Gasser Elbanna
Louise Placidet

Miniproject: Hopfield Networks

Introduction

In this project we will be exploring some aspects and properties of Hopfield Networks. We will first take a look at the consequences of using patterns with values ${\xi^\mu}_i \in \{0,1\}$, and how it is possible to store images in this network. Then, we will consider an alternative to the definition of capacity, which considers the maximum number of patterns such that they are *effectively stored* with a probability of 0.99, and compare that to the theoretical capacity obtained from flipping pixels. Furthermore, we will take a look at binary neurons which have a specific gain function, which saturates according to the steepness β value. Lastly, we will consider correlated patterns and their effect on capacity.

1. Theory Questions

<u>Question 1.1</u>: While most machine learning based object categorization algorithms require training over hundreds or thousands of samples/images in very large datasets in addition to optimizing the weights of the model, one-shot learning learns information about object categories and defines its weights in one step from one training sample/image to then classify many new examples. As a result, one can consider the learning rule in a Hopfield model to be a one-shot learning given that the Hopfield model memorizes patterns from just one sample and there is no training set consisting of multiple examples. Also, the weights of the Hopfield model are analytically derived in one step.

<u>Question 1.2</u>: Replacing the values that the patterns take the binary values 0 or 1 leads to a new definition of the interaction weights:

$$w_{ij} = \frac{1}{N}\sum_{\mu=1}^{P} \xi i^{\mu} \xi j^{\mu} \in [0,1]$$

According to week 6, this modification allows us to have a more realistically biological interpretation, considering attractor memory with ‘low’ activity patterns. Indeed, having patterns with values equal to -1 or +1 corresponds to an activity level of 50% of the network, which is neurobiologically very high.

However, introducing patterns with values 0 and 1 means that the weights will have values between 0 and 1. As a result, if a bit is flipped in a pattern, this version of the Hopfield model will not allow correcting this. Indeed, the pattern will systematically converge to 1. Indeed, when running a simulation (cf our notebook) with patterns that only have 0 or 1 values, the dynamics of the network always return patterns with only 1 bits. This Hopfield model no longer has an inhibitory group, which would allow it to correct the flip bits.

<u>Question 1.3</u>: We are now considering P patterns ξ consisting of N pixels taking binary values 0 or 1. In order to store these images in the classic Hopfield network defined above, the weights in the weight matrix are:

$${p^\mu}_i = {\xi^\mu}_i - b \quad {p^\mu}_j = {\xi^\mu}_j - a$$

${p^\mu}_i = 2{\xi^\mu}_i - 1$ *(Assuming a & b = 1/2)*

$$w_{ij} = \frac{1}{2a(1-a)N}\sum_{\mu=1}^{P} ({\xi^\mu}_i - 1/2)\,({\xi^\mu}_j - 1/2)$$

Gasser Elbanna
Louise Placidet

$$\sigma_i = 0.5(S_i + 1)$$

$$h_i(t) = \sum_j w_{ij} S_j$$

$$h_i(t) = \frac{1}{a(1-a)N} \sum_j \sum_\mu ({\xi^\mu}_i - b){\xi^\mu}_j \sigma_i - \frac{1}{a(1-a)N} \sum_j \sum_\mu ({\xi^\mu}_i - b) a \sigma_i$$

Indeed, by modifying the weights formula, we are able to take into account the inhibitory effect and avoid the problem encountered in the previous question. This is coherent with Dale's law, stating that neurons make either excitatory or inhibitory synapses, never both.

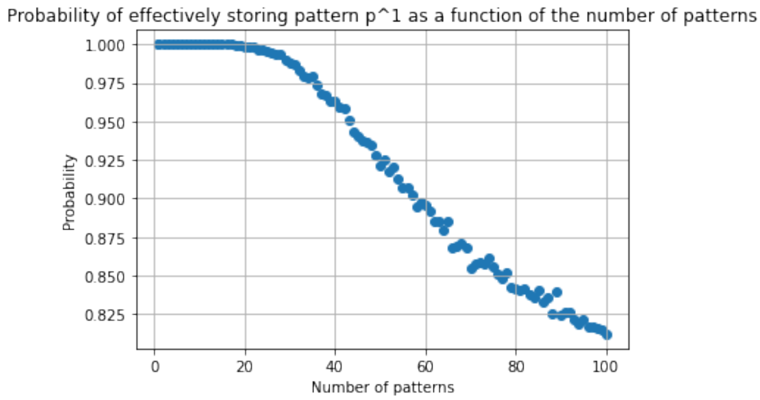
2. Capacity

Question 2.1:

In the context of this mini-project, the capacity is defined as the ratio P*/N, where P* is the maximum number of patterns that are *effectively stored* with a probability of 0.99. In this case, "effectively stored" indicates that after T time-steps, the network has the highest overlap with the pattern that was initially given to it.

In the case of the *Neuronal dynamics* book, capacity is defined as P*/N, where P* is the maximal number of patterns that a network can retrieve. A pattern is retrieved if you accept at most 1 bit wrong: this relies on the probability of having an error (meaning that for one neuron j, Sj(t=1) ≠ Sj(t=0)). However, they computed this probability using the Central Limit Theorem, which we cannot use. Indeed, with our newly defined patterns whose value is 0 or 1, the expectation is no longer equal to zero, but 0.5. Therefore, our variable no longer has a mean of zero, and we cannot apply the Central Limit Theorem.

Question 2.2:



From the plot above, one can conclude that the probability for pattern p^1 to be *effectively stored* decreases with the patterns P.

Question 2.3:

When computing the empirical capacity, we obtain 0.31 (this value shows variability due to the randomly generated patterns). For the theoretical capacity, one has to recompute the maximum number of patterns P* such that :

$$erf\left(\sqrt{\frac{N}{2(P-1)}}\right) = 1 - \frac{2}{N} = 0.98 \quad \text{with} \quad erf(x) = \frac{2}{\pi}\int_0^x e^{-t^2} dt$$

Gasser Elbanna
Louise Placidet

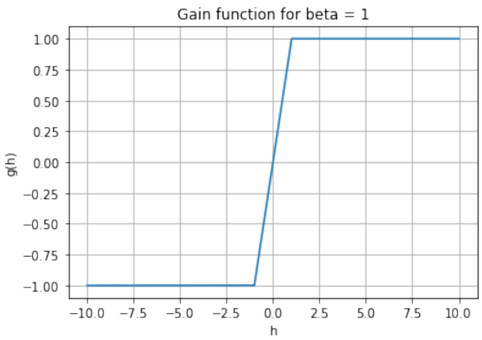
When solving for x in erf(x) = 1 - 2/N, we obtain according to wolframalpha, x = 1.64497635...

We can then deduce: $P* = 1 + \frac{N}{2x^2}$

This gives us the theoretical value of 0.19477, which is smaller than the empirical value computed experimentally. This indicates that the definition of “effectively stored” leads to a higher capacity than the theoretical definition of retrieving patterns with at most 1 bit wrong.

3. From binary neurons to saturated rectified linear neurons

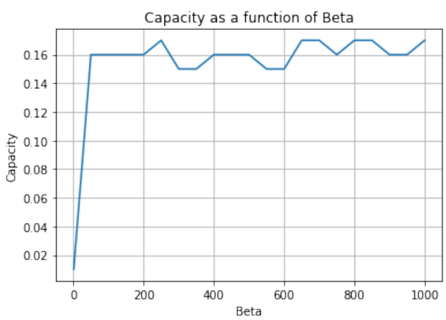
Question 3.1: With β=1, the g function plot is:

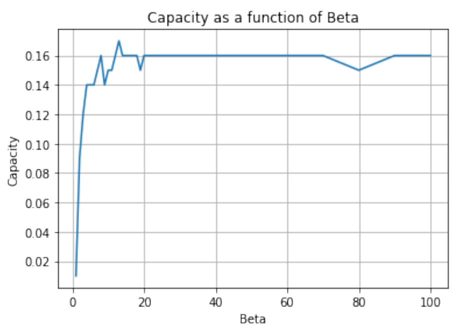


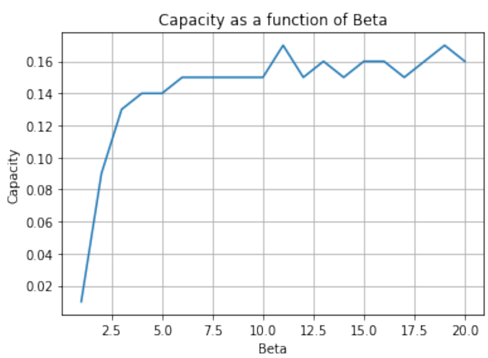
Question 3.2: For a Hopfield Network of N neurons with identical gain function g (and arbitrary β>0), the update rule is: $S_i(t+1) = min(1, max(-1, \beta h_i))$

$$= min(1, max(-1, \frac{\beta}{N}\sum_{j=1}^{N}\sum_{\mu=1}^{P} p^{\mu}{}_i p^{\mu}{}_j S_j(t)))$$

Question 3.3:



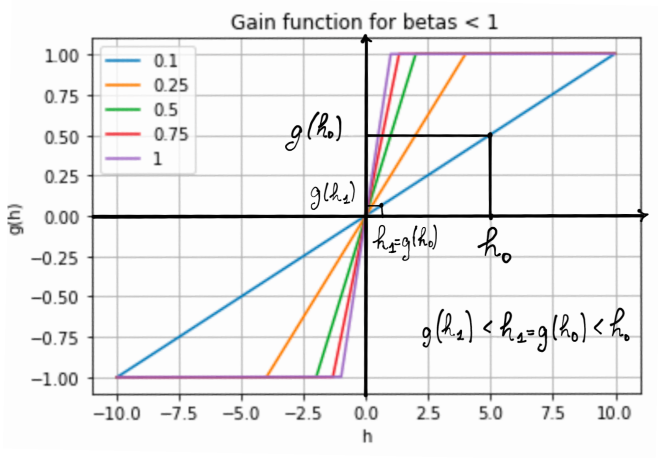




Gasser Elbanna
Louise Placidet

According to this plot, when the steepness increases, it does not seem to affect the capacity of the network. Indeed, this plot indicates that the capacity rapidly converges to 0.16 and oscillates around this value given the randomness of the patterns (starting from when β reaches 20, which is a relatively small value and we don't need to have β →∞).

Question 3.4: If ß<1, then we have the following plots:

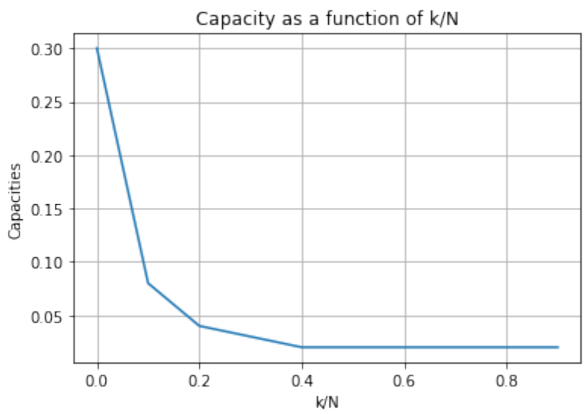


In the plot above, we notice that when β < 1, the gain function obtained from the input potential is smaller in value (given the slope, which decreases with β). This means when the steepness decreases, the activity computed from the input potential will consequently be lower. Indeed, this can be observed when running a simulation with β = 0.5, we can observe this decrease in activity over the iterations (see notebook for simulation). As a result, given that we reinject an input potential which is getting smaller and smaller, we observe an activity which converges to 0.

Question 3.5:

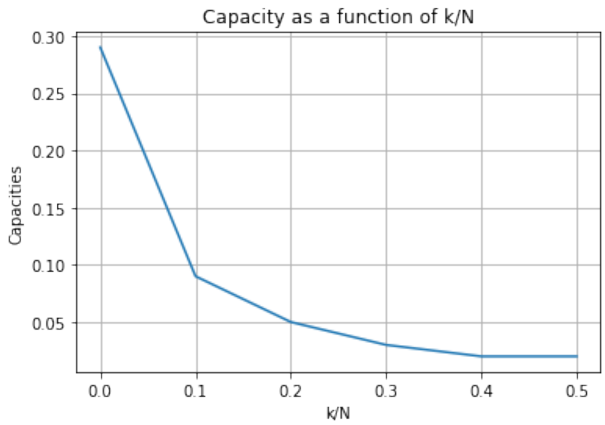
If we have ß grow to infinity, our g function becomes very similar to the sign function previously used. We are now in a deterministic case. In fact we wouldn't need to reach infinity since it converges considerably fast as illustrated in 3.3 (at ß=20).

4. Correlated patterns



When increasing the correlation between the different patterns, the capacity decreases exponentially. This indicates that correlation between patterns impacts the capacity.

Gasser Elbanna
Louise Placidet



Conclusion

In this project we examined the capacity of a Hopfield Network which has a more biological realism (Generalized Hopfield networks).
Accordingly, we noticed that by changing the pattern values from {-1,+1} to {0,+1}, in order to make it more biologically accurate, led to all patterns converging to 1 and the neurons didn't get corrected (flipped) throughout the simulation. Thus, the weight formula had to be adjusted, resulting in the ability to make apparent excitatory and inhibitory interactions.
Moreover, it was shown that the probability of effectively storing the first pattern decreased by increasing the number of patterns. Also, we observed that the capacity computed empirically (i.e. using our simulation), is quite similar to the theoretical capacity defined in the book and exercises.
Then, we examined a different gain function in which we used several hyperparameters (β), which represent the steepness, to test its consequences in a Hopfield network. Consequently, we observed that the capacity rapidly converged and stagnated to a specific value as β increased (>1) and as β tends to infinity, the gain function behaves similarly as a sign function. On the other hand, when β<1, we noticed that it led to a decrease in the population activity.
Finally, we used correlated patterns in our simulations to see its effect on capacity. As expected, we found that the capacity decreased drastically with increasing the correlated pixels in patterns.