Artificial Neural Networks and Deep Architectures, DD2437

Short report on lab assignment 3

Hopfield Networks

Ravi Bir—ravib@kth.se
Bharat Sharma—bsharma@kth.se
Qiao Ren—qiaor@kth.se

March 1, 2020

1 Main objectives and scope of the assignment

Our major goals in the assignment were

- explain the principles underlying the operation and functionality of auto associative networks.
- train the Hopfield network.
- explain the attractor dynamics of Hopfield networks the concept of energy function
- demonstrate how auto associative networks can do pattern completion and noise reduction.
- investigate the question of storage capacity and explain features that help increase it in associative memories.

2 Methods

Jupyter Notebook, Numpy, Matplotlib

3 Results and discussion - Hopfield Networks

3.1 Convergence and Attractors

The network was able to store all the patterns successfully.

No,with Little model update, x2d didn't converge to the stored pattern. Only x1d and x3d were able to converge to their stored patterns x1 and x3.

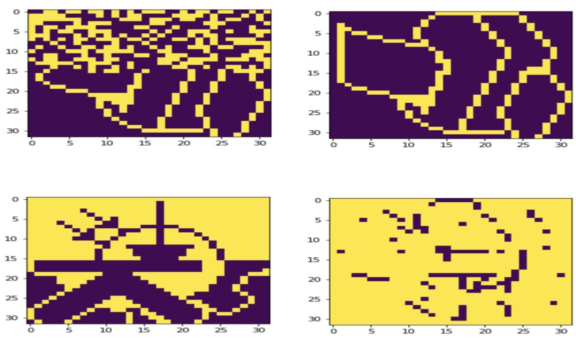
By trying out every possible combination of input patterns, we found that they converged to a total of 14 attractors.

When we make the distorted patterns dissimilar to the stored ones by more than 50 percent, the patterns don't converge to the given attractors anymore, they converge to different patterns.

3.2 Sequential Update

The first 3 patterns p1, p2 and p3 are stable as they converge to the original patterns when 30-40 percent of their bits are reversed.

In batch mode, the network was able to retrieve the complete p1 pattern from p10 but nothing from p11.



Figur 1: p10 and p11 input and output batch mode

In sequential mode, the network was able to retrieve p1 from p10 and was able to retrieve p3 from p11, but not p2 from p11.

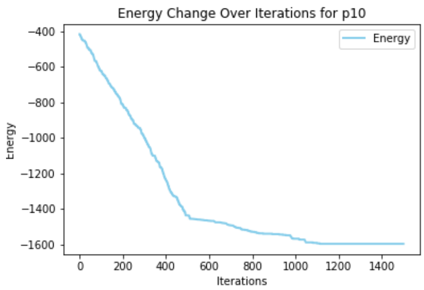
3.3 Energy

For the sequential Hopefield model, where the weights are updated one at a time, we can implement an energy function. It will always decrease as the state changes. Since it has to have a minimum at least somewhere the dynamics must end up in an attractor. Below is a table summarising the energy at different attractors and distorted patterns.

Image	Type	Energy
p1	Attractor	-1439.390625
p2	Attractor	-1365.640625
p3	Attractor	-1462.25
p10	Distorted	-1596.011718
p11	Distorted	-173.5

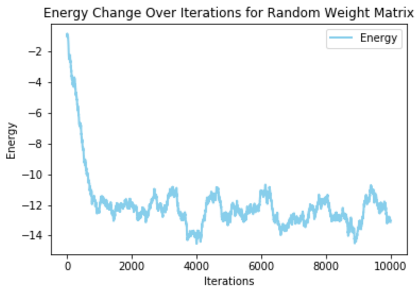
Tabell 1: Energy at Attractors and Distorted Patterns)

Below is a graph showing how the energy changes as the number of iterations increases for p10. It decreases quickly as first, but eventually the rate of energy decrease slows down until it reaches itâs final value of around -1600 and remains unchanged.

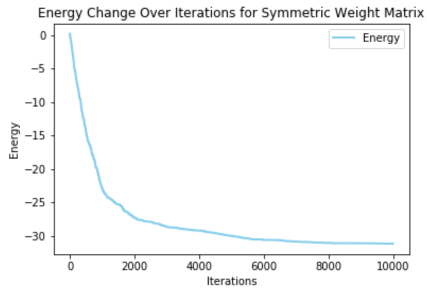


Figur 2: p10 Energy vs Iterations

When the weight matrix is set to normally distributed random numbers, the energy keeps oscillating. It increases and decreases and does not converge to a value. When the weight matrix becomes symmetric, the rate of energy decrease is quick at first, but then is slows down and eventually converges to a value.



Figur 3: Random Weights



Figur 4: Symmetric Weights

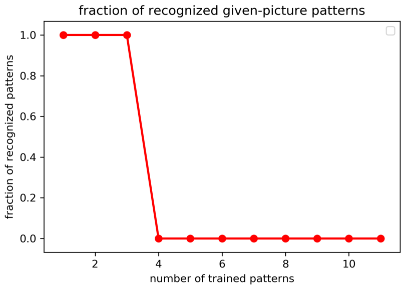
3.4 Distortion Resistance

The patterns are able to converge to the original patterns for up to 45 percent noise. After 45 percent the networks are not able to retrieve the original patterns.

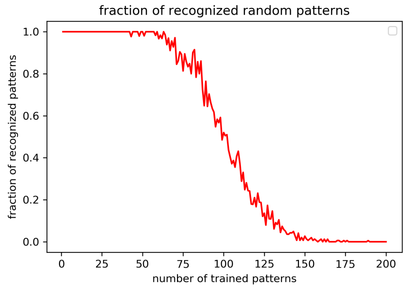
Figur 5: Distortion Resistance p1-p3

Beyond 55 percent noise, the network starts to converge to the opposite patterns.

p1 and p3 are able to retrieve the original patterns for up to 45 percent noise, whereas p2 was only able to retrieve up to 30 percent noise. Hence p1 and p3 are better attaractors in this case.



Figur 6: given images

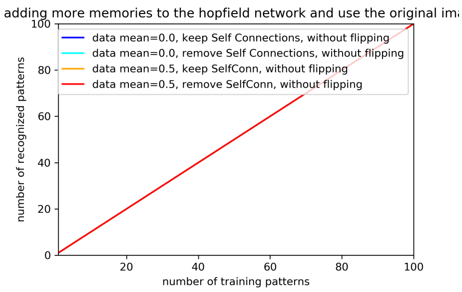


Figur 7: random images

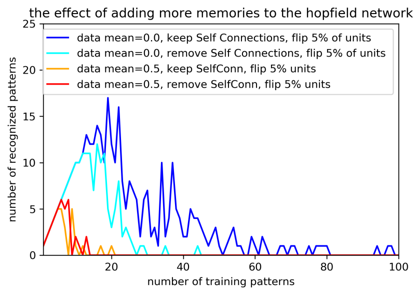
3.5 Capacity

Figure 6: For the given patterns, the capacity of the network is to store maximum 3 patterns. When more than 3 patterns are fed into the weight matrix, the performance drops to zero abruptly. The capacity performance is defined as: the amount of patterns that the network is able to recognize divided by the amount of patterns that are used for training the weight matrix.

Figure 7: For random patterns, the capacity of the network is to store maximum 50 patterns. The fraction of recognized patterns drops down when the training patterns is more than 50.



Figur 8: recover patterns from undistorted arrays

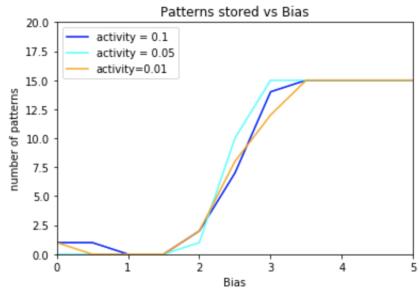


Figur 9: recover patterns from distorted arrays (5 percent of units are flipped)

Figure 8 shows that the number of stable patterns increases with the number of patterns used for training the network.

Figure 9: The largest capacity is the at peak of each line. When there is no bias in data (the probability of getting 1 and -1 is 50 and 50 percent), keeping self connections has larger capacity than removing self connections. When data has bias (in this case, the 1 has larger probability to appear), removing self connections has similar capacity as keeping self connections. Bias in the data reduces the capacity of the network.

3.6 Sparse Patterns



Figur 10: Capacity vs Activity vs Bias

From our graph we can see that from the bias values that we tested, as bias increases, the number of patterns also increases. This trend can be seen for every activity level. The highest number of patterns that can be stored (15) can be seen at a bias of around 3.4 for each activity level.

4 Final Remarks

We feel that Lab 3 aided us a lot in improving our understanding of auto-associative networks. The visualisation of the patterns helped a great deal. It was well structured and easy to follow.