Computational Neuroscience: Neuronal Dynamics of Cognition

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

A: ASSOCIATIVE MEMORY in a Network of Neurons

Wulfram Gerstner

EPFL, Lausanne, Switzerland

1 Introduction
- networks of neuron
- systems for computing
- associative memory

2 Classification by similarity

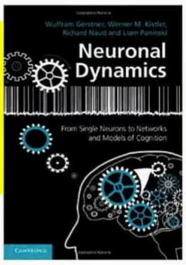
3 Detour: Magnetic Materials

4 Hopfield Model

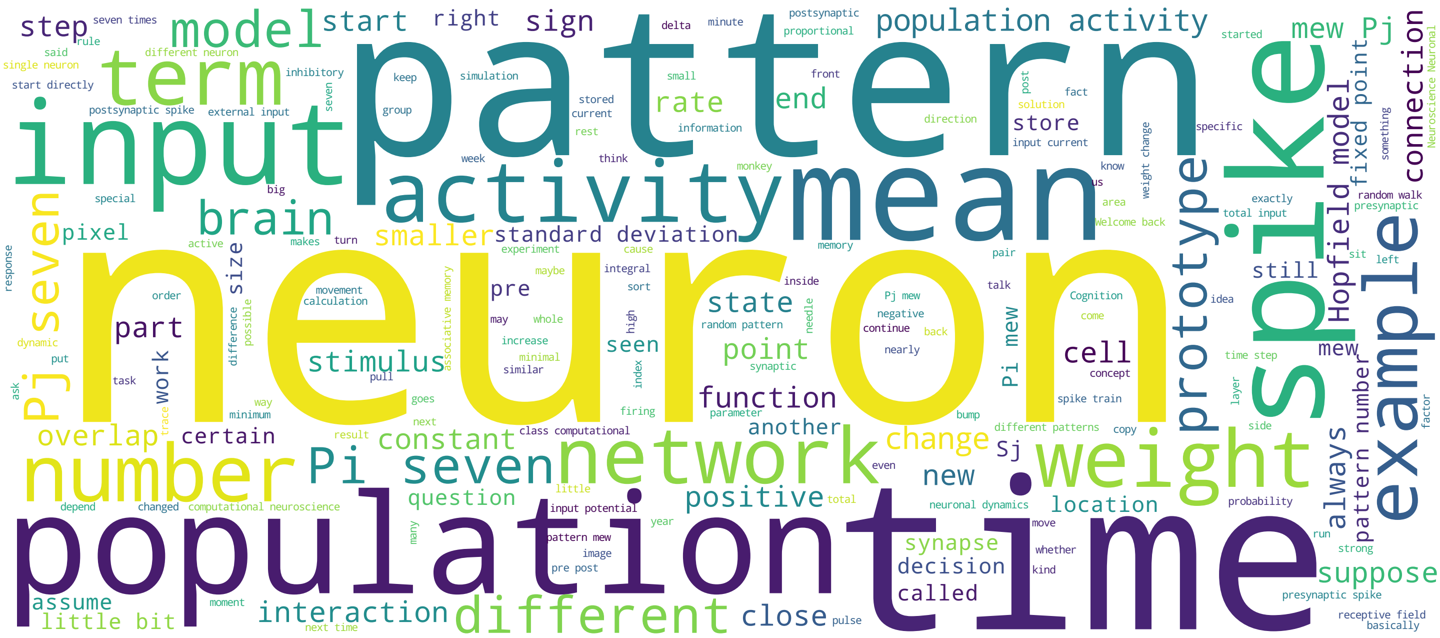
5 Learning of Associations

6 Storage Capacity

Reading for this week:
NEURONAL DYNAMICS
- Ch. 17.1 - 17.2.4



Cambridge Univ. Press

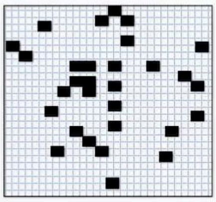


Search MOOC

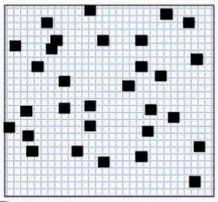
Video

EPFL

6. learning of several prototypes



Prototype $\vec{p}^1$



Prototype $\vec{p}^2$

interactions

$$(1) \quad w_{ij} = \frac{1}{N} \sum_{\mu} p_i^{\mu} p_j^{\mu}$$

Sum over all prototypes

Question: How many prototypes can be stored?

dynamics

$$S_i(t+1) = \mathrm{sgn}[\sum_j w_{ij} S_j(t)]$$

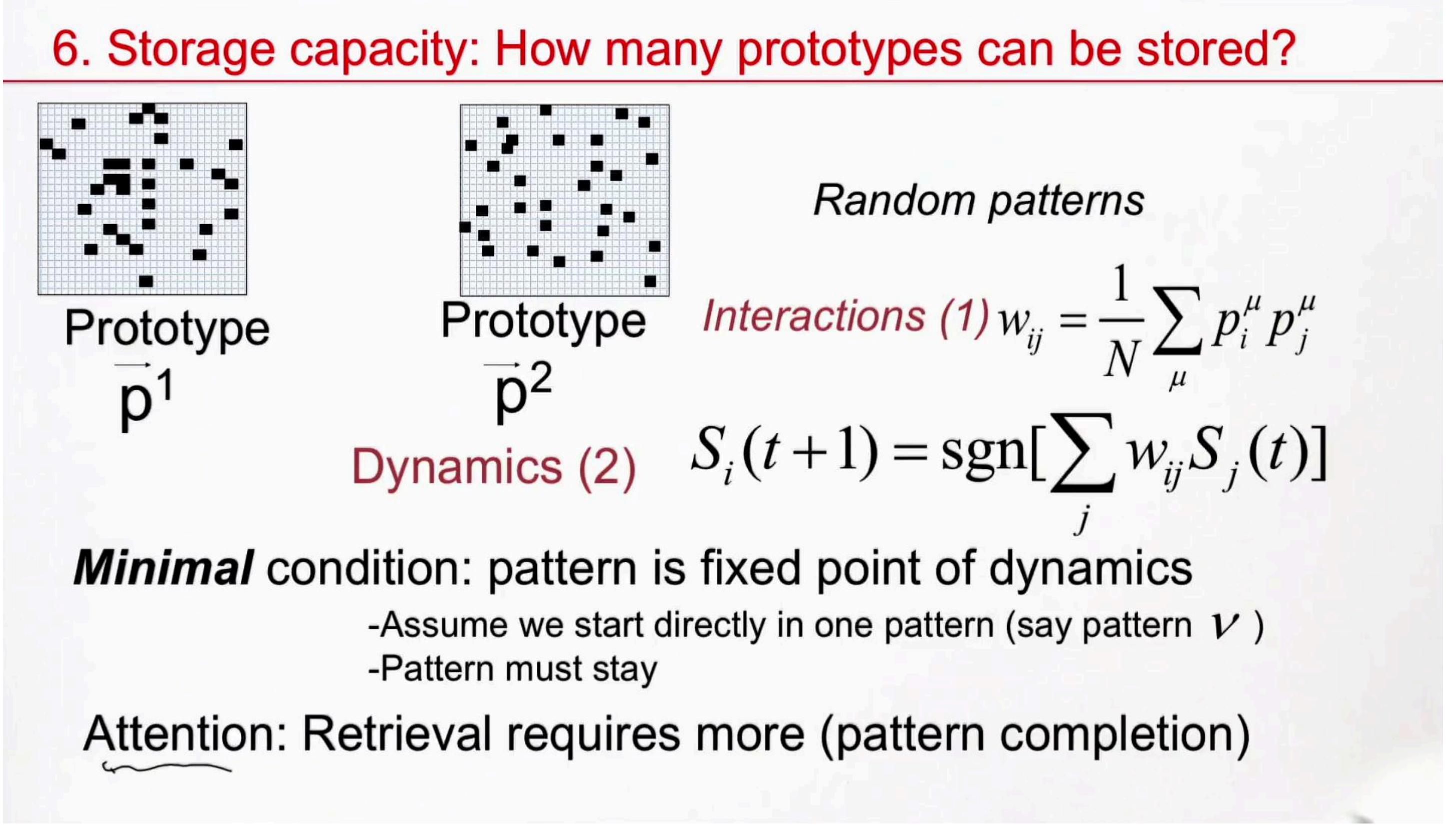
all interactions with i

So welcome back to the class computational neuroscience, we look again at a model of associative memory. And the question we would like to ask now is, we have seen that we can store several prototypes, but how many prototypes can really be stored so here again, the rules of the game in the Hopfield model, either dynamics, binary state variables. And these dynamics includes weights. And the weights are just the sum over the different contributions of the different patterns. And that added here, a one over and in front, which has no real meaning, because I look at the sign afterward, but it simplifies a little bit the calculations.

Notes

Summary

0m 08s



So what we want is that we start in an image that's noisy, that sort of noisy version of one of the prototypes, and then it corrects and skips spec, the pure prototype. That's the task. However, that's a little bit hard to calculate. So we make a really minimal condition. Suppose you start directly in the prototype. So you start in prototype number two, then at least, if you run the dynamics, you should not move away from the prototype, you should stay in the prototype. Okay, so we assume we started one of the prototypes. And the minimum condition is that the pattern should stay attention retrieval really requires more be monitored on completion. But this is really a minimal condition.

Notes

Summary

0m 49s

6. Storage capacity: How many prototypes can be stored?

-Assume we start directly in one pattern (say pattern 7)
-Pattern must stay

Interactions (1)

$$w_{ij} = \frac{1}{N} \sum_{\mu} p_i^{\mu} p_j^{\mu}$$

$$S_i(t+1) = \mathrm{sgn}[\sum_j w_{ij} S_j(t)]$$

$$= \mathrm{sgn}\Big[\sum_j \frac{1}{N} \sum_{\mu=1}^{M} p_i^{\mu} p_j^{\mu} p_j^{7} \Big] \quad \mu=7$$

$$= \mathrm{sgn}\Big[\sum_j \frac{1}{N} p_i^{7} \cdot p_j^{7} \cdot p_j^{7} + \sum_{\substack{\mu \\ \mu \neq 7}} \frac{1}{N} \sum_j p_i^{\mu} p_j^{\mu} p_j^{7} \Big]$$

So let's look at this minimal condition and assume that we start directly in one of the patterns, say pattern number seven. What does this mean? This means that for each j, the value of Sj is equal to Pj of patterns seven so at the time, each neuron is in this state that's consistent with pattern number seven. Now, what do we know about the weights? Well, for the weights, we have this formula, which I just plug in. So I have a 1 over N sum over mew Pi mew Pj mew. And then I copied a sign function, and summation signs and close the brackets. So suppose we have M patterns, so that sum runs over all the patterns from one to M. M is, for example, 25. And as we sum up, you have miracle 1234567. But seven is special. And then we continue 89, 10 until M 7 is Special, Let’s pull out this special pattern. And therefore, I speed it up, I say in this sum if the case new equals seven, then in this sum of the case, I sum of the mew, but mew could not be equal to seven. Especially all the rest so let's look at the first part sign off, I have a sum over J one over N. And nowhere, I've mew equals seven, so I have Pi seven, Pj seven, Pj seven. And then after rest, which I can just copy, one over N sums over J Pi mew Pj mew Pj seven.

Notes

Summary

1m 41s

6. Storage capacity: How many prototypes can be stored?

-Assume we start directly in one pattern (say pattern 7)
-Pattern must stay

$$S_i(t+1) = \mathrm{sgn}[\sum_j w_{ij} S_j(t)]$$

$$= \mathrm{sgn}\left[\sum_j \frac{1}{N} \sum_{\mu=1}^{M} p_i^\mu p_j^\mu p_j^7\right]$$

$\mu = 7$

$$= \mathrm{sgn}\left[\sum_j \frac{1}{N} p_i^7 \underbrace{p_j^7 \cdot p_j^7}_{=1} + \sum_{\mu \neq 7} \frac{1}{N} \sum_j p_i^\mu p_j^\mu p_j^7\right]$$

$$= \mathrm{sgn}\left[p_i^7 + \frac{1}{N} p_i^7 \sum_{\mu \neq 7} \sum_j p_i^7 p_i^\mu p_j^\mu \cdot p_j^7\right]$$

Interactions (1)

$$w_{ij} = \frac{1}{N} \sum_\mu p_i^\mu p_j^\mu$$

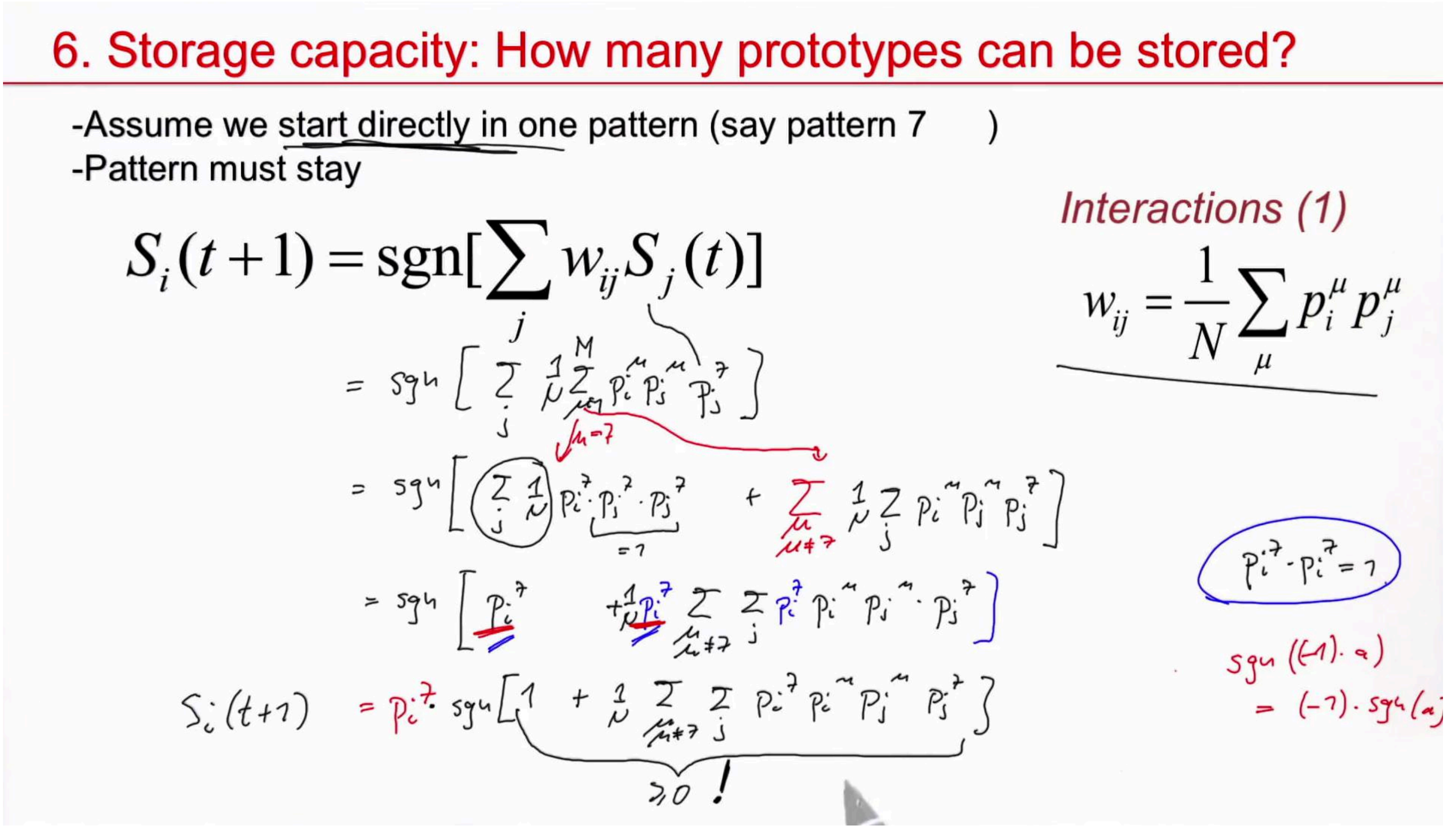
$$p_i^7 \cdot p_i^7 = 1$$

Now let's look at this term here. There's something interesting, because PJ seven times PJ seven, this is the square. So PJ is plus or minus one, suppose it's minus one, minus one time minus one. There's one if it's one at one, time one. So whatever the value, the square, PJ seven times PJ seven is one. So this is always equal to one. I have the end of these terms, because of some over G, so in the end, the one over and we'll go because of N terms and what remains is just Pi seven. Now let's look at the other term. Plus and I have a sum overall patterns mew, except pattern seven, either one over N there for the sum over j. And then I've Pi mew Pj mew Pj seven. So this term looks complicated. But for the moment, let's make it just a bit more complicated. The reason why I want to do this will become clear in a minute. So we have seen before that Pj seven times seven is always one. Same is true for Pi seven times Pi seven is always equal to one. So I can block this in, I can always multiply with one. That's right, one of these Pi seven here, and the other Pi seven here, nothing has changed, I've multiplied with one. Now the advantage is the value C I have a term here Pi seven.

Notes

Summary

3m 47s

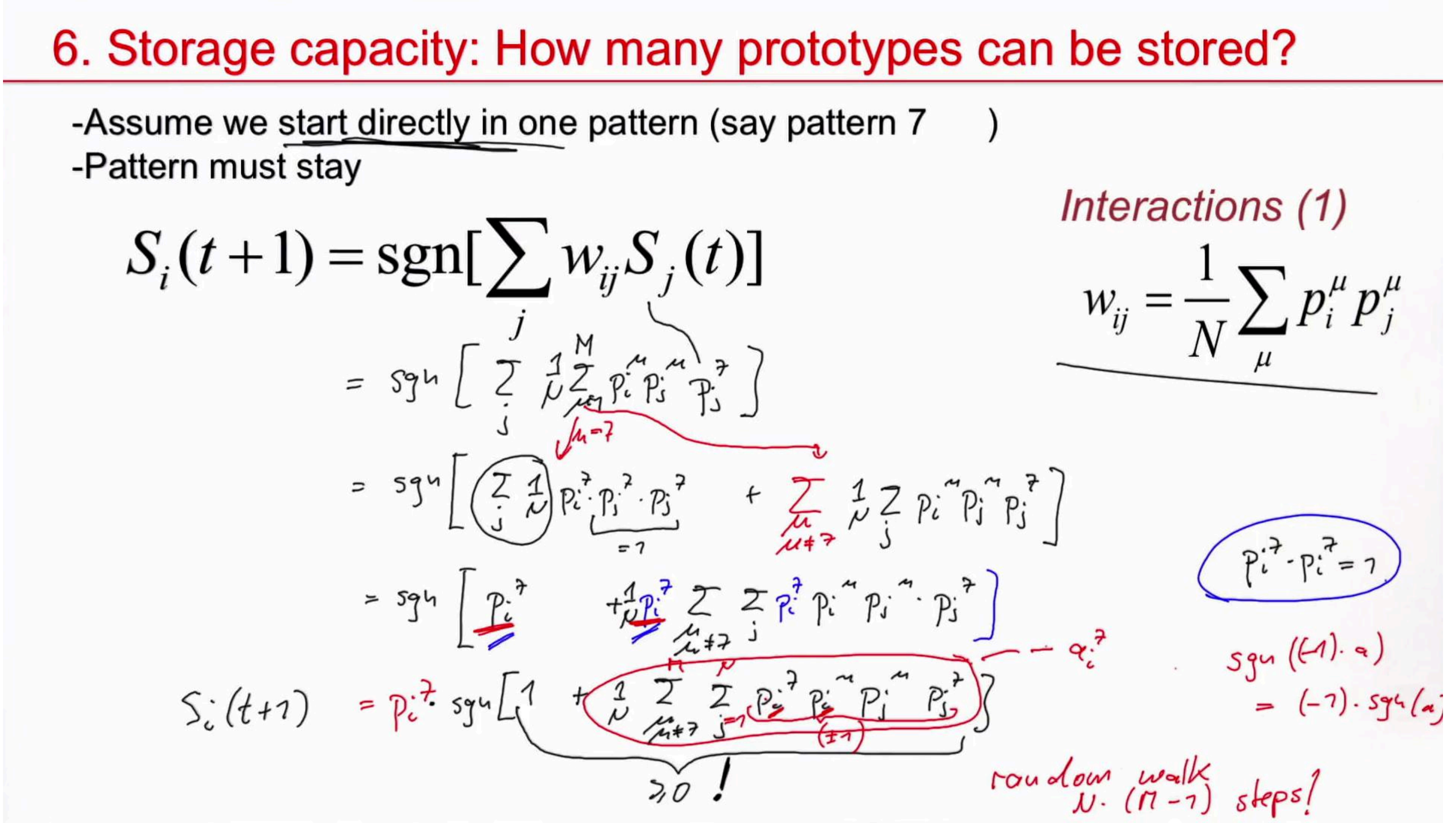


And if term here Pi seven, it's the same, no little aside, think of sign off minus one time A, that's the same as minus one times sign of A. That means if I have a number minus one, I can just pull it out, if I have number plus one, I can pull it out anyway. So I can take this Pi seven, invited in front, equal's Pi seven. And then I have to sign I've pulled this out, this is one plus one over N I've pulled this out the sum of mew, but not mew equal seven, some over j Pi seven Pi mew Pj mew Pj seven. And that close. This is the end of the calculation may look horrifying to you. But it's actually quite illuminating. Let me try to explain it a little bit. So he said, we start directly in one of the patterns. So we had that Sj at time t is equal to the pattern Pj seven. Now, what defined is that Si at time t plus one in the next time step is equal to Pi seven? Well, if this guy here is positive, then the sign of this is just one. So mew and I in the next time step is impaired on seven stays and impaired on seven if determine square brackets are larger than zero, this is what you want. But we don't necessarily have this.

Notes

Summary

5m 35s

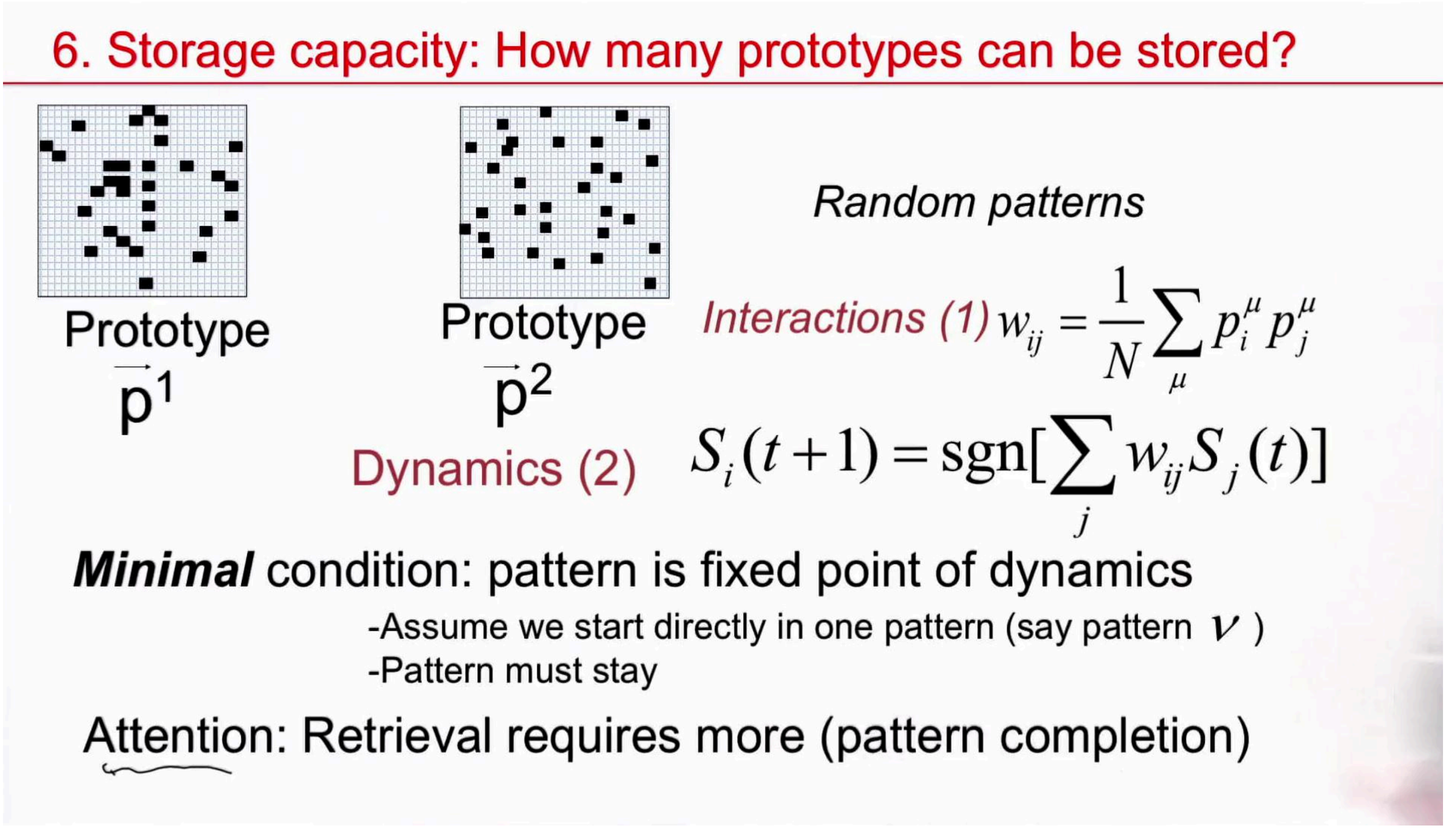


So I can rewrite this by saying this whole term here that's called this minus Ai seven just a constant A parameter A in the need to index i because i is here, i somewhat j goes, i jump over i sum over mew. Mew goes but the seven remains. So I have two indices. And this red thing, I define it as minus Ai seven. So what can we say about this? This guy here, whether it's multiplication, I have here plus-minus one plus-minus one plus-minus one plus-minus one. So the whole thing inside here is plus minus one. And then I sum. I sum over N of these plus-minus ones, and that sum over N minus one. Because I don't have this pattern numbers seven in there. So basically, what we have is plus one minus one plus one plus one minus one plus one. So I have a random walk, I go up and down plus one minus one in a random fashion. And I do this over how many steps? Well, its N times N minus one steps. So what we want is that this random walk stays limited. So that this part here inside the square brackets is positive so let's look at it.

Notes

Summary

7m 29s



At the next slide. The minimum condition that we had is we start in a pattern say pattern mew now case pattern seven. And then we want this pattern stays.

Notes

Summary

9m 09s

Q: How many prototypes can be stored?

A: If too many prototypes, errors (wrong pixels) show up.
The number of prototypes M that can be stored
is proportional to number of neurons N;
memory load = M/N

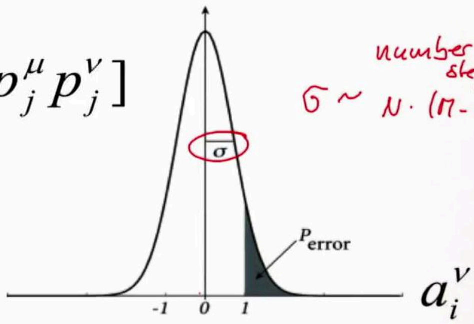
$$S_i(t+1) = p_i^\nu \operatorname{sgn}\Big[1 + \tfrac{1}{N} \sum_{\mu=1,\mu\neq\nu}^{M} \sum_{j=1}^{N} p_i^\mu p_i^\nu p_j^\mu p_j^\nu\Big]$$

$$= p_i^\nu \operatorname{sgn}[\underbrace{1 - a_i^\nu}_{>0!}]$$

Error-free if

$$S_i(t+1) = p_i^\nu$$

Gaussian



number of steps
$\sigma \sim N \cdot (N-1)$

Image: Neuronal Dynamics, Gerstner et al., Cambridge Univ. Press (2014),

And we find that this is possible. If these variables Ai mew now for arbitrary pattern mew now case it was seven is smaller than one. Because we want this to be positive, that's what we want. And therefore a must be smaller than one. Now for a random walk, which is completely unbiased, the mean will always be zero. But if you do 500 steps, you may actually end up close this here, but not exactly as you and you have a certain whisk of this distribution. And this whisk or standard deviation is proportional to the number of steps. How many steps do we have? Well, we have N times N minus one steps.

Notes

Summary

9m 23s

Q: How many prototypes can be stored?

A: If too many prototypes, errors (wrong pixels) show up.
The number of prototypes M that can be stored
is proportional to number of neurons N;
memory load = M/N

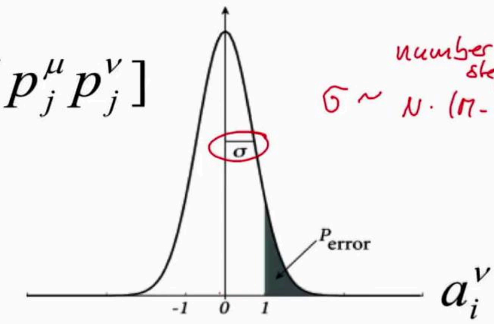
$$S_i(t+1) = p_i^\nu \operatorname{sgn}[1 + \tfrac{1}{N} \sum_{\mu=1,\mu\neq\nu}^{M} \sum_{j=1}^{N} p_i^\mu p_i^\nu p_j^\mu p_j^\nu]$$

$$= p_i^\nu \operatorname{sgn}[\underbrace{1 - a_i^\nu}_{>0!}]$$

Error-free if

$$S_i(t+1) = p_i^\nu$$

Gaussian



number of steps

$\sigma \sim N \cdot (M-1)$

Image: Neuronal Dynamics, Gerstner et al., Cambridge Univ. Press (2014),

So before the random walk with N times N minus one steps, the standard deviation is proportional to the square root of the number of steps. But then as a factor one over N in front so I want A to be smaller than one, which means that standard deviation has to be small. The standard deviation with step size one over N is one over N times original standard deviation. Which basically makes N minus one over N so what we see here is that if the number of patterns M number of patterns is of the same order as the number of neurons, then we run into a problem. So basically, if you've worked with 10,000 neurons, the number of patterns should be much smaller, for example, 200, then the standard deviation will be very small. And for nearly all the pixels for nearly all the neurons, we will be error-free. The storage capacity, asked the question, how many prototypes can be stored? The answer is we can store a number of prototypes that should be smaller than the number of neurons in the network significantly smaller, and then we will be fine. So we work with random patterns. In this case, we can have a number of patterns, M that small compared to the total number of years.

Notes

Summary

10m 22s

Q: How many prototypes can be stored?

A: If too many prototypes, errors (wrong pixels) show up.
The number of prototypes M that can be stored
is proportional to number of neurons N;
memory load = M/N

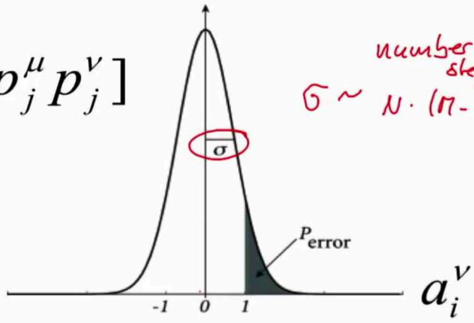
$$S_i(t+1) = p_i^\nu \operatorname{sgn}[1 + \tfrac{1}{N} \sum_{\mu=1,\mu\neq\nu}^{M} \sum_{j=1}^{N} p_i^\mu p_i^\nu p_j^\mu p_j^\nu]$$

$$= p_i^\nu \operatorname{sgn}[\underbrace{1 - a_i^\nu}_{>0!}]$$

Error-free if

$$S_i(t+1) = p_i^\nu$$

Gaussian



number of steps
$\sigma \sim N \cdot (M-1)$

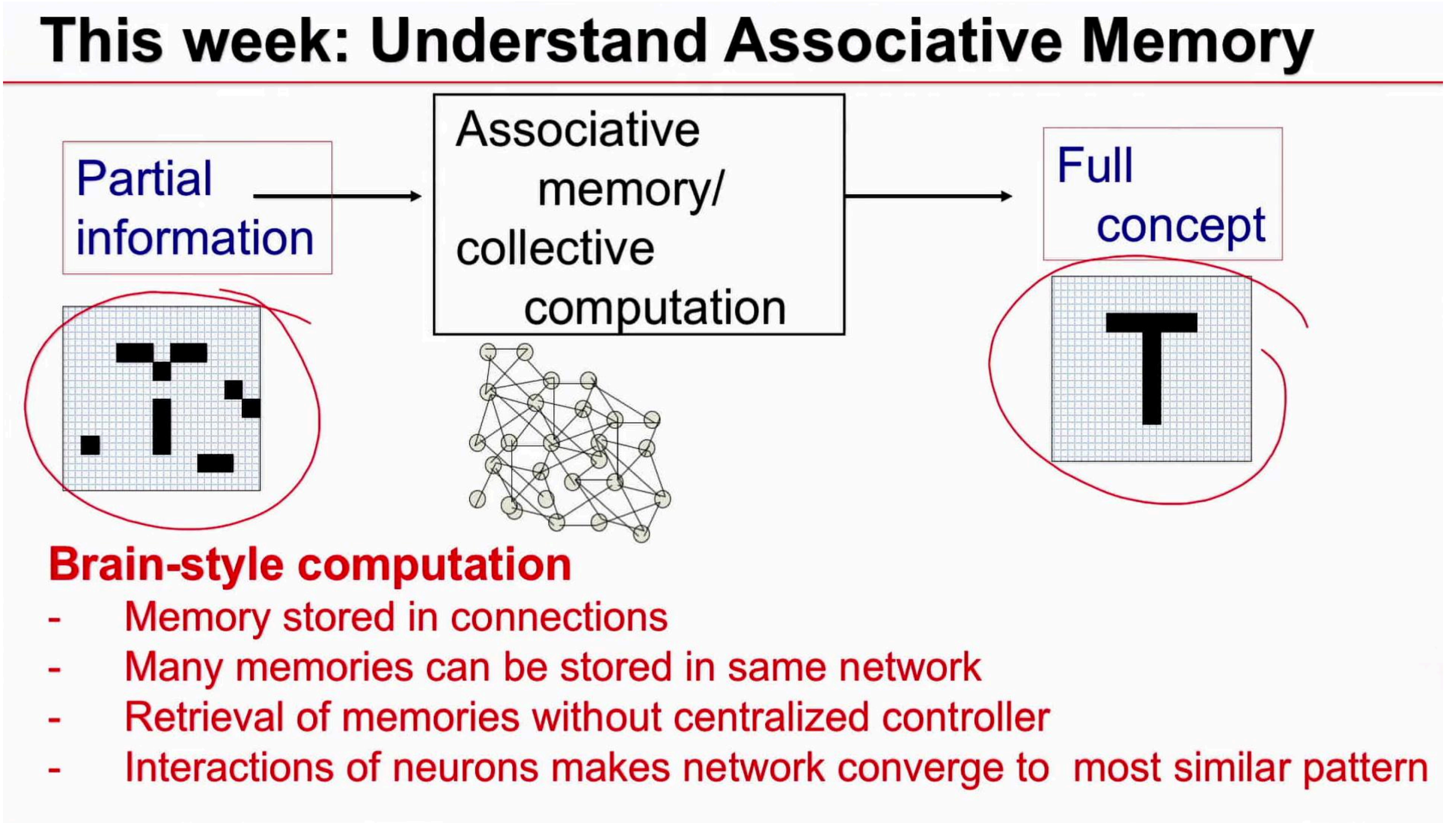
Image: Neuronal Dynamics, Gerstner et al., Cambridge Univ. Press (2014),

But at least if you have 10,000 years, we can store 200 patents, which is quite a lot. Now imagine the size of our brain. So it's not surprising that we can store a huge number of different patterns. In our brain, the size of the English vocabulary has been estimated to 100,000 words will be 100,000 different items. It's a lot of different concepts.

Notes

Summary

12m 20s



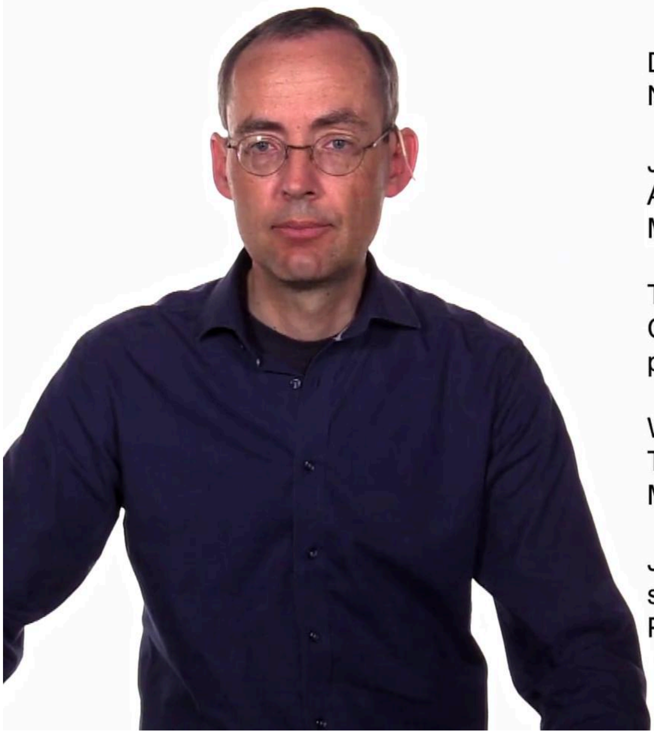
But why not we are able, in principle to store this in our brain so what I wanted to convey this week is this idea that memories are stored in the network of neurons more precisely in the connections between the neurons and we have we can retrieve this memory. And in the same network of neurons, we can store many, many memories. Specifically, in the Hopfield model. Each memory concept, each memory pattern is a random pattern. It's highly processed, but for these random patterns in a network of N neurons, we can store maybe in a network of 10,000 neurons, we would be able to store 200 or five from different patterns. Now importantly, there are simple computer algorithms that can retrieve a full concept from partial information. However, what I've shown here is that in the Hopfield model, we don't need a central controller. It's just the interaction between the neurons that makes the network converge to the most similar pattern that restores the full concept from partial information.

Notes

Summary

12m 49s

References: Associative Memory Models



D. J. Willshaw, O. P. Bunemann and H. C. Longuet-Higgins (1969)
Non-holographic associative memory. Nature 222, pp. 960–962

J. A. Anderson (1972)
A simple neural network generating an interactive memory.
Math. Biosc. 14, pp. 197–220

T. Kohonen (1972)
Correlation matrix memories. IEEE trans. comp. C-21,
pp. 353–359.

W. A. Little (1974)
The existence of persistent states in the brain.
Math. Biosc. 19, pp. 101–120.

J.J. Hopfield (1982) Neural networks and physical
systems with emergent collective computational abilities.
Proc. Natl. Acad. Sci. USA 79, pp. 2554–2558

So a model of memory models associative memory, models of auto-associative memories has a long tradition. I've presented here a version of the Hopfield model. But the tradition goes back much further. As you see on this slide there is David Willshaw, Kohonen, Anderson, and many others who have contributed to our understanding of how the brain could potentially form memories.

Notes

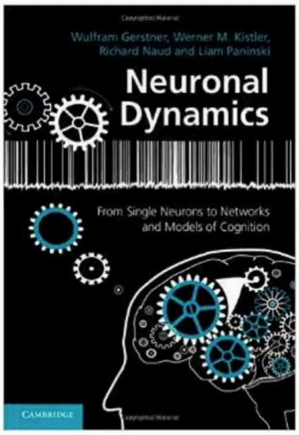
Summary

13m 57s

The end

Documentation:
http://neuronaldynamics.epfl.ch/

Online html version available



Reading for this week:
NEURONAL DYNAMICS
- Ch. 17.1 - 17.2.4

Cambridge Univ. Press

You find all the information in the book on neuronal dynamics for which you also have an online version. See you again next week.

Notes

Summary

14m 30s