, that we will call *capacity from random curves*. In the end it was selected the exponent that made the capacity from random curves at different N overlapping in the best way. In particular, only the exponents $0.5, 1.0, 1.5, 2.0, 2.5, 3.0$ have been tried for all models, both in the uncorrelated and in the random features case. In all of them the curves were overlapping only for one exponent, meanwhile for all other exponents the curves did not overlap at all, thus it was not needed a sophisticated method to choose the exponent more compatible

to the results.
The scalings

$$P = \alpha_{P,\frac{1}{2}} N^{\frac{1}{2}} \quad \text{for } r = 2 \tag{3.3}$$

$$P = \alpha_{P,\frac{3}{2}} N^{\frac{3}{2}} \quad \text{for } r = 4 \tag{3.4}$$

make the numerical curves at different N overlapping at best. For this reason, we conclude that the those are the scalings compatible with numerical results.
We observe that this exponents are lower than in the point stability case analysis of the previous section, in which $P = O(N)$ for $r = 2$ and $P = O(N^3)$ for $r = 4$. However, it is remarkable that increasing from $r = 2$ to $r = 4$ the scaling grows from sublinear to superlinear.



(a) **(b)**

Figure 3.1. Capacity from random curves in the uncorrelated patterns Hopfield model. (a) $r = 2$ model, that is the original Hopfield. (b) $r = 4$ model. We can observe that in the first case the correct scaling with N of the capacity from random to make the curves overlap is $\beta = 0.5$ and then sublinear, instead with $r = 4$ the capacity from random grows up to $\beta = 1.5$, thus becoming superlinear.

Random features Hopfield, analysis of patterns, $r = 2$ and $r = 4$

In the random features case, for patterns retrieval, from the numerical analysis in Fig. 3.2 it seems that the scaling for each r doesn't change from the uncorrelated scenario.

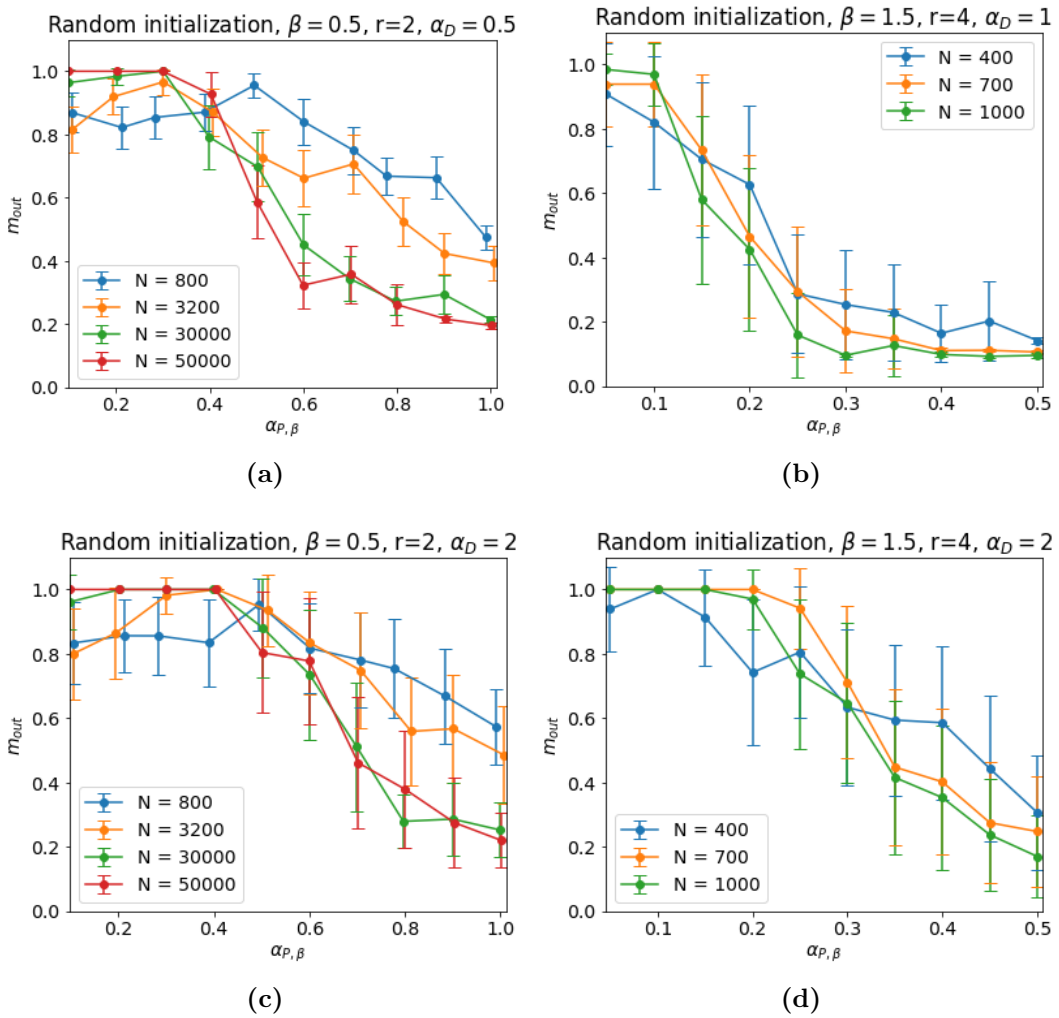


Figure 3.2. Curves of capacity from random of patterns in the random features Hopfield model, (a) with r=2, (b) with r=4. The scalings seems to be the same as in the uncorrelated case. Another effect is that increasing α_D also the value of the critical capacity from random grows, as for normal capacity in the SNR analysis.

In fact, also in this case as for the uncorrelated Hopfield model we obtain

$$P = \alpha_{P,\frac{1}{2}} N^{\frac{1}{2}} \quad \text{for } r = 2 \tag{3.5}$$

$$P = \alpha_{P,\frac{3}{2}} N^{\frac{3}{2}} \quad \text{for } r = 4 \tag{3.6}$$

both of which with a D scaling of $D = \alpha_{D,1} N$. Other exponent pairs (β_P, β_P) were also tried, but in no other case the curves overlap.
An interesting observation is that for $r = 4$, the scaling of P for point stability of patterns, in Chapter 2 were found to be $\beta_P = 3$ in the uncorrelated Hopfield, and the lower value $\beta_P = 2$ in the random features case. Instead, now, for the phenomenon of retrieval from random, the scalings in the two cases seems to be the same.
At qualitative level, another phenomenon observed is that, as for capacity and point

stability analysis of the previous section, increasing the parameter $\alpha_{D,1}$ (and so the number of factors), it increases also the critical value α^c_{P,β_P}, after which it's not possible retrieval from random. Therefore, more factors imply less correlation between patterns, and then the retrieval mechanism brakes down later, also when the system has a random initialization.

Random features Hopfield, factors analysis, $r = 2$ and $r = 4$

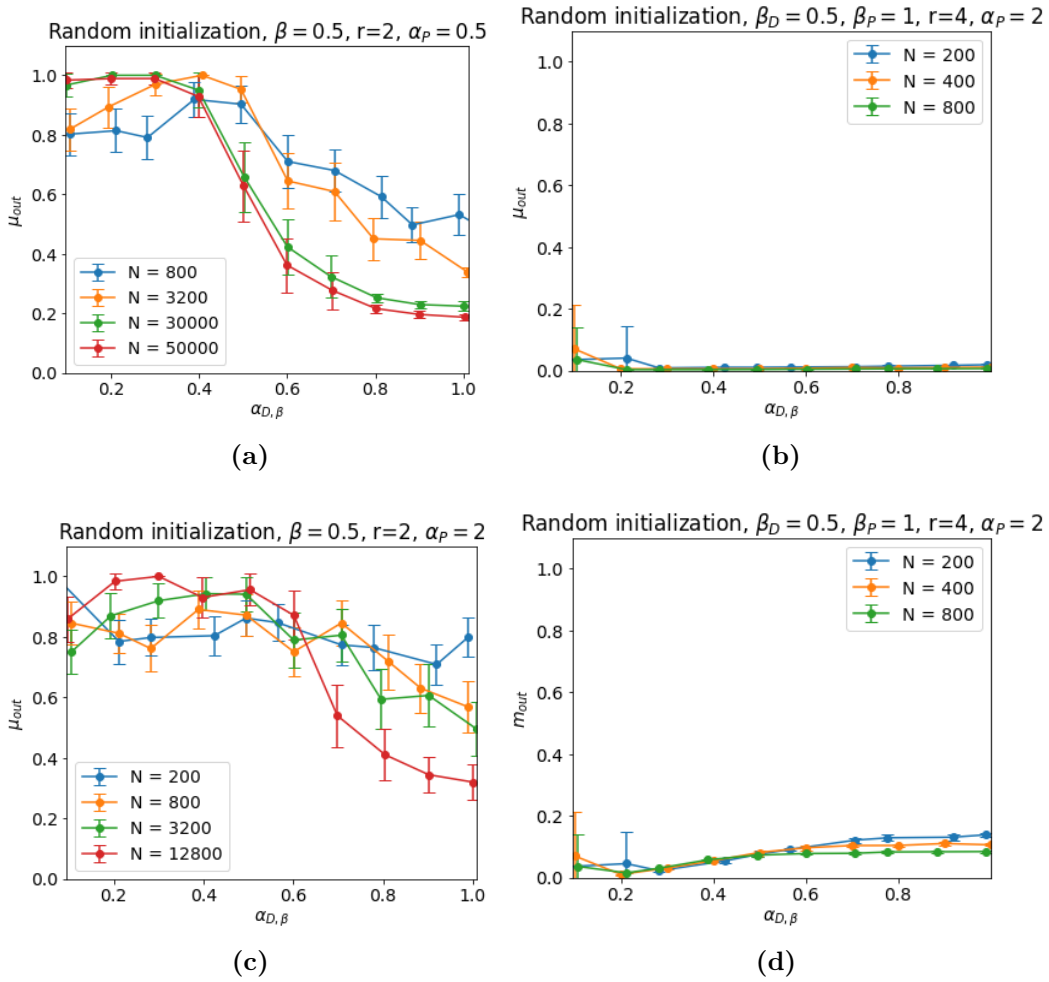


Figure 3.3. Curves of capacity from random of factors in the random features Hopfield model. (a) For $r = 2$ the curves obtained follow the same scaling $\beta_D = 0.5$ as for patterns in r=2. We observe that the curves in general and the approximate critical value is the same inverting the values of α_P and α_D. It is true also for a bigger value of α_P (c). (b) In the $r = 4$ case it seems that it is not possible to have retrieval from random of factors. (d) Also patterns are not dynamically reached, then the final states are uncorrelated to patterns too.

In the $r = 2$ case, it is found in Fig. 3.3 (a) and (c) that the scaling for D to have retrieval from random of factors it's $\beta_D = 0.5$, just like the value of β_P obtained in

the previous analysis for patterns:

$$D = \alpha_{D,\frac{1}{2}} N^{\frac{1}{2}} \text{ for r=2} \tag{3.7}$$

using the linear scaling $P = \alpha_P N$. Again, other couples of exponents (β_P, β_D) were tried, but no other one resulted in overlapping curves.
Another interesting fact found is that not only the scaling seems the same, but also the results are interchangeable: swapping the scalings β_P and β_D of P and D and the values of parameters $\alpha_{P,\beta_P} \Leftrightarrow \alpha_{D,\beta_D}$, the curve of retrieval from capacity from random for patterns becomes very similar to the curve for retrieval from random of factors, showing a sort of symmetry between the two cases.
Instead, for r=4, from Fig. 3.3 (b) and (d) it seems that it's not possible to have retrieval from random of factors for all the scalings tried in the numerical analysis. Thus, it is not possible to assess if factors retrieval from random is not possible at all in the $r = 4$ case, or if the appropriate scaling is substantially different from the ones tried. Anyway, the symmetry between factors and patterns present in the $r = 2$ model is not valid anymore.

3.2 Dynamical trajectories and spurious attractors

In the previous section we obtained the scaling exponents to have the retrieval from random phenomenon, namely β_P and β_D, for all the models investigated in this work, as well as the critical values of parameters α_β that separate the retrieval from random phase from the other one. Now, we study the way memories are retrieved starting from a random initial configuration. The process of retrieval, which we will refer to as *dynamical trajectory*, is analyzed showing the energy $E(t)$ and the magnetization $m(t)$ (or $\mu(t)$ for factors) at any instant t, starting from random until the final configuration. In fact, $T = 0$ dynamics of a discrete model with finite N variables always have a final exit state in which the model stops autonomously to move.
For each choice of parameters, 5 samples of disorder are generated, and for each of them 5 dynamical trajectories are produced starting from different random configurations $\vec{\sigma_0}$. The faded lines in the dynamical trajectory graphs correspond to each of this individual run, and the bold lines are the mean values for each time t. Due to the fact that trajectories end at different times, to prevent longer dynamics from dominating the averages at long times, in computing the means all trajectories already ended are still considered as being on their last value of E and m.
Looking at dynamical trajectories before, near and after the critical parameter $\alpha^c_{P,\beta}$ (or $\alpha^c_{D,\beta}$) it's possible to have a view of the reason why the systems fails to retrieve memories after the critical value. In particular, the main suggestion from this graphs is that in all cases the system have a dynamical transition from converging to stable states at low $\alpha^c_{P,\beta}$ ($\alpha^c_{D,\beta}$), to converge to metastable states at high $\alpha^c_{P,\beta}$ ($\alpha^c_{D,\beta}$).

Uncorrelated patterns Hopfield, $r = 2$ and r=4

For uncorrelated patterns, in Fig. 3.4 it is presented the result for the $r = 2$ model, instead in Fig. 3.5 it is presented the result in the $r = 4$ case.

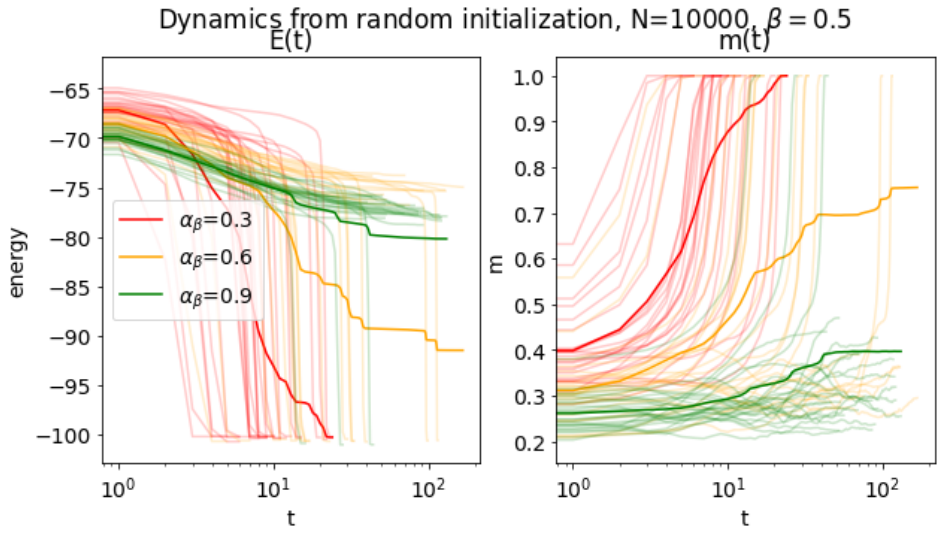


Figure 3.4. Dynamical trajectories of $r = 2$ uncorrelated Hopfield model, for three different parameter values, corresponding to before, near and after the critical capacity from random in Fig. 3.1 (a). Dynamics for all parameters starts inside a plateau, but then inside the capacity from random phase ($\alpha_\beta = 0.3$) dynamics suddenly decrease the energy and magnetize to a memory. Instead, outside the retrieval from random phase ($\alpha_\beta = 0.9$) dynamics remain trapped in the plateau until the exit state. At the boundary ($\alpha_\beta = 0.6$) some dynamics follow a steep energy decrease, meanwhile others remain trapped.

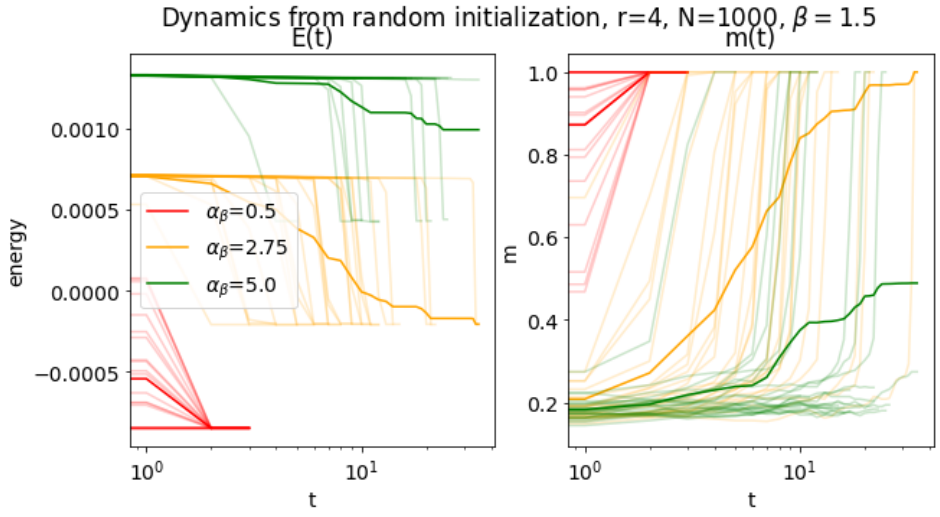


Figure 3.5. Dynamical trajectories of $r = 4$ uncorrelated Hopfield model, for three different parameter values, corresponding to before, near and after the critical capacity from random in Fig. 3.1 (b). The qualitative behaviour is the same of the $r = 2$ case. However, wee can see that for $\alpha_\beta = 0.5$ it is also present *one-shot magnetization*: starting from a random point, after only one dynamical step the system converges to a pattern.

Random features Hopfield, analysis of patterns, $r = 2$ and $r = 4$

The same numerical simulations of the uncorrelated case are repeated for retrieval of correlated patterns in the $r = 2$ (Fig. 3.6) and $r = 4$ (Fig. 3.7) cases.

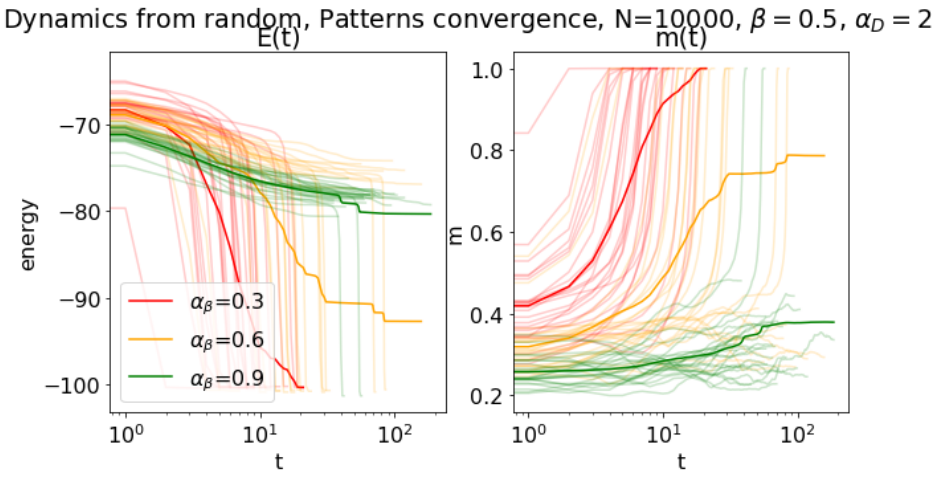


Figure 3.6. Dynamical trajectories for pattern convergence of $r = 2$ random features Hopfield model, for three different parameter values, corresponding to before, near and after the critical capacity from random in Fig. 3.2 (c). We can observe that the dynamical trajectories obtained are similar to the $r = 2$ uncorrelated case in Fig. 3.4, in line with the prediction that at high enough α_D values the number of factors D is so big that patterns are practically uncorrelated.

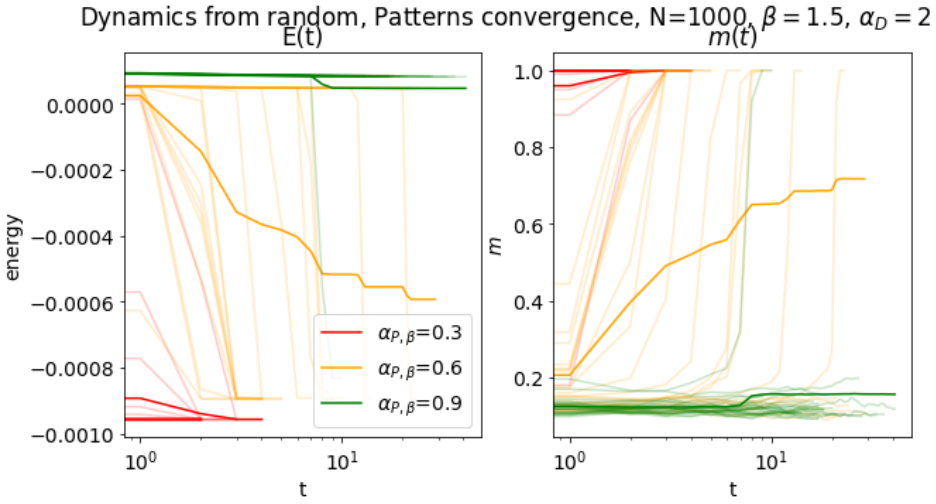


Figure 3.7. Dynamical trajectories of $r = 4$ uncorrelated Hopfield model, for three different parameter values, corresponding to before, near and after the critical capacity from random in Fig. 3.2 (d). We can observe that the result is similar to the $r = 4$ uncorrelated Hopfield, showing that also for an higher r value, in the α_D big case, the patterns are almost uncorrelated and so the system behaves similarly to that case.

Random features Hopfield, factors analysis, r=2

In Sec. 3.1, we have seen that in the 4-spins case it was not possible to find a regime in which factors retrieval from random happen. Then, only the $r = 2$ case is presented in Fig. 3.8.

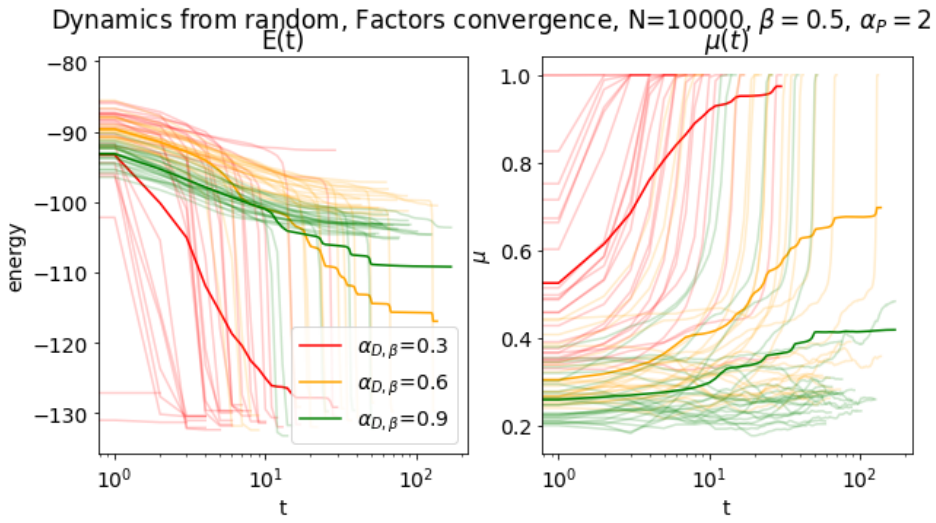


Figure 3.8. Dynamical trajectories for factors convergence of $r = 2$ random features Hopfield model, for three different parameter values, corresponding to before, near and after the critical capacity from random in Fig. 3.3 (c). We can observe that the dynamical trajectories obtained are similar to the $r = 2$ uncorrelated case in Fig. 3.4 as well as to the $r = 2$ random features case in Fig. 3.6. The difference is that in this case the system is converging to factors instead of patterns, and so it is not trivial that the dynamical behaviour in this kind of analysis looks the same. Again, as stated in Fig. 3.3, it seems that pattern retrieval from random and factors retrieval from random have the same dynamical functioning by exchanging the parameters $\alpha_P \Leftrightarrow \alpha_D$.

The dynamical trajectories behavior observed in all models analyzed

For all models considered, both for patterns and for factors (where they are global attractors), the behaviour of the *dynamical trajectories* is the same in the three different phase diagrams zones:

1. Inside the retrieval from random phase, the dynamics explore states with similar values of energy and magnetization, and then it suddenly finds a memory over which magnetize. We will call *plateau* this initial behaviour of almost constant energy. Then at a certain time, a steep decreasing of the energy happens. This phenomenon could be due to the fact that even though there exist some directions to magnetize a memory in the phase space with a random initial configuration, they are a few compared to the total number. So, even if the temperature is zero and so the entropy makes no effects on the exploration of the configurations space made by the system, the fact that only a few specific paths can move the system towards a memory creates an initial plateau in which the system takes a long time to find the right path.

2. Near the critical value, dynamics moves initially in a energy and magnetization plateau. After this plateau, some dynamics follow a steep decrease of energy and magnetize as in the previous case, instead others remain inside the plateau. To understand properly what happens exactly *at* the critical value of the capacity from random this analysis is not sufficient, because we do not have a precise enough value of the critical capacity from random.

3. In the phase where there's not retrieval from random, dynamics are constrained inside the plateau, and only a tiny fraction shows a consistent decrease in energy and magnetize to a memory. It is possible that this behaviour is due to the presence of energy barriers: the few possible directions to escape the plateau and magnetize present in the retrieval from random phase now are typically not present at all, making those systems with a $T = 0$ dynamics (in which it is impossible to make a move that increases the energy) unable to escape the plateau. In the case in which there is an energetic barrier separating high-energy minima from the lower energy memory-correlated minima are metastable spurious states.

To study the cause of the results observed with the dynamics a direct analysis of the energetic landscape is performed in the next section. In particular, by combining the results of dynamical trajectories and the analysis of the landscape it is possible to have a broader view on the phenomena taking place in the retrieval of memories from random initial configurations.

3.3 Analysis of the energy landscape

Finally, for each model are presented some examples of *random walking*, a procedure whose aim is to get a graphical intuition of the structure of the energy landscape, such as the amount of *roughness*, that is used in this context in a qualitative way referring to the density and depth of uninformative minima that are present in the configurations space.
The reason behind this method is to analyze with another approach why after a critical parameter α^c_{P,β_P} (or α^c_{D,β_D}) there is not anymore retrieval from random, with the aim of combining the results with the analysis of dynamical trajectories in the previous section.
The algorithm to produce the random walking graphs is the same for patterns or factors (changing the vectors considered in all steps from $\vec{\xi}^\mu$ to $\vec{f}^k$):

1. The system starts from an *informative minimum*, corresponding to a pattern $\vec{\xi}^\mu$ randomly selected.

2. The nearest $\vec{\xi}^\nu$ is selected, intended as the one with the maximum dot product $\vec{\xi}^\mu \cdot \vec{\xi}^\nu$.

3. A list of spins to be changed is constructed, taking all the spins misaligned in the two vectors.

4. The spins of the system are moved from $\vec{\xi}^\mu$ to $\vec{\xi}^\nu$, changing for every step a single spin taken randomly from the list of opposed spins. The variable that

changes on the x axis is the Hamming distance d from the initial configuration, and for each d it is computed the energy of the system E in that configuration.

5. Once the next pattern is reached, the procedure starts again, without considering the previous one in the search for the nearest pattern.

An example of this procedure repeated for 5 patterns can be observed in Fig. 3.9.

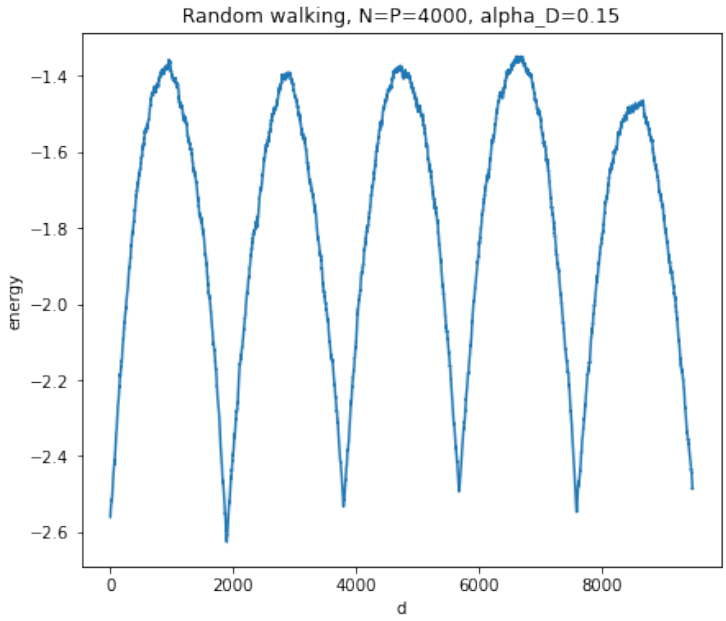


Figure 3.9. Starting from a random factor minimum (leftmost part of the energy curve), the system moves following the *random walking* algorithm towards the nearest factor, with d the number of spin changed from the start. From left to right, initially energy grows in a fair smoothly way, then arrives to a zone where it's rough, after which it falls sharply in the next minimum, again with a smoother behaviour. In this example the procedure is repeated 5 times, so we can see 6 minima. The model is the $r = 2$ random features Hopfield and we are in the range of parameters of high α and low α_D, outside the factors retrieval from random phase.

For each of the four models analyzed in the present work, using the appropriate scalings β_P and β_D obtained in the Sec. 3.1, the same 3 different sets of parameters of the Sec. 3.2 are chosen, that are one before, another in the nearby and the last after the critical transition of the retrieval from random. In particular, for each of the 3 different choice of parameters it is presented a single *random walking*.
However, for each of this plots, it is presented only a focus on the area in the middle of two memories. This choice is done because in very high dimensional space (such as the configurations space, that has a number of dimensions equal to N) a random initialized system is with very high probability in a place approximately perpendicular to patterns or factors. So, the random walking plots focuses on the area in the middle of two memories, emphasizing the landscape of the typical starting random initialization. As discussed before, convergence from random initialization is a sufficient condition to have non-zero basins of attraction, and so when it fails after

the critical value of α_β it means that the system it's not capable of reaching the neighborhood of the memories, namely it is trapped in the area almost perpendicular to all other ones, justifying the choice of presenting only that area in the plot.
Note that this trajectory in general it's not the only or the simplest one for the system to magnetize, and so the energy climb represented in the graphs are not impossible to overcome, as the system in principle can follow other dynamical trajectories to magnetize to a memory. The fact is that the system can overcome some roughness of the landscape thanks to high dimensionality and consequent possible loopholes in the energetic climbs, but over a certain roughness extension or depth this loopholes are typically not found anymore, and so we talk about *energy barriers*. So, the correct point of view of those graphs is to estimate visually in which way changing a selected parameter affects the landscape, and to view the landscape modification that prevents the system to perform retrieval from random.

Uncorrelated patterns Hopfield, $r = 2$ and $r = 4$

In Fig. 3.10 it is shown the projection of the energy landscape before, near and after the critical value of convergence from random. We note that for every graph presented for each model the scale in the x and in the y axis is maintained constant in order to make it possible to compare visually and qualitatively the landscape structure, in particular the amount of roughness. In Fig. 3.11 it is presented the $r = 4$ case.

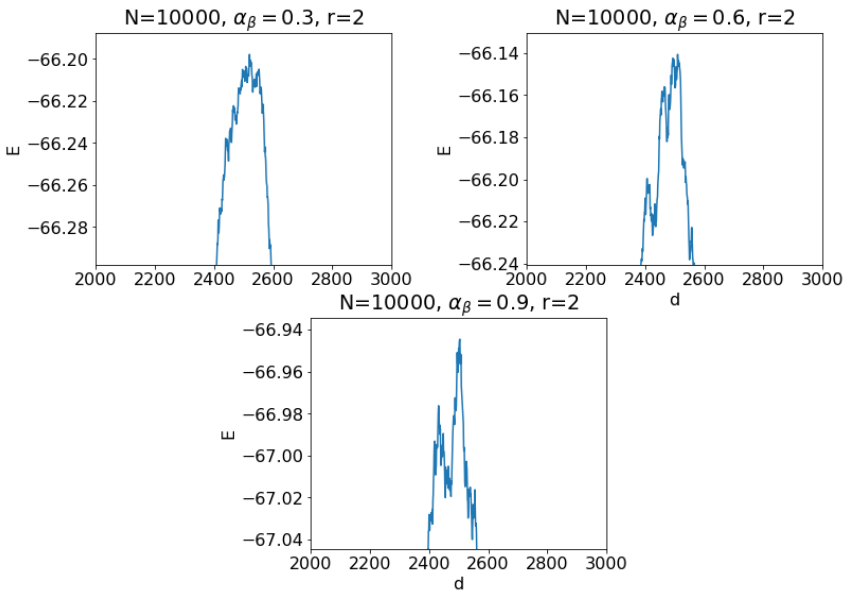


Figure 3.10. For every model the three graphs correspond to before, near and after the dynamical transition. In this figure it is investigated the $r = 2$ uncorrelated patterns model.

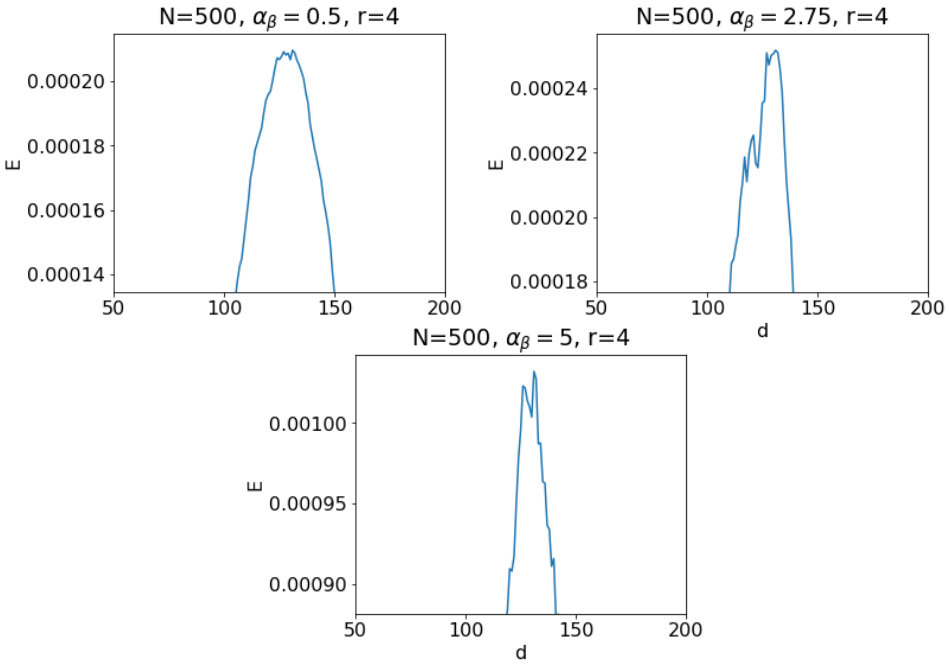


Figure 3.11. Uncorrelated patterns $r = 4$ model.

Random features Hopfield, analysis of patterns, $r = 2$ and $r = 4$

In Fig. 3.12 it is shown the $r = 2$ case, in Fig. 3.13 the $r = 4$ model.

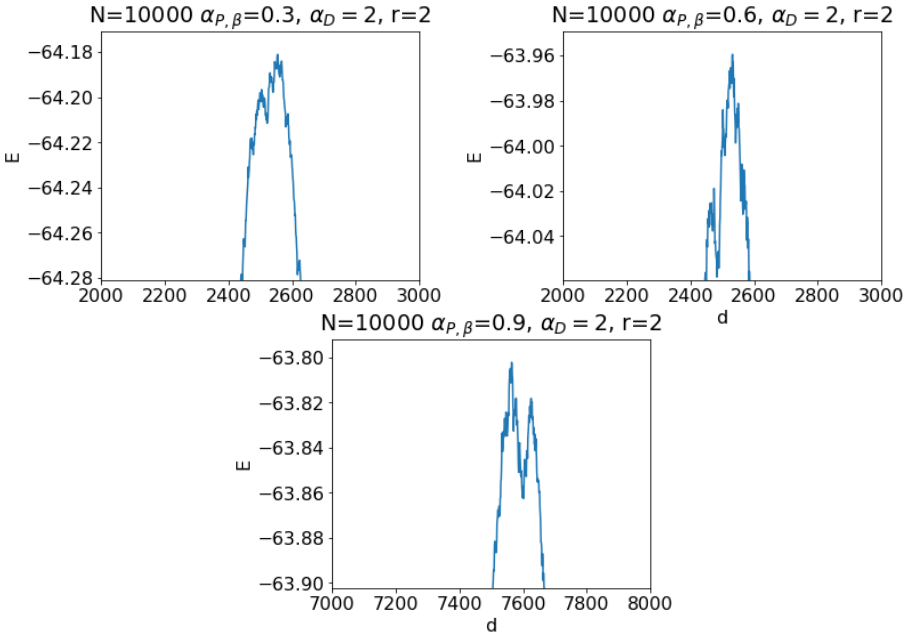


Figure 3.12. Random features $r = 2$ model, retrieval from random of pattern, with the scaling $\beta_P = 0.5$.

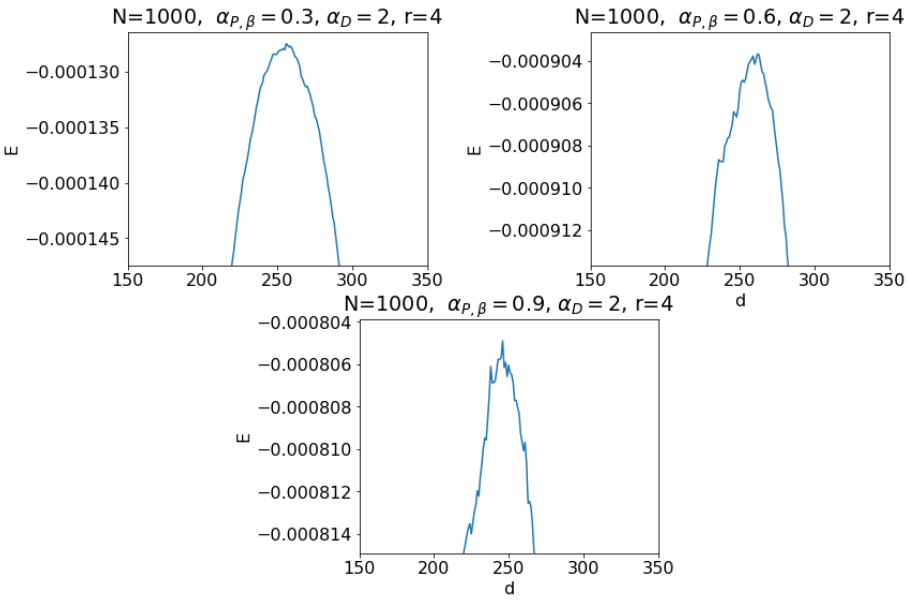


Figure 3.13. Random features $r = 4$ model, retrieval from random of pattern, with the scaling $\beta_P = 1.5$.

Random features Hopfield, factors analysis, $r = 2$
In Fig. 3.14 it is shown the $r = 2$ case

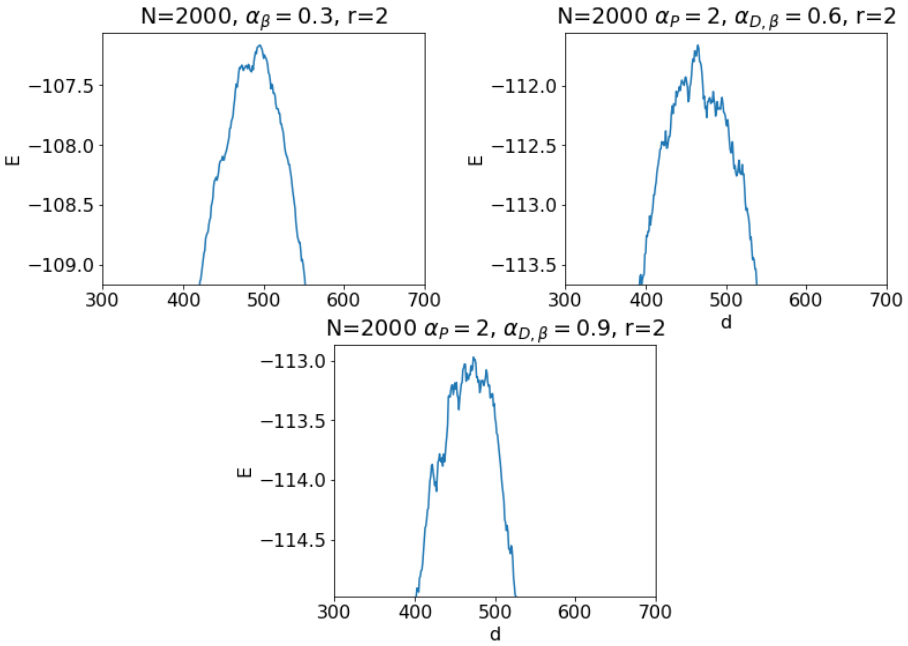


Figure 3.14. Random features 2-spins model, retrieval from random of factors, with the scaling $\beta_D = 0.5$.

For all models analyzed, despite the different model definitions, power r of the energy and patterns or features retrieval, the structure of the landscape changes in all cases in the same qualitatively way.
In the first example inside the retrieval from random phase both the extension and the depth of minima in the rough area is smaller compared to other models with increasing $\alpha_{P,\beta}$ (or $\alpha_{D,\beta}$) parameters. We have a visual confirmation that increasing P (or D in factors retrieval from random) increases the amount of roughness, and this is consistent with the guess made in the dynamical trajectories analysis: retrieval from random is possible until the landscape become so rough that trajectories on average are no longer able to find a path without energetic barriers to magnetize to memories, and the system has a dynamical transition from converging to memories to be trapped in spurious states.

Chapter 4

Conclusion

4.1 Results

4.1.1 SNR method and the study of the capacity

In the Chapter 2 of this thesis the signal-to-noise ratio (SNR) approach has been used to obtain the spinodal lines of point stability of memories.
The result of the SNR method is an equation of the form $F(\alpha_P, \alpha_D)$, using which the spinodal line is the implicitly defined curve in the (α_P, α_D) phase space that solve of the equation

$$\mathcal{H}(F(\alpha_P, \alpha_D)) = C \tag{4.1}$$

where $\mathcal{H}(x) \coloneqq \int_x^\infty Dx$ is the error function and $Dx \coloneqq e^{-\frac{x^2}{2}} dx$ is the Gaussian measure.

The $r = 2$ case

In the $r = 2$ case both patterns and factors have point stability with a scaling

$$\begin{cases} P = \alpha_P N \\ D = \alpha_D N \end{cases} \tag{4.2}$$

although in two different sectors of the phase diagram (α_P, α_D).
The factors capacity spinodal line in Fig. 2.2 matches well the phase diagram obtained from the simulation with the C parameter equal to the value in the uncorrelated Hopfield model case.
Also the patterns capacity spinodal line presented in Fig. 2.4 has a behaviour compatible with simulations, however the degree of similarity is lower than in the factors case using the same value of C as before. In particular, the discrepancy is bigger for small values of α_D and then it tends to decrease for bigger $\alpha_D \gg 1$.

The $r = 4$ case

Increasing r from the original Hopfield $r = 2$ value to the $r = 4$ case, two different scaling regimes appear for the storage of factors and patterns.
For the capacity of factors, the right scaling to obtain a spinodal line not dependent

on the size N is again the linear one (Eq. 4.2). The phase diagram of factors point stability (Fig. 2.6) resembles very much the one with $r = 2$, however for $r = 4$ the spinodal line provided by the SNR method has a worse match with numerical simulations.
In the patterns capacity, the application of the SNR method requires in principle the evaluation of thousands of terms. Using the scheme presented in the Box 2.2.1 to carry on the computation efficiently it is possible to restrict the calculation to only 80 terms, but still too many to obtain the function $F(\alpha_P, \alpha_D)$ manually. However, the exponents of this model can be predicted with the SNR method without carrying out the complete computation. A possible scaling of parameters in this model is

$$\begin{cases} P = \alpha_P N^2 \\ D = \alpha_D N \end{cases} \tag{4.3}$$

This scaling is validated through numerical results, even though it has not been demonstrated to be the only possible one.

4.1.2 The numerical exploration of dynamics and energetic landscape

The scaling relations for retrieval from random

To study the global attraction property, unlike in the case in which the system was initialized over a memory, now the starting state $\vec{\sigma}_0$ is a random point in the configurations space. In this part of the work, the scalings with N of the parameters obtained are presented in the form

$$\begin{cases} P = \alpha_{P,\beta_P} N^{\beta_P} \\ D = \alpha_{D,\beta_D} N^{\beta_D} \end{cases} \tag{4.4}$$

Uncorrelated patterns Hopfield, $r = 2$ and r=4
In the uncorrelated patterns case, it has been shown in Fig. 3.1 that the correct scalings to have retrieval from random of patterns (there are not factors in these models) are

$$P = \alpha_{P,\frac{1}{2}} N^{\frac{1}{2}} \text{ for } r = 2 \tag{4.5}$$

$$P = \alpha_{P,\frac{3}{2}} N^{\frac{3}{2}} \text{ for } r = 4 \tag{4.6}$$

Random features Hopfield, analysis of patterns, $r = 2$ and r=4
In the correlated case, from the numerical analysis in Fig. 3.2, it seems that the scalings needed to have retrieval of patterns from random don't change from the uncorrelated scenario for each r. And so also in this case

$$P = \alpha_{P,\frac{1}{2}} N^{\frac{1}{2}} \text{ for } r = 2 \tag{4.7}$$

$$P = \alpha_{P,\frac{3}{2}} N^{\frac{3}{2}} \text{ for } r = 4 \tag{4.8}$$

both of them with a D scaling of $D = \alpha_{D,1}N$.
Another phenomenon observed is that, as for capacity and point stability analysis, increasing the parameter $\alpha_{D,1}$ (and so the number of factors), it increases the critical value α^c_{P,β_P} after which it's not possible anymore retrieval from random.

Random features Hopfield, factors analysis, $r = 2$ and $r = 4$
In the $r = 2$ case, it is found in Fig. 3.3 that the scaling for D to have retrieval from random of factors it's equal to the previous analysis of the scaling for P:

$$D = \alpha_{D,\frac{1}{2}} N^{\frac{1}{2}} \text{ for } r = 2 \tag{4.9}$$

Not only the scaling seems the same, but also the results are interchangeable: swapping the scalings β_P and β_D of P and D and the values of parameters $\alpha_{P,\beta_P} \Leftrightarrow \alpha_{D,\beta_D}$, the results are similar, showing some sort of symmetry.
For r=4, it seems that it is not possible to have retrieval from random of factors for all the scalings tried in the numerical analysis, and so there is not a symmetry in the 4-spins model between patterns and factors retrieval from random.

The *dynamical trajectories* and the role of spurious attractors

With the appropriate scalings of parameters of the models obtained in the previous study, the dynamical trajectories have been analyzed. For all models, both for patterns and for factors (when the latter are global attractors), the dynamical behaviour is the same when varying the models in each of the three different phases:

1. Inside the retrieval from random phase, the dynamics explore states with similar values of energy and magnetization, and then it suddenly finds a memory over which magnetize. At the same time, it happens a sudden decreasing of the energy.

2. At the critical value, dynamics move in an energy and magnetization plateau. After this plateau, some dynamics follow a sudden decrease of energy and magnetize, instead others remain inside the plateau.

3. In the phase where there's not retrieval from random, dynamics are constrained inside the plateau, and only a tiny fraction shows a downhill in energy and magnetize to a memory.

The analysis of the landscape

The final analysis focuses on the structure of the energetic landscape in the typical random initialization area and for parameters choices such that the systems are inside, at the boundary or outside the retrieval from random phase.
The result is that, for each model, in the first example inside the retrieval from random phase both the extension and the depth of minima in the rough area is smaller compared to other models with increasing α_{P,β_P} (or α_{D,β_D}) parameter. So, increasing patterns (or factors) increases the amount of roughness, and retrieval from random is possible until the landscape become so rough that trajectories on average are no longer able to find a path without energetic barriers to magnetize to memories.

4.2 Discussion of results

4.2.1 The spinodal lines with the SNR method

The signal to noise ratio (SNR) method has been applied by Gardner to p-spin Hopfield models [Gar87]. The novelty of the present work consists in the application of the similar polynomial energy model [KH16], but to a new class of data structure, in which it is present a combinatorial correlation (as defined in Eq. 1.37). In particular, it is presented and utilized a scheme (see Box 2.2.1) to efficiently count the different noise and signal contributions with this kind of correlation between patterns.
In all cases except the last one it was possible to obtain the spinodal line of point stability of patterns or factors in the phase diagram (α_P, α_D). In all cases, analytical predictions are compatible to numerical simulations, except for the case of 4-spin model point stability of patterns, in which it was not possible to compute the line due to a proliferation of terms.
For $r = 2$, there is a qualitative symmetry in the storage of patterns and factors. From the phase diagrams of this two cases (Fig. 2.2 and 2.4) it is possible to observe that by exchanging the two axis $\alpha_P \Leftrightarrow \alpha_D$ the two spinodal lines have a similar behaviour.
With respect to the replica method, in the cases in which the result was computed, the SNR method has proven to be simpler to carry on, and the predictions are compatible with the numerical results. However, it also has some problems:

- There is a parameter C, that is needed in order to compute the spinodal lines, and that cannot be provided by this method because it comes from the thermodynamics of the models. However, its value can be derived for the original Hopfield model, and using this number we obtain a good enough prediction for spinodal lines comparing to numerical results.

- There is a case, the patterns point stability for 4-spins random features model, in which the proliferation of terms makes the computation not possible to carry out. This is a remarkable limitation in a method that is useful if it is simpler than replica computation. Maybe a diagrammatic implementation of SNR could help to manage the proliferation of terms, or another possibility could be that there are further simplifications possible in the scheme of computations presented (2.2.1), that avoid to take into account many irrelevant terms.

4.2.2 The scalings for storage

To have the equation for the spinodal line not depending on the size N, there are precise scalings of P and D with respect to N to be chosen, depending on the results of the SNR computation. In all cases, the scalings predicted are consistent with numerical results. The linear scaling is the correct one for the 2-spins model both for patters and factors, and in the 4-spins model for factors.
For the point stability of patterns in the 4-spins random features model, even though the computation of all terms of the SNR method has not been completed, it was

possible to make the prediction of the scaling

$$\begin{cases} \mathrm{P} = \alpha_P N^2 \\ \mathrm{D} = \alpha_D N \end{cases} \tag{4.10}$$

Then, with respect to the 2-spin case, patterns can be stored with a superlinear scaling in the 4-spins random features model, even if with a lower exponent compared with the scaling $P = \alpha_P N^3$ of the uncorrelated patterns 4-spins Hopfield.
Both in 2-spins and 4-spins models, the scaling exponents of P for patterns storage are compatible with

$$\begin{cases} r-1 & \text{for uncorrelated patterns} \\ r/2 & \text{with random features} \end{cases} \tag{4.11}$$

In particular, the first relation was demonstrated to be correct in [Gar87] for all r values, instead the second one is obtained in this work only for $r = 2$ and $r = 4$. Applying the same types of reasoning as in Sec. 2.3.2 to bigger even r values it is possible to understand if the second relation of 4.11 is valid only for the models investigated or it is a general one.

4.2.3 The retrieval from a random initialization

In addition to the behaviour of retrieving a memory from a corrupted version of it, in literature it is already known that there are different scalings regime in which the Hopfield model converges to memories even starting from a random initial state.
However for the 2-spins model this phenomenon needs a sublinear scaling $P = \alpha_{P,\frac{1}{2}} N^{\frac{1}{2}}$ to happen, and in this work it has been confirmed that this is true both for the uncorrelated and for the random features Hopfield. In the latter, not only patterns but also factors are retrieved from random if D has a sublinear scaling instead of P. Then we observe again a qualitatively symmetrical behaviour of patterns and factors in the 2 spins model, as we have seen for the storage.
For the 4-spins model, the factors are not retrieved from random for all numerically tested scalings. There are two possibilities: or the correct couple of exponents β_P and β_D are an exotic one, or it is not possible at all to converge to factors from an initial random configuration in the $r = 4$ case. Anyway, the symmetry between patterns and factors of the 2-spins case is not valid anymore.
Instead, for retrieval from random of patterns in the 4-spins model, we have obtained that this phenomenon happen with a superlinear scaling $P = \alpha_{P,\frac{3}{2}} N^{\frac{3}{2}}$, both in the uncorrelated and random features cases. The remarkable result is that with this increasing in the r value we go from a sublinear to a superlinear scaling, and passing from uncorrelated to correlated patterns results only in a prefactor in the maximum number of patterns P.

4.2.4 The dynamical transition in the random initialized models

Finally it is analyzed the dynamics when the system starts from a random point in the configuration space, that is typically in the zone almost perpendicular to all patterns. The numerical behaviour of the dynamics is the same for all models

analyzed.
We observe that at first all dynamics move in an area in which the energy decreases slowly. However, for the set of parameters such that retrieval for random is possible, at a certain time we observe a steep descent in energy and the system magnetize, instead outside this set of parameters dynamics typically remain trapped in such energy plateau. Moreover, all landscapes show some degree of roughness, but it qualitatively increases as models stop to retrieve memories.
This results are compatible with the presence of an energy landscape in which only a few directions allow the system to magnetize. Then, after a dynamical transition, dynamics are typically trapped in an energetic plateau because they are no longer able to find a path without energy climbs to magnetize. It is even possible that such paths do not exist at all anymore, making the dynamics trapped in metastable states.

4.3 New developments

4.3.1 The study of the generalization

As described in Sec. 1.1.1 the most important feature for a machine learning model is the generalization: after the exposure to the training dataset, the model is useful if it can solve the task on new and unseen data. From the point of view of the hidden manifold model [Gol+20], as described in Sec. 1.5, a possible definition of generalization in the Hopfield model could be done with the use of the generalization vectors

$$\vec{g}^s = \text{sign}\left(\frac{1}{\sqrt{D}}\sum_k^D c_k^s f_i^k\right) \tag{4.12}$$

where c_k^s are new randomly generated coefficients, meanwhile the factors $\vec{f}^k$ are the same ones from which patterns were built. In this framework, the patterns $\vec{\xi}^\mu$ are the visible data, there is an intrinsic space of features $\vec{f}^k$ to be learnt, and if the model learns the features, it could be possible that it also develops attractors corresponding to generalization vectors $\vec{g}^s$, thus generalizing.
In particular, both the SNR computation of capacity and the numerical analysis of dynamics can be applied without modifications, making this investigation a natural continuation of this work.

4.3.2 The continuous variables model

Modern machine learning models are based on continuous variables, and so it seems natural to extend the Hopfield networks of this thesis to continuous variables. Furthermore, the connection between Hopfield and Transformers [Ram+20] has been suggested for the continuous formulation of the Hopfield model.
Switching to continuous variables can be a difficult operation, and some problems can appear and some choices has to be done. One problem is that the state of the system $\vec{x} \in \mathbb{R}^N$ can have a norm $|\vec{x}| \to \infty$. A possible solution is to introduce the spherical constraint $|\vec{x}|^2 = N$, even if from the p-spin model [Nis01] we know that it is possible that at low temperatures the condensation phenomenon could occur,

where a single variable has $x_i \sim O(\sqrt{N})$ and all others are approximately null. Another possibility is to introduce a regularization term in the Hamiltonian that penalises configurations with high values of $|\vec{x}|$, as done for example in the exponential continuous Hopfield model [LM23]:

$$E(\vec{x}) = -\frac{1}{\lambda}\log\sum_{\mu=1}^{P} e^{\lambda\,\vec{x}\cdot\vec{\xi}^{\mu}} + \frac{1}{2}|\vec{x}|^2 \tag{4.13}$$

Despite the differences between binary and continuous variables and the difficulties in changing the systems from the former to the latter, it is surely an important direction that can be considered for further studies of modern Hamiltonian-based systems.

Bibliography

[Ros58] F. Rosenblatt. "The perceptron: A probabilistic model for information storage and organization in the brain. https://doi.org/10.1037/h0042519". In: *Psychological Review* 65(6) (1958).

[EA75] S F Edwards and P W Anderson. "Theory of spin glasses". In: *Journal of Physics F: Metal Physics* 5.5 (May 1975), p. 965. DOI: 10.1088/0305-4608/5/5/017. URL: https://dx.doi.org/10.1088/0305-4608/5/5/017.

[TAP77] D. J. Thouless, P. W. Anderson, and R. G. Palmer. "Solution of 'Solvable Model of a Spin Glass'". In: *Philosophical Magazine* 35.3 (1977), pp. 593–601. DOI: 10.1080/14786437708235992.

[Par80] G Parisi. "A sequence of approximated solutions to the S-K model for spin glasses". In: *Journal of Physics A: Mathematical and General* 13.4 (Apr. 1980), p. L115. DOI: 10.1088/0305-4470/13/4/009. URL: https://dx.doi.org/10.1088/0305-4470/13/4/009.

[Hop82] J J Hopfield. "Neural networks and physical systems with emergent collective computational abilities." In: *Proceedings of the National Academy of Sciences* 79.8 (1982), pp. 2554–2558. DOI: 10.1073/pnas.79.8.2554. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.79.8.2554. URL: https://www.pnas.org/doi/abs/10.1073/pnas.79.8.2554.

[AGS85] Daniel J. Amit, Hanoch Gutfreund, and H. Sompolinsky. "Storing Infinite Numbers of Patterns in a Spin-Glass Model of Neural Networks". In: *Phys. Rev. Lett.* 55 (14 Sept. 1985), pp. 1530–1533. DOI: 10.1103/PhysRevLett.55.1530. URL: https://link.aps.org/doi/10.1103/PhysRevLett.55.1530.

[Gar87] E Gardner. "Multiconnected neural network models". In: *Journal of Physics A: Mathematical and General* 20.11 (Aug. 1987), p. 3453. DOI: 10.1088/0305-4470/20/11/046. URL: https://dx.doi.org/10.1088/0305-4470/20/11/046.

[GC99] Zhi-Hong Guan and Guanrong Chen. "On delayed impulsive Hopfield neural networks1This work is supported by the National Natural Science Foundation of China, the China Natural Petroleum Corporation, and the HUST Foundation1". In: *Neural Networks* 12.2 (1999), pp. 273–280. ISSN: 0893-6080. DOI: https://doi.org/10.1016/S0893-6080(98)00133-6. URL: https://www.sciencedirect.com/science/article/pii/S0893608098001336.

[Nis01] H Nishimori. "Statistical Physics of Spin Glasses and Information Processing: An Introduction". In: 111 (2001).

[KH16] Dmitry Krotov and John J. Hopfield. "Dense Associative Memory for Pattern Recognition". In: *CoRR* abs/1606.01164 (2016). arXiv: 1606.01164. URL: http://arxiv.org/abs/1606.01164.

[Méz17] Marc Mézard. "Mean-field message-passing equations in the Hopfield model and its generalizations". In: *Phys. Rev. E* 95 (2 Feb. 2017), p. 022117. DOI: 10.1103/PhysRevE.95.022117. URL: https://link.aps.org/doi/10.1103/PhysRevE.95.022117.

[Vas+17] Ashish Vaswani et al. "Attention is All you Need". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[Gol+20] Sebastian Goldt et al. "Modeling the Influence of Data Structure on Learning in Neural Networks: The Hidden Manifold Model". In: *Phys. Rev. X* 10 (4 Dec. 2020), p. 041044. DOI: 10.1103/PhysRevX.10.041044. URL: https://link.aps.org/doi/10.1103/PhysRevX.10.041044.

[Ram+20] Hubert Ramsauer et al. "Hopfield Networks is All You Need". In: *CoRR* abs/2008.02217 (2020). arXiv: 2008.02217. URL: https://arxiv.org/abs/2008.02217.

[LM23] Carlo Lucibello and Marc Mézard. "The Exponential Capacity of Dense Associative Memories". In: *arXiv preprint arXiv:2304.14964* (2023).

[Neg+23] Matteo Negri et al. *Storage and Learning phase transitions in the Random-Features Hopfield Model*. 2023. arXiv: 2303.16880 [cond-mat.dis-nn].