The Role of Hippocampus in Signal Processing and Memory

Lyudmila Kushnir

Submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

under the Executive Committee

of the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2016

ABSTRACT

The Role of Hippocampus in Signal Processing and Memory

Lyudmila Kushnir

Historically, there have been two lines of research on mammalian hippocampus. The first one is concerned with the role of hippocampus in formations of new memories and owes its origin to the seminal study by Brenda Milner and William Scoville of a single memory disorder patient, widely known as H.M. The second line of research views the hippocampus as the brain area concerned with orienting and navigating in space. It started with John O'Keefe's discovery of place cells, pyramidal neurons in the CA3 area of hippocampus, that fire when the animal enters a particular place in its environment.

I argue that both lines of discoveries seem to be consistent with a more general view of hippocampus as a brain area strongly involved in the integration of sensory, and possibly internal, information.

The first part of the thesis presents an investigation of the effect of limited connectivity constraint on the model network in the framework of pattern classification. It is shown that feed-forward neural classifiers with numerous long range connections can be replaced by networks with sparse feed-forward connectivity and local recurrent connectivity without sacrificing the classification performance. The limited connectivity constraint is relevant for most biological networks, and especially for the hippocampus.

The second part describes a decoding analysis from the calcium signal recorded in mouse dentate gyrus. The animal's position can be decoded with approximately 10cm accuracy and the neural representation of position in the dentate gyrus have close to maximal dimensionality. The analysis also suggests that cells with single firing field and cells with multiple firing fields contribute approximately equal amount of information to the decoder.

Contents

List of Figures

Acknowledgements

I am deeply grateful to my advisors. To Stefano Fusi for helping me to find my way in science and to appreciate the high dimensionality of the world.

To René Hen, who introduced me to the field of experimental neuroscience, without which my education would have been somewhat one-sided.

I thank Mazen Kheirbek, without whom the first experimental collaboration of my life wouldn't have been possible, and I wish him great success in his future scientific endeavors.

And, of course, I am thankful to Fabio Stefanini, who has been an invaluable companion on the rough path of neural decoding.

I also thank my committee members: Larry Abbott, Ken Miller, and especially Howard Eichenbaum for agreeing to be on my committee.

I thank everybody in the Center for Theoretical Neuroscience and all the members of René Hen's group for informative discussions.

I thank my advisors in the Physics graduate program of the University of Illinois at Urbana-Champaign: Eduardo Fradkin and John Stack, in particular for their help and support at the difficult transition point of my scientific career.

Many thanks to my friends in the Center for Theoretical Neuroscience: Dina Obeid, Fabio Stefanini, Misha Tsodyks and Raoul-Martin Memmesheimer; to my friends in New

York: Ulkar Agaeva, Shlomo Harnas, Andrey Manakov, Victoria Tarakanova, Vasily Dzyabura, Dasha Silinskaia and Timur Gatanov. Also, to my friends who were elsewhere for most of my PhD but who were still with me in a sense: Andrey Volkov, Karine Kozlova, Alexey Arbuzov, Ekaterina Kaurova, Yin Yuan, Dong-Ha Oh, Ivan Sadovsky, Tatiana Pavlova, Daniyar Nurgaliev, Hien Nguyen, Arely Cortés-Gonzalez, Tomoki Ozawa, Reza Vafabakhsh, Julia Koschinsky, Galina Fomina, Kateryna Fomina and members of СДЛМ3.0 chat: Ivan Gordeli, Tolya Dymarsky, Alexander Alexandrov and Tanya Artemova.

And, of course, to my parents, Irina Divari and Vladimir Kushnir, and my husband, Vasily Pestun.

Chapter 1

Introduction

1.1 Hippocampus role in integration of information

Historically, there have been two lines of research on mammalian hippocampus. The first one is concerned with the role of hippocampus in formations of new memories and owes its origin to the seminal study by Brenda Milner and William Scoville [1] of a single memory disorder patient, widely known as H.M. This patient had his medial temporal lobe structures surgically removed for the relief of medically intractable epilepsy. The surgery was immediately followed by a severe persistent anterograde amnesia, while sparing the short-term memory and long-term memories that were formed before the surgery. This finding strongly suggested a crucial role of hippocampus in creation of episodic memories.

The second line of research views the hippocampus as the brain area concerned with orienting and navigating in space. It started with John O'Keefe's discovery of place cells [2], pyramidal neurons in the CA3 area of hippocampus, that fire when the animal enters a particular place in its environment.

Both lines of discoveries seem to be consistent with a more general view of hippocampus as a brain area strongly involved in the integration of sensory, and possibly internal, information [3], [4], [5],[6],[7]. Indeed, both processes - navigating in space and forming new episodic memories can be seen as merging different aspects of experience together to form an integrated representation - of a particular location in one case and of an episode in the other. One could argue, that in both cases there is an added aspect to the perception of information - a sense of recognition of a particular place and a "feel" of the episodic memory. The later appeared to be missing in the patient H.M., whose hippocampus was surgically removed (see below). Quoting Dr. Susanne Corkin [8] "...he has memories of his childhood, (. . .) although these memories seem to be semanticized."

1.1.1 The role in memory

The importance of hippocampus in formation of new episodic memories was first pointed out in [1], based on a study of memory disorder patient, widely known as patient H.M. In 1953 this patient had undergone a surgical resection of the anterior two thirds of his hippocampi, parahippocampal cortices, entorhinal cortices, piriform cortices, and amygdalae [9] as a treatment for a severe epilepsy, that could not be controlled otherwise. Although the lesion extended to almost entire medial temporal lobes, his profound memory deficits described below are believed to be due to hippocampal removal and not removal of other structures based on the studies of other patients [10],[11].

The most profound deficit that was observed in patient H.M. immediately after the surgery was the inability to form new explicit memory, referred to as anterograde amnesia [1]. This impairment was long-lasting and expanded to all kinds of memory tests, all stimulus materials and all sensory modalities [8](review),[12],[13]. Some of his memories about the events that happened before the surgery were also compromised (retrograde

amnesia), but to a much lesser extend [14].

A fact, that appeared paradoxical at the time of discovery, was that H.M. was still able to acquire new sensorimotor skills, as was first demonstrated by Brenda Milner in the mirror-tracking task [15]. Further studies [16],[17] indicate that despite relatively poor initial performance (probably due to side effects of the antiepileptic drug on the cerebellum), H.M. improved consistently over several days of testing and was able to retain this new skill for years [18]. He still didn't show any recollection of having done the task before or feeling of familiarity.

Another implicit memory effect, that was observed to be spared in the patient H.M., is the repetition priming of certain forms (see [8] for review). In [19] authors report an interesting difference in H.M.'s scores on two kinds of repetition priming - word stem completion priming and perceptual identification priming. The first effect consists in the increased likelihood that a subject will complete a three-letter stem to a word, that was previously studied. Perceptual identification priming is a decreased latency to identify previously studied word compared to unstudied words. H.M. did not show word stem completion priming to the words that came into common usage after the onset of his amnesia, while exhibiting robust perceptual priming to both kinds of words - common before or after the onset of his amnesia.

As pointed out in [8], H.M. showed normal priming in category exemplar production task, but not in category decision task (M. M. Keane et al., unpublished data).

In contrast, the difference in H.M.'s (and other patients with medial temporal lobe lesions) performance on the tasks goes the other way around - his performance on category sorting task (assigning given words to one of the given categories) was close to control levels, while he his score on category fluency task (mean number of examples named from a given category in 1 min) was less than half of that of healthy participants [20].

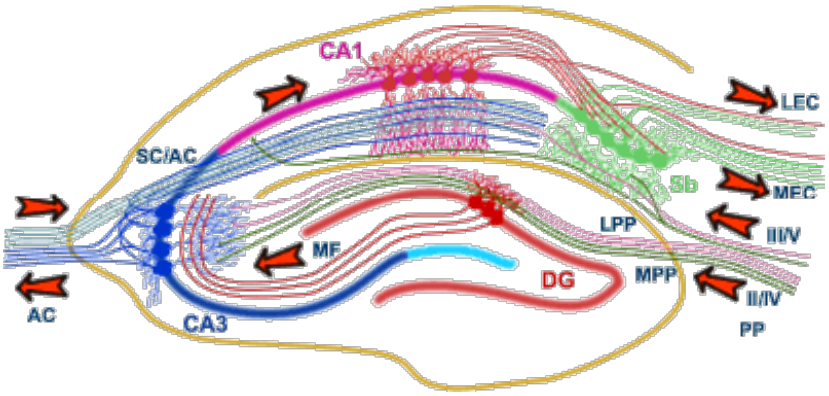

Figure 1.1: The hippocampal Network: The hippocampus forms a principally unidirectional network, with input from the Entorhinal Cortex (EC) tht forsms connections with the Dentate Gyrus (DG) and CA3 pyramidal neurons via the Perforant Path (PP - split into lateral and medial). CA3 neurons also receive input from the DG via the Mossy Fibres (MF). They send axons to CA1 pyramidal cells via the Schaffer Collateral Pathway (SC), as well as to CA1 cells in the contralateral hippocampus via the Associational Commisural (AC) Pathway. CA1 neurons also receive inputs direct from the Perforant Path and send axons to the Subiculum (Sb). These neurons in turn send the main hippocampal output back to the EC, forming a loop.

1.1.2 The role in coding of space

The second line of research is along investigating the observed strong spatial modulation of the activity in various hippocampal areas [2],[21],[22]. Spatially tuned cells observed in the areas CA1 and CA3 (see figure 1.1) got the name "place cells" for their selective firing in one or a few regions of the environment [2],[21]. In contrast, firing patterns of the cells in the entorhinal cortex, constitute a hexagonal grid [22] and are referred to as "grid cells". Investigations of the spatial tuning properties of the principal neurons in the dentate gyrus, granule cells are more sparse due to technical difficulties [23], but those that exist report multiple firing fields for these cells [24],[25]. Numerous studies of firing patterns of hippocampal neurons are reviewed in [26].

Although firing of place cells in CA1 and CA3 areas are often very localized, it is not nearly as reliable in the time domain [27]. In other words, the cell doesn't reliably fire every time the animal crosses the firing field of the cell, even if when the cell does fire, the animal is usually located within the firing field. The authors of [27] compared the distribution of the discharges of a typical place cell during passes through the cell's firing field to a model with Poisson variance of the location specific firing rate. Their finding was that variability in the firing of place cells is much higher than expected from a random model. These poses a question of whether the animals position can be reliably decoded from the activity of place cells.

1.1.3 Neurophysiology of the hippocampus.

The most studied areas of the hippocampus are CA3, CA1 and the dentate gyrus. The CA3 area has strong recurrent connectivity and is often thought of as an attractor network with different attractors corresponding to stored representations of sensory experiences ([28],[29]). Besides strong recurrent input, each CA3 pyramidal cell receives an input from the layer II of entorhinal cortex either directly or via the perforant path through the dentate gyrus.

The principal cells of the dentate gyrus are granule cells, whose mossy fiber axons synapse onto CA3 principal neurons, pyramidal cells. The mossy fibers terminate with very strong synapses - activating one is enough to drive the pyramidal neuron [30]. Also, the connectivity is remarkably sparse - a typical pyramidal neuron is connected to approximately 50 granule cells [31]. Such a sparse connectivity looks even more unusual in the light of extremely low (1%- 5%) activity levels observed in the granule cell layer [23]. Experiments addressing this issue are relatively few compared to the investigations of the CA3 and CA1 area, in part due to technical difficulties of working in the area - the cells are small,

densely packed and sparsely firing [32], [33]. The sparse activity implies that a substantial number of pyramidal cells receive zero input from the dentate gyrus for a typical pattern of activity.

We model sparse activity and sparse, but strong feedforward connectivity in the framework of pattern classification problem (see section 2.4.4) and demonstrate that high classification performance can still be achieved.

The functional role of the dentate gyrus is usually seen as separating neural representations of similar environments if this separation is behaviorally relevant - pattern separation [23],[34], [35]. The theoretical framework for this separation was provided in [36], where the authors demonstrate that even a random entorhinal cortex to dentate gyrus connectivity would suffice to increase the separation of the entorhinal cortex correlated activity patterns. It also suggests that the level of activity in the dentate gyrus controls the trade off between the animals ability to generalize and to discriminate between similar environments.

There is also another prominent neurophysiological feature of the dentate gyrus that should be mentioned. The dentate gyrus is one of only two areas in mammalian brain, that integrate newly born neurons throughout animals life. Adult neurogenesis is an abundant process in non-mammalian vertebrates, but was spared by the evolution only in the dentate gyrus and the olfactory bulb. [23]. What is different about the computation in these areas compared to other regions of the mammalian brain, that makes integration of new units necessary, (or at least helpful) is not understood. It was also a largely controversial question in the near past, whether the adult neurogenesis has any functional significance at all, given it's relatively low rates - 0.004% in middle aged rodents. However, evidence points to its functional relevance - aberrant adult neurogenesis has been argued to contribute to a growing list of psychiatric and neurological conditions, (see [23] for review).

1.2 Generalization - main feature of information integration

Both lines of hippocampus research described above can be seen in the framework of information integration [37],[3],[6],[5],[4]. The role of hippocampus in memory is consistent with the view that hippocampus is involved in relating different aspects of experiences to each other and finding the "correct place" for the new memory in the structure of existing knowledge before storing it. For example, the most striking deficit of the patient H.M. and other patients with hippocampal damage, namely the inability to form new episodic memories, can be seen as inability to integrate different sensory inputs together with internal preexisting associations to form an episodic memory. A few less profound deficits are also consistent with this point of view. In [20] the authors report, that H.M. scored substantially lower than controls on category fluency but not on category sorting task. Category fluency task requires a subject to name as many examples as he can of a given category (for example, birds or musical instruments), while category sorting task requires to assign each item on the list to one of the suggested categories. One could argue, that the former task relies more on the integration of information (in this case internal) to recall the examples one after the other, while the latter task could be performed only using the already organized long term memory storage, that is believed to be mediated by the cortex. Another similar argument is that later observations of the patient H.M. demonstrate his preserved ability to acquire new semantic information if he could relate it to some preoperational knowledge. For example, when solving cross-word puzzles, he could learn that polio vaccine was invented by Salk, even though it was announced after his lesion (in 1955 the operation was in 1953), supposedly because he already had a concept of the disease in the long-term memory [38]. In [8], Dr. Corkin claims that the information in the H.M.'s brain is fragmented and lacking in detail, in contrast to absent. The evidence she provides is the patient's performance on "Famous Faces Test II", where he was shown photographs

of people who were famous in decades following his operation and asked to name them. H.M. could not do it even after being given a few semantic cues, however, he named 18 out 36 individuals using phonemic cues. He still could not say anything about these people, which is consistent with the traces of the information acquired after 1953 being present, but lacking any organization.

The role of hippocampus in space encoding could also be seen in the framework of information integration, as opposed to internal map or "brain's GPS" view. Different location in space are inevitably associated with different patterns of sensory inputs and building a representation of the space is in a sense the same process as organizing this different patterns into a meaningful structure representing the topology of the environment.

1.3 Classification as a minimal problem in generalization

Information integration or information organization are still vaguely defined concepts and it is unclear how to model them mathematically. Our understanding of how the information is represented in the brain and what are the relevant organizational structures are still very limited. However, I will try to argue that there is an aspect of an information integration in general that is easy to formulate mathematically and without which is extremely hard to imagine any information processing.

This general aspect is classification of incoming signals, which is a necessary step in "making sense " out of received information and relating it to previously acquired knowledge. The brain receives a high dimensional input from the sensory systems and process them in the context of it's own internal states. It is hard to argue, that multiple combinations of sensory input patterns and internal signal can "mean the same thing" to the brain, and that neural representations of these patterns can nevertheless be very far from each other in the neural space. When we consider a specific brain area, presumably representing a specific

stage in information processing, "different signals mean the same thing" can be put into more concrete terms - different inputs into a brain area (both sensory and internal) elicit the same output. Learning in the brain area can then be seen as adjusting the connectivity of the network in such a way that the input patterns that belong to the same class (mean the same thing to the animal) elicit the same output, while patterns that belong to different classes elicit different outputs.

When the hippocampus is viewed as a temporary storage of episodic memories, the classification problem that it has to be able to solve is classifying the patterns of sensory inputs over different modalities together with the internal signals into categories, so that the new memories can be successfully integrated into the long term structure. For example, forming an episodic memory of a conversation with a newly met person presumably requires integration of visual and auditory information together with emotional signals from the limbic system, in order to create the concept of that person and recall the entire experience later, when the person's name is mentioned, for example.

In the space coding framework, the classification problem for hippocampus would be, again, integrating inputs across the modalities and categorizing them as corresponding to a specific location in space. One can imagine that different instances of the animal being at a certain location will lead to very different inputs into hippocampus because of different directions of sight, positions of the nose or internal variables. Still, these different input patterns should be perceived as similar when the space is concerned. Learning the representation of space is, in a sense, learning to divide different input patterns into categories corresponding to different spacial locations.

It should be emphasized, that we view categorization of input patterns not necessarily as a final description of the functional role of the hippocampus, but as minimal task that

its organization should allow it to preform. Also, it is natural to assume that this dimensionality reduction process is necessary to compress the information before it is stored in memory.

1.3.1 Theoretical models of the classification problem.

The first model of the neural network, that performed binary classification of the input patterns - *perceptron*, was proposed in 1957 by Rosenblatt, [39]. Its operation is extremely simple - a fully connected readout evaluates the weighted sum of the input components and compares it to the threshold (*linear threshold readout*). If the result is positive, the input patterns is classified as belonging to one class, and if it is negative - as belonging to the other class. Training the perceptrons means tuning the coefficients (weights) with which the different input components enter the sum to lead a specified binary output on a specified set of input patterns (training set). These coefficients can be thought of as the strength of the synaptic connections from the model input neurons to the model readout neuron.

It was also shown by Rosenblatt [39] that if a set of weights leading a desired output on the training set exists, it can be found by an online learning rule that "learns" one pattern at a time. The capacity of the perceptron, meaning the maximal number of training patterns the network can learn was derived by Cover in [40] and is equal to $2N$, where N is the number of input components. It should be stressed, that this result is only valid for the set of uncorrelated patterns. In a more general case, N is the dimensionality of the input representation.

The learning rule proposed by Rosenblatt, which is often called "perceptron learning rule", although is guaranteed to find a solution for the weights is slow often slow and requires a multiple presentation of the same training pattern. There have been proposed numerous

alternative learning rules that lead the same scaling of the capacity with the dimensionality of the input. A particularly simple one that has an advantage of biological plausibility is a Hebb-like learning rule, that we use in the theoretical investigation of chapter 2. As shown there, this learning rule leads to a linear scaling of the perceptrons capacity for a finite tolerated error rate, although the coefficient in front of N is less than 2.

In this light, the sparse feedforward connectivity from the dentate gyrus to the CA3 area mentioned above is very puzzling - the dentate gyrus representations are thought to be high dimensional (decorrelated), which means that the classification capacity of an isolated CA3 readout neuron is severely limited by the number of synapses it receives (which is of the order of 50). If the brain would rely on a computation performed by a single CA3 neuron, the number of distinct classes of responses of the CA3 area would only be of the order of 2×50, which looks too small (this argument does not take into account the existence of direct connection from the entorhinal cortex to the dentate gyrus). Apparently, the computation is carried out by the population of CA3 neurons, but it is not clear how.

In the following chapter we propose a way to integrate the computations of many CA3 readouts via recurrent dynamics and show that these solves the problem posed by limited connectivity. The classification capacity of the proposed network grows linearly with the total number of units even when the number of connections per unit remains unchanged.

1.4 Thesis structure

In the second chapter, 'Classifiers with limited connectivity' we investigate theoretically the effect of limited connectivity constraint on the model network in the pattern classification framework. The limited connectivity constraint is relevant for most biological networks, and especially for the hippocampus. We think of the dentate gyrus as an input layer and the CA3 - a layer of readouts that performs classification of the dentate gyrus activity patterns.

We model the activity patterns in dentate gyrus as random and uncorrelated, which is consistent with its suggested role in pattern separation. The feedforward connectivity from the dentate gyrus to the CA3 area has been observed to be extremely sparse, about 50 incoming synapses per CA3 neuron, and the activity in the dentate gyrus - to be extremely sparse. We model both of these features and show that the classification capacity can still exhibit a favorable scaling with the number of neurons, not connections per neuron.

In the third chapter 'Decoding position form dentate gyrus calcium recordings' I discuss the decoding analysis performed in collaboration with Fabio Stefanini of the data collected by Mazen Kheirbek under supervision of René Hen and Stefano Fusi. We decode the spatial position of an animal freely exploring its environment ($50cm \times 50cm$ box).

Besides being interesting in itself, the study described in chapter 3 can be seen as providing a justification for one of the assumptions made in the theoretical model of chapter 2, namely the assumption that the dentate gyrus activity patterns are uncorrelated. The decoding analysis described in chapter 3 allows to estimate the dimensionality representation of space, and it turns out to be similar to the number of locations within the box defined with the decoding accuracy. This means that the representation has a maximal possible dimensionality. The fact that the performance of linear and non-linear decoders lead similar results is also consistent with this conclusion.

Chapter 2

Classifiers with limited connectivity

The capacity of a perceptron scales linearly with the number of synaptic connections that converge to a single readout [40], thus limited feedforward connectivity in the hippocampus is puzzling - it is unclear how the brain can take advantage of the large number of neurons in the dentate gyrus (about a million in the rodent) when the connectivity to the downstream CA3 area is so low (about 50 converging synapses in the rodent). In this chapter we address the problem posed by extremely sparse feedforward connectivity from the dentate gyrus to the CA3 area of mammalian hippocampus. As discussed in the introduction, we believe it is fruitful to think of the hippocampal network in the framework of pattern classification, which immediately makes one think of the basic model for pattern classification, namely the perceptron [41].

The following should not be viewed solely as a model of hippocampus, however, but rather as a theoretical investigation of the classification performance of perceptron-like neural networks under limited connectivity constraint, which is relevant in many brain areas, especially when long-range connections are considered.

One possible way to overcome the limitations of classification capacity imposed by

sparse connectivity is to combine multiple sparsely connected perceptrons together and determine the collective decision by majority vote, as in committee machines. The number of classifiable random patterns would then grow linearly with the number neurons, even if the number of connections per neuron remains fixed ([42]). However, implementing the majority vote on the neural level would require a fully connected downstream unit, and the problem would simply be moved to the next layer.

We propose a different approach in which the readout is implemented by connecting multiple perceptrons in a recurrent attractor neural network. We show with analytical calculations that the number of random classifiable patterns can grow unboundedly if the number of both input units and perceptrons grow in proportion to each other, while the connectivity of each perceptron, recurrent connectivity and the connectivity of downstream readout all remain finite.

Our solution is still valid even when the input neural representations are sparse, which is surprising given the limited connectivity constraint. Paradoxically, in the case of sparse input representations, the capacity can be made very similar to the dense case, while majority vote scheme implies a much lower capacity.

2.1 Review of the results on perceptron-based classification models

In this section we review the previous results obtained for the capacity of network classifiers based on linear threshold units. We first talk about the first network model for classification introduced by Rosenblatt in 1957, [39], which consists of N input units and a fully connected readout. The famous $2N$ result for the classification capacity of the perceptron discussed further, implies the linear scaling of the capacity not with the number of input units, but

with the number of connections.

Also reviewed here are the results for multi-layered classification networks. Although some consider a case of limited feedforward connectivity, the final classification decision is taken based on majority vote of the ensemble of readouts (committee machine). This voting procedure is assumed to be easily implementable, but it nevertheless requires an additional readout with fully connectivity. As the number of committee members also have to increase unboundedly to achieve an arbitrarily large classification capacity, introducing this additional readout defeats the point of having a network classifier with limited connectivity.

2.1.1 Cover's Capacity

A classical result of Cover [40] is that the expectation value of the maximal number $P_{\max}$ of random $N-dimensional$ patterns, which can be separated into two classes by a linear threshold readout is $2N$.

This result is based on a theorem du to Winder, Joseph, Cameron, Perkins, Schlafli and others [43][44][45][46], that states that the number of inequivalent ways to linearly separate generic P vectors in $\mathbb{R}^N$ is given by

$$C(P,N) = 2\sum_{k=0}^{N-1}\binom{P-1}{k} \tag{2.1}$$

To estimate the maximal capacity Cover compared the combinatorial number $C(P,N)$ with the number of ways to randomly assign labels ± 1 to P patterns, which is equal to 2^P. The function

$$\frac{C(P,N)}{2^P} \tag{2.2}$$

for large N and P has a step-like behavior, being almost a constant except for a sudden

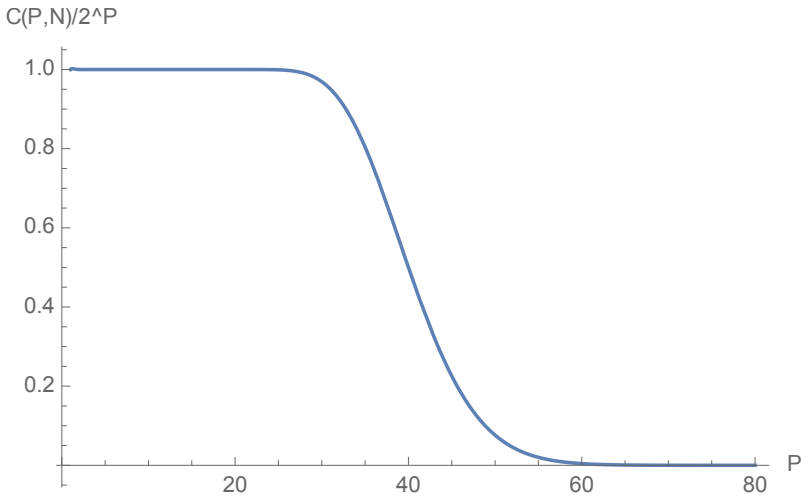

Figure 2.1: Illustration of the Cover's capacity. The probability for a perceptron to learn P patterns without errors is plotted as a function of P for the dimensionality of input $N = 20$. We see that classification performance drops very sharply around $P = 2N$. For larger N the drop becomes even more step-like.

drop at

$$P_{\max} = 2N \tag{2.3}$$

More precisely,

$$\frac{C(P,N)}{2^P}\bigg|_{P=2N} = \frac{1}{2} \tag{2.4}$$

rapidly going to 1 for $P < P_{\max}$ and to 0 for $P > P_{\max}$, see figure 2.1

The computed capacity corresponds to the case of perceptron network architecture consisting of N input units and 1 fully connected readout. Also notice that Cover's capacity computation is a function of the architecture of the network only and not the training algorithm. The capacity achieved when restricted to a concrete algorithm might be smaller than the theoretical limit.

2.1.2 Capacity bounds on associative memory in recurrent networks

In [47] Hopfield introduced a content-addressable memory model as a dynamical physical system in which memory pattern ν is represented by a stable attractor point s^ν in the state space of the dynamical system. The time evolution of the dynamical system is a flow in state space defined by equations of motions (possibly stochastic). If the initial state s is sufficiently closed to an attractor point s^ν, the flow in time t will bring the trajectory $s(t)$ to s^ν.

We say that this dynamical system has capacity P if any prescribed set of P points in its state space can be made to be the set of the stable attractors by adjusting the connectivity of the network. This adjustment of the strength of connections models biological learning.

Hopfield [47] considered a neural network modeled by a stochastic dynamics of N units. The state of unit i at time t is represented by a variable $s_i(t)$ taking value in $\{0, 1\}$. The network evolves according to a transition rule

$$s_i \mapsto \text{sign}\left(\sum_{j=1}^{N} w_{ij} s_j\right) \tag{2.5}$$

Where w_{ij} is a connection strength between i and j units.

The stable attractor state s_i has to satisfy the equation

$$s_i = \text{sign}\left(\sum_{j=1}^{N} w_{ij} s_j\right) \tag{2.6}$$

Given a set of P patterns represented by P states s^ν, the learning rule considered in [47] (after Hebb [48] and Cooper [49]) is

$$w_{ij} = \sum_{\mu=1}^{P} (s_i^\mu - \frac{1}{2})(s_j^\mu - \frac{1}{2}) \tag{2.7}$$

Assuming that the pattern representation variables s_i^ν are randomly distributed with equal probability of taking values 0 and 1, we find that the expectation value of the current into the i-th unit

$$h_i^\nu = \sum_{j=1}^{N} w_{ij} s_j^\nu \tag{2.8}$$

is equal to

$$\langle h_i^\nu \rangle = N(s_i - \frac{1}{2}) \tag{2.9}$$

The non-zero expectation value of h_i^ν comes from the $\mu = \nu$ term in the sum of (2.7).

The diagonal term $\mu = \nu$ is the *signal* term and the contribution of the non-diagonal terms $\mu \neq \nu$ is the *noise*.

When the noise is not too large compared to the signal (the number of patterns P is not too large), (2.9) together with (2.7) implies that the patterns s^ν are fixed points of the recurrent dynamics. A more involved computation required to ensure that the fixed points are stable.

In [50] [51] Amit, Gutfreund and Sompolinsky have analyzed the capacity of stochatstic Hopfield model with temperature parameter β^{-1}. In [50] it was found that for finite number P of patterns and large number of neurons $N \to \infty$ there exists a critical temperature $\beta_c^{-1} = 1$ and the second-order phase transition from a disordered state at higher temperature $\beta^{-1} > \beta_c^{-1}$ to a phase with $2P$ degenerate ground states, each one correlated with one of the trained patterns s^ν. In [51] the analysis was performed in the *scaling limit* $P = \alpha N$ for a constant α by the replica method [52] and mean field theory. It was found that for $\alpha < \alpha_c$ with $\alpha_c \simeq 0.14$ the model exhibit associative memory and at low temperature there exists $2P$ dynamically stable degenerate states.

Hence the conclusion of Amit-Gutfreund-Sompolinsky was that Hopfield model with

Hebb learning rule has a capacity which scales linearly with N

$$P_{\max} \simeq \alpha_c N, \qquad \alpha_c \simeq 0.14 \tag{2.10}$$

However, in [53] [54] [55] Gardner showed that there could exists specially designed learning rule in which the capacity of Hopfield model can scale faster with N and achieves a bound of

$$P_{\max} \simeq 2N \tag{2.11}$$

Let us call *suitable* those network weight parameters (w_{ij}) which produce necessary attractors in the state space to store a given set of patters. Gardner's approach computes the relative volume of the suitable weight parameters to the total volume of the weight parametric space over some prior measure as a function of the number of stored patterns. As the number P of stored patterns increases, the volume of suitable weight parameters decreases, and at certain threshould value $P_{\max}$ goes to zero. Gardner's approach determines only possibility of existence of suitable weight parameters and its relative volume in the parametric space for a given set of patterns, but not actually the learning rule.

In [56] Abbott and Kepler determined efficient learning rules which achieve the Cover's bound for recurrent Hopfield network (see also review on learning algorithms [57])).

2.1.3 Capacity bounds for multi-layered networks

In [58] Mitchison and Durbin considered a multi-layered network of an N-units input layer, a single layer of M intermediate linear threshold readouts and a final layer of S output units (also linear threshold), under the assumtions that all the weights in the network are plastic and that

$$N \geq M \geq S \tag{2.12}$$

They derived a lower and upper bound on the maximal capacity $P_{\max}$

$$2N < P_{\max} < N\gamma \log \gamma; \qquad \gamma = 1 + M/S \tag{2.13}$$

Mitchison and Durbin also considered a simpler case of multi-layer network: N input units, M intermediate units and $S = 1$ final readout unit. The weights between N-layer and M-layer were assumed plastic, but the weights between the M-layer and the final S-readout were fixed to 1. With this architecture the final readout unit performs a consensus operation (majority vote): the final output is 1 if majority of the intermediate units are 1, and 0 otherwise. Such network is called a *committee machine* [59]. Even for the restricted case of committee machine exact combinatorial computation analgous to Cover [40] is hard. Still, numerical estimates showed that theoretical capacity (not specifying a learning rule) of committee machine asymptotically scales as

$$P_{\max} \simeq 2MN \tag{2.14}$$

Kwon and Oh [42], and Monasson and Zucchini [60] analyzed the capacity of a committee machine by the asymptotic behavior of order parameters. Their results are in agreement with the bounds of Mitchison and Durbin. The most relevant result in the present context is the classification capacity for a committee of readouts with non-overlapping connections derived in [42] $P_{\max} \simeq 8\sqrt{2}/\pi N\sqrt{\log M}$. Again, the calculation does not provide a set of weights (or a learning rule) to achieve this capacity.

2.2 Fully connected readout

In this section we derive the classification capacity of a single linear threshold readout achieved with the simple learning rule that we employ throughout this chapter. We assume

that the input patterns and labels are random and uncorrelated, meaning that the activity of each input unit as well as the label is chosen independently, that makes calculations analytically tractable. We use simple Hebbian-like learning rule, that is not optimal and thus leads a capacity that is lower than Cover's $2N$ result [40] but has the same scaling with the number of inputs.

2.2.1 Input statistics

We assume that pairs (ξ^μ, η^μ) of a pattern ξ^μ and a label η^μ for $\mu = 1 \dots P$ are drawn from a random ensemble of P pairs (pattern, label). The pattern components ξ_i^μ on all N input units and label η^μ are random mutually independent variables. We assume that each component ξ_i^μ is activated to 1 with probability f called *coding level* and otherwise is 0, and that label η^μ takes value in one of the two classes $\eta = +1$ with probability y and $\eta = 1$ otherwise:

$$\xi_i^\mu = \begin{cases} 1, & \text{with probability } f \\ 0, & \text{with probability } 1-f \end{cases} \qquad \eta^\mu = \begin{cases} 1, & \text{with probability } y \\ -1, & \text{with probability } 1-y \end{cases} \tag{2.15}$$

2.2.2 Learning rule and the synaptic current

The Hebb-like learning rule, that we use to train the weights $\{w_i\}$ of the classifier is:

$$w_i = \frac{1}{\sqrt{P}} \left(\sum_{\mu=1}^{P} (\xi_i^\mu - f)(\eta^\mu + 1 - 2y) - (1-f)(1-2y) \right) \tag{2.16}$$

In the case of equal probability of a pattern to belong to either class, $y = \frac{1}{2}$, the learning rule simplifies to:

$$w_i = \frac{1}{\sqrt{P}} \sum_{\mu=1}^{P} (\xi_i^\mu - f)\eta^\mu \tag{2.17}$$

When one of the learned patterns $\{\xi_i^\nu\}$ is presented, the response of the linear threshold readout is given by sign $\left(\sum_{i=1}^{N} w_i \xi_i^\nu - \theta\right)$. Here and in all that follows we set the threshold θ to zero. Plugging in the expression for $\{w_i\}$ from the learning rule, we can write for the *synaptic current* h^ν,

$$h^\nu = \sum_{i=1}^{N} w_i \xi_i^\nu \tag{2.18}$$

as follows

$$h^\nu = \sum_{i=1}^{N} w_i \xi_i^\nu = \sum_{i=1}^{N} \frac{1}{\sqrt{P}} \left(\sum_{\mu=1}^{P} (\xi_i^\mu - f)(\eta^\mu + 1 - 2y) - (1-f)(1-2y) \right) \xi_i^\nu \tag{2.19}$$

We split the sum over patterns into the contribution from the same pattern $\{\xi^\nu\}$ that is presented, and the other terms, to get

$$h^\nu = \frac{1}{\sqrt{P}} \left((1-f)\eta^\nu \sum_{i=1}^{N} \xi_i^\nu + \sum_{i=1}^{N} \left(\sum_{\mu \neq \nu}^{P} (\xi_i^\mu - f)(\eta^\mu + 1 - 2y) \right) \xi_i^\nu \right) \tag{2.20}$$

Here we also used that on the same patterns $(\xi_i^\nu)^2 = \xi_i^\nu$ because ξ_i^ν takes value 0 or 1. The first term is *the signal term* and the second term is *noise term*.

We see that the synaptic current depends on the number n^ν of input units that are active

$$n^\nu = \sum_{i=1}^{N} \xi_i^\nu \tag{2.21}$$

The value of n^ν is in *binomial distribution* of N trials with probability f that we denote by $\mathbf{B}(N, f)$.

From the equation (2.20) for the current we find

$$h^\nu = \frac{1}{\sqrt{P}}(1-f)n^\nu\eta^\nu + 2\sqrt{f(1-f)y(1-y)n^\nu}z^\nu \tag{2.22}$$

where we have introduced *noise random variable* z^ν with the zero mean and unit variance coming from the summation over μ in the second term in (2.20). The coefficient is concluded from the fact that each individual term $(\xi_i^\mu - f)\eta^\mu$ has variance

$$[f(1-f)^2 + (1-f)f^2][y(2-2y)^2 + (1-y)4y^2] = 4f(1-f)y(1-y) \tag{2.23}$$

and the fact that the ξ_i^μ variables are mutually independent. By central limit theorem the noise variable z^ν can be approximated as Gaussian in the limit $P \to \infty$ with finite f and n^ν.

If a pattern belongs to either class with equal probability ($y = \frac{1}{2}$), the expression for h^ν simplifies to

$$\boxed{h^\nu = \frac{1}{\sqrt{P}}(1-f)n^\nu\eta^\nu + \sqrt{f(1-f)n^\nu}z^\nu} \tag{2.24}$$

The mean value of n^ν over the binomial distribution $\mathbf{B}(N, f)$ is determined by the number of inputs N and the coding level f

$$\langle n_k^\nu \rangle = Nf \tag{2.25}$$

2.2.3 Classification capacity of a fully connected readout

As the number of learned patterns P increases, the signal (the first term) in (2.24) decreases like $1/\sqrt{P}$, while the noise term (the second term) in (2.24) remains constant. To characterize the performance of a single readout we compute the average sign of the input

current h^ν over different realizations of the input patterns:

$$\langle \mathrm{sgn}(h^\nu) \rangle_{n^\nu, z^\nu} = \eta^\nu \langle \mathrm{erf} \frac{\sqrt{n^\nu (1-f)}}{\sqrt{8Pfy(1-y)}} \rangle_{n^\nu} \tag{2.26}$$

There are two limits in (2.26) that can be analyzed analytically and that we will discuss.

Dense regime

One limit that we call *a dense regime*, is when the mean value of the active input units is large: $\langle n^\nu \rangle = Nf \gg 1$. In this case the approximation plugging in $\langle n^\nu \rangle$ instead of n^ν gives a reasonable estimate and (2.26) can be simplified to

$$\langle \mathrm{sgn}(h^\nu) \rangle_{n^\nu, z^\nu} = \eta^\nu \mathrm{erf} \sqrt{\frac{N(1-f)}{8Py(1-y)}} \tag{2.27}$$

The above expression characterizes the classification performance of the classifier, as it measures the proportion of times that the sign of the current coincides with the class of the input pattern. To express this result in the terms of classification capacity, we have to specify a *tolerated error rate* ϵ, which is equivalent to the requirement $\langle \mathrm{sgn}(h^\nu) \eta^\nu \rangle_{n^\nu, z^\nu} \geqslant 1 - 2\epsilon$. Then, solving (2.27) for P gives single unit capacity in the dense regime:

$$P = \frac{1-f}{8y(1-y)[\mathrm{erf}^{-1}(1-2\epsilon)]^2} N \tag{2.28}$$

Sparse regime

Another limit that we call *a sparse regime* is when the number of input units n^ν is equal to 0 or 1 in the vast majority of cases ($\langle n^\nu \rangle = Nf \ll 1$). In this case the equation (2.26)

should be rewritten as

$$\langle \mathrm{sgn}(h^\nu) \rangle_{n^\nu, z^\nu} = Nf \mathrm{erf} \sqrt{\frac{(1-f)}{8Pfy(1-y)}} \eta^\nu \tag{2.29}$$

Since this expression is only valid in the limit $Pfy \gg 1$ (Gaussian noise assumption, see (2.24)) and $Nf \ll 1$, we can only apply this analysis to the case high tolerated error rate ϵ, namely in the case when

$$1 - 2\epsilon \ll Nf \tag{2.30}$$

The error function can be approximated by a linear function of its argument without lost of generality to get

$$\langle \mathrm{sgn}(h^\nu) \rangle_{n^\nu, z^\nu} = Nf \sqrt{\frac{(1-f)}{2\pi Pfy(1-y)}} \eta^\nu \tag{2.31}$$

Which leads to the capacity in the sparse regime ($Nf \ll 1$), for high tolerated error rate (2.30)

$$P = \frac{(1-f)Nf}{2\pi y(1-y)(1-2\epsilon)^2} N \tag{2.32}$$

If the condition (2.30) is not satisfied, we can not compute the capacity P in the sparse regime explicitly, but there is still some analysis we can carry out. First, it is clear, that the classification accuracy can not differ from chance level ($\frac{1}{2}$) by more than $1 - \mathrm{e}^{-Nf}$, which is the probability to have at least one active input in a pattern. So, if the tolerated error rate $\epsilon < \mathrm{e}^{-Nf} - \frac{1}{2} \approx \frac{1}{2} - Nf$, the capacity, as defined here is zero. In the intermediate case, when $\epsilon > \mathrm{e}^{-Nf} - \frac{1}{2}$ but (2.30) does not hold, we can put an upper bound $P_{\max} < \frac{A}{fy}$, where $A \gg 1$ is a constant.

Having low input sparseness is harmful for the capacity, low output sparseness - helpful.

See [61] and [62] for alternative analysis of stochastic training of perceptron.

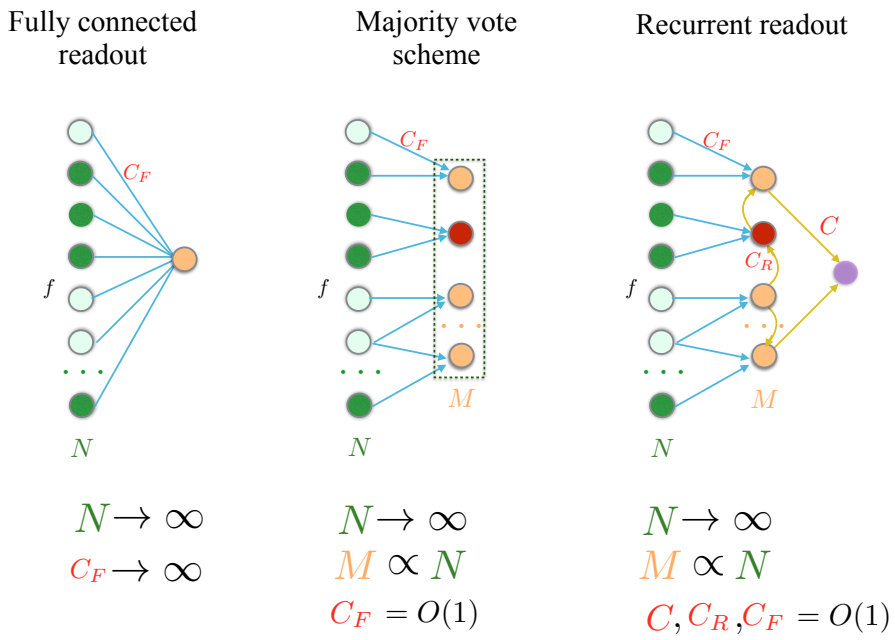

Figure 2.2: Network architecture for the cases of fully connected readout, majority vote scheme and recurrent readout. Only the last one can be considered as a classifier with limited connectivity.

2.3 Sparsely connected readouts: majority vote

We consider the network shown in the middle of figure 2.2 which consists of the input layer (green) and the readout layer (orange). The collective decision of the ensemble of readout units is determined by majority vote.

2.3.1 Network topology

The *input layer* of N neurons is presented with a random and uncorrelated patterns $\xi^\mu = (\xi_i^\mu)_{i=1\ldots N}$ from a set of P patterns $(\xi^\mu)_{\mu=1\ldots P}$ that the network has to classify.

The *readout layer* $\mathbf{M}$ consists of $M = |\mathbf{M}|$ linear threshold readouts. Each readout unit is connected to a randomly chosen C_F out of N input units. Hence, the *feedforward*

connectivity C_F is the number of feedforward inputs that each readout receives. The C_F is an important parameter in the problem as it determines the classification capacity (see previous section) of a readout unit considered in isolation.

2.3.2 Scaling regime

We will keep the connectivity parameters C_F, C_R and C and coding level f to fixed constant values, while sending both the number of input units N, the number of intermediate readouts M and the number of patterns P to infinity

$$\boxed{P, M, N \to \infty; \qquad f, C_F, C_R, C \quad \text{are constant}} \tag{2.33}$$

This is different from classical linear threshold classifier (perceptron), considered in the previous section, where the number of connections received by the readout is always equal to the number of input units. We want to recover the linear scaling of the capacity with the number of input units, that is known to hold for the perceptron [40], in this limited connectivity case.

2.3.3 Input statistics

As in the previous section, we assume that pairs (ξ^μ, η^μ) of a pattern ξ^μ and a label η^μ for $\mu = 1 \dots P$ are drawn from a random ensemble of P pairs (pattern, label). The pattern components ξ_i^μ on all N input units and labels η^μ are random mutually independent variables. We assume that each component ξ_i^μ is activated to 1 with probability f called *coding level* and otherwise is 0, and that label η^μ takes value in one of the two classes $\eta \pm 1$

with equal probability:

$$\xi_i^\mu = \begin{cases} 1, & \text{with probability } f \\ 0, & \text{with probability } 1-f \end{cases} \qquad \eta^\mu = \begin{cases} 1, & \text{with probability } \frac{1}{2} \\ -1, & \text{with probability } \frac{1}{2} \end{cases} \tag{2.34}$$

2.3.4 Single readout

Here and everywhere further, we assume that the output sparseness parameter $y = \frac{1}{2}$ to simplify the formulas.

We will start by ignoring the readout connectivity in the intermediate layer for now and will include it in the next section. Each intermediate layer unit, that we from now on call *intermediate readout* is identical to the readout unit considered in the previous section with the difference that the role of N is now played by C_F - number of units that the readout is actually connected to. The synaptic current into the readout k now becomes

$$h_k^\nu = \frac{1}{\sqrt{P}} \left((1-f)\eta^\nu \sum_{i \in I_k} \xi_i^\nu + \sum_{i \in I_k} \left(\sum_{\mu \neq \nu}^{P} (\xi_i^\mu - f)\eta^\mu \right) \xi_i^\nu \right) \tag{2.35}$$

The only difference from equation (2.20) is the input indices run over I_k, which is the subset of C_R out of N inputs that are connected to the readout k.

The number of active inputs n^ν will now acquire a readout index

$$n_k^\nu = \sum_{i \in I_k} \xi_i^\nu \tag{2.36}$$

And so will the noise variable z_k^ν

$$z_k^\nu = \frac{1}{\sqrt{f(1-f)n_k^\nu}} \sum_{i \in I_k} \left(\sum_{\mu \neq \nu}^{P} (\xi_i^\mu - f)\eta^\mu \right) \xi_i^\nu \tag{2.37}$$

So, the synaptic current into the readout k in response to the input pattern ν is now

$$\boxed{h_k^\nu = \frac{1}{\sqrt{P}}(1-f)n_k^\nu \eta^\nu + \sqrt{f(1-f)n_k^\nu} z_k^\nu} \tag{2.38}$$

Now n_k^ν is drawn from a binomial distribution $\mathbf{B}(C_f, f)$ of C_F trials with probability f,

$$\langle n_k^\nu \rangle = C_F f \tag{2.39}$$

and z_k^ν is gaussian with zero mean and unit variance.

As before, we can compute the expectation value of the sign (h_k^ν) over different realizations of input patterns.

$$\langle \mathrm{sgn}(h_k^\nu) \rangle_{n_k^\nu, z_k^\nu} = \left\langle \mathrm{erf} \frac{\sqrt{n_k^\nu (1-f)}}{\sqrt{2Pf}} \right\rangle_{n_k^\nu} \eta^\nu \tag{2.40}$$

Which can be simplified in the limit $Pf \gg n_k^\nu$ which is valid by the assumption of our scaling regime $P \to \infty$ and $C_F = \mathrm{const}$:

$$\langle \mathrm{sgn}(h_k^\nu) \rangle_{n_k^\nu, z_k^\nu} = \sqrt{\frac{2(1-f)}{\pi P f}} \left\langle \sqrt{n_k^\nu} \right\rangle_{n_k^\nu} \eta^\nu \tag{2.41}$$

The expectation value $\langle \sqrt{n_k^\nu} \rangle$ is computed in the binomial distribution $\mathbf{B}(C_F, f)$

$$\langle \sqrt{n_k^\nu} \rangle = \sum_{n=0}^{C_F} \binom{C_F}{n} f^n (1-f)^{C_F - n} \sqrt{n} \tag{2.42}$$

In the limit of large C_F and finite $\lambda = C_F f$ the binomial distribution $\mathbf{B}(C_F, f)$ turns into Poisson distribution $\mathbf{P}(\lambda)$. Consequently,

$$\langle \sqrt{n_k^\nu} \rangle = \sum_{n=0}^{\infty} \frac{1}{n!} \lambda^n e^{-\lambda} \sqrt{n}, \qquad (C_F \text{ is large, } \lambda = C_F f \text{ is finite}) \tag{2.43}$$

Dense regime

In the dense regime of large C_F and $C_F f \gtrsim 1$

$$\langle \sqrt{n_k^\nu} \rangle = \sqrt{C_F f} \tag{2.44}$$

Sparse regime

In the sparse regime of large C_F and $C_F f \lesssim 1$

$$\langle \sqrt{n_k^\nu} \rangle = C_F f \tag{2.45}$$

For illustration we display the exact $\langle \sqrt{n} \rangle$ in Poisson distribution, in Binomial distribution at $C_F = 50$ and the approximation

$$\langle \sqrt{n} \rangle_\lambda = \min(\lambda, \sqrt{\lambda}) = \begin{cases} \lambda, & \lambda < 1 \quad \textit{(sparse)} \\ \sqrt{\lambda}, & \lambda > 1 \quad \textit{(dense)} \end{cases} \tag{2.46}$$

that we use through out the text. At $C_F = 50$ there is no practical difference between Poisson and binomial distribution, and the elementary approximation (2.46)

differs from the exact expectation value for no more than 23% achieved at the boundary of the sparse and dense regime $\lambda = 1$ as $\langle \sqrt{n} \rangle_{\lambda=1} = 0.773$

2.3.5 Majority rule for the ensemble of readouts

It follows from the above, that as the number of patterns learned by the network grows, the probability of a single readout to classify a pattern correctly approaches the chance level. However, there is always a slight tendency towards the correct answer (the expected value of the sign of the margin is positive for any finite P, however large), that can be utilized

by having a growing number of such readouts that take a collective decision by majority vote. This scheme is known by the name of committee machine and has been shown to largely exceed the performance of a single classifier. It is important to note that in order for the capacity of a committee machine to keep increasing as new members are added, the members responses should be sufficiently independent from each other. In the case of limited connectivity, that we consider, the correlations automatically become smaller and smaller as we increase the number of input units. This happens because the probability of a typical pair of readouts to have a common input and thus correlated responses decreases. In order for correlations not to be a limiting factor of the classification capacity, we need to increase the number of input units linearly with the number of readouts. If one introduces some other mechanism of reducing the correlations between the responses of readouts with common inputs (like making different readouts learn different sets of patterns), a sublinear scaling of the number of input units N with the number of readouts M will be sufficient.

The collective signal of M readouts is given by

$$r^\nu = \frac{1}{M}\sum_{k=1}^{M} r_k^\nu, \qquad r_k^\nu = \mathrm{sgn}(h_k^\nu) \tag{2.47}$$

where h_k^ν is given in (2.24). Positive $r^\nu \eta^\nu$ means that the pattern ν is classified correctly.

Let us start with independent readouts. In this case r^ν can be though of as drown from a normal distribution with the mean given by (2.41) and the variance

$$\mathbf{cov}(r^\nu, r^\nu) = \frac{1}{M}(1 + \mathcal{O}(P^{-1})) \tag{2.48}$$

The probability p_{correct} to classify a pattern correctly ($r^\nu \eta^\nu > 0$) can then be easily computed, and the requirement $p_{\mathrm{correct}} > 1 - \epsilon$ leads to the expression for classification capacity

$$P_{\max} = \frac{\langle\sqrt{n}\rangle^2}{f}\frac{1-f}{\pi(\mathrm{erf}^{-1}(1-2\epsilon))^2}M \tag{2.49}$$

This result only holds for the case of independent readouts, which can be achieved either by a special mechanism that makes different readouts learn different patterns, or by simply making the connections non-overlapping (no input unit is connected to more than one readout). The latter case implies a linear scaling of the number of input units with the number of readouts $N = C_F M$.

We assume the reasoning below to be applied to a test pattern ν but do not explicitly write the label symbol ν to make equations cleaner.

To derive an analogous expression for the overlapping case without a decorrelation mechanism, we need to compute the variance

$$\mathbf{cov}(r^\nu, r^\nu) = \langle (r - \langle r\rangle)^2\rangle \tag{2.50}$$

of the mean readout variable r^ν defined by (2.47) with more precision:

$$\begin{aligned}\mathbf{cov}(r^\nu, r^\nu) = \frac{1}{M^2}\sum_{k=1}^{M}\sum_{l=1}^{M}\mathbf{cov}(r_k^\nu, r_l^\nu) = \\ = \frac{1}{M^2}\sum_{k=1}^{M}\mathbf{cov}(r_k, r_k) + \frac{1}{M^2}\sum_{k=1}^{M}\sum_{l=1}^{M}(1-\delta_{kl})\mathbf{cov}(r_k, r_l) \overset{M\to\infty}{=} \\ = \frac{1}{M}\mathbf{cov}(r_k, r_l)_{k=l} + \mathbf{cov}(r_k, r_l)_{k\neq l}\end{aligned} \tag{2.51}$$

where we split the sum to the diagonal and the non-diagonal term. We want to compute the covariance keeping the terms of the order $1/M$, $1/N$ and $1/P$ assuming that M, N, P scale linearly $M, N, P \to \infty$.

The diagonal term contains the factor $1/M^2$ and M terms, therefore we simply take the

leading contribution as in (2.48)

$$\frac{1}{M^2}\sum_{k=1}^{M}\mathbf{cov}(r_k, r_k) = \frac{1}{M}\left(1 + \mathcal{O}(P^{-1})\right) \tag{2.52}$$

In the non-diagonal term there are $M(M-1)$ equivalent terms with common factor $1/M^{-2}$ which gives $(1-1/M)$ coefficient. Therefore, for the desired precision we shall compute covariance for two different readouts k and l

$$\mathbf{cov}(r_k^\nu, r_l^\nu)_{k\neq l} = \langle r_k r_l\rangle - \langle r_k\rangle\langle r_l\rangle \tag{2.53}$$

keeping terms of order $1/M, 1/N$ or $1/P$.

The covariance $\mathbf{cov}(r_k, r_l)$ vanishes if r_k and r_l are independent variables, which is the case when readout k and readout l *do not share common input*.

If readout k and readout l *share commont input*, then the covariance $\mathbf{cov}(r_k, r_l)$ can be contributed by the correlation in the current (2.38).

Let $|I_{kl}|$ be the number of common inputs to readout k and l in the architecture of the network $|I_{kl}| = |I_k \cap I_l|$. Let $n_{kl} \leq |I_{kl}|$ be the number of active neurons in common input set I_{kl}. In the limit $N \to \infty$ and fixed $|I_k| = |I_l| = C_F$ the probability to have overlap of size $|I_{kl}|$ is approximated by

$$\text{Prob}(|I_k \cap I_l| = |I_{kl}|) = \frac{1}{|I_{kl}|!}\left(\frac{C_F^2}{N}\right)^{|I_{kl}|} \tag{2.54}$$

Recalling, that we are interested in the limit of large number of stored patterns P, that will imply the large number of the input units N, we will ignore the probability of two readouts to have more than $|I_{kl}| = 1$ common inputs. The estimate (2.54) implies that restriction to $|I_{kl}| = 1$ is consistent for $C_F^2/N \ll 1$. Therefore, we restrict to the case

$|I_{kl}| = 1$ with

$$\text{Prob}[|I_{kl}| = 1] = \frac{C_F^2}{N} \tag{2.55}$$

Even though for our estimates it is sufficient to consider $|I_{kl}| = 1$ common input, we will keep $|I_{kl}|$ explicitly in the equations below to keep track of the meaning of the respective terms.

Consequently $C_F - |I_{kl}|$ is the total number of *non-common inputs* for the readout k or readout l defined by network architecture. Some of those neurons might be active or non-active, so let $n'_k \leq C_F - |I_{kl}|$ and $n'_l \leq C_F - |I_{kl}|$ be the respective total number of independent *active non-common inputs* for readout k or readout l.

When the readouts share n_{kl} active inputs, the noise term in (2.38) for each readout should be split into two contributions: one comes from the common input in I_{kl}, we call it z_{kl}, and the other one is independent from the other readout, called respectively z_k or z_l from the complementary non-common inputs in I_k or I_l but not in I_{kl}.

Hence for the purpose of estimating the correlator (2.53) we shall write the current (2.38) in terms of independent random variables n_{kl}, n'_k, n'_l and z_{kl}, z_k, z_l

$$\begin{aligned} h_k &= \frac{1}{\sqrt{P}}(1-f)(n'_k + n_{kl})\eta + \sqrt{f(1-f)n_{kl}}z_{kl} + \sqrt{f(1-f)n'_k}z_k \\ h_l &= \frac{1}{\sqrt{P}}(1-f)(n'_l + n_{kl})\eta + \sqrt{f(1-f)n_{kl}}z_{kl} + \sqrt{f(1-f)n'_l}z_l \end{aligned} \tag{2.56}$$

The variables n_{kl}, n'_k, n'_l are in the independent binomial distributions $n_{kl} \in \mathbf{B}(|I_{kl}|, f)$ and $n'_k, n'_l \in \mathbf{B}(C_F - |I_{kl}|, f)$. The random variables z, z_k, z_l are in the independent standard Gaussian distributions.

The covariance (2.53)

$$\mathbf{cov}(r_k, r_l) = \langle \text{sgn}(h_k), \text{sgn}(h_l) \rangle - \langle \text{sgn}(h_k) \rangle \langle \text{sgn}(h_l) \rangle \tag{2.57}$$

where the average is over n_{kl}, n'_k, n'_l and z, z_k, z_l is contributed only by the cases with $n_{kl} > 0$. For $n_{kl} = 0$ the covariance vanishes because the remaining random variables are independent (2.57).

For $n_{kl} > 0$ the signal term can be ignored compared to noise (including the signal term will lead to corrections suppressed by $\frac{1}{P}$) but because of (2.55) we need to keep only $\mathcal{O}(1)$ terms in (2.57) in the scaling limit $N, M, P \to \infty$. We can drop the second term in (2.57) because it is of order $\mathcal{O}(1/P)$.

First we integrate over z_k and z_l and find

$$\begin{aligned} \langle \mathrm{sgn}(h_k) \rangle_{z_k} &= \mathrm{erf}\left(\frac{z_{kl}}{\sqrt{2}} \sqrt{\frac{n_{kl}}{n'_k}} \right) \\ \langle \mathrm{sgn}(h_l) \rangle_{z_l} &= \mathrm{erf}\left(\frac{z_{kl}}{\sqrt{2}} \sqrt{\frac{n_{kl}}{n'_l}} \right) \end{aligned} \tag{2.58}$$

To proceed further we need to use the table integral[1]

$$\int_0^\infty \mathrm{erf}(az)\mathrm{erf}(bz)e^{-c^2z^2}dz = \frac{1}{c\sqrt{\pi}} \tan^{-1} \frac{ab}{c\Delta} \qquad \Delta = \sqrt{a^2 + b^2 + c^2} \tag{2.59}$$

which for our purposes is conveniently presented as

$$\mathbf{cov}\left(\mathrm{erf}\left(\frac{az}{\sqrt{2}} \right), \mathrm{erf}\left(\frac{bz}{\sqrt{2}} \right) \right)_{z \in \mathbf{N}(0,1)} = \frac{2}{\pi} \tan^{-1} \frac{1}{\sqrt{(a^{-2}+1)(b^{-2}+1) - 1}} \tag{2.60}$$

This leads to the result

$$\mathbf{cov}\left(\mathrm{erf}\left(\frac{z_{kl}}{\sqrt{2}} \sqrt{\frac{n_{kl}}{n'_k}} \right), f\mathrm{erf}\left(\frac{z_{kl}}{\sqrt{2}} \sqrt{\frac{n_{kl}}{n'_l}} \right) \right)_z = \frac{2}{\pi} \tan^{-1} \frac{1}{\sqrt{(n'_k/n_{kl}+1)(n'_l/n_{kl}+1) - 1}} \tag{2.61}$$

[1] See equation 18 on page 158 in [63]

and finally to

$$\mathbf{cov}(r_k, r_l)_{k\neq l} = \frac{2}{\pi}\left\langle \theta(n_{kl}) \tan^{-1} \frac{1}{\sqrt{(n'_k/n_{kl}+1)(n'_l/n_{kl}+1)-1}} \right\rangle_{n_{kl}\in\mathbf{B}(|I_{kl}|,f), n'_k, n'_l \in \mathbf{B}(C_F-|I_{kl}|)}$$

where $\theta(n_{kl})$ is the step function (as explained after equation (2.57)).

In the approximation $C_F^2/N \ll 1$ we consider only the case of $|I_{kl}| = 1$ and then the probability that the single common input neuron in the overlap $I_k \cap I_l$ is active, i.e. $n_{kl} = |I_{kl}| = 1$, is f. Therefore, the average over $\mathbf{B}(|I_{kl}|, f)$ gives factor f and leads to

$$\boxed{\begin{array}{c} \mathbf{cov}(r_k, r_l)_{k\neq l} = \dfrac{1}{N}\varphi_{C_F,f} \quad \text{where} \\ \varphi_{C_F,f} = \dfrac{2fC_F^2}{\pi}\left\langle \tan^{-1} \dfrac{1}{\sqrt{(n'_k+1)(n'_l+1)-1}} \right\rangle_{n'_k, n'_l \in \mathbf{B}(C_F-1,f)} \end{array}} \tag{2.62}$$

so that each n'_k and n'_l are independently drawn from the binomial distribution $\mathbf{B}(C_F-1, f)$.

Dense regime

In the dense regime ($C_F f \gtrsim 1$) from (2.62) we find

$$\varphi_{C_F,f} = \frac{2C_F}{\pi} \tag{2.63}$$

and

$$\langle \mathbf{cov}(r_k, r_l)_{k\neq l} = \frac{2C_F}{\pi N} \tag{2.64}$$

Sparse regime

In the sparse regime ($C_F f \lesssim 1$ and $C_F \gg 1, f \ll 1$) from (2.62) we find

$$\varphi_{C_F,f} = fC_F^2 \tag{2.65}$$

and

$$\mathbf{cov}(r_k, r_l)_{k\neq l} = \frac{fC_F^2}{N} \tag{2.66}$$

Correction to classification capacity

Hence, the diagonal (2.48) and the non-diagonal term (2.62) in (2.51) we obtain corrected standard deviation σ_r of the collective signal r^ν (2.47)

$$\sigma_r = \sqrt{\frac{1}{M} + \frac{1}{N}\varphi_{C_F,f}} \tag{2.67}$$

where $\varphi_{C_F,f}$ is in (2.62)(2.63)(2.65). The mean of r^ν is like in (2.41)

$$\mu_r^\nu = \sqrt{\frac{2(1-f)}{\pi P f}}\langle\sqrt{n_k^\nu}\rangle\eta^\nu \tag{2.68}$$

where $\sqrt{n_k^\nu}$ is computed in (2.42)(2.44)(2.45).

This implies the classification capacity in case when no care is taken to decorate the readouts, and the connections overlap by chance, is

$$P_{\max} = \frac{\langle\sqrt{n}_k\rangle^2}{f}\frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi}\frac{M}{1+\frac{M}{N}\varphi_{C_F,f}} \tag{2.69}$$

where $\varphi_{C_F,f}$ is in (2.62)(2.63)(2.65).

If both the number of input units N and the number of readouts M should increase in proportion to each other, the capacity P increases linearly with N and M.

In the dense limit, using (2.44) and (2.63) it simplifies to

$$P_{\max} = \frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi}\frac{C_F M}{1+\frac{M}{N}\frac{2C_F}{\pi}} \tag{2.70}$$

and in sparse limit it simplifies to

$$P_{\max} = \frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi} \frac{MC_F^2 f}{1+\frac{M}{N}C_F^2 f} \tag{2.71}$$

2.3.6 Optimizing the architecture for a given number of units

We see from the previous expression, that classification capacity of our version of a committee machine depends on both, number of input and readout units. In biological as well as in machine learning context it is natural to constrain the total number of units and ask what will be the way to divide them between input and readout layer that maximizes the classification capacity. (With the caveat of keeping the input units independent from each other, not sure anymore this is such a natural thing to do). This leads to the relation between number of input units N and number of readouts M as well as the expression of $P_{\max}$ in terms of total number of units $(M+N)$:

$$\begin{aligned} N &= M\varphi_{C_F,f}^{\frac{1}{2}} \\ P_{\max} &= \frac{\langle\sqrt{n_k}\rangle^2}{f} \frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi} \frac{M+N}{\left(1+\varphi_{C_F,f}^{\frac{1}{2}}\right)^2} \end{aligned} \tag{2.72}$$

where $\langle\sqrt{n_k}\rangle$ is in (2.42)(2.44)(2.45) and $\varphi_{C_F,f}$ is in (2.62)(2.63)(2.65).

2.4 Sparsely connected readout: recurrent dynamics

The majority rule scenario already overcomes the limitations of readouts connectivity, but this is not the final answer to constructing a classifier with limited connectivity. The reason is that we still need to implement the majority rule and bring the classification signal to the level of a single unit. The naive way to do it would require another final readout that

would have to sample the entire population of M intermediate layer readouts. Since M has to scale linearly with the number of learned patterns P, the connectivity of the final readout would also have to scale linearly with P and would exceed any predetermined limit for sufficiently large number of learned patterns.

To implement the majority vote of the intermediate readouts while keeping the connectivity of any unit in the network limited, we introduce the recurrent connectivity in the readout layer. Our goal is to have two attractor states of the intermediate layer dynamics, that will be far away from each other (in terms of the Hamming distance), and to have the slight imbalance in the feedforward input, determined by the class of the presented pattern, biasing the network to choose one or the other. The fact that the attractors are far away and do not become closer when the number of learned patterns P increases, implies that the final readout will be able to discriminate between this states, and thus indicate the class of the presented pattern, even if it's connectivity doesn't scale with P. It turns out that for two-way classification it is enough to have random recurrent connectivity with sufficiently large but not increasing with P number of connections per unit, and the weights of these recurrent connection don't have to be tuned (no learning required for recurrent connections).

Let C_R be the *number of recurrent connections* per unit in the intermediate layer and let α be weights on recurrent connections. The recurrent connections weights are assumed to be symmetric.

We want to compute the probability of the network of recurrently connected readouts to go to the correct attractor (the one assigned to the class of the input pattern presented) as a function of the number of input units N, readout units M and various parameters of the network.

2.4.1 Network topology

The recurrent readout network show on the right of figure 2.2 consists of the input layer (green), the intermediate readout layer (orange) and the final readout unit (purple). As before, the *input layer* of N neurons is presented with a random and uncorrelated patterns $\xi^\mu = (\xi_i^\mu)_{i=1\ldots N}$ from a set of P patterns $(\xi^\mu)_{\mu=1\ldots P}$ that the network has to classify.

The readout layer that we sometimes call *intermediate layer* $\mathbf{M}$ consists of $M = |\mathbf{M}|$ linear threshold readouts. Each readout unit is connected to a randomly chosen C_F out of N input units. Hence, the *feedforward connectivity* C_F is the number of feedforward inputs that each readout receives. The C_F is an important parameter in the problem as it determines the classification capacity (see section 2.2) of a readout unit considered in isolation. The intermediate layer is recurrently connected for the purpose that will be explained later. The number of recurrent connections a typical readout receives is denoted by C_R. For the case of binary classification, the probability that two readouts are connected is the same for each pair. The recurrent connections are not plastic and can be chosen to be all of equal strength.

The *final layer* consists of a single readout unit that is connected to a randomly chosen subset of C readout units in the second layer, with the strength of all connections taken equal.

Let M be the number of readout units in the intermediate layer, and J_{kl} the connectivity matrix of the recurrent network in the intermediate layer for $k, l \in [1 \ldots M]$ so that

$$J_{kl} = \begin{cases} 1 & \text{if readout } k \text{ and } l \text{ are connected} \\ 0 & \text{if readout } k \text{ and } l \text{ are not connected} \end{cases} \tag{2.73}$$

Let the C_R be the number of recurrent connections per unit

$$\sum_{l=1}^{M} J_{kl} = C_R, \qquad \forall k \in [1 \dots M] \tag{2.74}$$

We will carry this calculation in the *mean field* approximation. Let m_k^ν be *the average activity* of the unit k in the recurrent network in response to the input pattern ν in the recurrent dynamics.

Discrete time dynamical model

We model the recurrent dynamics as a probabilistic dynamical process in discrete time t with the probabilistic transition rule from a network state at time t to a network state at time $t+1$. Let $s_k(t) \in [0,1]$ for $k \in [1 \dots M]$ be the dynamical variable describing the state of neuron k at time t in recurrent network.

Let $\tilde{h}_k^\nu$ be the total input current in the neuron k

$$\tilde{h}_k^\nu(t) = \sum_{l=1}^{M} \alpha J_{kl} s_l^\nu(t) + h_k^\nu \tag{2.75}$$

where the first term describes the current from the recurrent connections and the second term describes the constant in time current from the input layer.

The probabilistic transition rule from the state at time t to the state to time $t+1$ is

$$s_k(t+1) = \begin{cases} 1, & \text{with probability } \frac{1}{1+e^{-2\beta\tilde{h}_k^\nu(t)}} \\ -1, & \text{with probability } \frac{e^{-2\beta\tilde{h}_k(t)}}{1+e^{-2\beta\tilde{h}_k^\nu(t)}} \end{cases} \tag{2.76}$$

Here β is the *inverse temperature parameter* for the statistical model of the recurrent

dynamics. The recurrent dynamics has joint probability stationary distribution

$$\text{Prob}[\{s_k\}] \sim \exp\left(\sum_{k=1}^{M} \beta s_k \tilde{h}_k\right) \tag{2.77}$$

In certain regimes (see below) we will be able to approximate the probabilistic dynamics or equivalently, the stationary distribution, by *mean field* method.

2.4.2 Regimes

We consider two different regimes of the recurrent dynamics described above: *uniform regime* and *two subnetworks regime*.

Uniform regime is defined as a regime where all the recurrently connected readouts when averaged over the statistical ensemble, follow the same dynamical trajectory, and considering them all as identical gives a good approximation for the network's evolution. In terms of mean field approximation, we define a uniform regime as a regime when introducing one order parameter m is enough.

The two subnetworks regime, on the other hand is characterized by the necessity to split the population $\mathbf{M}$ of recurrently connected readouts into two populations

- population $\mathbf{M}_{\text{f}}^{\nu}$ of *free* units, that receive zero feedforward input
- population $\mathbf{M}_{\text{IR}}^{\nu}$ of *input receiving* units, that receive non-zero feedforward input

We remark that the identity of free units and input receiving units depends on the input pattern ν.

It is clear that dense input representation $C_F f \gg 1$ is sufficient condition for the recurrent network of the intermediate layer to be in the uniform regime, because the proportion of units with zero feedforward input is negligible.

The sparse input representation $C_F f \lesssim 1$ requires more careful treatment. If the input representations is sparse $C_F f \lesssim 1$ but the dynamical noise in the intermediate layer is high enough to "mix" the two populations, then the entire network can be considered as one. It will become clear from the section 2.4.3 that the high noise requirement is $\beta^{-1} \gg \sqrt{f}$. But if noise is low, $\beta^{-1} \ll \sqrt{f}$, and input representation is spare $C_F f \lesssim 1$, the recurrent network is in the two subnetworks regime.

The two subnetworks regime is somewhat complicated in the general case, but there is an assumption, that makes it tractable. If the feedforward connections are very strong relative to the recurrent ones and dynamical noise is not too high, the input receiving units can be considered enslaved to the input, namely $s_k^\nu = \text{sign}\,(h_k^\nu)$, for $k \in \mathbf{M}_{\text{IR}}^\nu$. This case is considered in the section 2.4.4. There, high and low dynamical noise subsections refer to comparison between the noise and external input to the subnetwork of free units, that is coming from the enslaved ones.

To summarize, the *uniform regime*, studied in section 2.4.3 is characterized by

$$\begin{gathered} \text{high dynamical noise} \quad \beta^{-1} \gg \sqrt{f} \\ \text{or} \\ \text{low dynamical noise} \quad \text{and} \quad \text{dense input} \quad C_F f \gg 1 \end{gathered} \tag{2.78}$$

In section 2.4.4 we consider a special case of *two subnetwork regime* when the following conditions are met

$$(\text{sparse input} \quad C_F f \lesssim 1) \quad \text{and} \quad (\text{low noise} \quad \beta^{-1} \ll \sqrt{f}) \quad \text{and} \quad (C_R \alpha \mathrm{e}^{-C_F f} \ll \sqrt{f}) \tag{2.79}$$

There is also an intermediate regime, when the last condition is not satisfied. In this case two subnetwork are coupled and this makes it harder to track analytically. We discuss

it qualitatively in the end of the section 2.4.4.

2.4.3 Uniform regime

The following is under the assumptions (2.78)

Let m^ν be the stochastic average activity of the recurrent network state variables for the recurrent dynamics described in (2.76). In uniform regime the average activity of the network m^ν is the only order parameter for the *mean field equation*. The mean field equation is

$$m^\nu = \frac{1}{M}\sum_{k=1}^{M} \tanh\left(\beta\left(C_R \alpha m^\nu + h_k^\nu\right)\right) \tag{2.80}$$

The standard deviation σ_h of h_k^ν is given from (2.38) by

$$\sigma_{h_k^\nu} = \sqrt{f(1-f)n_k^\nu} \tag{2.81}$$

Consider the mean field equation (2.80). When $\mu_h \ll \sigma_h$ (which is true for sufficiently large number of learned patterns P), this equation has three solutions (two attractors and one unstable) if the following conditions are met

$$\begin{aligned} &\beta^{-1} < C_R\alpha \\ &\sigma_h \lesssim \sqrt{\frac{2}{\pi}} C_R \alpha \end{aligned} \tag{2.82}$$

In the dense regime $C_F f \gg 1$, see (2.44), we can approximate the external current h_k^ν to be drawn from the normal distribution with mean μ_h and standard deviation σ_h

$$\sigma_h = \sqrt{f^2(1-f)C_F}$$

from (2.44).

The meaning of the first condition is that the effective temperature β^{-1} should be smaller than the typical recurrent network current scale $C_R\alpha$. The meaning of the second condition is that the noise in σ_h in the source should be smaller then the typical recurrent network current $C_R\alpha$.

The second condition needs to be imposed independently in the *low dynamical noise* limit $\beta^{-1} \ll \sigma_h$. In this limit we can replace $\tanh(\beta x)$ by by $\mathrm{sgn}(x)$ and apply (2.41) to get (2.82). In the *high dynamical noise* limit $\beta^{-1} \gg \sigma_h$ the first condition automatically implies the second condition.

The graphical representation of the mean field equation under the above conditions is shown on figure 2.4.

The sigmoid curve on the plot represents the right-hand side of equation (2.80) as a function of the average activity m, that we denote by $f(m)$, and the straight line at slope 1 represents the left-hand side of the equation. The two stable solutions at $m \approx 1$ and $m \approx -1$ correspond to the two attractor states mentioned above. We want the network to evolve to the $m \approx 1$ state when the input pattern is positive (the feedforward currents h_k^ν are drawn from the distribution with a positive mean) and to the $m \approx -1$ state when the input pattern is negative (h_k^ν are drawn from the distribution with a negative mean). This becomes possible if we initialize the network close to $m = 0$ (see section 2.6), which is typically to the right of the point of unstable equilibrium m_u for positive patterns and to the left of m_u for negative patterns.

To proceed further, we will need to estimate the distribution of the value of average activity at the point of unstable equilibrium, m_u^ν over different realizations of $\{h_k^\nu\}$ from the same class of patterns η^ν. This can be done at two limiting cases that we call *high dynamical noise* $\beta^{-1} \gg \sigma_h$ and *low dynamical noise* $\beta^{-1} \ll \sigma_h$.

High dynamical noise

The results of this section are valid for both dense and sparse regimes. Under the following assumptions

$$\begin{aligned} & C_R\alpha > \beta^{-1} \\ & \text{for dense input:} \quad C_F f \gg 1 \quad \beta^{-1} \gg \sqrt{f^2(1-f)C_F} \\ & \text{for sparse input:} \quad C_F f \lesssim 1 \quad \beta^{-1} \gg \sqrt{f} \end{aligned} \tag{2.83}$$

The condition on β^{-1} follows from the expression for σ_h in (2.81). Since $n_k^\nu \in \mathbf{B}(C_F, f)$, we have in dense regime the typical $\sqrt{n_k^\nu} \simeq \sqrt{C_F f}$ and in sparse regime $f \ll 1$ and the typical non-zero $\sqrt{n_k^\nu} \simeq 1$.

The above conditions imply that near $m = 0$ and we can replace the hyperbolic tangent in the mean field equation with its argument to get the equation for unstable equilibrium m_u^ν

$$m_u^\nu = \frac{1}{M}\sum_{k=1}^{M} \beta(C_R\alpha m_u^\nu + h_k^\nu)$$

Solving this equation gives

$$m_u^\nu = -\frac{1}{C_R\alpha - \beta^{-1}}\frac{1}{M}\sum_{k=1}^{M} h_k^\nu$$

Which leads the following expressions for the mean μ_u and standard deviation σ_u of m_u

$$\begin{aligned} \mu_u &= -\frac{1}{C_R\alpha - \beta^{-1}}\mu_h \\ \sigma_u &= \frac{\sigma_h}{C_R\alpha - \beta^{-1}}\sqrt{\frac{1}{M} + \frac{C_F}{N}} \end{aligned} \tag{2.84}$$

Where the μ_h and σ_h^2 are the mean and the variance the feedforward current h_k^ν and for

large P are given by (see (2.38) and (2.44))

$$\mu_h = \frac{1}{\sqrt{P}} f(1-f) C_F \eta^\nu$$
$$\sigma_h^2 = \mathbf{cov}(h_k^\nu, h_k^\nu) = C_F f^2 (1-f) \tag{2.85}$$

To derive (2.84) we need the variance of the mean source current

$$\bar{h}^\nu = \frac{1}{M} \sum_{k=1}^{M} h_k^\nu \tag{2.86}$$

Like in (2.51) the variance of $\bar{h}^\nu$ is contributed by the diagonal terms and the non-diagonal terms, in the limit $M \to \infty$ approximated by

$$\mathbf{cov}(\bar{h}^\nu, \bar{h}^\nu) = \frac{1}{M} \mathbf{cov}(h_k^\nu, h_k^\nu) + \mathbf{cov}(h_k^\nu, h_l^\nu)_{k \neq l} \tag{2.87}$$

For the non-diagonal terms, neglecting $\frac{1}{P}$ corrections coming from the signal, using representation (2.56), we find in the same way as in the computation of (2.57)

$$\mathbf{cov}(h_k^\nu, h_l^\nu)_{k \neq l} = f(1-f) \langle n_{kl} \rangle \tag{2.88}$$

from the covariance of the z_{kl} terms. Here $\langle n_{kl} \rangle$ is the expectation value of the number of common active input neurons for readouts k and l (see details in section 2.3.5). Therefore, comparing with (2.38) we find

$$\mathbf{cov}(h_k^\nu, h_l^\nu)_{k \neq l} = \mathbf{cov}(h_k, h_k) \frac{\langle n_{kl} \rangle}{\langle n_k \rangle} \tag{2.89}$$

The $\langle n_{kl} \rangle$ in the limit $N, M \to \infty$ and finite C_F, f can be estimated as

$$\langle n_{kl} \rangle = fN \left(\frac{C_F}{N} \right)^2 = \frac{C_F^2}{N} f \tag{2.90}$$

which gives

$$\frac{\langle n_{kl} \rangle}{\langle n_k \rangle} = \frac{C_F}{N} \tag{2.91}$$

and therefore

$$\mathbf{cov}(h_k^\nu, h_l^\nu)_{k \neq l} = \frac{C_F}{N} \mathbf{cov}(h_k, h_k) \tag{2.92}$$

Hence (2.87) reduces to

$$\mathbf{cov}(\bar{h}^\nu, \bar{h}^\nu) = \left(\frac{1}{M} + \frac{C_F}{N} \right) \sigma_h^2 \tag{2.93}$$

and implies (2.84).

We can now compute the probability that the network of recurrent readouts evolves to the attractor that corresponds to the class of the input pattern. This happens when the initial state of the network, characterized by the average activity m_0 is on the correct side of the unstable equilibrium m_u (see figure 2.4, namely

$$(m_0^\nu - m_u^\nu)\eta^\nu > 0 \tag{2.94}$$

We will assume that the average activity of the recurrent network in initial state m_0 is distributed as if each unit was set to be in one of the two states with equal probabilities, namely

$$m_0 \sim \mathcal{N}(0, \frac{1}{M}) \tag{2.95}$$

for large M. We will discuss the initialization of the network further in the section 2.6.

Condition (2.94) is satisfied with probability

$$\text{prob}[(m_0^\nu - m_u^\nu)\eta^\nu > 0] = \frac{1}{2}\left(1 + \text{erf}\left(\frac{|\mu_u|}{\sqrt{2\left(\sigma_u^2 + \frac{1}{M}\right)}}\right)\right)$$

Which leads to the capacity of the network classifier in the uniform regime for high dynamical noise $\beta\sigma_h \ll 1$

$$P_{\max} = \frac{(1-f)}{2[\text{erf}^{-1}(1-2\epsilon)]^2}\frac{C_F M}{1 + \frac{M}{N}C_F + \frac{(C_R\alpha - \beta^{-1})^2}{C_F f^2(1-f)}} \tag{2.96}$$

When the last term in the denominator can be ignored ($C_R\alpha \approx \beta^{-1}$), the answer differs from the majority vote result for the dense case only by numerical factors $2/\pi$, while being valid even in the sparse regime. However, making the last term in the denominator negligible requires more and more fine tuning as f approaches zero.

Low dynamical noise and dense input

The results of this section are only valid in the dense case ($C_F f \gg 1$), when h_k^ν is a gaussian variable.

We consider the limit of low dynamical noise

$$\beta^{-1} \ll \sqrt{f^2(1-f)C_F} \tag{2.97}$$

which implies that for almost all the readouts $\beta h_k^\nu \gg 1$ we can replace hyperbolic tangent in (2.80) by the sign function to get

$$m_u^\nu = g(m_u^\nu) \tag{2.98}$$

where

$$g(m) = \frac{1}{M}\sum_{k=1}^{M} \text{sign}(C_R \alpha m + h_k^\nu) \tag{2.99}$$

This is a stochastic function over different realizations of $\{h_k^\nu\}$. The mean $\langle g(m)\rangle$ can be found by integrating over the distribution of h_k^ν (see (2.85))

$$\langle g(m)\rangle = \text{erf}\left(\frac{C_R \alpha m + \mu_h}{\sqrt{2}\sigma_h}\right) \tag{2.100}$$

Where, again μ_h and σ_h are the mean and standard deviation of h_k^ν respectively as in (2.85).

Now it is easy to derive the second condition of (2.82) for the existence of three solutions to the mean field equation (2.80). Indeed, in the low noise approximation, the equation becomes:

$$m = \text{erf}\left(\frac{C_R \alpha m + \mu_h}{\sqrt{2}\sigma_h}\right)$$

and it has three solutions if the derivative of the right hand side at $m = 0$ is larger than one (see figure 2.4). Assuming $\mu_h/\sigma_h \ll 1$, which is true for sufficiently large number of patterns, and recalling that the derivative of the error function at zero is equal to $2/\sqrt{\pi}$ immediately leads the second line of (2.82).

We now return to estimating the distribution of m_u, the unstable solution. For $\mu_h \ll \sigma_h$, which is always the case if the number of stored patterns P is large enough, we assume that $C_R \alpha m_u$ is also small compared to σ_h and check the self-consistency later. Then, we can use the approximation for the error function at small arguments to get

$$\langle g(m)\rangle = \sqrt{\frac{2}{\pi}}\frac{C_R \alpha m + \mu_h}{\sigma_h} \tag{2.101}$$

the variance of $g(m)$ can be written as as sum of the diagonal and the non-diagonal terms

$$\mathbf{cov}(g(m), g(m)) = \frac{1}{M} + \mathbf{cov}(\mathrm{sgn}(C_R\alpha m + h_k), \mathrm{sgn}(C_R\alpha m + h_l))\rangle_{k\neq l} \tag{2.102}$$

which is similar to the expression (2.51) for variance of $\frac{1}{M}\sum_{k=1}^{M}\mathrm{sgn}(h_k)$ computed previously in (2.67), with the only difference that here the distribution of h_k is shifted by $C_R\alpha m$. However, because the mean of the distribution did not change the final result and $C_R\alpha m_u + \mu_h$ is still negligible compared to σ_h, we can write

$$\mathbf{cov}(g(m), g(m)) = \frac{1}{M} + \frac{\varphi_{C_F,f}}{N} \tag{2.103}$$

As a sum of large number M of weakly correlated terms, $g(m)$ can be assumed to be normally distributed and can be written as

$$g(m) = \sqrt{\frac{2}{\pi}}\frac{C_R\alpha m + \mu_h}{\sigma_h} + \sqrt{\frac{1}{M} + \frac{\varphi_{C_F,f}}{N}}z \tag{2.104}$$

Where z is a gaussian variable with zero mean and unit variance.

Plugging the expression for $g(m)$ into the equation (2.98), and solving for m_u we get

$$m_u = -\frac{1}{\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sigma_h} - 1}\sqrt{\frac{2}{\pi}}\frac{\mu_h}{\sigma_h} + \frac{1}{\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sigma_h} - 1}\sqrt{\frac{1}{M} + \frac{1}{N}\varphi_{C_F,f}}z \tag{2.105}$$

where $\varphi_{C_F,f}$ is (2.62)(2.63).

So, the distribution of m_u which is the unstable (close to zero) solution to the equation

(2.98) is normal with the mean and standard deviation

$$\mu_u = -\frac{1}{\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sigma_h} - 1}\sqrt{\frac{2}{\pi}}\frac{\mu_h}{\sigma_h}$$
$$\sigma_u = \frac{1}{\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sigma_h} - 1}\sqrt{\frac{1}{M} + \frac{1}{N}\varphi_{C_F,f}} \tag{2.106}$$

where μ_h and σ_h can be taken from (2.85).

Again, assuming the average activity of the initial state of the network $m_0 \sim \mathcal{N}(0, \frac{1}{M})$, leads to classification capacity, obtained similarly to (2.96)

$$P_{\max} = \frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi}\frac{C_F M}{1 + \frac{M}{N}\frac{2}{\pi}C_F + \left(\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sigma_h} - 1\right)^2} \tag{2.107}$$

where we have used the dense regime approximations for $\varphi_{C_F,f}$ (2.63).

The capacity is decreased compared to the majority vote scenario (2.70).

2.4.4 Two subnetworks regime

The results of this section are valid if

$$(\text{sparse input} \quad C_F f \lesssim 1) \quad \text{and} \quad (\text{low noise} \quad \beta^{-1} \ll \sqrt{f}) \quad \text{and} \quad (C_R\alpha \mathrm{e}^{-C_F f} \ll \sqrt{f}) \tag{2.108}$$

In this section we consider sparse input representation, which is characterized by the average number of active units connected to a single readout being of the order, or even less than one $C_F f \lesssim 1$. And the strength of feedforward connections are so strong, that the state of the units that receive a non-zero feedforward input are set at $s_k^\nu = \mathrm{sign}\,(h_k^\nu)$ and neither the dynamical noise nor the recurrent input is enough to flip them. This regime might be relevant for describing the mammalian hippocampal network, where the activity

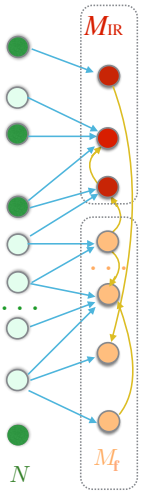

Figure 2.3: Illustration for the two subnetwork regime. The orange circles represent free units of the intermediate layer, all their feedforward inputs are silent for the given input patterns. They participate in the recurrent dynamics analyzed in section 2.4.4 The red circles denote input receiving units in the same layer.

level in the dentate gyrus was estimated to be around 1%-5%, [23] and the feedforward connectivity from dentate gyrus to CA3 to be around $C_F = 50$ incoming synapses per CA3 cell [31].

In this case a substantial fraction of input currents h_k^ν is equal to zero, and it would be wrong to assume that h_k^ν comes from a normal distribution as we did before.

We consider the readouts with zero feedforward input separately. We call them *free units* and denote by $\mathbf{M}_{\rm f}^\nu \subset \mathbf{M}$. Notice that that the identities of free units will depend on the input pattern ν.

We assume that the feedforward connections are much stronger than the recurrent ones, so that those units that receive a non-zero input. We call them *input receivers* and denote by $\mathbf{M}_{\rm IR}^\nu$. The input receivers are enslaved to the input: the state of the unit k in response to input pattern ν is given by $s_k^\nu = \mathrm{sign}\,(h_k^\nu)$.

In the Poisson regime $C_F \gg 1$ and $C_F f$ finite (2.43) the average number is the number

of free readout units is

$$M_{\rm f} = |\mathbf{M}_{\rm f}| = e^{-C_F f} M \tag{2.109}$$

and the average number of input receiving units is

$$M_{\rm IR} = |\mathbf{M}_{\rm IR}| = M - M_{\rm f} = (1 - e^{-C_F f}) M \tag{2.110}$$

The typical number of free units linked to a given free unit is $e^{-C_F f} C_R$.

Then the mean field equation for the subnetwork of free units reads:

$$\tilde{m}^\nu = \frac{1}{M_{\rm f}} \sum_{k=1}^{M_{\rm f}} \tanh\left(\beta\left(C_R \alpha e^{-C_F f} \tilde{m}^\nu + H_k^\nu\right)\right) \tag{2.111}$$

and $\tilde{m}^\nu$ is the average activity of the subnetwork of free readouts. The index k runs over all the free units and the index l runs over the input receivers. The H_k^ν (not to be confused with h_k^ν) denotes the external input to the subnetwork of free units coming from the enslaved input receivers

$$H_k^\nu = \sum_{l=1}^{M_{\rm IR}} \alpha J_{kl} \mathrm{sign}\left(h_l^\nu\right) \tag{2.112}$$

the summation is, again, over the input receiving units.

On average, the free unit k receives C_R inputs, and $(1 - e^{-C_F f}) C_R$ of them come from input receivers. So the above sum will have on average $C_R(1 - e^{-C_F f})$ terms. Assuming that this is a large number, H_k^ν is a gaussian variable with the mean and standard deviation given in the leading order by

$$\begin{aligned} \mu_H &= \alpha C_R (1 - e^{-C_F f}) \langle \mathrm{sign}\left(h_k^\nu\right) \rangle_{n_k \neq 0} \\ \sigma_H &= \alpha \sqrt{C_R (1 - e^{-C_F f})} \end{aligned} \tag{2.113}$$

The conditions for having three fixed points of the free subnetwork dynamics are analogous to (2.82)

$$\begin{aligned} &\beta^{-1} < C_R \alpha e^{-C_F f} \\ &\sqrt{C_R}\frac{e^{-C_F f}}{\sqrt{1-e^{-C_F f}}} > f(\beta) = O(1) \end{aligned} \tag{2.114}$$

And in the limit $\beta\sigma_h \gg 1$, the second condition is $\sqrt{C_R}\frac{e^{-C_F f}}{\sqrt{1-e^{-C_F f}}} > \sqrt{\frac{\pi}{2}}$

To find the point of unstable equilibrium $\tilde{m}_u$ we consider two limiting regimes.

High free recurrent dynamical noise

Suppose we are in the regime of high free recurrent dynamical noise $\beta^{-1} \gg \sigma_H$ which means that

$$\beta^{-1} \gg \alpha\sqrt{C_R(1-e^{-C_F f})} \tag{2.115}$$

Such regime allows three fixed points by (2.82) if β^{-1} is in the window

$$\alpha\sqrt{C_R(1-e^{-C_F f})} \ll \beta^{-1} < C_R \alpha e^{-C_F f} \tag{2.116}$$

This window is non-empty if C_R is sufficiently large but $C_F f$ of order of 1

$$C_R > e^{2C_F f} - e^{C_F f} \tag{2.117}$$

In this regime of high free noise (2.115) the mean field equation for the free subnetwork (2.111) can be approximated by:

$$\tilde{m}_u^\nu = \beta C_R \alpha e^{-C_F f}\tilde{m}_u + \beta\frac{1}{M_{\rm f}}\sum_{k=1}^{M_{\rm f}}\sum_{l=1}^{M_{\rm IR}} J_{kl}\alpha\,\mathrm{sign}\,(h_l^\nu) \tag{2.118}$$

Each input receiving unit has C_R outgoing connections and approximately $e^{-C_F f}C_R$ of

them terminate on a free unit. Since the strength of these connections is α, the above equation can be rewritten as

$$\tilde{m}_u = \beta C_R \alpha e^{-C_F f} \tilde{m}_u + \beta C_R \alpha e^{-C_F f} \frac{1}{M_{\rm f}} \sum_{l=1}^{M_{\rm IR}} \mathrm{sign}\,(h_l) \tag{2.119}$$

Recalling that $M_{\rm f} = M e^{-C_F f}$ and $M_{\rm IR} = M(1 - e^{-C_F f})$ and solving for $\tilde{m}_u$ leads

$$\begin{aligned} \tilde{m}_u^\nu &= -\frac{\beta C_R \alpha}{\beta C_R \alpha e^{-C_F f} - 1}(1 - e^{-C_F f})\bar{r}^\nu & (2.120)\\ \bar{r}^\nu &= \frac{1}{M_{\rm IR}} \sum_{k=1}^{M_{\rm IR}} \mathrm{sgn}(h_k^\nu) & (2.121) \end{aligned}$$

To find the mean and variance of $\bar{r}^\nu$ we need to compute two quantities: $\langle \mathrm{sign}\,(h_k^\nu)\rangle_{k\in\mathbf{M}_{\rm IR}}$ and $\mathbf{cov}(\mathrm{sign}\,(h_k^\nu), \mathrm{sign}\,(h_l^\nu))_{k\neq l;k,l\in\mathbf{M}_{\rm IR}}$. Since both averages are zero for zero h_k^ν and the probability of h_k^ν to differ from zero is given by $(1 - e^{-C_F f})$

$$\begin{aligned} \langle \mathrm{sign}\,(h_k^\nu)\rangle_{k\in\mathbf{M}_{\rm IR}} &= \frac{1}{1 - e^{-C_F f}} \langle \mathrm{sign}\,(h_k^\nu)\rangle_{k\in\mathbf{M}} \\ \mathbf{cov}(\mathrm{sign}\,(h_k^\nu), \mathrm{sign}\,(h_l^\nu))_{k\neq l;k,l\in\mathbf{M}_{\rm IR}} &= \frac{1}{(1 - e^{-C_F f})^2} \mathbf{cov}(\mathrm{sign}\,(h_k^\nu), \mathrm{sign}\,(h_l^\nu))_{k\neq l;k,l\in\mathbf{M}} \end{aligned} \tag{2.122}$$

which leads the expressions for the mean and the variance of $\bar{r}^\nu$ expressed in term of r^ν

$$\langle \bar{r^\nu} \rangle = \frac{1}{1 - e^{-C_F f}} \langle r^\nu \rangle = \frac{1}{1 - e^{-C_F f}} \sqrt{\frac{2}{\pi}} \frac{\langle \sqrt{n_k^\nu} \rangle}{\sqrt{Pf}} \eta^\nu \tag{2.123}$$

$$\mathbf{cov}(\bar{r}^\nu, \bar{r}^\nu) = \frac{1}{M_{\rm IR}} + \frac{\varphi_{C_F,f}}{(1 - e^{-C_F f})^2} \frac{1}{N} \tag{2.124}$$

Here r^ν denotes the average $\mathrm{sign}\,(h_k)^\nu$ over all readouts, considered in the section 2.3.5 for which the mean and variance is computed in (2.67)(2.68) and $\varphi_{C_F,f}$ from 2.62. We ignored

the factor $\sqrt{1-f}$ in $\langle r^\nu \rangle$, because f is small in the sparse regime.

From the mean and variance of $\bar{r}^\nu$ follows the mean and variance of the distribution of the $\tilde{m}_u^\nu$ in (2.120) by the linear relation. Then, following the same line of arguments as in section 2.4.3 around equation (2.96), but applying to the free recurrent network with the number of units $M_{\rm f} = Me^{-C_F f}$, leads the expression for the capacity of network classifier with sparse input representation $C_F f \lesssim 1$ and strong feedforward synapses:

$$\begin{aligned} P_{\max} &= \frac{1}{\pi[\mathrm{erf}^{-1}(1-2\epsilon)]^2} \frac{MC_F^2 f}{\gamma + \frac{M}{N} C_F^2 f} \\ \gamma &= 1 - \frac{2\beta C_R \alpha - e^{C_F f}}{(\beta C_R \alpha)^2} \end{aligned} \tag{2.125}$$

We used sparse regime $\langle \sqrt{n_k^\nu} \rangle = fC_F$ in (2.45) and sparse $\varphi_{C_F,f} = fC_F^2$ from 2.65. The parameter γ was computed as follows:

$$\gamma = \frac{M}{M_{\rm f}} \left(\frac{\beta C_R \alpha}{\beta C_R \alpha e^{-C_F f} - 1} \right)^2 + \frac{M}{M_{\rm IR}} (1 - e^{-C_F f})^2 \tag{2.126}$$

where the first term comes from the initialization noise (2.95) on the free subnetwork, and the factor that multiplies $\frac{M}{M_{\rm f}}$ in the first term comes from the factor (2.120) relating $\tilde{m}_u^\nu$ to $\bar{r}^\nu$, and the second term comes from the principal contribution $\frac{1}{M_{\rm IR}}$ to $\mathbf{cov}(\bar{r}^\nu, \bar{r}^\nu)$ in (2.124).

Introducing a new parameter

$$\Delta = \beta C_R \alpha e^{-C_F f} - 1 \tag{2.127}$$

allows to express the relation of γ to other parameters in the problem in a more clear way:

$$\gamma = 1 - e^{-C_F f} \left(1 - \frac{\Delta^2}{(\Delta+1)^2} \right), \qquad 1 - e^{-C_F f} < \gamma < 1 \tag{2.128}$$

For the bounds on γ we used condition (2.114). The first condition for having three fixed points (2.114) in terms of Δ is $\Delta > 0$. The lower bound on γ corresponds to majority vote with units that receive zero input not voting (and not increasing the variance). The $\gamma = 1$ corresponds to majority rule.

When $C_F f \ll 1$, γ can be made of the order $C_F f$ and this will allow for very small f without sacrificing the capacity. In this regime

$$P_{\max} = \frac{1}{\pi[\mathrm{erf}^{-1}(1-2\epsilon)]^2} \frac{MC_F}{1+\frac{MC_F}{N}} \tag{2.129}$$

However, this would mean $\beta C_R \alpha = 1 + \Delta$ with $\Delta \ll 1$, which is kind of a fine tuning.

The counterintuitive increase in classification capacity relative to majority vote can be clarified. The signal is now $\bar{r}^\nu$, the average of the sign of the input current taken over only those units that receive a non-zero input, while in the majority rule scenario, the average r^ν was taken over all the units. So the signal to noise ratio of the information coming form the input layer is improved.

However, the free neurons that do not receive any information and whose state is completely random in the case of majority vote scenario, still contribute to the noise in the recurrent readout scheme by their random initial state. Let us now consider the subnetwork of free units in the recurrent readout scheme. When the noise is too low $\beta C_R \alpha e^{-C_F f} \gg 1$, the fluctuations of the external input, coming from the input receivers, is as important as fluctuations in the initial state and there is no improvement compared to the majority vote, as seen from (2.125). When the noise increases (β decreases), but no too much, so that there are still three fixed points of the recurrent dynamics ($\beta C_R \alpha e^{-C_F f} > 1$) and the noise is not enough to flip input receiving units, the situation changes. Now the role of initial condition is diminished relative to the external input that is not affected by noise, and the decrease in the variance of external signal becomes more important than the same amount

of variance in the initial condition. This makes it possible to outperform the majority rule.

If we now fix the total number of units (input plus readout), and divide them between layers in the optimal way as described in section 2.3.6, the expression for capacity becomes

$$P_{\max} = C_F^2 f \frac{1}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2 \pi} \frac{M+N}{\left(\gamma^{\frac{1}{2}} + \varphi_{C_F,f}^{\frac{1}{2}}\right)^2}$$

and the optimal relation between the number of readouts M and the number of input units N is

$$N = M\gamma^{-\frac{1}{2}} \varphi_{C_F,f}^{\frac{1}{2}} \tag{2.130}$$

Low free recurrent dynamical noise

Now we assume the regime of *low free recurrent dynamical noise* $\beta^{-1} \ll \sigma_H$ where $\sigma_H = \alpha\sqrt{C_R(1-e^{-C_F f})}$ from (2.113) is the variance of the current H (2.112) from input receiving neurons. In this regime the mean field equation (2.111) can be rewritten as

$$\tilde{m}_u^\nu = \langle g(\tilde{m}_u^\nu) \rangle \tag{2.131}$$

where

$$g(\tilde{m}_u^\nu) = \frac{1}{M_\mathrm{f}} \sum_{k=1}^{M_\mathrm{f}} g_k(\tilde{m}_u^\nu), \qquad g_k(\tilde{m}_u^\nu) = \mathrm{sign}\left(C_R e^{-C_F f} \tilde{m}_u^\nu + \tilde{H}_k^\nu\right) \tag{2.132}$$

Here $\tilde{H}_k$ is rescaled by α^{-1} compared to H_k of (2.112)

$$\tilde{H}_k^\nu = \sum_{l=1}^{M_\mathrm{IR}} J_{kl} \mathrm{sign}\,(h_l^\nu) \tag{2.133}$$

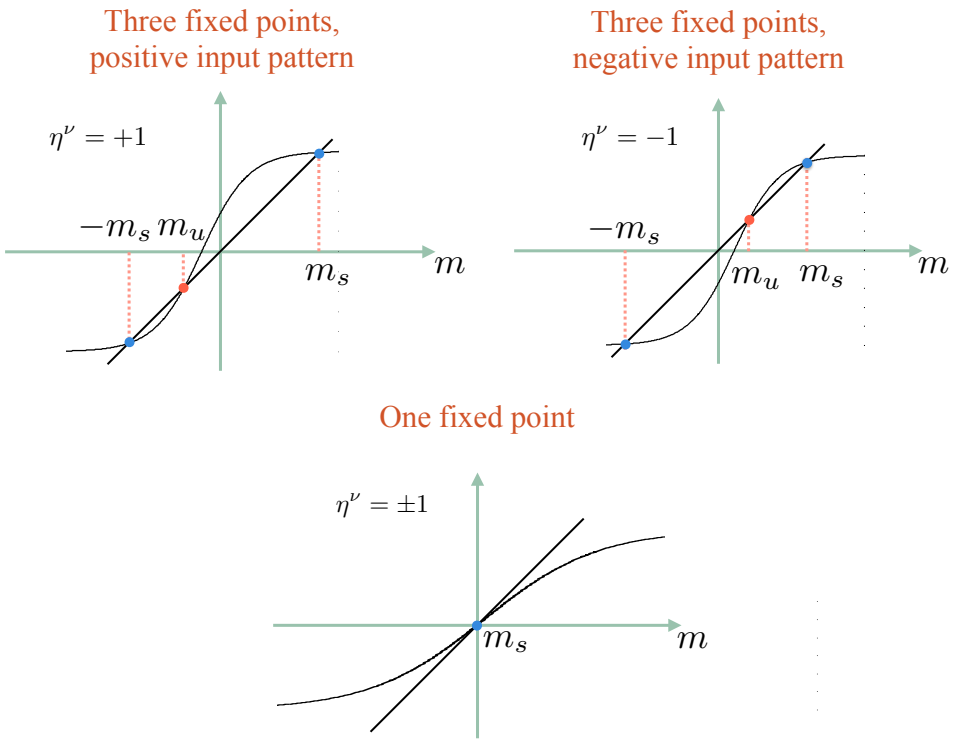

Figure 2.4: Graphical solution for the mean field equation 2.80. Two cases are possible depending on the values of equation parameters. The lower panel represents a regime with only one solution, which is stable. This regime is not suitable for our purposes. In the top two panels the equation has two stable solutions (blue dots) and the external input determines the location of the intermediate unstable solution (the red dot). If initialized at $m = 0$, the network will evolve to the right stable solution for a "positive" input pattern and to the left stable solution for a negative input pattern. m_s denotes the absolute value of average network activity for the stable solutions, and m_u - for the unstable one.

In the averaging in the mean field equation (2.131) the input current $\tilde{H}_k^\nu$ defined by h_l^ν is treated as a fixed external parameter. Consequently, mean field equation defines average activity $\tilde{m}_u^\nu$ as implicit function of all inputs (h_l^ν). This implicit function and the probability distrubution of input variables (h_l^ν) induces the probability distribution on the mean activity $\tilde{m}_u^\nu$. We proceed to find it, and with this understanding we omit explicit pattern reference ν but assuming it implicitly in all equations below.

On average each free subnetwork unit receives $C_R(1 - e^{-C_f f})$ connections from input receivers. Therefore the mean of $\tilde{H}$ is

$$\langle \tilde{H}_k \rangle = C_R(1 - e^{-C_F f}) \langle \text{sign}\,(h_l) \rangle_{l \in \mathbf{M}_{\text{IR}}} \tag{2.134}$$

and the variance of $\tilde{H}$ is

$$\mathbf{cov}(\tilde{H}, \tilde{H}) = C_R(1 - e^{-C_F f}) \tag{2.135}$$

For small $\tilde{m}_u$ from (2.132) and (2.41) we compute the mean of $g(\tilde{m}_u)$

$$\langle g(\tilde{m}_u) \rangle = \sqrt{\frac{2}{\pi}} \frac{C_R e^{-C_F f} \tilde{m}_u + C_R(1 - e^{-C_F f}) \langle \text{sign}\,(h_l) \rangle_{l \in \mathbf{M}_{\text{IR}}}}{\sqrt{C_R(1 - e^{-C_F f})}} \tag{2.136}$$

Next we want to compute the variance of $g(\tilde{m}_u)$. For this, we want to compute the correlation of individual terms k and p in (2.132)

$$\mathbf{cov}(g_k(\tilde{m}_u), g_p(\tilde{m}_u))_{k \neq p} = \mathbf{cov}\left(\text{sign}\left(C_R e^{-C_F f} \tilde{m}_u + \tilde{H}_k\right), \text{sign}\left(C_R e^{-C_F f} \tilde{m}_u + \tilde{H}_p\right)\right)_{k \neq p} \tag{2.137}$$

There are two cases of contributions to the correlation, that we will call case I and case II, see figure 2.5.

The case I contribution to this correlation comes from the free units k and p being connected to the same input receiving unit r. We neglect the probability that the overlap

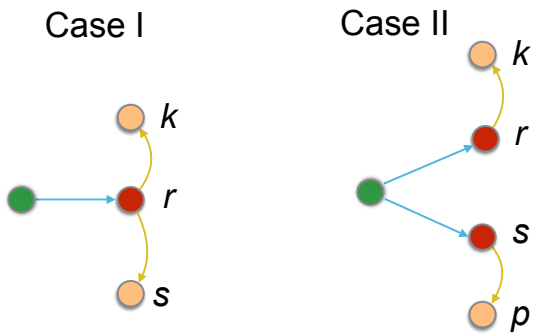

Figure 2.5: Two sources of input correlations for the subnetwork of free units (orange circles), referred in the text as case I and case II. On the left diagram two free units are connected to the same input receiving unit in the readout layer (red circle). On the right diagram there is no input receiving unit that is connected to both free units, but the correlation arises from an active unit in the input layer (green circle), which is connected to the two free units indirectly. The probabilities to observe these cases and their contribution to input correlations are computed in section 2.4.4, subsection "Low free recurrent dynamical noise"

will be over more than one input receiving unit since we keep connectivity C_R fixed when we scale the number of units M. The probability of case I contribution is

$$p_{\mathrm{I}} = \frac{C_R}{M} C_R (1 - e^{-C_F f}) \tag{2.138}$$

The case II contribution comes from the possibility that there is an input layer unit that is active and connects to both via different input receiving neurons. The probability of case II contribution

$$p_{\mathrm{II}} = f \frac{C_F^2}{N} C_R^2 (1 - e^{-C_F f})^2 \tag{2.139}$$

To derive these probabilities, recall that the average number of connections from a free unit k to input receiving units is $C_R(1 - e^{-C_F f})$ and the probability of any two readouts to be connected to the same active input is $f C_F^2 / N$ by (2.55).

In the case I the relevant correlation is

$$\mathbf{cov}^{\mathrm{I}}(g_k(\tilde{m}_u), g_p(\tilde{m}_u)) = \mathbf{cov}\left(\begin{array}{c}\operatorname{sign}\left(C_R e^{-C_F f}\tilde{m}_u + \operatorname{sign}(h_r) + \sqrt{C_R(1-e^{-C_F f})}z_k\right), \\ \operatorname{sign}\left(C_R e^{-C_F f}\tilde{m}_u + \operatorname{sign}(h_r) + \sqrt{C_R(1-e^{-C_F f})}z_p\right)\end{array}\right)_{k\neq p} = \\ = \frac{2}{\pi}\mathbf{cov}\left(\frac{C_R e^{-C_F f}\tilde{m}_u + \operatorname{sign}(h_r)}{\sqrt{C_R(1-e^{-C_F f})}}, \frac{C_R e^{-C_F f}\tilde{m}_u + \operatorname{sign}(h_r)}{\sqrt{C_R(1-e^{-C_F f})}}\right) = \frac{2}{\pi}\frac{1}{C_R(1-e^{-C_F f})} \tag{2.140}$$

We approximated $C_R - 1$ by C_R and used error function integral (2.136, 2.41) at small argument on standard Gaussian variables z_k and z_p to transform the first line to the second line.

In the case II the relevant correlation

$$\begin{aligned}&\mathbf{cov}^{\mathrm{II}}(g_k(\tilde{m}_u), g_p(\tilde{m}_u)) = \\ &= \mathbf{cov}\left(\begin{array}{c}\operatorname{sign}\left(C_R e^{-C_F f}\tilde{m}_u + \operatorname{sign}(h_r) + \sqrt{C_R(1-e^{-C_F f})}z_k\right), \\ \operatorname{sign}\left(C_R e^{-C_F f}\tilde{m}_u + \operatorname{sign}(h_s) + \sqrt{C_R(1-e^{-C_F f})}z_p\right)\end{array}\right)_{k\neq p, r\neq s} = \\ &= \frac{2}{\pi}\left\langle \frac{C_R e^{-C_F f}\tilde{m}_u + \operatorname{sign}(h_r)}{\sqrt{C_R(1-e^{-C_F f})}}\frac{C_R e^{-C_F f}\tilde{m}_u + \operatorname{sign}(h_s)}{\sqrt{C_R(1-e^{-C_F f})}}\right\rangle = \\ &= \frac{2}{\pi}\frac{1}{C_R(1-e^{-C_F f})}\frac{2}{\pi}\left\langle \tan^{-1}\sqrt{\frac{1}{(n_r+1)(n_s+1)-1}}\right\rangle_{n_r,n_s\in\mathbf{B}(C_F-1,f)} = \\ &= \frac{2}{\pi}\frac{1}{C_R(1-e^{-C_F f})}\frac{\varphi_{C_F,f}}{f C_F^2}\end{aligned} \tag{2.141}$$

where n_r and n_s are from binomial distribution on $C_F - 1$ trials with probability f computed as in (2.62) from the correlation of the $\operatorname{sign}(h_r)$ and $\operatorname{sign}(h_s)$.

Now we can compute 2.137 in the leading order as $p_{\mathrm{I}}, p_{\mathrm{II}}$ probability weighted sum of

the contributions from case I and case II:

$$\mathbf{cov}(g_k(\tilde{m}_u), g_p(\tilde{m}_u)) = p_{\mathrm{I}}\mathbf{cov}^{\mathrm{I}}(g_k(\tilde{m}_u), g_p(\tilde{m}_u)) + p_{\mathrm{II}}\mathbf{cov}^{\mathrm{II}}(g_k(\tilde{m}_u), g_p(\tilde{m}_u)) = \\ = \frac{2}{\pi}\frac{C_R}{M} + \frac{2}{\pi}\frac{\varphi_{C_F,f}}{N}C_R(1 - e^{-C_f f}) \quad (2.142)$$

At the diagonal terms we have simply

$$\mathbf{cov}(g_k(\tilde{m}_u), g_k(\tilde{m}_u)) = 1 \quad (2.143)$$

Alltogether, combining the contribution from diagonal and non-diagonal terms as in (2.87) we find

$$\mathbf{cov}(g(\tilde{m}_u), g(\tilde{m}_u)) = \frac{1}{M_{\mathrm{f}}} + \frac{2}{\pi}C_R\left(\frac{1}{M} + \frac{\varphi_{C_F,f}}{N}(1 - e^{-C_f f})\right) \quad (2.144)$$

We will assume that that $C_F f \lesssim 1$ so that $M_{\mathrm{f}} \simeq M$ and that even though C_R does not scale linearly with M, N, P still

$$C_R e^{-C_F f} \gg 1 \quad (2.145)$$

then we can, in fact, drop the diagonal term in (2.144) and take the approximation

$$\mathbf{cov}(g(\tilde{m}_u), g(\tilde{m}_u)) = \frac{2}{\pi}C_R\left(\frac{1}{M} + \frac{\varphi_{C_F,f}}{N}(1 - e^{-C_f f})\right) \quad (2.146)$$

Given the mean (2.136) and the variance (2.146) of $g(\tilde{m}_u)$, and recalling that $g(\tilde{m}_u)$ is a sum of large number of random variables with low correlation we approximate $g(\tilde{m}_u)$ in terms of standard Gaussian random variable z and plug into the mean field equation

(2.131), restoring the label ν

$$\tilde{m}_u^\nu = \sqrt{\frac{2}{\pi}} \frac{C_R e^{-C_F f} \tilde{m}_u^\nu + C_R(1 - e^{-C_F f}) \langle \text{sign}\,(h_l^\nu) \rangle_{l \in \mathbf{M}_{\text{IR}}}}{\sqrt{C_R(1 - e^{-C_F f})}} + (\mathbf{cov}(g(\tilde{m}_u^\nu), g(\tilde{m}_u^\nu)))^{\frac{1}{2}} z \tag{2.147}$$

The solution is

$$\tilde{m}_u^\nu = \mu_u + \sigma_{\tilde{m}_u^\nu} z \tag{2.148}$$

where under the assumption (2.145)

$$\mu_u^\nu = -\sqrt{\frac{2}{\pi}} e^{C_F f} \frac{\langle \sqrt{n} \rangle}{\sqrt{Pf}} \eta^\nu \tag{2.149}$$

(small f approximation to (2.41))

and

$$\sigma^2_{\tilde{m}_u^\nu} = e^{2C_F f}(1 - e^{-C_F f}) \left(\frac{1}{M} + \frac{\varphi_{C_F, f}}{N}(1 - e^{-C_F f}) \right) \tag{2.150}$$

for which we used (2.146).

To find the pattern capacity we need to add the initialization noise on M_{f} variables like in (2.95) to get effective variance of $\tilde{m}_u^\nu$

$$\begin{aligned} \sigma^2_{\text{init}} + \sigma^2_{\tilde{m}_u} &= \frac{1}{M_{\text{f}}} + e^{2C_F f}(1 - e^{-C_F f}) \left(\frac{1}{M} + \frac{\varphi_{C_F, f}}{N}(1 - e^{-C_f f}) \right) = \\ &= e^{2C_F f} \left(\frac{1}{M} + \frac{\varphi_{C_F, f}}{N}(1 - e^{-C_f f})^2 \right) \end{aligned} \tag{2.151}$$

and require that the initial state of the network is on the correct side of unstable equilibrium with probability $1 - \epsilon$. Again, the difference difference between the initial state m_0 and the unstable equilibrium m_u can be assumed to be Gaussian with the mean (2.149) and

variance (2.151), which leads

$$P_{\max} = \frac{\langle\sqrt{n}\rangle^2}{f} \frac{1}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi} \frac{M}{1+\frac{M}{N}(1-\mathrm{e}^{-C_F f})^2 \varphi_{C_F,f}} \tag{2.152}$$

or in the sparse approximation (2.42) for $\langle\sqrt{n}\rangle$ and $\varphi_{C_F,f}$

$$P_{\max} = \frac{1}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi} \frac{MC_F^2 f}{1+\frac{M}{N}(1-\mathrm{e}^{-C_F f})^2 C_F^2 f} \tag{2.153}$$

Compared to the majority rule this regime gives advantage for relatively low number of inputs N, while the high noise regime (2.125) gives advantage for relatively low number of readouts M.

Besides the uniform and two subnetwork regimes, there is also an intermediate case that we do not analyze here. It is realized when the relative strength of the feedforward and recurrent connections is such, that input receiving units are follow the sign of the external input in the beginning of patterns presentation, when the recurrent network is in the disordered state, but are flipped when the subnetwork of free units synchronizes and the recurrent inputs become more consistent.

2.5 Summary of limited connectivity results

uniform high noise (2.96)

$$P_{\max} = \frac{(1-f)}{2[\mathrm{erf}^{-1}(1-2\epsilon)]^2} \frac{MC_F}{1+\frac{M}{N}C_F + \frac{(C_R\alpha-\beta^{-1})^2}{C_F f^2(1-f)}} \tag{2.154}$$

uniform low noise, dense (2.107)

$$P_{\max} = \frac{1-f}{\pi[\mathrm{erf}^{-1}(1-2\epsilon)]^2}\frac{MC_F}{\gamma_{\mathrm{un}} + \frac{M}{N}\frac{2}{\pi}C_F} \tag{2.155}$$

$$\gamma_{\mathrm{un}} = 1 + \left(\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{(C_F f^2(1-f))^{\frac{1}{2}}} - 1\right)^2 \tag{2.156}$$

2-network high noise; sparse (2.125)

$$\begin{aligned} P_{\max} &= \frac{1}{\pi[\mathrm{erf}^{-1}(1-2\epsilon)]^2}\frac{MC_F^2 f}{\gamma + \frac{M}{N}fC_F^2} \\ \gamma &= 1 - \frac{2\beta C_R\alpha - e^{C_F f}}{(\beta C_R\alpha)^2} \end{aligned} \tag{2.157}$$

2-network low noise; sparse (2.153)

$$P_{\max} = \frac{1}{\pi[\mathrm{erf}^{-1}(1-2\epsilon)]^2}\frac{MC_F^2 f}{1 + \frac{M}{N}(1-\mathrm{e}^{-C_F f})^2 C_F^2 f} \tag{2.158}$$

For small f the results are very suggestive when presented in terms of the *total number of feedforward connections* $\mathcal{C}_F = MC_F$ and the *density parameter of input representation* $\lambda = C_F f$. Denote once and for all a chosen *error tolerance parameter*

$$c_\epsilon = \frac{1}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2} \tag{2.159}$$

Majority dense (2.70):

$$\boxed{P_{\max} = \frac{c_\epsilon}{\pi}\frac{\mathcal{C}_F}{1 + \frac{2}{\pi}\mathcal{C}_F/N}} \tag{2.160}$$

Majority sparse (2.71):

$$\boxed{P_{\max} = \frac{c_\epsilon}{\pi}\frac{\mathcal{C}_F\lambda}{1 + \mathcal{C}_F\lambda/N}} \tag{2.161}$$

Recurrent uniform high noise (2.96):

$$\boxed{P_{\max} = \frac{c_\epsilon}{2} \frac{\mathcal{C}_F}{1 + \mathcal{C}_F/N}} \tag{2.162}$$

if we maximize in the limit $C_R \alpha \beta \to 1$.

Recurrent uniform low noise, dense (2.107):

$$\boxed{P_{\max} = \frac{c_\epsilon}{\pi} \frac{\mathcal{C}_F}{1 + \frac{2}{\pi}\mathcal{C}_F/N}} \tag{2.163}$$

if we maximize in the limit $\sqrt{\frac{2}{\pi}} C_R \alpha \to \sigma_h$

Recurrent two-subnetwork high noise, sparse (2.125):

$$\boxed{P_{\max} = \frac{c_\epsilon}{\pi} \frac{\mathcal{C}_F}{\lambda^{-1}(1 - e^{-\lambda}) + \mathcal{C}_F/N}} \tag{2.164}$$

if we maximize by adjusting $\Delta \to 0$ see (2.128)

Recurrent two-subnetwork low noise; sparse (2.153):

$$\boxed{P_{\max} = \frac{c_\epsilon}{\pi} \frac{\mathcal{C}_F \lambda}{1 + \lambda(1 - e^{-\lambda})^2 \mathcal{C}_F/N}} \tag{2.165}$$

2.6 Network initialization

We have assumed so far that there is a way to initialize the network of the recurrently connected readouts at the disordered state (meaning that every unit is up or down with probability 1/2) every time before the feedforward input is on. This may seem problematic because the disordered state is unstable and it is not clear how the network is brought there. In this section we suggest two ways of achieving this with a population of interneurons, whose input is extremely noisy before the feedforward input is on ($\xi_i = 0, i = 1 \dots N$) and

is equal to zero or canceled by the threshold when the feedforward input is on. The noise from the interneurons in the absence of the feedforward input should be strong enough, so that the only fixed point of the recurrent dynamics is the disordered state ($m = 0$), it should also be exactly balanced, so that the fixed point is exactly at $m = 0$.

Spontaneous activity

One way to initialize the recurrent network of the intermediate layer close to $m = 0$ state (disordered state) is to have the input layer being spontaneously active before an input pattern is presented. We want the the feedforward input provided by the spontaneous activity to be such, that $m = 0$ is the only fixed point of the dynamics. As discussed before, this is equivalent to violating one or both of the conditions (2.82). Since we still want the conditions to be satisfied during the presentation of a test pattern, and we can not change the inverse temperature parameter β, we require

$$\sigma_h^{sp} \gtrsim \sqrt{\frac{2}{\pi}} C_R \alpha$$

If the spontaneous activity of the input neuron i is ξ_i^{sp}, the feedforward current into the readout k is given by

$$h_k^{sp} = \sum_{i \in \mathbf{I_k}} w_i \xi_i^{sp}$$

with $w_i = \frac{1}{\sqrt{P}} \sum_{\mu=1}^{P} (\xi_i^\mu - f)\eta^\mu$ which for large number of patterns is distributed normally with the mean zero and standard deviation $\sqrt{f(1-f)}$.

We assume that ξ_i^{sp} are also Gaussian with mean zero and standard deviation ζ. Then, the variance of the spontaneous feedforward input is

$$\sigma_h^{sp} = \sqrt{f(1-f)C_F}\zeta$$

It is important that the distribution of ξ_i^{sp} is symmetric. If it is not, the feedforward spontaneous input, and consequently the initial activity of the recurrent network m, will have a systematic bias and this might become a limiting factor for the classification capacity.

In the uniform regime this way of network initialization imposes an additional constraint on the relative strength of the recurrent and feedforward connections. During the spontaneous activity, the standard deviation of the feedforward input should be large relative to the recurrent input, and during the presentation of a pattern - small:

$$\sqrt{\frac{\pi}{2}}\sqrt{C_F(1-f)f^2} < C_R\alpha < \sqrt{\frac{\pi}{2}}\sqrt{C_F(1-f)f}\zeta \tag{2.166}$$

This constrained is easier to satisfy for low f.

Population of interneurons

The other alternative to set the network at $m = 0$ state before the input is on, involves two recurrently connected populations of N_{int} interneurons each: *excitatory* and *inhibitory*. Both of them receive strong excitatory input from the input layer. This input is only present when the input pattern is on and puts both populations in the ordered state ($\psi_j^+ = 1$, $\psi_j^- = 1$). In the absence of the feedforward input the only fixed point is the disordered state. Each recurrent readout receives exactly C_{int} connections of strength w_+ from the excitatory population, and C_{int} connections with strength w_- from the inhibitory population. It is crucial that the noise in the number of incoming connections C_{int} is much lower than $\sqrt{C_{\text{int}}}$ expected by random connectivity. Another requirement is that

$$|w^+ - w^-|C_{\text{int}} \ll \mu_h$$

where μ_h is the mean of feedforward input to a recurrent readout (see (2.24)).

The total synaptic current into recurrent readout k then now looks like

$$h_k^{\text{total}} = h_k^{\nu} + \sum_{l=1}^{M} \alpha J_{kl} s_l + w^+ \sum_{j=1}^{C_{\text{int}}} \psi_j^+ - w^- \sum_{j=1}^{C_{\text{int}}} \psi_j^- \tag{2.167}$$

When the feedforward input is off, the only external input is from interneurons and is zero mean with the standard deviation $(w^+ + w^-)\sqrt{C_{\text{int}}}$. If the condition $(w^+ + w^-))\sqrt{C_{int}} > \sqrt{2/\pi} C_R \alpha$ is met, there is only one stable fixed point for the dynamics of the recurrent readout population with the average activity m_0 close to zero.

It is required for the unbounded grows of classification capacity, that $\langle m_0 \rangle = 0$ when the mean is taken over different presentations of learned input pattern. In the current scheme, the noise of the disorder is the activity of the interneuron population and the final size effects will change from one pattern presentation to another, and the expectation value of m_0 will be exactly zero. We should only make sure that the variance of m_0 is not much larger than $\frac{1}{M}$, so that the classification capacity is not lowered much compared to (2.125).

Assuming that $\beta(w^+ + w^-)\sqrt{C_{\text{int}}} \gg 1$, we can approximate the $\tanh(\dots)$ by the $\text{sign}\,(()\dots)$ and estimate the variance of m_0 from the mean field equation for the population of recurrent readouts in the absence of the feedforward input (assuming $\beta(w^+ + w^-)\sqrt{C_{\text{int}}} \gg 1$):

$$m_0 = \frac{1}{M} \sum_k \text{sign}\left((C_R \alpha m_0 + (w^+ + w^-)\sqrt{C_{\text{int}}} \epsilon_k) \right)$$

where ϵ_k is drown form the Gaussian distribution with zero mean and unit variance. As discussed before, the solutions of this equation will be distributed normally (for large M) and will have zero mean and standard deviation

$$\sigma_{m_0} = \frac{1}{1 - \sqrt{\frac{2}{\pi}} \frac{C_R \alpha}{(w^+ + w^-)\sqrt{C_{\text{int}}}}} \sqrt{\frac{1}{M} + \langle \text{sign}\,(\epsilon_k)\,\text{sign}\,(\epsilon_l) \rangle_{k \neq l}}$$

over different realizations of $\{\epsilon_k\}$. It is easy to see that for random connectivity the second term is of the order $\frac{C_{\text{int}}}{N_{\text{int}}}$, so for $(w^+ + w^-)\sqrt{C_{\text{int}}} \gg C_R\alpha$ and $N_{\text{int}} \sim C_{\text{int}}M$, the variance of m_0 is close to what was assumed in the derivation of classification capacity.

2.6.1 Simulations

The results of the simulation for the high noise uniform case are presented at the figure (2.6). We simulated the case of sparse ($C_F f = 2.5$) and dense ($C_F f = 10$) input representations. In agreement with our theoretical results, the parameters of the recurrent network of intermediate readouts can be found to lead virtually no difference in the classification capacity for sparse and dense regimes.

To estimate how the classification capacity depends on the network size we employed two procedures. In the first one, at each new step we chose the size of the network and tried to learn the number of patterns P which was slightly less than the estimated capacity for the previous (smaller) network. If we were able to classify the patterns with required accuracy (error rate of 10%, we increased P). The blue markers correspond to this procedure. The red markers were obtained for approaching the capacity from the other side. We started by trying to learn a number of patterns P which was larger than the theoretical estimate by a factor of 1.3. We then decreased P before the required accuracy was reached. We did not model the network initialization here. Instead the initial state of the units was chosen to be +1 or -1 randomly with equal probability.

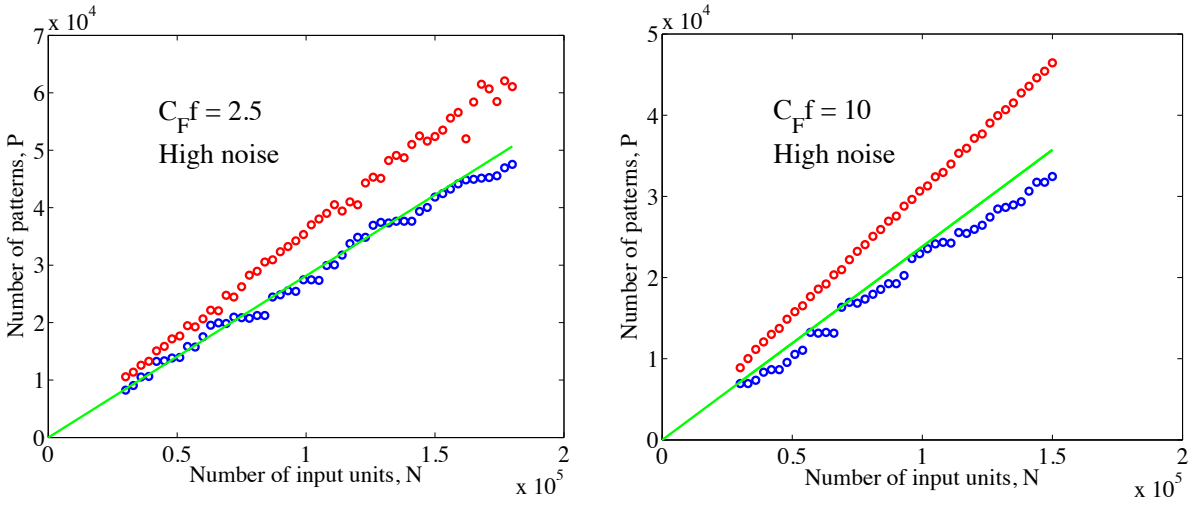

Figure 2.6: The simulation results for the high noise uniform case in the sparse (left) and dense (right) regimes. The green line is the theoretical prediction of formula 2.96. To estimate how the classification capacity depends on the network size we employed two procedures. In the first one, at each new step we chose the size of the network and tried to learn the number of patterns P which was slightly less than the estimated capacity for the previous (smaller) network. If we were able to classify the patterns with required accuracy (error rate of 10%, we increased P). The blue markers correspond to this procedure. The red markers were obtained for approaching the capacity from the other side. We started by trying to learn a number of patterns P which was larger than the theoretical estimate by a factor of 1.3. We then decreased P before the required accuracy was reached. We did not model the network initialization here. Instead the initial state of the units was chosen to be +1 or -1 randomly with equal probability.

2.7 Chapter conventions

Table 2.1: Notations

Notaion	Meaning
$\mu, \nu \in [1 \dots P]$	pattern indices
$i, j \in [1 \dots N]$	input neuron indices
$k, l \in [1 \dots M]$	medium layer readout indices
$\mathbf{N}(0,1)$	the normal distribution with mean 0 and variance 1
$\mathbf{B}(N,f)$	distribution of # of events in N trials with event probability f
$\mathbf{P}(\lambda)$	Poisson distribution with parameter λ
$\varphi_{C_F,f}$	covariance in the intermediate layer (2.62) and (2.63)(2.65)

The error function

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \tag{2.168}$$

Small x expansion

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \left(x - \frac{1}{3} x^3 + \dots \right) \tag{2.169}$$

Expectation value of $\mathrm{sgn}(x+a)$ in normal distribution is expressed in terms of the erf function as

$$\langle \mathrm{sgn}(x+a) \rangle_{\mathbf{N}(0,\sigma)} = \frac{1}{\sqrt{2\pi}} \int_{-a}^{a} e^{-\frac{1}{2\sigma^2} x^2} dx = \mathrm{erf}\left(\frac{a}{\sqrt{2}\sigma} \right) \tag{2.170}$$

Small a/σ expansion

$$\langle \mathrm{sgn}(x+a) \rangle_{\mathbf{N}(0,\sigma)} = \sqrt{\frac{2}{\pi}} \frac{a}{\sigma} + \dots \tag{2.171}$$

Suppose that a is a signal which scales with the number of patterns as

$$a = \frac{\tilde{a}}{\sqrt{P}} \tag{2.172}$$

and suppose we set threshold to the probability of the correct classification $p = 1 - \epsilon$. Then from the threshold equation

$$\mathrm{erf}\left(\frac{\tilde{a}}{\sqrt{2}\sigma\sqrt{P_{\max}}}\right) = 1 - 2\epsilon \tag{2.173}$$

we find capacity

$$P_{\max} = \frac{1}{(\mathrm{erf}^{-1}(1-2\epsilon))^2}\frac{\tilde{a}^2}{2\sigma^2} \tag{2.174}$$

2.8 Chapter conclusions

Our study shows that feed-forward neural classifiers with numerous long range connections connecting different layers can be replaced by networks with sparse long range connectivity and local recurrent connectivity without sacrificing the classification performance. Our strategy could be used in the future to design more general scalable network architectures with limited connectivity, which resemble more closely brain neural circuits dominated by recurrent connectivity.

We argue that this problem is especially relevant for the hippocampus as the observed dentate gyrus to CA3 connectivity is very sparse. Our model network also possesses other features of the hippocampus, in particular, extensive recurrent connectivity of the layer representing the CA3 area and low activity level in the input layer, that is meant to model the dentate gyrus.

Although we do not find a clear advantage of having sparse representation if the dentate gyrus, we show that the model network can operate in the regime when the average number

of active dentate gyrus cells connected to one CA3 readout is of the order of 1, or even less. Paradoxically, the classification capacity in this case can be made almost identical to the case of dense representations. One can argue that low activity level in the dentate gyrus has advantages that are beyond the presented framework (see, for example [36]), in which case our results should be considered as justification for the possibility to realize this advantages despite the limited connectivity.

One can argue that the two-subnetworks regime, described for the case of relatively very strong feedforward connections and sparse input representations (the regime observed in the case of mammalian hippocampus for input layer corresponding to the dentate gyrus and the readout layer - to the CA3 area), assumes a broader parameter regime for achieving high classification capacity. However, the precise analysis required to make this argument remains to be done.

Chapter 3

Decoding position from dentate gyrus calcium recordings

This project was done in collaboration with Fabio Stefanini, Mazen Kheirbek, René Hen and Stefano Fusi. The experiments were performed by Mazen Kheirbek, and the following analysis - by myself and Fabio Stefanini.

3.1 Chapter summary

Hippocampus has been long hypothesized to be involved in coding of space [2],[21]. Cells with specially tuned firing patterns were observed in the areas CA1, CA3 and the entorhinal cortex [26]. The dentate gyrus, however has been studied much less due to its sparse activity and other technical difficulties for electrophysiological recordings. Nevertheless a few electrophysiological studies address the spatial tuning of the dentate gyrus granule cells [23],[25] and suggest that they encode spatial information exhibiting firing patterns with multiple place fields [24],[32]. However, it remains unclear whether these tuning properties are stable enough to be used to decode the animal's position.

With development of calcium imaging techniques [64] and, more recently, miniaturized head-mounted microscopes [65], it became possible to record the activity of large number of cells simultaneously in a moving animal. The recordings analyzed here were performed by Mazen Kheirbek and is the first calcium imaging of the dentate gyrus activity.

In order to establish whether dentate gyrus granule cells encoded spatial information, we trained two separate decoders, a linear decoder using mean-square linear regression and a non-linear decoder consisting of a committee of binary classifiers trained on discretized locations within the cage. We then asked whether the decoders could predict the position of the mouse based on the recorded Ca2+ data on a single 200 ms time-bin basis.

We demonstrate that the animals position can be decoded from the recorded calcium signal with approximately 10cm accuracy and that the neural representation of position in the dentate gyrus have close to maximal dimensionality. Our analysis also suggests that cells with a single firing field within a box contribute the same amount of information to the decoder as cells with multiple firing fields.

An often encountered approach in the context of decoding animal's position from hippocampal activity is a bayesian framework [66],[67],[68]. This implies that the decoder has an access to the prior distribution of its locations, prior distribution of the population activity and the distribution of the neural activity at any given location. The output of a bayesian decoder is than a posterior distribution over the locations given the observed neural activity, and the predicted position corresponds to the maximum of this distribution.

Although, all three distributions can be estimated over the training data and performance can be cross-validated, both the training of the decoder and the decoding procedure are usually complicated and are hard to see as biologically plausible.

The advantage of the two decoders presented here is that both training and decoding procedures are relatively simple. The first decoder described below is an ensemble of linear

threshold classifiers that take a collective decision about the predicted location. The weights of these linear threshold classifiers (perceptrons) are the only parameters that are tuned during the training of the decoder. One can think of these weights as synaptic strengths of a downstream readout neuron. The weights can be learned using an online algorithm [39] (although we use another training algorithm to speed up training). The collective decision of the ensemble of perceptrons can be implemented with another layer of linear threshold units and some winner-take-all interaction between them.

The second decoder that we use is a linear regression decoder whose prediction is a weighted sum of the activities of the recorded neurons. Again, the training of the decoder can be done online, meaning that each new training pattern modifies the regression coefficients independently of the other training patterns.

After confirming that the position information is consistently represented in the signal, we turn to analyzing the spatial firing patterns of the recorded units. Because of the sparse activity in the dentate gyrus [23] it is reasonable to assume that the region of interest extracted from the calcium data correspond to single cells, not few overlapping neurons. Following the procedure described in [24] we divide the cells into two populations - cells with a single firing field cells, or single field cells and the remaining non-single field cells. We compare the amount of spatial information carried out by the two populations. Our conclusion is that there is either no significant difference between the two populations, or the slight difference is in favor of non-single field cells.

In agreement with [24], 30% to 40% of the cells were classified as single field. It should also be pointed out that there is an evidence that spatially tuned cells in the areas CA1 and CA2, and also in the dentate gyrus, that look like single field in a standard box often exhibit multiple firing fields in larger environments [25].

In the end of the chapter we determine the number of principal components that are

useful for position decoding. This number should be similar to the dimensionality of the spatial representation. This may be much smaller than the dimensionality given by the standard PCA analysis, namely the number of principal components with the explained variance higher than a chosen threshold. The difference is due to the fact that the variability of population activity along some directions in the neural space may be very large, but independent from the animal location (coding for other variables) and thus will not improve the cross-validated performance.

The estimated dimensionality of the representation of space in the dentate gyrus turns out to be maximal for the given accuracy, consistent with its hypothesized role in dimensionality expansion [36],[69],[70],[71],[72].

3.2 Description of the experiment and the acquired data

We used miniaturized head-mounted microscopes to perform functional Calcium imaging of dentate gyrus granule cells as mice foraged in an open field box. The size of the box was $50cm \times 50cm$ and the recording sessions were10 min long.

We started by extracting putative Ca2+ events from the fluorescence traces using published methods [73] and convolved the temporal events with decaying exponentials to extract putative signals for the granule cells (see figure 3.1).

The decoding seemed to work in five out of six animals who explored the box at least a little bit. Number of putative single units recorded in each animal, animal's mobility and the total number of events is summarized in the last three columns of the table of figure 3.4. The mobility was defined as percentage of time bins that belonged to intervals of mobility (see section 3.3.2). As expected the distribution of the visitation numbers over the discrete locations in the box were not uniform. All animals (except for the one mouse for which the decoding did not work and that was not included in the analysis) spent most time in the

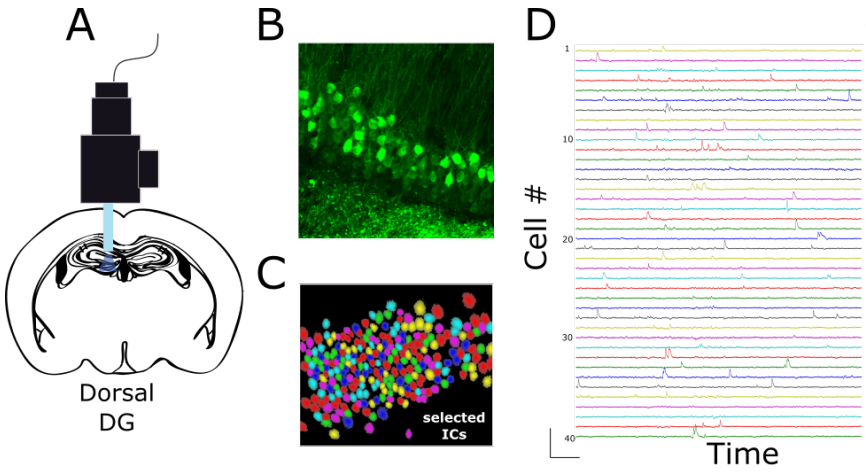

Figure 3.1: A, gradient index lens (GRIN) is implanted above the dorsal DG, and a miniaturized microscope is used to image the activity of DG GCs. B. AAVdj-CamKII-GCaMP6f expression in DG GCs for functional Ca2+ imaging. C. Example Isolated DG GC units from a representative imaging session. D, Standard deviation Ca2+ traces from the extracted independent components from C.

corners, some time close to the walls and less in the center.

3.3 Position decoding

In this section we describe our approach to decoding animal's position from the putative Ca2+ events in the dentate gyrus granule cells.

3.3.1 Cross-validation

Any neural decoding procedure is predicting animal's behavior or sensory stimulus by looking at the neural activity in a particular brain region. The way the neural activity is processed to obtain the prediction involves some number of free parameters that are being tuned to establish the correspondence between the predictor (in this case neural activity) and the predicted variable (in our case the animal's position). The procedure of finding

these free parameters is usually referred to as training the model or *training the decoder*.

To claim that a certain behavioral or sensory variable is represented by the recorded neural activity, the performance of the decoder should be *cross-validated*, meaning that the data used to train the the decoder can not be used in estimating the accuracy of the decoding.

Our data consists of a 10min - long calcium recordings with the measurements taken each 200ms, which makes 3000 data points. After filtering for speed (see section 3.3.2), about 2000 data points remain, depending on the animal. We divide these points into 5 equal sets of 400 data points and train 5 decoders each using 4 out of five sets (1600 recordings) together with the corresponding positions of the animal as training data. We then use a decoder trained on a given set of 1600 points to predict animal's position during the remaining 400 time bins based on the neural activity recorded during these time bins (which was not used to train the decoder).

In this way, we get the decoder predictions for the entire time of the recording, but never use the same data for training and prediction. Thus, our decoding is cross-validated.

3.3.2 Filtering for the animal's speed

It has bee hypothesized before [74] that the animals position is represented by neural activity in the hippocampus only when the animal is in motion. We confirm that the decoding performance increases when we first filter the data by the speed of the animal, and use only filtered data for training. We can still make predictions for the times when the animal is not moving or moving very slowly, and in the results section we present both, the performance of the decoder when the animal is moving fast, and when it is moving slowly or stands still.

The filtering procedure is as follows. We first choose a threshold for speed v_0, (most

our results are presented for two values of the threshold - 1cm and 2.5 cm) and a time duration t_0 (equal to 1s). The periods of duration longer than t_0 during which the speed is higher than v_0 we call *mobile* and include in the training of the decoder. The periods of duration longer than t_0 during which the animal's speed was less v_0 we call *immobile* and do not include in the training data for the decoder. When a mobile period is interrupted by a period shorter than t_0 during which the speed is less than v_0, this short period is still considered mobile and vice versa. Our results were obtained using $t_0 = 1s$ and one of two speed thresholds - $v_0 = 1cm/s$ or $v_0 = 2.5cm/s$ (we specify which one).

The decoding results are consistently better when the analysis is restricted to the time intervals classified as mobile according to the procedure described above. One of the hypothesis was that it is due to elevated activity levels when the animal is moving. However, we didn't observe a consistent correlation of the total number of events in a time bin (combined over all cells) with the animal's speed (see figure 3.2). Neither we observed a significant difference between the average activity of the population during periods of mobility and immobility (see figure 3.3). We conclude that the observed difference in decoding performance is due to decreased accuracy in representation of space in the immobile state. Whether a variable other than current location (future location for example) is represented by the same population of neurons in the immobile state remains to be determined (see also figure 3.11).

3.3.3 Description of the committee of perceptrons decoder (non-linear decoder).

We first divide the arena into 64 equally sized squares (8×8 grid), that we later refer to as *locations*. We then train a maximal margin linear threshold classifier for each pair of locations, which makes it $M = 63 \times 64/2$ binary classifiers. We often refer to this binary

Mouse	DG1	DG2	DG4	DG5	DG10
Speed-activity correlation	-0.07	-0.02	-0.03	0.28	0.37
p-value	7E-05	0.4	0.1	6E-54	4E-99

Figure 3.2: Coefficient of correlation between the animals speed and the number of events combined across the cells. Contrary to our expectation, a significant positive correlation is only observed for two out of five animals. The p-values are relative to the null hypothesis that there is no correlation between the two variables.

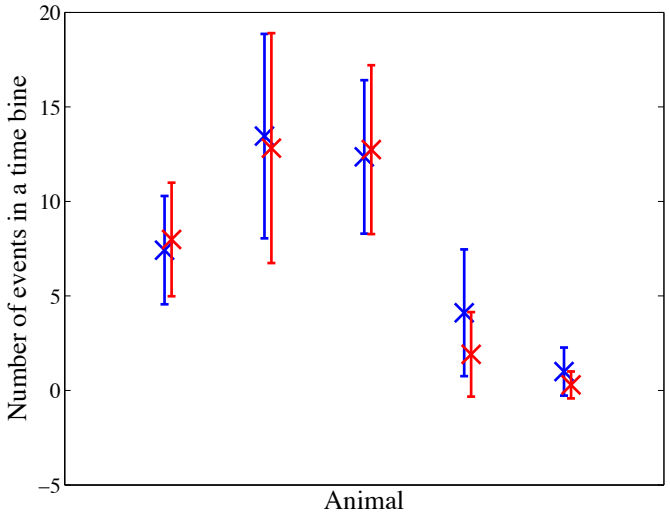

Figure 3.3: Mean and standard deviations of the total number of events in a time bin during intervals classified as mobile (blue) or immobile (red). Again, there is no significant difference in the activity between mobile and immobile states.

classifiers as *perceptrons*. Each perceptrons discriminates between a pair of locations. For training, for example the perceptron that discriminates between location 1 and location 42, we only use the training data recorded at the times when the animal was at these locations (and that belong to the training period for the given cross-validation run see section 3.3.1). Each time stamp of calcium recording contributes a neural pattern with an appropriate label. Because the animal spends unequal amount of time in different locations, the training sets for most binary classifiers are unbalanced, but we found that balancing them does not improved the cross-validated performance.

The decoded position for a given neural activity pattern from the test interval is presented, is determined as follows. Let the activity pattern be $\{r_i\}$ where i is the number of ROI, each of M binary classifiers gives a response

$$\eta_t^j = \text{sign}\left(\sum_{i=1}^{N} w_i^j r_{i,t} - \theta^j\right) = \pm 1 \tag{3.1}$$

For each location, let's say "42" there are 63 perceptrions each of which corresponds to the pair of locations with one location in the pair being "42". Namely, there is a perceptron that classifies a neural activity pattern as corresponding to location "42" or to location 1, another perceptron distinguishes 42 from 2, and so on. We then need to combine the responses of all the perceptrons to determine the decoded location. The way to do it is to count for every location the number of classifiers whose responses indicate that neural activity pattern corresponds to this location and choose the location with the highest score (the maximum score is 63).

The easy way to do it, is to introduce a *coding matrix*, whose columns are the ideal responses of the entire ensemble of perceptrons to a pattern corresponding to a given location (the actual responses could never be ideal because the classifiers always give +1 or -1, but virtually never 0). If we choose the first classifier to discriminate between location "1"

and location "2", the second - between "1" and "3"... the 64th - between "2" and "3" and so on, and assign output +1 to the first location in the pair and -1 to the second, the coding matrix will be:

$$\mathbf{C} = \begin{pmatrix} 1 & -1 & 0 & \cdots & 0 & 0 \\ 1 & 0 & -1 & \cdots & 0 & 0 \\ \cdots & & & & & \\ 1 & 0 & 0 & \cdots & 0 & -1 \\ \cdots & & & & & \\ 0 & 1 & -1 & \cdots & 0 & 0 \\ \cdots & & & & & \\ 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix}$$

If the outputs of all the perceptrons are written as M-dimensional row-vector η, then the decoded location is determined as a position of maximal element in the row vector of the product of the output and the coding matrix, $\eta_t \mathbf{C}$.

The decoded position is then taken to be the center of the $6cm \times 6cm$ square predicted by the decoder.

3.3.4 Description of the linear regression decoder.

To make sure that the statements, that we make about the accuracy of the decoding in different conditions or using different subsets of cells, are not the artifacts of the chosen decoder, but are actually representative of the neural representations of position, we introduce another decoder, and repeat the analysis. The second decoder is a linear regression decoder that predicts X and Y coordinates of the animal independently based on linear

regression model. The predicted x and y coordinates at moment t are given by

$$X_{pr}(t) = \sum_{n=1}^{N} a_n^{(x)} r^n(t) + x_0$$
$$Y_{pr}(t) = \sum_{n=1}^{N} a_n^{(y)} r^n(t) + y_0$$

where the sum is taken over the cells. The coefficients $a^{(x)_n}$ and $a^{(y)_n}$ are tuned to minimize the decoding error over the training set (again, the decoder's performance is cross-validated).

3.3.5 Lasso algorithm for regularization.

We were also able to improve the accuracy of the linear regression decoder by applying a regularization algorithm to avoid overfitting to the training data. This algorithm is called "lasso" (least absolute shrinkage and selection operator) and was introduced in [75] based on [76]. The procedure is to tune $a_n^{(x)}$ and $a_n^{(y)}$ not to minimize the mean squared error as done in the usual linear regression, but a cost function given by

$$\text{Cost}_x = \sum_{t=1}^{T} (X - X_{pr})^2 + \lambda \sum_{n=1}^{N} |a_n^{(x)}| \tag{3.2}$$

where λ is a parameter. The fact that the second sum contains the absolute values of the coefficients (L1 norm) and not the square for example, guarantees that the optimization procedure will find a solution for $\{a_n^{(x)}\}$ with some of the coefficients in the set $\{a_n^{(x)}\}$ being equal to zero. Which coefficients are non-zero (which cells are included in the decoder) is optimized to minimize the mean squared error on the training set. Changing the parameter λ changes the relative importance of the new term and thus the number of non-zero coefficients. The best cross-validated performance is observed for the values of λ corresponding

to between 100 and 200 non-zero coefficients (100 to 200 cells included in the decoding). When α is decreased and more cells are included, the decoder tends to overfit and the performance on the test set decreases.

The lasso algorithm also provides a way to rank the cells according to the amount of information they provide about the decoded variable (see section 3.4.2).

3.3.6 Results

We were able to decode position with accuracy of about 10cm in a 50cm by 50cm box for 5 out of 6 animals. Table 3.4 summarizes the decoding performance as measured by median decoding error for these animals. We also present the chance level for this measure and the p-value (see section 3.3.7).

Another way to represent the accuracy of the decoding is to compute the percentage of times that the predicted location is within 10cm from the actual location of the animal (*percentage close*). The choice of 10cm is motivated by the median errors across five animals. The decoding results relative to this measure of performance are summarized in the table on figure 3.5. The chance level for this measure and the p-values were computed in a similar way (see section 3.3.7).

The higher chance performance in the immobile periods is observed because the animal stops mostly in the corners and just knowing the prior distribution leads a relatively high performance. The fact that the chance performance is even higher than 25%, indicates that the mouse spends highly unequal amount of time in the four corners. The p-values higher than 0.5, especially for DG5 and DG10, is a reflection of the fact, that the decoder consistently predicts a wrong location (more often than by chance). See figure 3.11, panels a), b) and c)

The decoding performance seem to increase when we the threshold for classifying a time

period as mobile or immobile to $v_0 = 2.5cm$ as seen from figure 3.6 Another measure of the performance is the percentage of time the decoded location is within 10cm of the actual location. Increasing the threshold velocity increases the performance The performance of the linear regression decoder is summarized in the figure 3.7. The performance is slightly worse than in the case of ensemble of perceptrons, but still very significantly above chance.

3.3.7 Computing the chance level and p-value

The chance level for decoding performance reported in tables of figures 3.4,3.5 and 3.6 was computed as the median error distance (or percentage close) measured for the shuffled decoder's predictions. The p- values are relative to the null hypothesis that there is no information about animal's position in the recorded calcium signal, and the decoder's prediction are drawn randomly from a certain prior distribution. The p-values defined in this way can be estimated by performing many shufflings of the decoder's predictions and counting the number of times when the performance thus obtained is better than the performance computed on unshuffled predictions (when the actual distribution of the locations in the test data is used instead of the distribution of the predictions, the p-values are still very low). However there are substantial autocorrelations in both, the recorded calcium traces and the animals locations that should be taken into account.

If the prediction errors at different time points were independent form each other, p-value would have been easy to compute - we would have to shuffle the decoder's predictions many times and calculate the proportion of times when the shuffled predictions lead to a lower median error (or higher percentage close) than the actual ones. However, neither the animals positions, nor the decoder's predictions are independent when two proximal moments in time are considered, and thus the prediction errors are also correlated. The

Table 1, Median error of decoding over periods of mobility and immobility (threshold speed 1cm/s)

	Median Error, cm	Chance Error, cm	p-value	Median error, immobile	Chance Error, immobile	p-value, immobile	Number of cells recorded	Mobility	Total # events in 10 min
DG1	11.5	24.4	< 1E-04	24.6	27.4	0.05	300	0.4	23 255 (78 per cell)
DG2	10.5	25.3	< 1E-04	19.3	24.7	0.05	638	0.5	39 457 (62 per cell)
DG4	4.7	28.8	< 1E-04	24.0	31.3	0.0004	356	0.2	37 953 (107 per cell)
DG5	8.5	27.9	< 1E-04	33.0	28.8	0.36	432	0.8	11 243 (26 per cell)
DG10	15.0	22.0	< 1E-04	16.4	15.5	0.62	136	0.7	2 280 (17 per cell)

Figure 3.4: Table of the decoding results for the non-linear decoder. Cross-validated median decoding error was computed over 10min session of free exploration of a 50cm by 50cm box, separately for mobile and immobile periods. The reported chance level is the median error over the decoder predictions shuffled in time. The p-values were estimated as the percentage of shufflings that lead a lower median error than the unshuffled predictions. To compute the p-values we used only the data points separated by a time interval, longer than a certain length (τ_0). This thinning out of the data was done to get rid of autocorrelations. See section 3.3.7 and figure 3.8 for details.

Table 2, Accuracy of the decoding measured as the fraction of times when the decoding error is less than 10cm (threshold speed 1cm/s)

	Fraction Error< 10cm	Chance level	p-value	Fraction Error< 10cm, immobile	Chance level, immobile	p-value, immobile	Number of cells recorded	Mobility	Total # events in 10 min
DG1	0.47	0.14	<1E-4	0.21	0.15	0.002	300	0.4	23 255 (78 per cell)
DG2	0.49	0.16	<1E-4	0.37	0.37	0.4	638	0.5	39 457 (62 per cell)
DG4	0.69	0.16	<1E-4	0.3	0.38	0.56	356	0.2	37 953 (107 per cell)
DG5	0.57	0.14	<1E-4	0.18	0.38	0.98	432	0.8	11 243 (26 per cell)
DG10	0.34	0.16	<1E-4	0.29	0.37	0.96	136	0.7	2 280 (17 per cell)

Figure 3.5: Table of the decoding results for the non-linear decoder. Cross-validated performance is expressed as a fraction of times the decoded position is within 10cm of the actual one. The reported chance level is the median error over decoder predictions shuffled in time. The p-value was computed as the percentage of shufflings that lead a lower median error than the unshuffled predictions, when only the data points separated by a time interval longer than τ were included. This thinning out of the data was done to get rid of the correlations. See section 3.3.7 and figure 3.8. The higher chance performance in the immobile periods is observed because the animal stops mostly in the corners and just knowing the prior distribution leads a relatively high performance. The fact that the chance performance is even higher than 25%, indicates that the mouse spends highly unequal amount of time in the four corners. The p-values higher than 0.5, especially for DG5 and DG10, is a reflection of the fact, that the decoder consistently predicts a wrong location (more often than by chance). See figure 3.11, panels a), b) and c)

Table 3, accuracy of the decoding measured as fraction of times when the decoding error is less than 10cm (threshold speed 2.5cm/s)

	Fraction Error< 10cm	Chance level	p-value	Fraction Error< 10cm, immobile	Chance level, immobile	p-value, immobile	Number of cells recorded	Mobility	Total # events in 10 min
DG1	0.57	0.15	<1E-04	0.26	0.14	<1E-04	300	0.4	23 255 (78 per cell)
DG2	0.52	0.16	<1E-04	0.30	0.32	0.1	638	0.5	39 457 (62 per cell)
DG4	0.76	0.15	<1E-04	0.29	0.37	0.89	356	0.2	37 953 (107 per cell)
DG5	0.59	0.14	<1E-04	0.37	0.19	<1E-04	432	0.8	11 243 (26 per cell)
DG10	0.34	0.16	<1E-04	0.20	0.28	0.998	136	0.7	2 280 (17 per cell)

Figure 3.6: Table of the decoding results for the non-linear decoder. Cross-validated performance is expressed as a fraction of times the decoded position is within 10cm of the actual one. The threshold speed to classify the interval as mobile or immobile is $v_0 = 2.5cm/s$, see section 3.3.2

Table 4, Median error of decoding over periods of mobility with linear regression decoder (threshold speed 1cm/s)

	Median Error, cm	Chance Error, cm	p-value	Number of cells recorded	Mobility	Total # events in 10 min
DG1	16.1	21.6	7E-04	300	0.4	23 255 (78 per cell)
DG2	15.3	22.0	4E-05	638	0.5	39 457 (62 per cell)
DG4	17.3	25.8	0.01	356	0.2	37 953 (107 per cell)
DG5	15.2	25.5	8E-08	432	0.8	11 243 (26 per cell)
DG10	13.6	19.3	5E-03	136	0.7	2 280 (17 per cell)

Figure 3.7: Median decoding error for the linear regression. The decoder was trained and tested on mobile intervals only. The speed threshold $v_0 = 1cm/s$

animal's positions at consecutive moments are correlated because of the finite speed of the animal, and the decoder's predictions are correlated because of the time-correlations in the recorded calcium signal.

To find an upper bound on the p-values we employ the following procedure. We first find a minimal time τ_0 such that, if two time points are further away from each other than τ_0, the prediction errors measured at this time points can be considered to be independent from each other. We then thin out the data so that no two time points are closer than τ_0 apart and compute the p-value for the tinned out sample.

The problem now is to estimate τ_0. For that we need a measure of autocorrelations in the recorded activity. Since the recorded activity is a vector, we first project it onto a relevant direction in the neural space, that we choose to be the weight vector of one of the perceptrons described above (a binary decoder that discriminates between a randomly chosen pair of locations). Thus, we estimate the autocorrelation in the time series of $h_t = \sum_{n=1}^{N} W_n r_t^n$ (sum over neurons), where W_n are the weights of one of the perceptrons, and time points t are chosen to be further apart than τ from each other. As a measure of autocorrelation we use the Durbin-Watson statistics [77]

$$d(\tau) = \frac{\Sigma_t^{(\tau)}(h_t - h_{t-1})^2}{\sum_{t=1}^{T}(h_t - \langle h_t \rangle)^2}$$

which is equal to 2 for the uncorrelated sample. We then plot the dependence of the Durbin-Watson statistics d from τ (we chose a new perceptron for each τ), see figure 3.8. The value of τ for which the curves cross the $d = 2$ value (the red line) is a reasonable estimate for τ_0. We estimate $\tau_0 = 7$ in the case of mobile periods and $\tau_0 = 3s$ in the case of immobile periods for all the animals.

Durbin-Watson Statistics for estimating the autocorrelation time of calcium traces.

$$d = \frac{\sum_{t=2}^{T}(h_t - h_{t-1})^2}{\sum_{t=1}^{T}(h_t - \langle h \rangle)^2} \qquad h_t = \sum_{n=1}^{N} W_n r_t^n$$

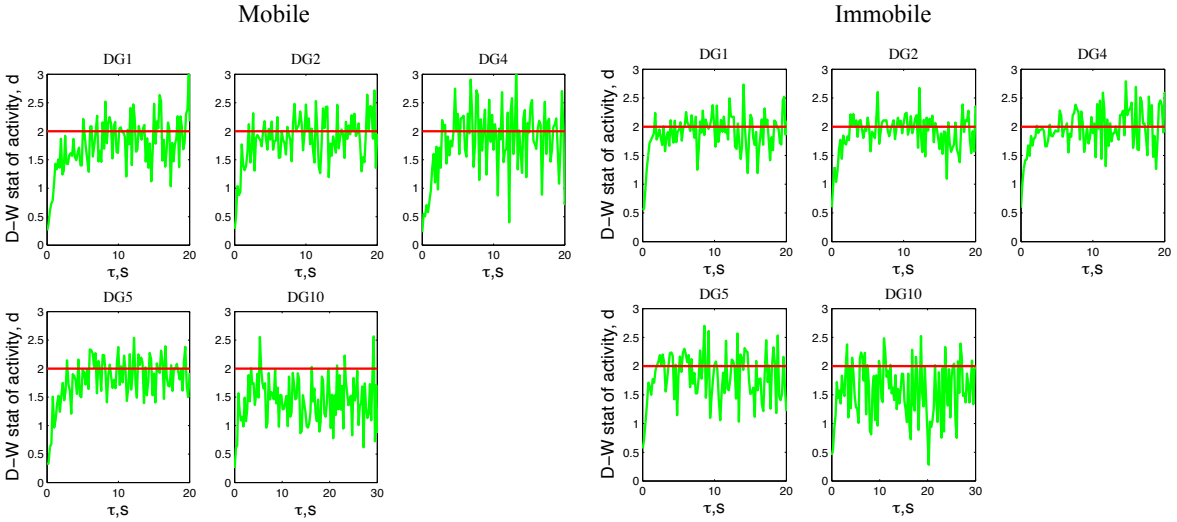

Figure 3.8: Estimating the autocorrelation time for the recorded calcium traces. We compute the Durbin-Watson statistics d of the population activity projected onto the weight vector of one of the perceptrons used in the non-linear decoder. The recorded activity was subsampled in such a way, that no two data points were separated by a time interval shorter than τ. The plots show the Durbin-Watson statistics for a thinned out data as function of τ. We used a weight vector from a different perceptron for each value of τ. Initially, the curves increase, indicating the drop in the autocorrelation of the thinned out sample with τ, and eventually oscillate around the value $d = 2$, that corresponds to no autocorrelation. We estimate the value of τ for which it happens in the mobile case as $\tau_0 = 7s$ and $\tau_0 = 3s$ for immobile case. We use these estimates to compute the upper bound on p-values for the decoding accuracy (see section 3.3.7). The curves seem to reach $d = 2$ value consistently faster for the periods of immobility (speed threshold $v_0 = 1cm/s$), but we did not investigate whether this difference is due to the real difference in autocorrelation times or because the immobile periods are shorter and more spread out, so that the same τ actually means longer average time-interval between the points.

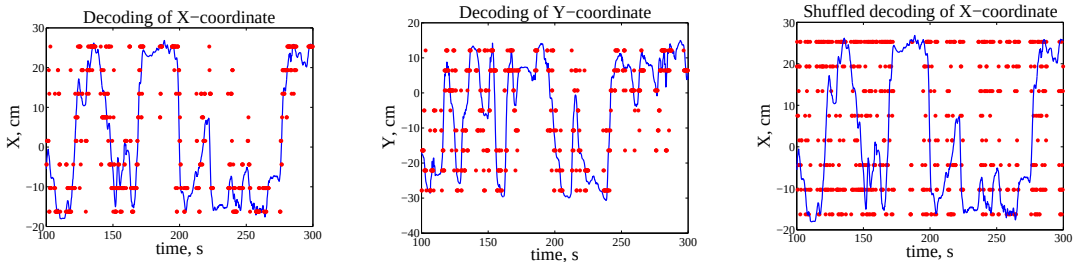

Figure 3.9: Actual and decoded trajectory for one of the animals. The predictions from the non-linear decoder. The right-most panel shows the shuffled predictions on top of the actual trajectory for comparison.

3.3.8 Visualizing the decoding performance

To give a feeling of how good the decoding is, we present two ways of visualizing the decoding results for the best animal. The later reveals an aspect of the decoder's performance that is not detectable with either of the measures reported above.

Figure 3.9 shows the actual (blue line) and the decoded (red dots) X-coordinates and Y-coordinates as functions of time. The last panel shows the actual X-coordinate and the decoder predictions shuffled in time.

Another useful representation of the decoder's performance are the distributions of the decoder's outputs for a given actual location of the animal, shown in figures 3.10 and 3.11. The animals actual location is represented by a red square frame, and the distribution of decoder's predictions is coded with color map. The intervals of mobility and immobility are analyzed separately. For both figures the speed threshold was chosen to be $v_0 = 2.5cm/s$. Figure 3.10 provides an example of good decoding of the corner location independently of the mobility. In figure 3.11 the distributions of the decoded locations are very different for

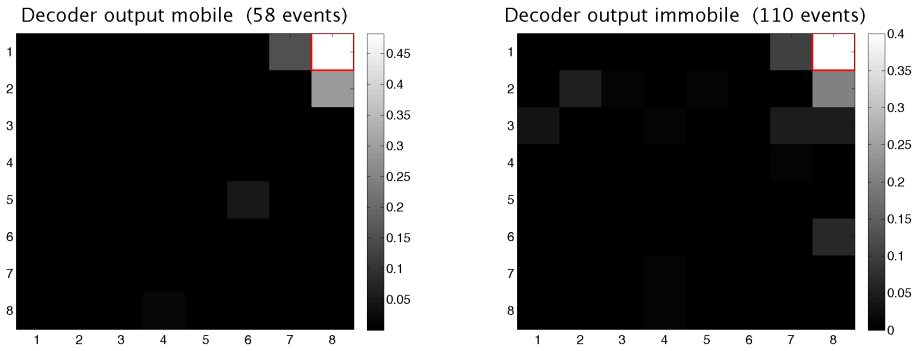

Figure 3.10: An example of good decoding of the corner location in both types of time intervals - mobile and immobile. Here the speed threshold was chosen to be $v_0 = 2.5cm/s$. The actual location of the mouse is represented by a red square frame, and the distribution of decoder's predictions is coded with color map.

mobile and immobile states (see captions).

3.4 Analysis of neural representations

In this section we concerned with some aspects of the representation of position in the dentate gyrus. We first discuss firing patterns of individual cells and divide the cells into two classes based on their firing properties, namely single field cells and non- single field cells. We then address the question of how distributed the neural code is by evaluating the decoding accuracy when a smaller number of cells, than were recorded, is used for decoding. This analysis also suggests that the two classes of cells contain similar amount of information about the position, when corrected for the number of cells in the group.

3.4.1 Single cell properties

This section is devoted to analyzing the firing properties of the individual cells recorded in the experiment. The question we are addressing is whether the observed spatial pattern of the cell's firing determines the amount of information that the cell contains about the animal's location. We classify all the recorded cells exceeding the activity threshold (10 events in a 10-min session) into two classes - cells with a single firing field, *single field cells* and cells with more than one firing field, or *non-single field cells*. It should be pointed out that any cell exceeding the activity threshold will have a certain number of firing fields, determined as described below. It should also be emphasized that the standard place cell criterion used for the CA3 and CA1 areas can not be applied in the dentate gyrus because of the extremely sparse activity (last column in the table of figure 3.4).

Following Leutgeb, [24] we identify the firing fields of a cell as follows. We first construct the firing rates for each location bin ($5cm \times 5cm$) as a total number of spikes occurred in this location divided by the total time that the animal spent there. We then convolve these discrete firing rates with a Gaussian filter to obtain a smooth firing rate map. The pixel corresponding to the global maximum of the firing map and the adjacent pixels with firing

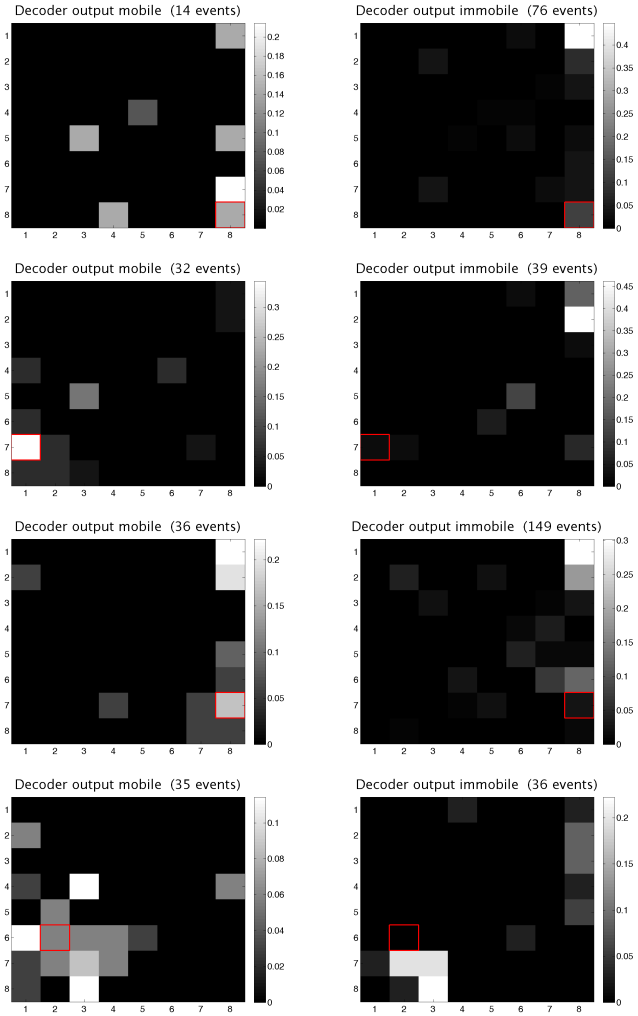

Figure 3.11: Decoding results in both types of time intervals - mobile and immobile for some locations. Here the speed threshold was chosen to be $v_0 = 2.5cm/s$. The actual location of the mouse is represented by a red square frame, and the distribution of decoder's predictions is coded with color map. (**a**)During mobile periods the maximum of the distribution of predicted locations is close to the actual position (red frame). Seeming worse decoding compared to figure 3.10 is probably due to a low number of occurrences (14). In contrast, for immobile periods the decoder consistently predicts another corner (more often than the actual location). (**b**)Good accuracy for mobile intervals, and a consistent prediction of another location during immobile intervals. (**c**) During mobile intervals the decoder's prediction is distributed along the wall, the actual position is predicted slightly less often than a neighboring corner. For immobile the decoder predicts neighboring corner more consistently. (**d**) In this case the decoding is better for immobile intervals.

rate exceeding 58% of the peak rate were considered to be the first firing field. These pixels were then deleted from the rate map and the procedure was iterated to find the second firing field, and so on until no pixels with the rate exceeding 60% of the highest peak for this cell were found. We also required that the firing field was no smaller than 9 pixels to remove the "half-fields" at the borders of the box.

Figure 3.12 shows a few examples of the firing patterns and corresponding firing rate maps for the cells classified as single field (top panel) and non-single field (bottom panel).

3.4.2 Decoding from a subset of cells

In this section we address the question of how distributed are the neural representation of position in the dentate gyrus. In order get an idea of how many cells is required to decode the animal's location with a given accuracy, we plot the dependence of the decoding performance (in this case median decoding error) from the number of cells included in the decoding analysis (both training and test).

Ranking of cells based on the non-linear decoder

How fast the decoding performance increases with the number of cells used for the decoding depends on the order in which the cells are included. On figure 3.13 we plot the dependence of the median error vs. number of cells when the cells are included in random order (blue solid line) and when the cells are included in the order based on the absolute value of the weights assigned to the cells (dashed blue line). We also present the curves for population of single field cells (two red lines) and non single field cells (two green lines). For all three populations the ordering according to the weights leads a much faster drop in median decoding error, compared to the random order.

Some care should be taken in the ranking of the cells according to the decoder's weights.

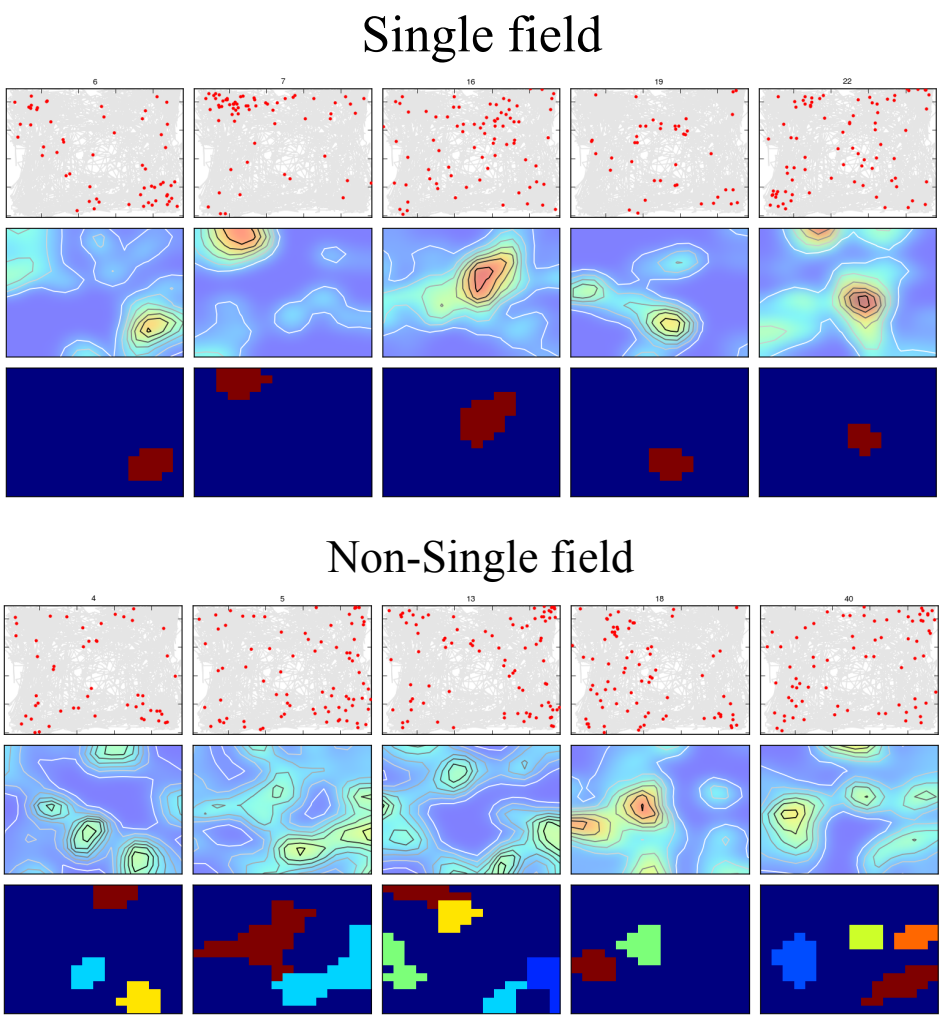

Figure 3.12: Firing patterns (top rows) of example single field and non-single field cells with corresponding firing rate maps (middle rows) and extracted firing fields (see section 3.4.1)

As discussed in section 3.3.3, each cell gets $M = 64 \times 63/2$ weights, each assigned by one of M perceptrons. Also there are five sets of the training data, because of our cross-validation procedure. All these $5M$ numbers should be combined to give an average weight assigned to the cell. We do it in two steps. The first step is to normalize the weights assigned to all the cells by a given perceptron, in such a way that the maximum weight each perceptron gives is equal to 1. This step is needed because the uniform rescaling of the weights for all the cells (and the threshold) does not change the output of the perceptron (see (3.1)), what matters for estimating the importance of a cell for the given perceptron, is its weight relative to the weights assigned to other cells by the same perceptron. The second step is to simply average the absolute value of the weights assigned to the cell over all the perceptrons and all the training sets. The cell with the highest average absolute value of the weight is considered the most informative for position decoding.

The ranking of a cell is correlated with the total number of events However, for the population of non-single field cells, the ordering according to the weights still leads a slightly faster drop in the decoding error with the number of cells included that ordering based on activity levels, as illustrated in figure 3.15. The statistical significance of this difference remains to be determined.

Ranking of the cells based on the linear regression decoder

In the case of linear regression decoder the decoding coefficients $a_n^{(x)}$ and $a_n^{(y)}$ assigned to a cell n do not reflect its importance for the decoding. However, the most informative cells for this decoder can be picked with the lasso algorithm described in section 3.3.5. Changing the parameter λ in front of the penalty term in (3.2) changes the relative importance of the number of non-zero coefficients in the cost function, which is the same as number of cells used decoding. Large value of λ means small number of cells, and vice versa.

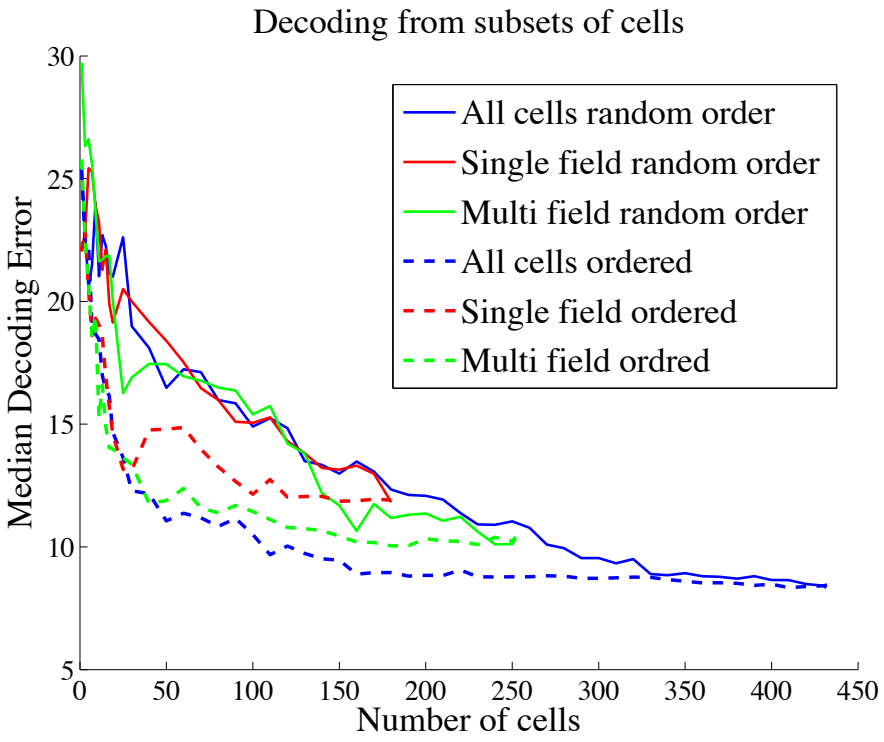

Figure 3.13: Median decoding error as a function of the number of cells used in the decoding. The solid curves correspond to including the cells in a random order, the dashed lines represent the order based on the absolute value of the weights assigned to the cell by the non-linear decoder (see section 3.4.2). The blue lines are for all the recorded cells that fired more than 10 events during 10min session, red lines - for the cells classified as single field, and the green lines - for the cells that passed the activity threshold but were not classified as single field (see section 3.4.1).

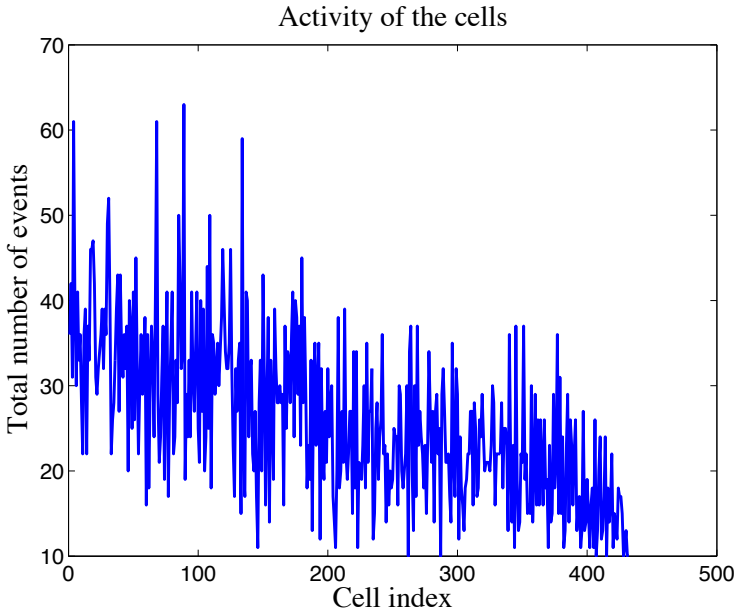

Figure 3.14: Total number of events the cell fires in the course of 10min session is plotted as a function of the rank, assigned to the cell by the non-linear decoder (see figure 3.13 and section 3.4.2) . The most informative cells are in the beginning. There is a clear correlation between the importance of the cell for the decoder and its overall activity, however the fluctuations are still large. The plots for only single field or only non-single field cells look very similar.

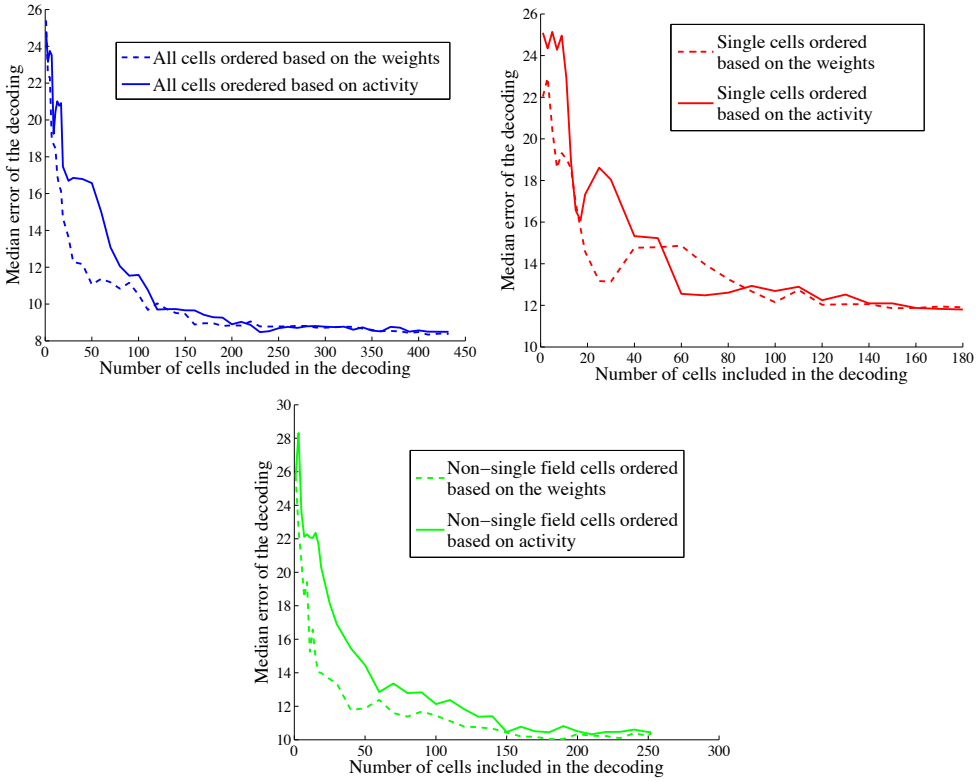

Figure 3.15: Median decoding error as a function of the number of cells used in the decoding for two ways of determining the order in which cells are included. The dashed lines correspond to including the cells according to the weights assigned by the non-linear decoder, the solid lines correspond to ordering cells based on their activity. The analysis is performed either for all the cells (left upper panel), or for single field (right upper panel) and non-single field cells separately. When cells are ordered based on the decoder's weights the median error drops slightly faster, but the statistical significance of this difference remains to be determined.

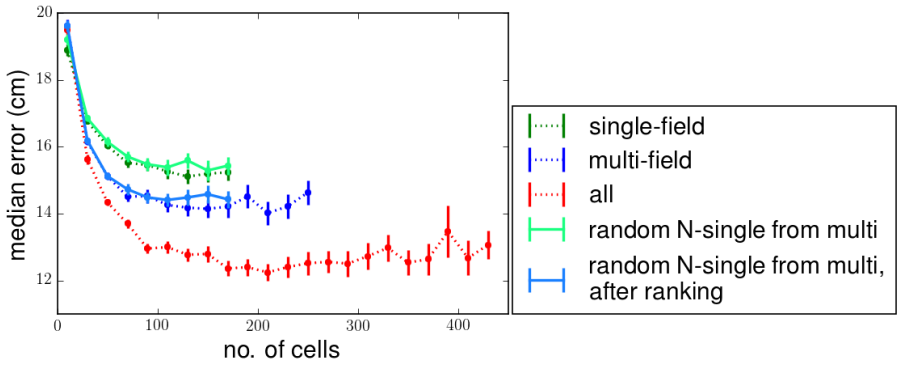

Figure 3.16: Median decoding error as a function of the number of cells for the linear-regression decoder. The cells were chosen using the lasso algorithm (see section 3.3.5). The red curve corresponds to the case when the algorithm chooses from the entire population of cells. For the dark blue curve the pool of cells was restricted to non-single field population. The dark green corresponds to single field cells. The apparent group difference in decoding performance can be explained by the difference in the number of cells in the groups. When we choose a random subset of non-single field cells matched in number to the single field group and run the lasso algorithm, the result is almost indistinguishable from the single-field cells (the light green curve). When the most active non-single field cells in the number matching the single field population were considered, the curve (light blue) followed the non-single field curve produced by lasso. This plot is consistent with both groups being equally useful for the decoding and a strong correlation between the activity level of a cell and the amount of information it contains about position.

We confirmed that this procedure also leads to a faster drop in the decoding error compared to including cells in a random order.

The performance of the linear regression decoder as a function of the number of cells for the two groups is shown on figure 3.16

3.4.3 Principal component analysis

We plot the median decoding error as a function of the number of principal components included in the decoding 3.17. The principal components are added in the order of decreasing explained variance. For both decoders the performance increases faster with the number

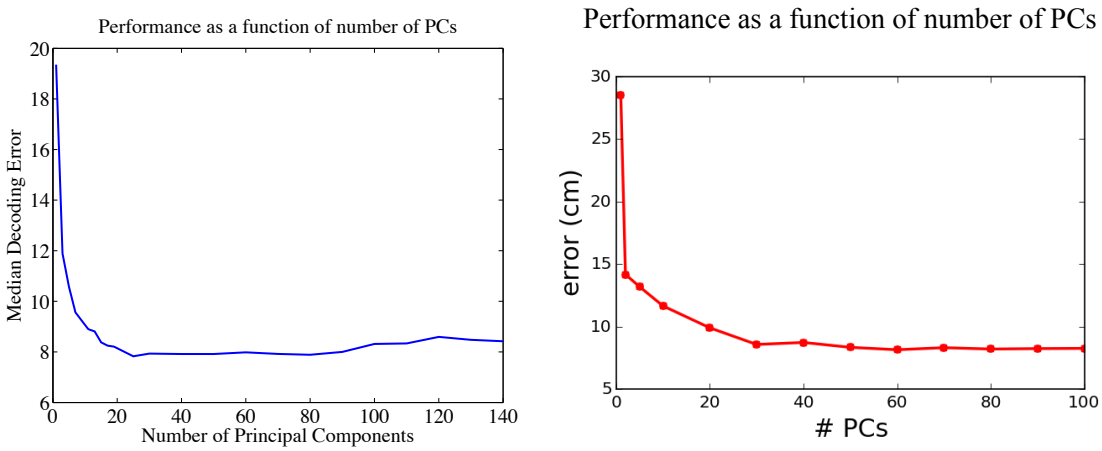

Figure 3.17: Median decoding error as a function of number of included principal components of the population activity. The principal components are added in the order of decreasing explained variance. The left panel corresponds to the non-linear decoder and the right panel - to the linear regression. Including 25 principal components seems to be enough to achieve the best decoding performance for the first decoder and 30 - for the second.

of principal components than with the number of cells, even cells are ordered from most informative to least informative. If we assume that the ranking of the cells is approximately correct, this the evidence for a distributed spatial code.

By looking at the plots on 3.17, one can see that including 25 principal components seems to be enough to achieve the best decoding performance for the committee of perceptrons decoder and 30 principal components - for the linear regression.

The number of principal components after which the curve flattens is consistent with the assumption that the position is represented with accuracy about 10 cm. Indeed, in $50cm \times 50cm$ box, there are 25 $10cm \times 10cm$ squares, and the maximal dimensionality of the representation that discriminates only between these squares is 25 [71]. This argument suggests that the representations in the dentate gyrus have maximal dimensionality, consistent with its hypothesized role in dimensionality expansion [36],[69],[70],[72].

3.5 Chapter conclusions

We demonstrated for the first time that animal's position can be decoded from the activity of dentate gyrus granule cells in a mouse. With two independent decoders, one linear and one non-linear, we were able to achieve a cross-validated accuracy of approximately 10cm in a $50cm \times 50cm$ open field box. Similar accuracy of a linear and non-linear decoder, and also principal component analysis, indicate that the spatial representation in the dentate gyrus has maximal dimensionality. We also divided the recorded cells into two groups based on their firing patterns - single field cells and non-single field cells, following [24]. Comparing the performance of a decoder, which is constraint to use only single field or non-single field cells, we conclude that there is no or little difference in the amount of information carried by each group. This result is consistent between both decoders.

Chapter 4

Conclusion

In this thesis I described two projects related to mammalian hippocampus. Although one of the projects addresses a purely theoretical problem, that can be formulated without any reference to the hippocampus, it was inspired by the neurophysiological peculiarity of the hippocampal structure, namely, the very sparse connectivity between dentate gyrus granule cells and CA3 pyramidal neurons.

In chapter 2 I presented a theoretical proof of principle, that neural networks with limited connectivity are plausible in the framework of pattern classification. This is a novel result, as previous network models for pattern classification required fully connected readouts in order to exhibit a favorable scaling of the classification capacity with the size of the network. We argue that this problem is especially relevant for the hippocampus as the observed dentate gyrus to CA3 connectivity is very sparse. Our model network also possesses other features of the hippocampus, in particular, extensive recurrent connectivity of the layer representing the CA3 area and low activity level in the input layer, that is meant to model the dentate gyrus.

Although we do not find a clear advantage of having sparse representation if the dentate

gyrus, we show that the model network can operate in the regime when the average number of active dentate gyrus cells connected to one CA3 readout is of the order of 1, or even less. Paradoxically, the classification capacity in this case can be made almost identical to the case of dense representations. One can argue that low activity level in the dentate gyrus has advantages that are beyond the presented framework (see, for example [36]), in which case our results should be considered as justification for the possibility to realize this advantages despite the limited connectivity.

The second project presented here is concerned with the decoding of animal's position from the dentate gyrus calcium traces. We show that position of a mouse during free foraging in a $50cm \times 50cm$ open field box can be decoded with the accuracy of around 10cm. This is the first calcium imaging in the dentate gyrus and the first time that position was decoded from the dentate gyrus activity. It should also be stressed, that in contrast to bayesian decoders often used in this context, both of our decoders are relatively simple, and can be implemented in a neural network. Training of the decoders can also be done in a biologically plausible way.

The theoretical model of chapter 2 assumes random and uncorrelated patterns in the dentate gyrus. One of the results of our decoding analysis supports this assumption. In chapter 3 we estimate the dimensionality of the representation of space in the dentate gyrus by determining the number of principal components required to achieve the full decoding performance. We find that this representation have maximal dimensionality, which indicates that the patterns of neuronal activity representing different locations are not correlated.

References

[1] W. B. Scoville and B. Milner, "Loss of recent memory after bilateral hippocampal lesions," *Journal of neurology, neurosurgery, and psychiatry* **20** (1957), no. 1 11.

[2] J. O'Keefe and J. Dostrovsky, "The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat," *Brain research* **34** (1971), no. 1 171–175.

[3] H. Eichenbaum, "A cortical–hippocampal system for declarative memory," *Nature Reviews Neuroscience* **1** (2000), no. 1 41–50.

[4] H. Eichenbaum, "Time cells in the hippocampus: a new dimension for mapping memories," *Nature Reviews Neuroscience* **15** (2014), no. 11 732–744.

[5] H. Eichenbaum, M. Sauvage, N. Fortin, R. Komorowski, and P. Lipton, "Towards a functional organization of episodic memory in the medial temporal lobe," *Neuroscience & Biobehavioral Reviews* **36** (2012), no. 7 1597–1608.

[6] P. Lipton and H. Eichenbaum, "Complementary roles of hippocampus and medial entorhinal cortex in episodic memory," *Neural plasticity* **2008** (2008).

[7] A. D. Redish, *Beyond the cognitive map: from place cells to episodic memory*. MIT Press Cambridge, MA, 1999.

[8] S. Corkin, "What's new with the amnesic patient hm?," *Nature Reviews Neuroscience* **3** (2002), no. 2 153–160.

[9] J. C. Augustinack, A. J. van der Kouwe, D. H. Salat, T. Benner, A. A. Stevens, J. Annese, B. Fischl, M. P. Frosch, and S. Corkin, "Hm's contributions to neuroscience: A review and autopsy studies," *Hippocampus* **24** (2014), no. 11 1267–1286.

[10] W. Penfield and B. Milner, "Memory deficit produced by bilateral lesions in the hippocampal zone," *AMA Archives of Neurology & Psychiatry* **79** (1958), no. 5 475–497.

[11] W. Penfield and G. Mathieson, "Memory: autopsy findings and comments on the role of hippocampus in experiential recall," *Archives of Neurology* **31** (1974), no. 3 145–154.

[12] B. Milner, S. Corkin, and H.-L. Teuber, "Further analysis of the hippocampal amnesic syndrome: 14-year follow-up study of hm," *Neuropsychologia* **6** (1968), no. 3 215–234.

[13] S. Corkin, "Lasting consequences of bilateral medial temporal lobectomy: Clinical course and experimental findings in hm," in *Seminars in Neurology*, vol. 4, pp. 249–259, 1984.

[14] H. Sagar, N. Cohen, S. Corkin, and J. Growdon, "Dissociations among processes in remote memory.," *Annals of the New York Academy of Sciences* (1985).

[15] B. Milner, "Physiologie de l'hippocampe: Colloque international, no. 107, editions du centre national de la recherche scientifique, paris, 1962. 512 pp. 58nf," 1965.

[16] S. Corkin, "Acquisition of motor skill after bilateral medial temporal-lobe excision," *Neuropsychologia* **6** (1968), no. 3 255–265.

[17] R. Shadmehr, J. Brandt, and S. Corkin, "Time-dependent motor memory processes in amnesic subjects," *Journal of Neurophysiology* **80** (1998), no. 3 1590–1597.

[18] J. D. Gabrieli, S. Corkin, S. F. Mickel, and J. H. Growdon, "Intact acquisition and long-term retention of mirror-tracing skill in alzheimer's disease and in global amnesia.," *Behavioral neuroscience* **107** (1993), no. 6 899.

[19] B. R. Postle and S. Corkin, "Impaired word-stem completion priming but intact perceptual identification priming with novel words: evidence from the amnesic patient hm," *Neuropsychologia* **36** (1998), no. 5 421–440.

[20] H. Schmolck, E. A. Kensinger, S. Corkin, and L. R. Squire, "Semantic knowledge in patient hm and other patients with bilateral medial and lateral temporal lobe lesions," *Hippocampus* **12** (2002), no. 4 520–533.

[21] J. O'keefe and L. Nadel, *The hippocampus as a cognitive map*, vol. 3. Clarendon Press Oxford, 1978.

[22] T. Hafting, M. Fyhn, S. Molden, M.-B. Moser, and E. I. Moser, "Microstructure of a spatial map in the entorhinal cortex," *Nature* **436** (2005), no. 7052 801–806.

[23] L. J. Drew, S. Fusi, and R. Hen, "Adult neurogenesis in the mammalian hippocampus: why the dentate gyrus?," *Learning & Memory* **20** (2013), no. 12 710–729.

[24] J. K. Leutgeb, S. Leutgeb, M.-B. Moser, and E. I. Moser, "Pattern separation in the dentate gyrus and ca3 of the hippocampus," *science* **315** (2007), no. 5814 961–966.

[25] E. Park, D. Dvorak, and A. A. Fenton, "Ensemble place codes in hippocampus: Ca1, ca3, and dentate gyrus place cells have multiple place fields in large environments," *PLoS One* **6** (2011), no. 7 e22349–e22349.

[26] E. I. Moser, E. Kropff, and M.-B. Moser, "Place cells, grid cells, and the brain's spatial representation system," *Annu. Rev. Neurosci.* **31** (2008) 69–89.

[27] A. A. Fenton and R. U. Muller, "Place cell discharge is extremely variable during individual passes of the rat through the firing field," *Proceedings of the National Academy of Sciences* **95** (1998), no. 6 3182–3187.

[28] D. Marr, D. Willshaw, and B. McNaughton, *Simple memory: a theory for archicortex*. Springer, 1991.

[29] E. T. Rolls, A. Treves, and E. T. Rolls, *Neural networks and brain function*. Oxford university press Oxford, 1998.

[30] N. Spruston and C. McBain, "Structural and functional properties of hippocampal neurons," *The Hippocampus Book* (2007) 133–201.

[31] D. G. Amaral, N. Ishizuka, and B. Claiborne, "Chapter neurons, numbers and the hippocampal network," *Progress in brain research* **83** (1990) 1–11.

[32] M. Jung and B. McNaughton, "Spatial selectivity of unit activity in the hippocampal granular layer," *Hippocampus* **3** (1993), no. 2 165–182.

[33] J. F. Guzowski, J. A. Timlin, B. Roysam, B. L. McNaughton, P. F. Worley, and C. A. Barnes, "Mapping behaviorally relevant neural circuits with immediate-early gene expression," *Current opinion in neurobiology* **15** (2005), no. 5 599–606.

[34] M. R. Hunsaker and R. P. Kesner, "The operation of pattern separation and pattern completion processes associated with different attributes or domains of memory,"

Neuroscience & Biobehavioral Reviews **37** (2013), no. 1 36–58.

[35] R. C. O'reilly and J. L. McClelland, "Hippocampal conjunctive encoding, storage, and recall: avoiding a trade-off," *Hippocampus* **4** (1994), no. 6 661–682.

[36] O. Barak, M. Rigotti, and S. Fusi, "The sparseness of mixed selectivity neurons controls the generalization–discrimination trade-off," *The Journal of Neuroscience* **33** (2013), no. 9 3844–3856.

[37] M.-B. Moser, D. C. Rowland, and E. I. Moser, "Place cells, grid cells, and memory," *Cold Spring Harbor perspectives in biology* **7** (2015), no. 2 a021808.

[38] B. G. Skotko, E. A. Kensinger, J. J. Locascio, G. Einstein, D. C. Rubin, L. A. Tupler, A. Krendl, and S. Corkin, "Puzzling thoughts for hm: Can new semantic information be anchored to old semantic memories?," *Neuropsychology* **18** (2004), no. 4 756.

[39] F. Rosenblatt, *The perceptron, a perceiving and recognizing automaton Project Para.* Cornell Aeronautical Laboratory, 1957.

[40] T. M. Cover, "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition," *IEEE Transactions on Electronic Computers* **EC-14** (1965), no. 3 326–334.

[41] M. Minsky and S. Papert, *Perceptrons.* 1969.

[42] C. Kwon and J. Oh, "Storage capacities of committee machines with overlapping and non-overlapping receptive fields," *Journal of Physics A: Mathematical and General* **30** (1997), no. 18 6273.

[43] R. D. Joseph and L. Hay, "The number of orthants in n-space intersected by an s-dimensional subspace,".

[44] R. O. Winder, "Single stage threshold logic," *Switching Circuit Theory and Logical Design, 1961. SWCT 1961. Proceedings of the Second Annual Symposium on* (1961) 321–332.

[45] D. Perkins, D. Willis, and E. Whitmore, "Division of space by concurent hyperplanes," *Internal Rep. Lockheed Aircraft Corp., Missiles and Space Division, Sunnyvale, Calif* (1959).

[46] J. G. Wendel, "A problem in geometric probability," *Math. Scand* **11** (1962) 109–111.

[47] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the national academy of sciences* **79** (1982), no. 8 2554–2558.

[48] D. O. Hebb, *The organization of behavior: A neuropsychological theory*. Psychology Press, 2005.

[49] L. N. Cooper, F. Liberman, and E. Oja, "A theory for the acquisition and loss of neuron specificity in visual cortex," *Biological Cybernetics* **33** (1979), no. 1 9–28.

[50] D. J. Amit, H. Gutfreund, and H. Sompolinsky, "Spin-glass models of neural networks," *Phys. Rev. A* **32** (Aug, 1985) 1007–1018.

[51] D. J. Amit, H. Gutfreund, and H. Sompolinsky, "Storing infinite numbers of patterns in a spin-glass model of neural networks," *Physical Review Letters* **55** (1985), no. 14 1530.

[52] D. Sherrington and S. Kirkpatrick, "Solvable model of a spin-glass," *Phys. Rev. Lett.* **35** (Dec, 1975) 1792–1796.

[53] E. Gardner, "The space of interactions in neural network models," *Journal of physics A: Mathematical and general* **21** (1988), no. 1 257.

[54] E. Gardner and B. Derrida, "Optimal storage properties of neural network models," *Journal of Physics A: Mathematical and General* **21** (1988), no. 1 271.

[55] E. Gardner, "Maximum storage capacity in neural networks," *EPL (Europhysics Letters)* **4** (1987), no. 4 481.

[56] L. Abbott and T. B. Kepler, "Optimal learning in neural network memories," *J. Phys. A: Math. General* **22** (1989) 711–717.

[57] L. F. Abbott, "Learning in neural network memories," *Network: Computation in neural systems* **1** (1990), no. 1 105–122.

[58] G. Mitchison and R. Durbin, "Bounds on the learning capacity of some multi-layer networks," *Biological Cybernetics* **60** (1989), no. 5 345–365.

[59] N. J. Nilsson, *Learning machines*. New York, 1965.

[60] R. Monasson and R. Zecchina, "Weight space structure and internal representations:

a direct approach to learning and generalization in multilayer neural networks," *Physical review letters* **75** (1995), no. 12 2432.

[61] H. Sompolinsky, N. Tishby, and H. S. Seung, "Learning from examples in large neural networks," *Phys. Rev. Lett.* **65** (Sep, 1990) 1683–1686.

[62] H. S. Seung, H. Sompolinsky, and N. Tishby, "Statistical mechanics of learning from examples," *Phys. Rev. A* **45** (Apr, 1992) 6056–6091.

[63] M. Geller and E. Ng, "A table of integrals of the error function. II. Additions and corrections," *J. Res. Natl. Bur. Stand* **75** (1971) 149–163.

[64] C. Stosiek, O. Garaschuk, K. Holthoff, and A. Konnerth, "In vivo two-photon calcium imaging of neuronal networks," *Proceedings of the National Academy of Sciences* **100** (2003), no. 12 7319–7324.

[65] K. K. Ghosh, L. D. Burns, E. D. Cocker, A. Nimmerjahn, Y. Ziv, A. El Gamal, and M. J. Schnitzer, "Miniaturized integration of a fluorescence microscope," *Nature methods* **8** (2011), no. 10 871–878.

[66] E. N. Brown, L. M. Frank, D. Tang, M. C. Quirk, and M. A. Wilson, "A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells," *The Journal of Neuroscience* **18** (1998), no. 18 7411–7425.

[67] M. Wilson and B. Mcnaughton, "Dynamics of the hippocampal ensemble code for space (vol 261, pg 1055, 1993)," *Science* **264** (1994), no. 5155 16–16.

[68] K. Zhang, I. Ginzburg, B. L. McNaughton, and T. J. Sejnowski, "Interpreting neuronal population activity by reconstruction: unified framework with application to hippocampal place cells," *Journal of neurophysiology* **79** (1998), no. 2 1017–1044.

[69] B. Babadi and H. Sompolinsky, "Sparseness and expansion in sensory representations," *Neuron* **83** (2014), no. 5 1213–1226.

[70] M. Rigotti, O. Barak, M. R. Warden, X.-J. Wang, N. D. Daw, E. K. Miller, and S. Fusi, "The importance of mixed selectivity in complex cognitive tasks," *Nature* **497** (2013), no. 7451 585–590.

[71] P. Gao and S. Ganguli, "On simplicity and complexity in the brave new world of large-scale neuroscience," *Current opinion in neurobiology* **32** (2015) 148–155.

[72] S. Fusi, M. E.K., and M. Rigotti, "Why neurons mix: high dimensionality for higher cognition," *Current Opinion in Neurobiology* (in press).

[73] Y. Ziv, L. D. Burns, E. D. Cocker, E. O. Hamel, K. K. Ghosh, L. J. Kitch, A. El Gamal, and M. J. Schnitzer, "Long-term dynamics of ca1 hippocampal place codes," *Nature neuroscience* **16** (2013), no. 3 264–266.

[74] B. McNaughton, C. Barnes, and J. O'keefe, "The contributions of position, direction, and velocity to single unit activity in the hippocampus of freely-moving rats," *Experimental Brain Research* **52** (1983), no. 1 41–49.

[75] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)* (1996) 267–288.

[76] L. Breiman, "Better subset regression using the nonnegative garrote," *Technometrics* **37** (1995), no. 4 373–384.

[77] J. Durbin and G. S. Watson, "Testing for serial correlation in least squares regression. i," *Biometrika* **37** (1950), no. 3-4 409–428.