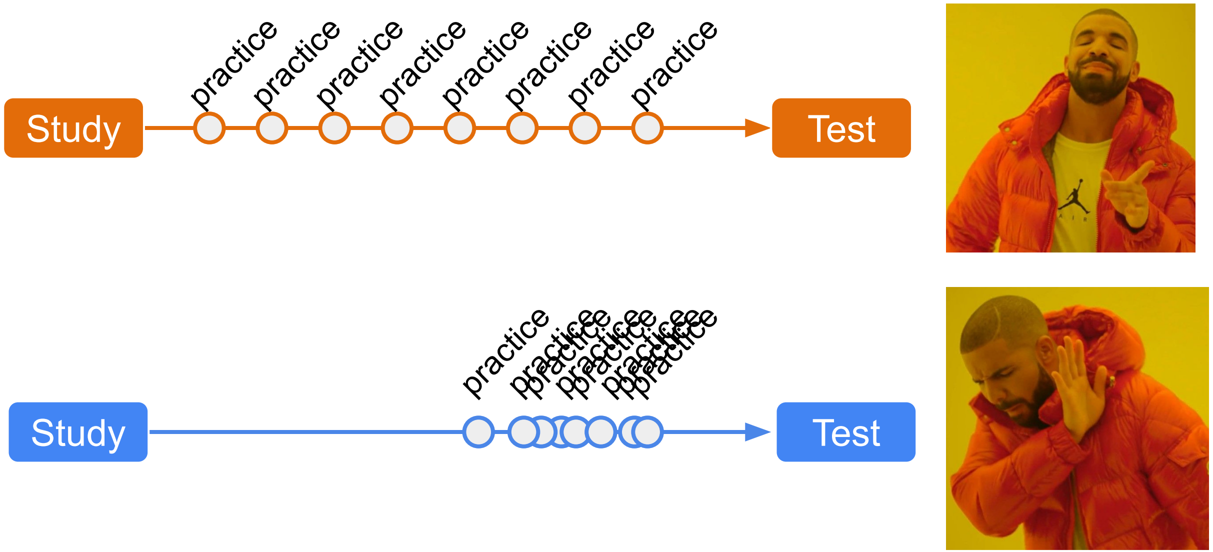
What is the spacing effect, and why does it exist?

Andrea Stocco,
taking a lot of ideas from Christian Lebiere

What is the spacing effect?

- When learning new things, you often **practice** them **multiple times**
- The **farthest** apart the practices, the **longer** the skills are retained

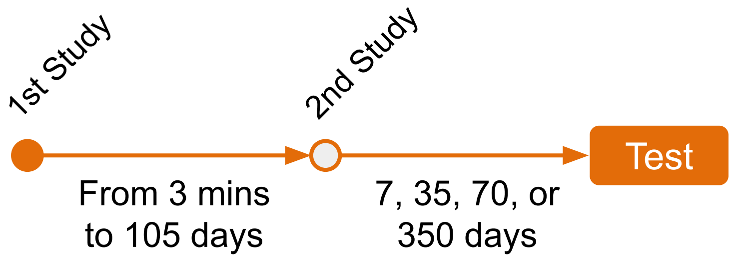


The spacing effect /2

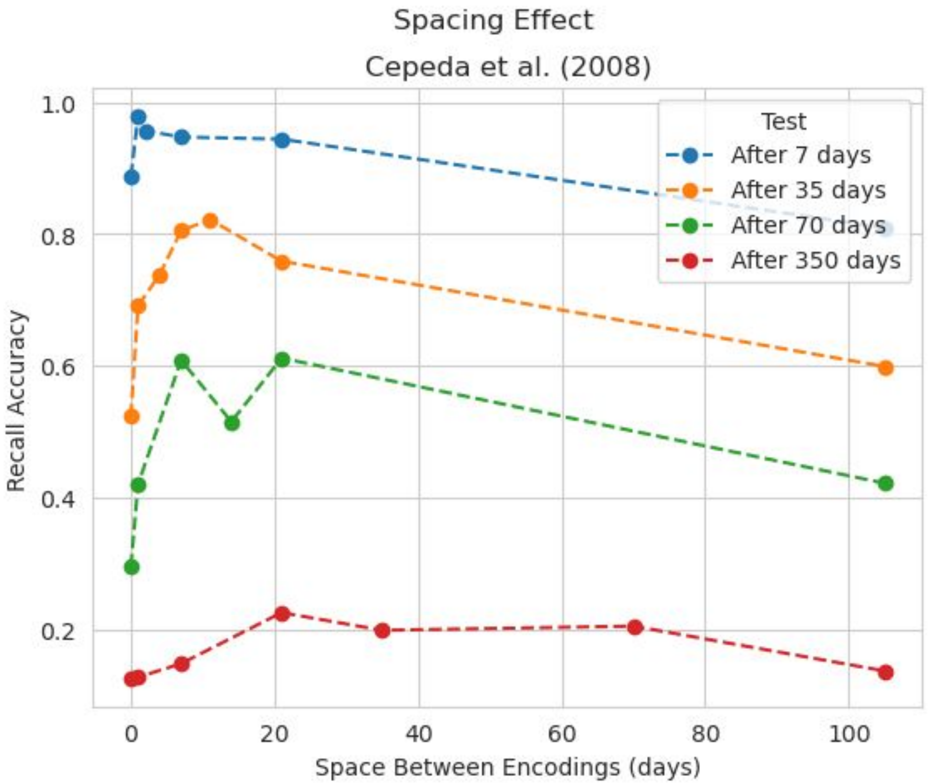
- AKA the **spaced practice** benefit, **spaced repetition** effect, etc.
- Was first observed by Ebbinghaus himself in 1884
  - First modern study of memory ever
- Mostly studied for declarative memories, but…
- … Has been shown for **procedural** memories as well
  - Real-life skills, like CPR
  - Complex skills, like surgical practice
  - Motor skills in athletes
- We have good descriptions, no good explanations

Cepeda's experiment (2008)

- One of my 6 all-time favorite memory papers
- Had participants memorize trivia questions and answers
  - E.g., "which country consumes the most hot sauce per capita? Norway"
- Each trivia was studied twice
- Systematically varied the **spacing** between the two study sessions and the **retention interval** before test.



Results from Cepeda et al., 2008



A model of long-term memory

- The model we use in the lab, first proposed by Anderson & Schooler (1991)
    - #2 on my favorite papers on memory
- Basically, a computational version of the Multiple Trace Theory (MTT)
    - See Stocco et al., 2023, *CoBB*, for the details
- Every time you encode something, it leaves a **trace** in your brain
- A memory is the result of the **accumulation of traces.**

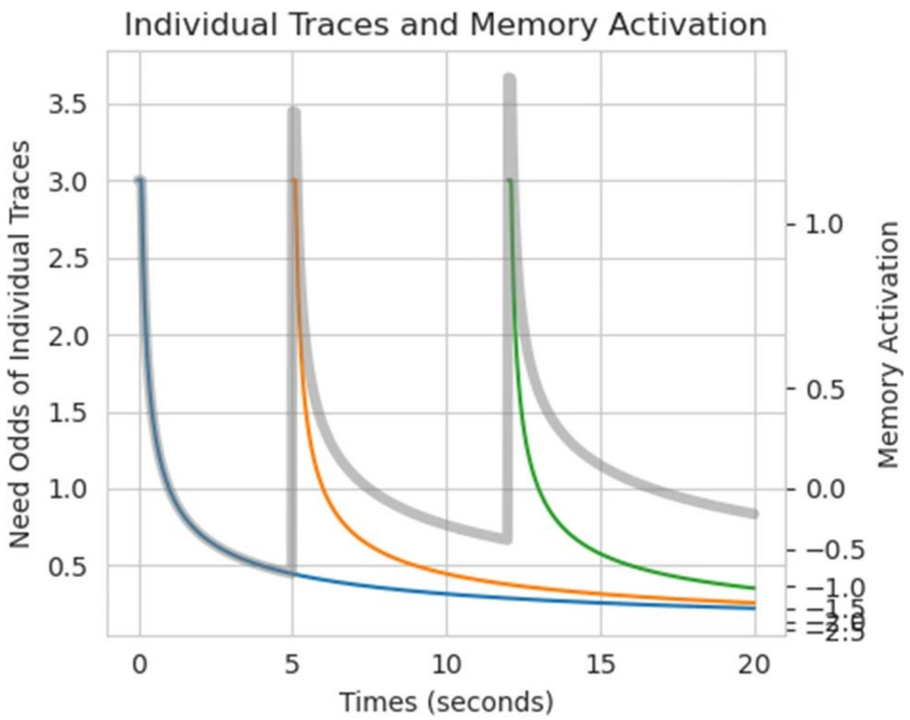
The model: Assumptions

- Odds of retrieving a trace decrease as a **power function** over time:

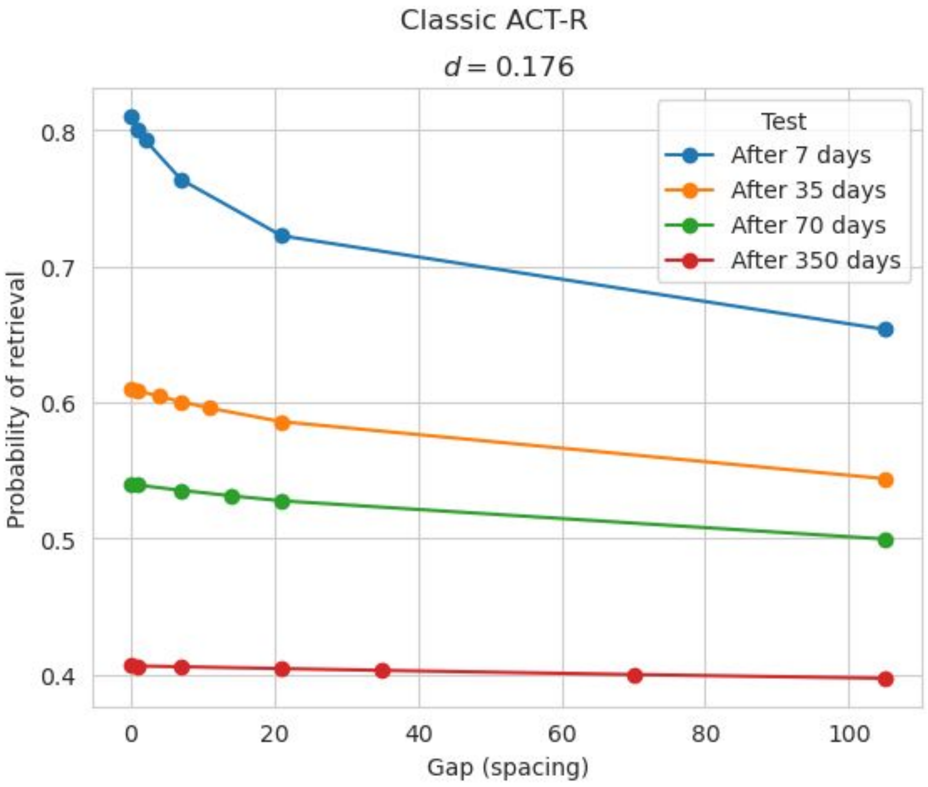
$$\text{Odds} = t^{-d}$$

- Activation of a memory is the log sum of traces

$$A(m) = \log(\ t_1^{-d} + t_1^{-d} + \ldots + t_N^{-d})$$



The model does not produce a spacing effect



Phil Pavlik's extension (Pavlik & Anderson, 2005)

Activation of memory *m* is the log sum of traces decaying with rate *d*

$$A(m) = \log(\ t_1^{-d} + t_1^{-d} + \ldots + t_N^{-d}\ )$$

Phil Pavlik's idea: traces decay at **different rates**

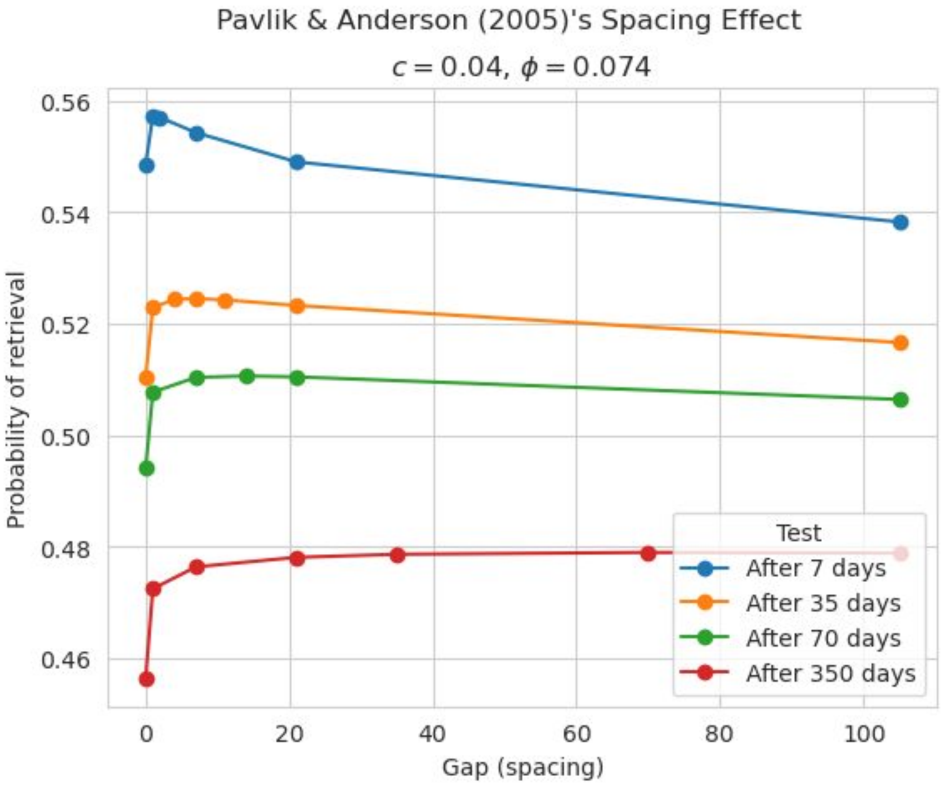
$$A(m) = \log(\ t_1^{-d(1)} + t_1^{-d(2)} + \ldots + t_N^{-d(N)}\ )$$

The *i*-th trace's decay rate $d(i)$ depends on the residual activation $A(m)$:

$$d_i = ce^{A(m)} + \varphi$$

This is the model used by Holly; φ is her **Speed of Forgetting**

Pavlik’s model



New Approach

Alternative model

- Came from discussions with Christian Lebiere
- Dissatisfied with all existing models of the spacing effect
  - Pavlik
  - Mozer and O'Reilly
  - Cepeda himself
  - Walsch's Predictive Performance Equations
- These models are purely **descriptive**; the do not explain **why** the spacing effect should happen

Alternative model

Pavlik's idea: traces decay at different rates

$$A(m) = \log(\ t_1^{-d(1)} + t_2^{-d(2)} + \ldots + t_N^{-d(N)}\ )$$

Alternative: Different traces are **weighted different**

$$A(m) = \log(\ w_1 t_1^{-d} + w_2 t_2^{-d} + \ldots + w_N t_N^{-d}\ )$$

But how is w computed?

A free-energy interpretation of trace weight

- **Free energy principle**: The brain maintains homeostasis by minimizing the **surprisal of new stimuli:** –log *P*(stimulus)
  - “Free” as in free speech, not as in free beer
- In the case if memory, the **surprisal** of each trace should be the degree of (un)predictability of the new trace:

$$w_{trace} = \text{surprisal of } m = -\log P(m)$$

- You can also think of it as **predictive coding**: The brain is trying to maximize successful predictions of the next events
- Also an optimal encoding: How many resources should you invest in the new trace?

Calculate the ideal weight

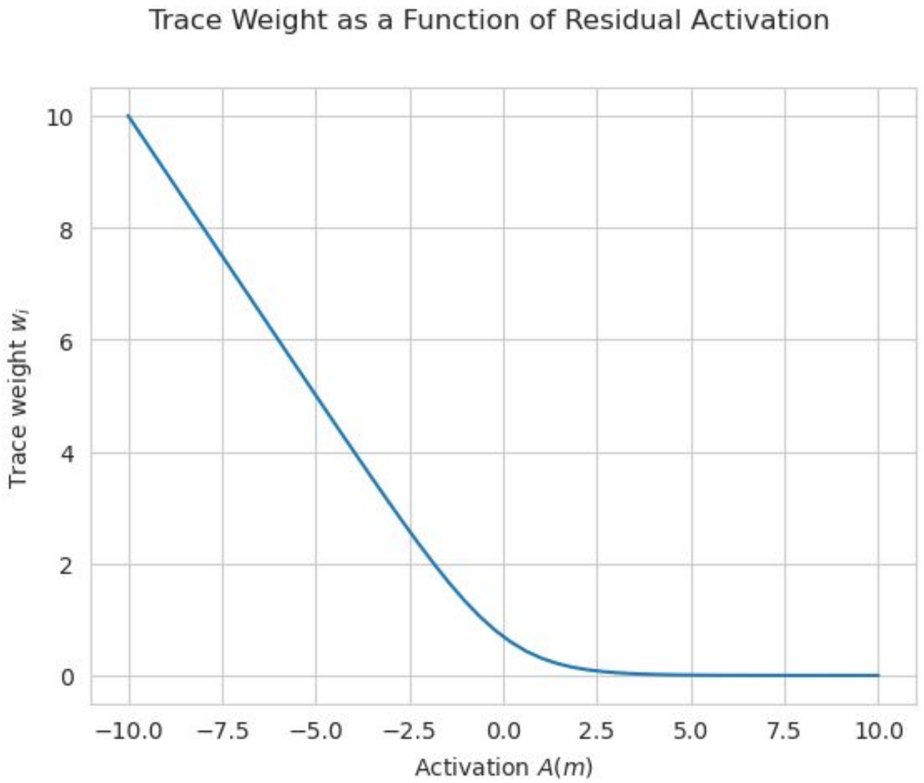
Activation $A(m)$ is the log odds of m

Odds of $m = e^{A(m)}$

Probability is odds / (1 + odds). So:

$$
\begin{aligned}
w \quad &= -\log P(m) = -\log [e^{A(m)} / (1 + e^{A(m)})] \\
&= -\log [1 / (1 + e^{-A(m)})] \\
&= \log(1 + e^{-A(m)})
\end{aligned}
$$

Trace weight *w* is softplus function of (neg) activation



Weight model is simpler than Pavlik & Anderson

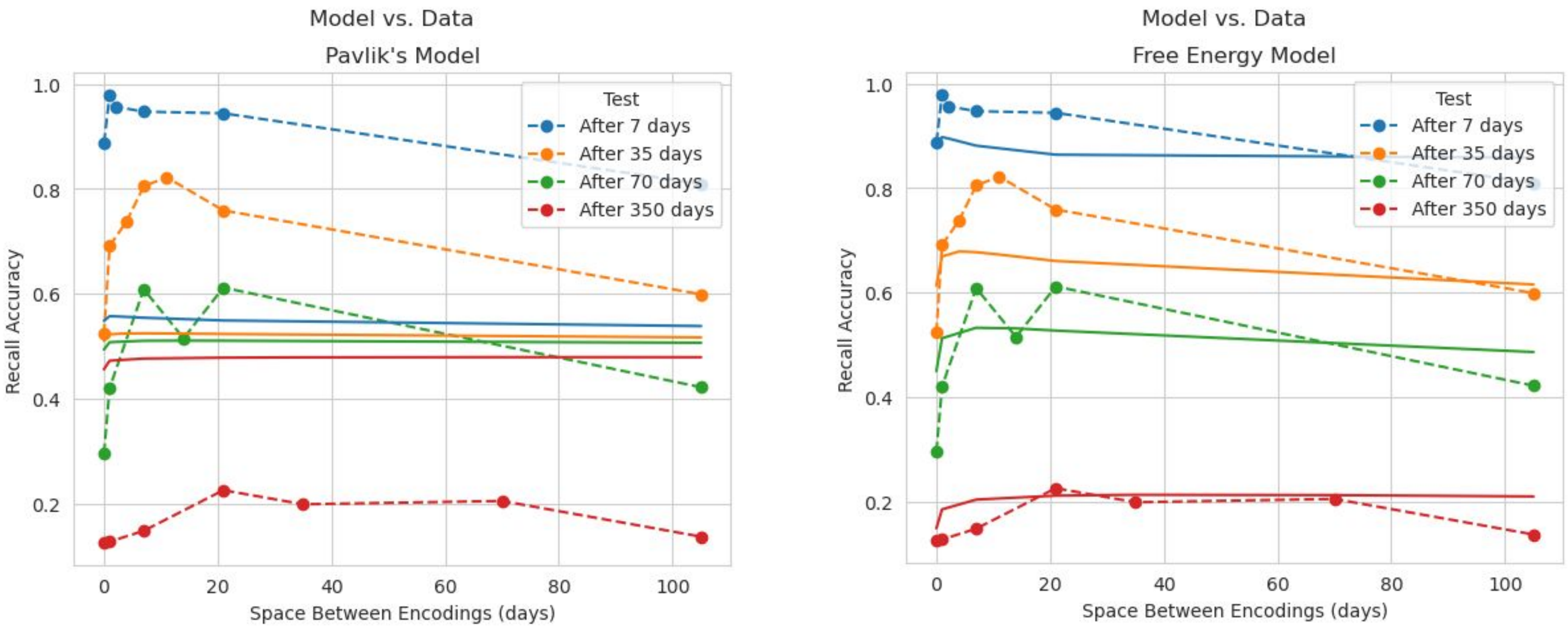
- Pavlik and Anderson's model has **two** free parameters, c and φ
- The weighted trace model has **one** free parameter, d
- Technically, the initial weight of the first trace, w_1, is also a parameter – but…
  - It represents the surprisal of the first time we encounter a new fact
  - It is not entirely free, and reasonable estimates can be made for its value. For example, an estimate of how likely we are going to see something new, given the context.

Testing the model

Test #1: Fit

- Compared Pavlik's model against the Free Energy model
- Found the parameters of the model that best fit Cepeda's dataset
- Used Bayesian Adaptive Directed Search (BADS) algorithm.

Pavlik & Anderson vs. Weighted Traces



Test #2: Flexibility

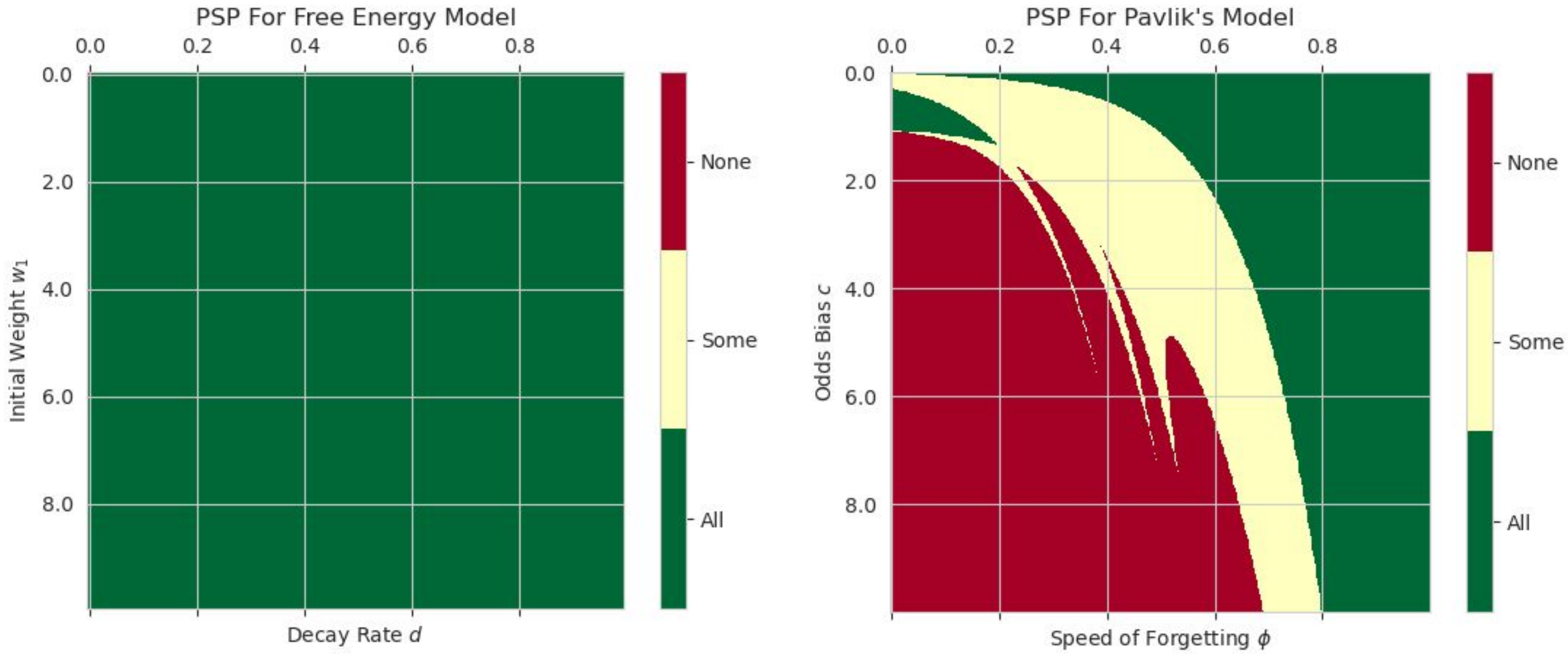
- Ideally, an effect should be a **natural consequence** of the model
    - The main effect should show up no matter what the parameter values are
- **Parameter Space Partitioning**: Explore how many possible qualitatively different patterns are generated by the model.
    - Myung & Pitt, 2006
- Models that generate **fewer** patterns are better

Testing the model

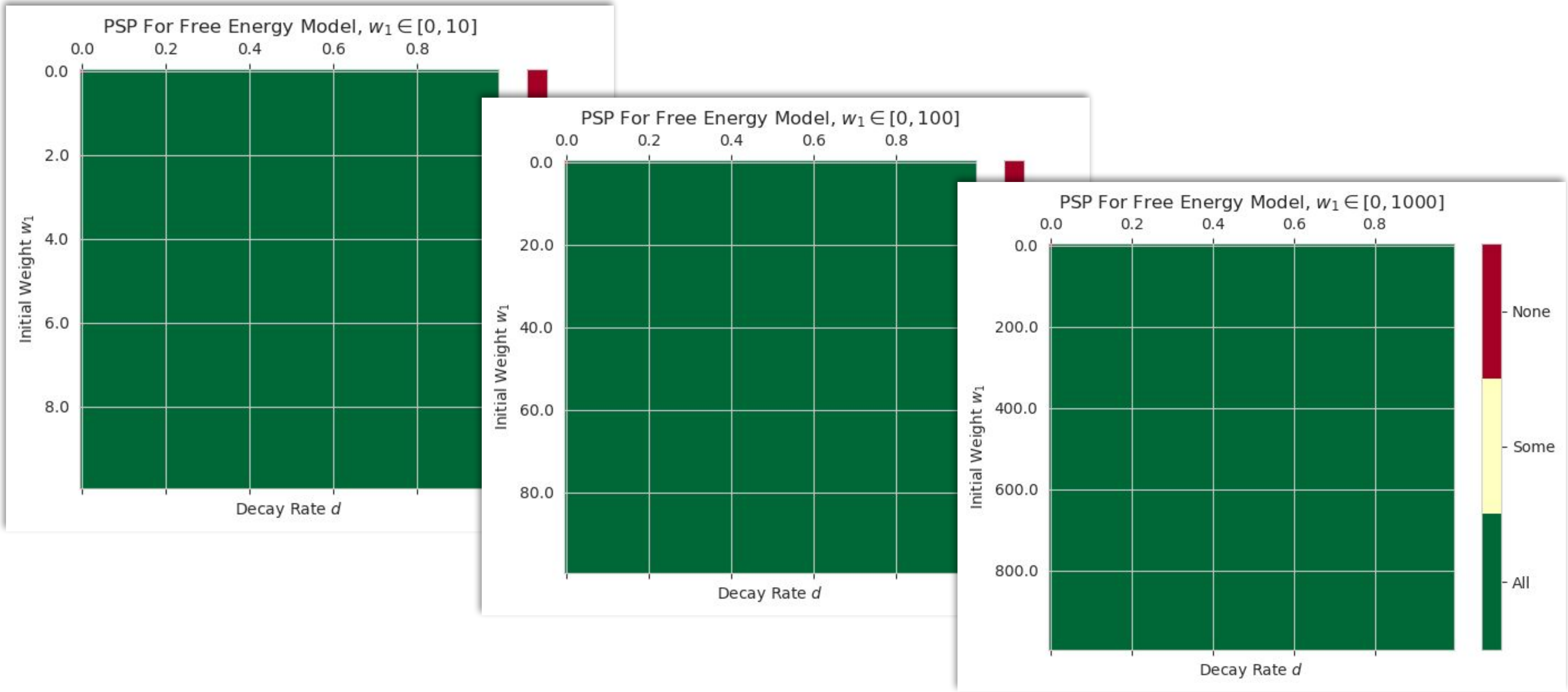
- Free Energy: two parameters w_1 and d.
- Pavlik model: two parameters, c and φ
- Examined similar value ranges:
  - $w_1 = c = [0, 10]$
  - $d = \varphi = [0, 1]$
- Qualitative scale, color-coded as traffic light

2 : **None** of the curves shows spacing effect
1 : **Some** curves show spacing effect
0 : **All** of the curves show spacing effect

Parameter Space Partitioning Results



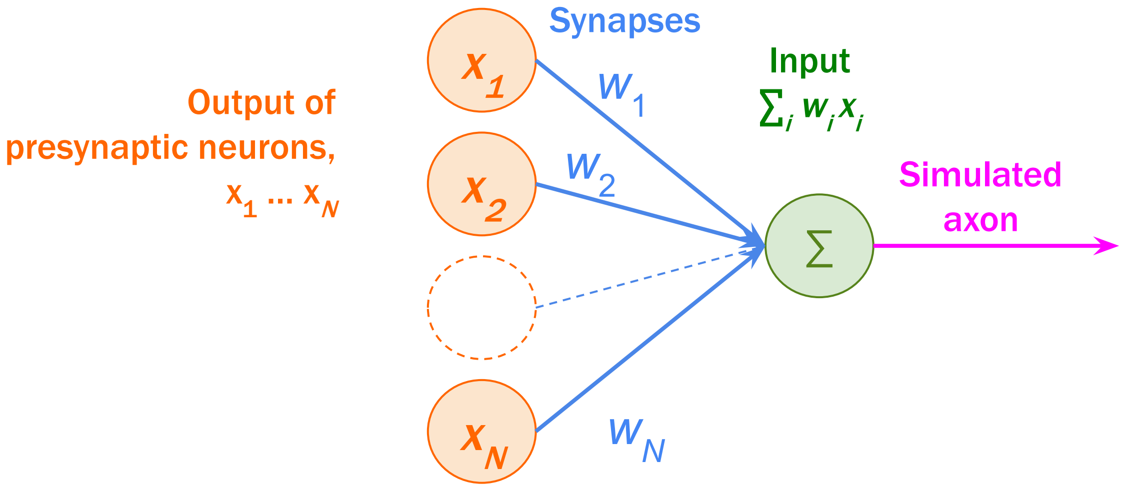
Spacing effect persists across wider ranges of w_1



Neural interpretation

Crash course in neural networks

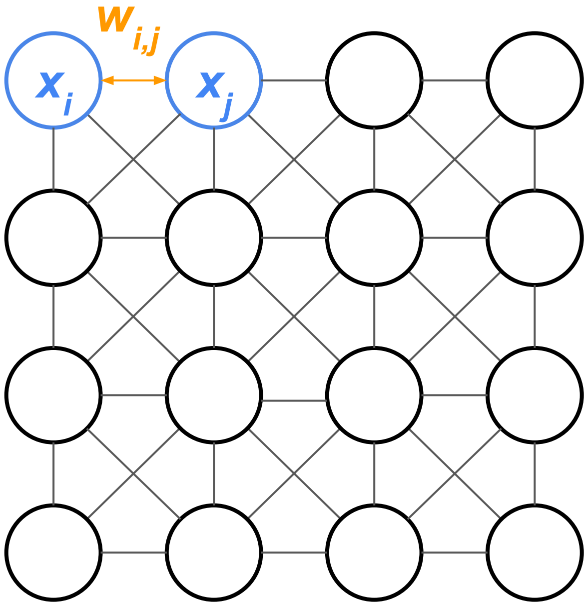
- McCulloch-Pitts neuron
- Input is sum of weighted outputs of presynaptic neurons
- Output is +1 or -1, depending on whether the input is > or < 0.



$$y = \begin{cases} \mathbf{+1}, \text{ if input} > 0 \\ \mathbf{-1}, \text{ if input} < 0 \end{cases}$$

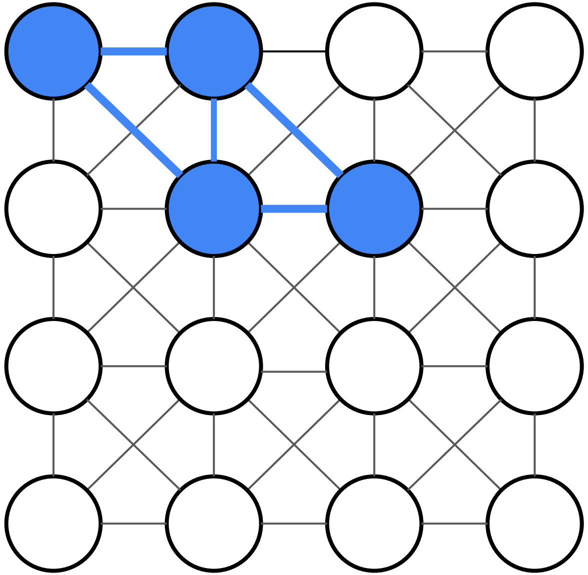
A neural model of memory: The Hopfield network

- Fully interconnected *N* neurons
- Binary activation: $x = \{-1, 1\}$
- Symmetrical synapses: $w_{i,j} = w_{j,i}$
- **Standard model** of hippocampus (Rolls & Treves, 1998; Weber et al., 2017)



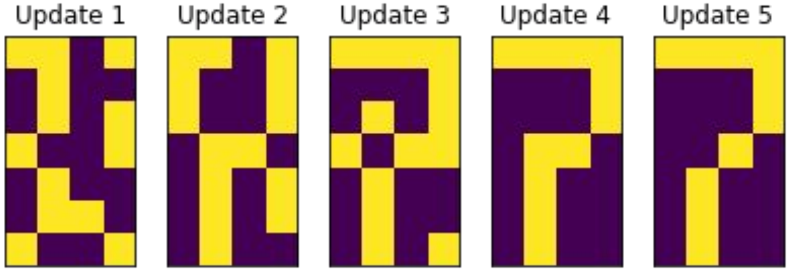
Memory as a Hopfield network

- Hebbian learning: $\Delta w_{i,j} = x_i x_j$
- When both neurons are "on", synapses are strengthened
- Memories are "stored" in the synapses between neurons



Why is it a good model of the hippocampus?

- Hopfield networks **remember** their memories
- Just a few neurons being active triggers the retrieval of the closest memory
- Here is an example of a network that has a memory representing “7” that gets recreated.

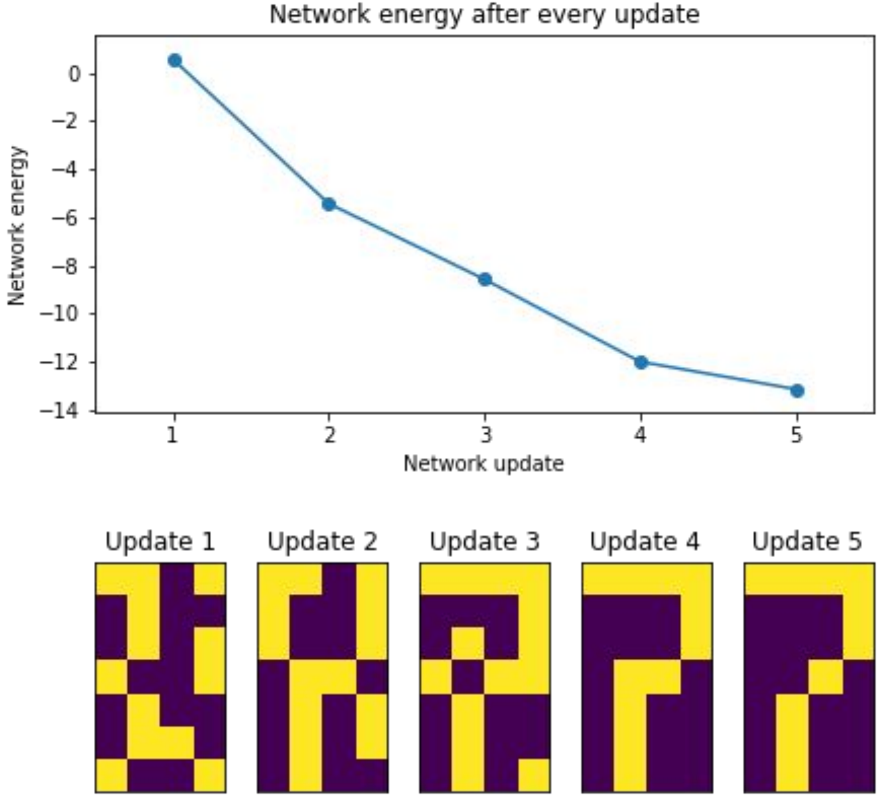


How do Hopfield networks remember?

- The network has an intrinsic "energy" H

$$H(m) = -\sum_i \sum_j w_{i,j}\, x_i\, x_j$$

- The network moves to states with lower energy
- **Memories are the states with the lowest energy**



What is the network energy?

- The probability that a network will remember a memory is inversely proportional to its energy

$$P(m) = 1 / (1 + e^{H(m)})$$

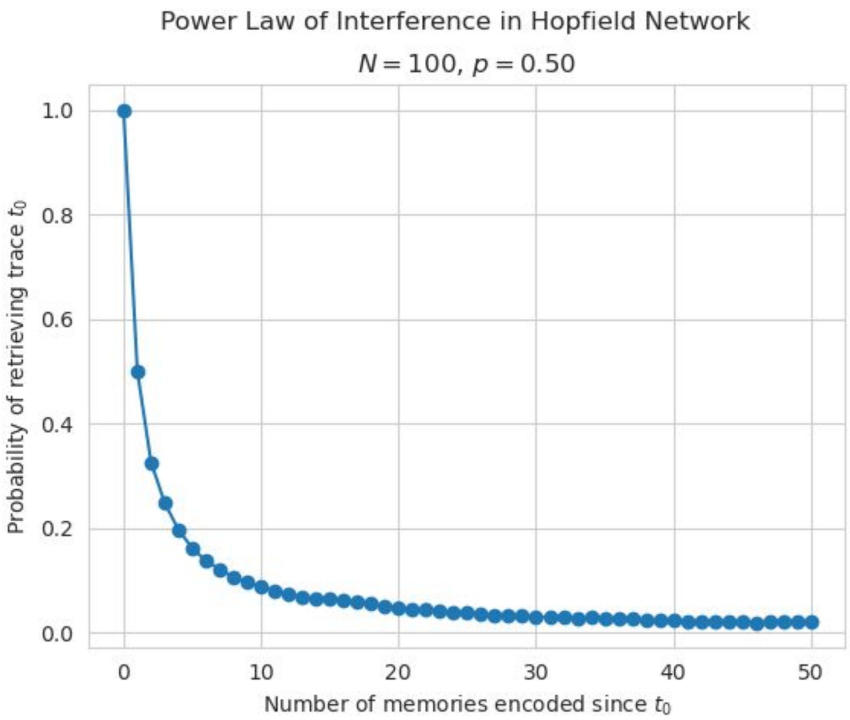
- Analogous to ACT-R, where

$$P(m)\ 1 / (1 + e^{-A(m)})$$

- So, Hopfield energy $H(m) \approx$ ACT-R Activation $A(m)$

Interference in Hopfield ~ Decay in ACT-R

- Model with $N = 100$ neurons, each with p prob. of being “on”
- As new memories are learned, older memories **increase** their energy
- This is **interference** from synapses changing values
- Interference decays with power function

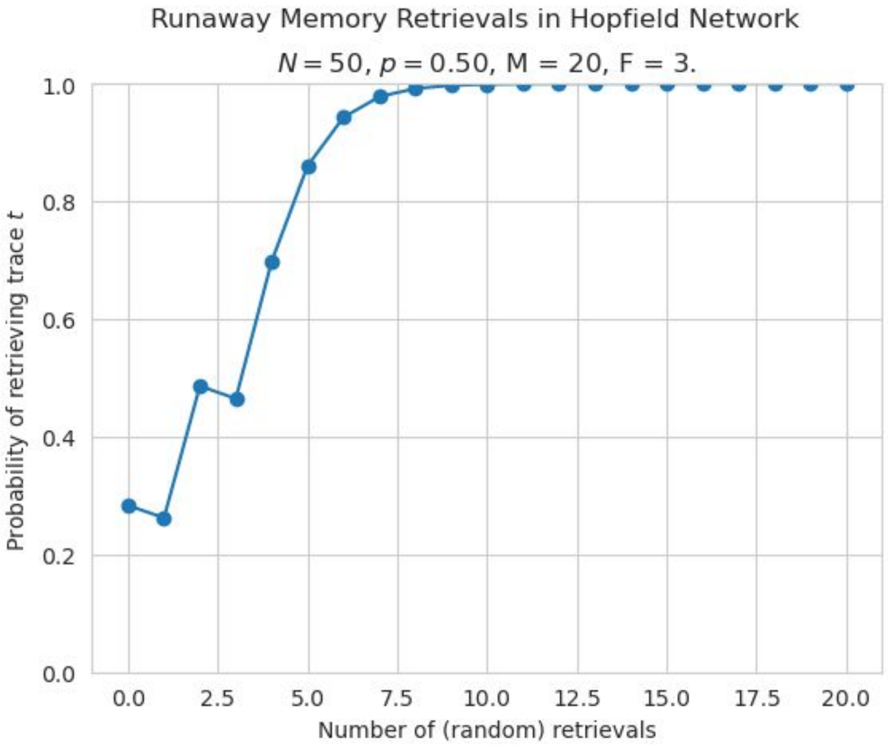


Traces in Hopfield networks

- Traces are identical “patterns” that are learned.
- Multiple traces increase synaptic weights and lowers the memories’ energy
- Memories with multiple traces are more likely to be **remembered**

Runaway energy: Retrievals makes memories unstable

- If every retrieval triggers Hebbian learning, weights for the most active memories grow unbounded
  - Two-term rule, $\Delta w_{i,j} = x_i\, x_j /\ N$
- Retrieval probability goes into positive feedback loop
- Shown: a network with *M* traces, one of which has already been retrieved *F* times



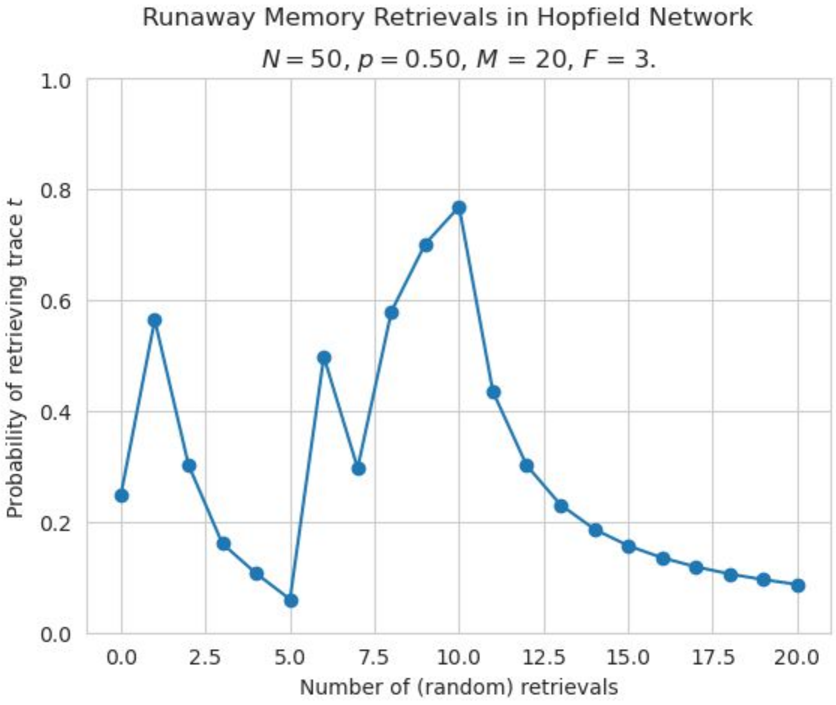
Scaling synapses by surprisal

- Runaway energy can be controlled by **scaling synaptic updates** in proportion to their **surprisal** $-\log P(m)$
    - Equivalent to the free energy model
- Synaptic weights are adjusted based on the three term rule:
    - $\Delta w_{i,j} = -\log P(m)\ x_i\, x_j$
- This minimizes changes to the network
- … Now you see where “**free energy**” comes from!!

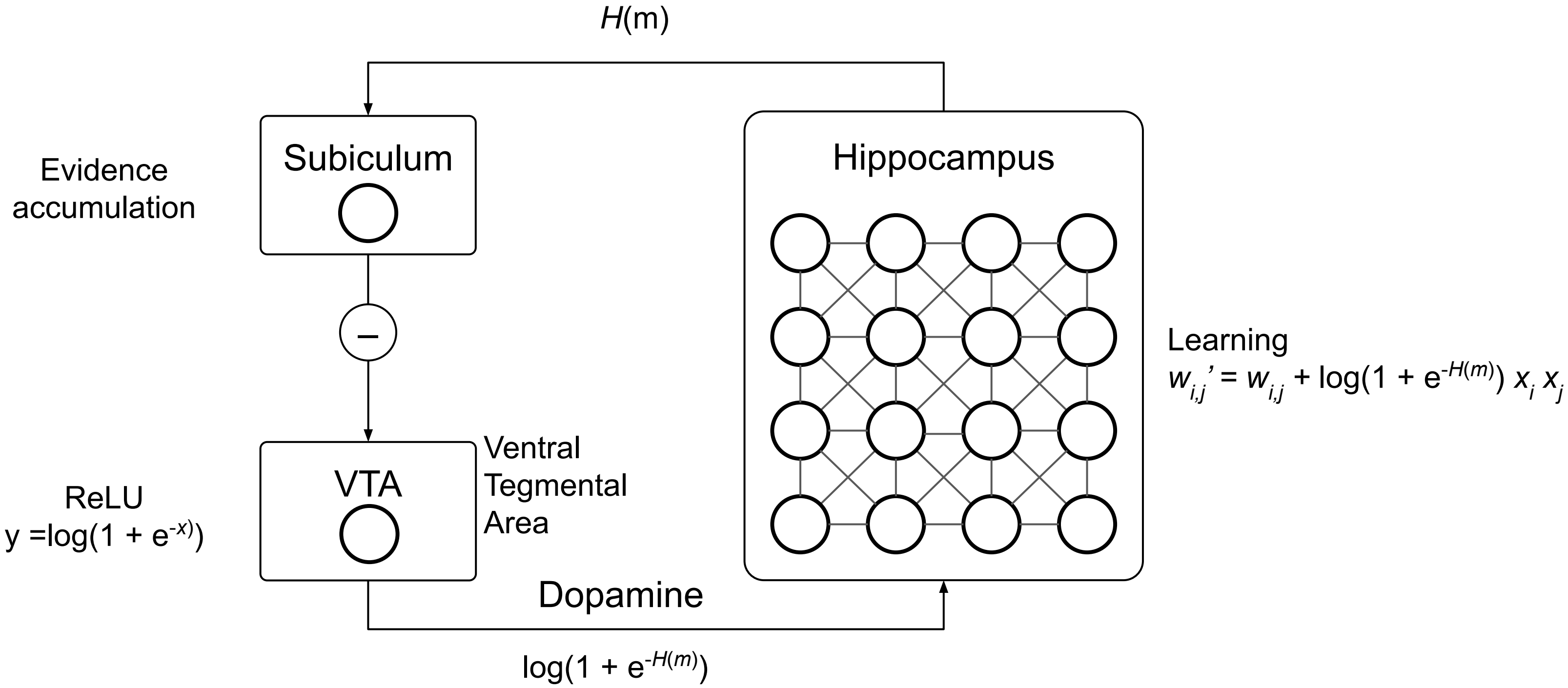
Free-energy scaling counters runaway effect

$$\Delta w_{i,j} = -\log P(m)\ x_i\ x_j$$

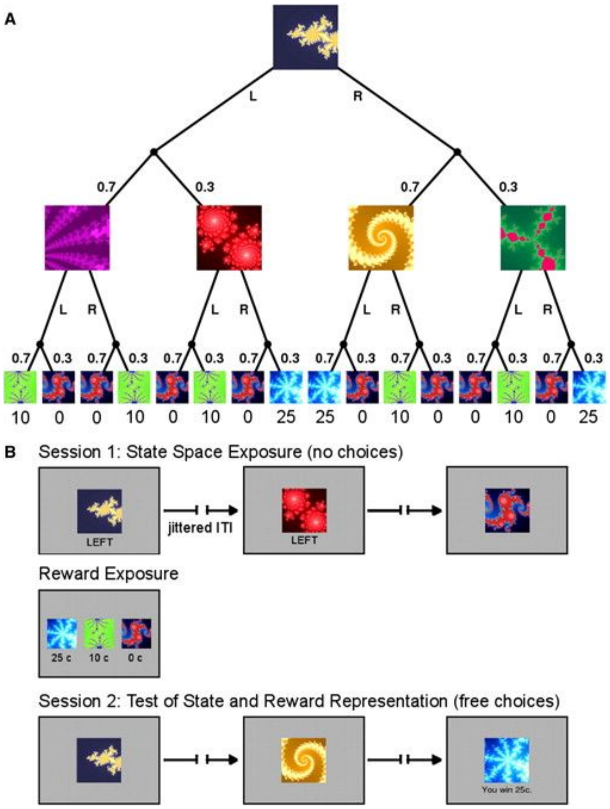
- The three-term rule is typically understood as the **neuromodulatory effect of dopamine** on synaptic plasticity
- Free energy makes memories **stable**

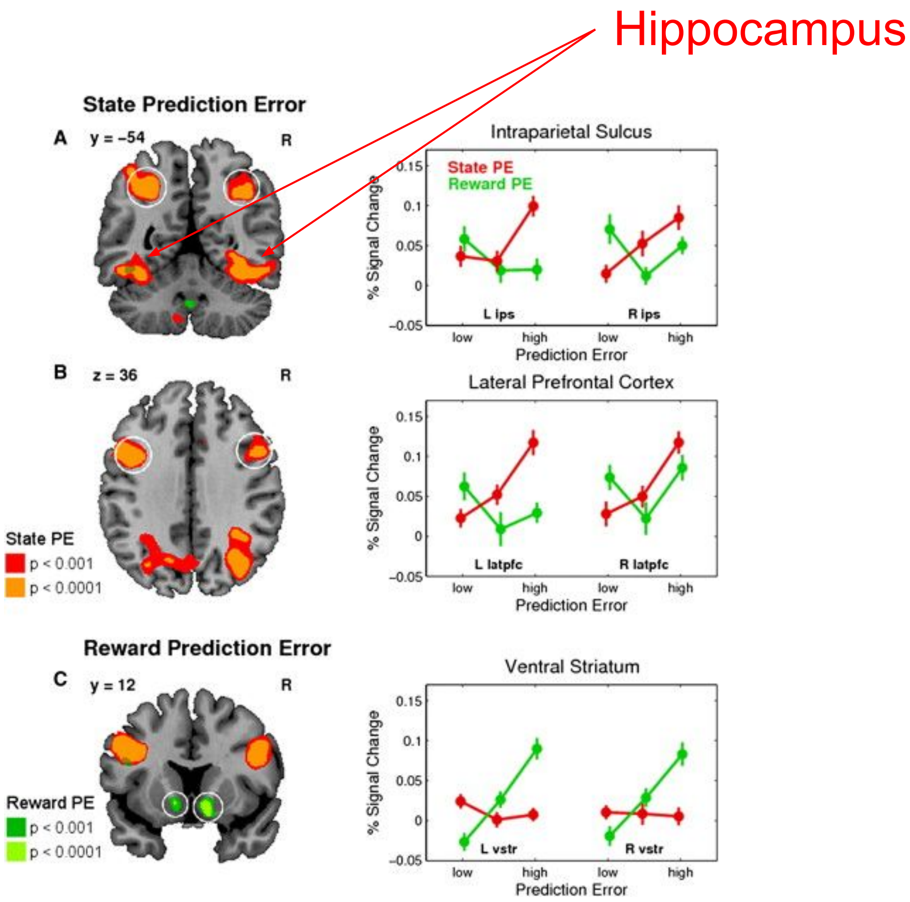


A possible neural Implementation



Evidence from fMRI





Take home messages

- Spacing effect is a consequence of minimizing free energy
- Avoids runaway effects in memory and makes the hippocampus stable
- Once traces are scaled by surprisal, spacing effect is **unavoidable**
- And also… ACT-R ~ Hopfield model of hippocampus!

My favorite papers in memory research

6. Cepeda et al., 2008: **Spacing effects in learning.**
5. Brewer & Treyens, 1981: **Role of schemata in memory for places.**
4. Milner & Scoville, 1957: **Loss of recent memory after hippocampal lesions.**
3. Craik & Lockhart, 1972: **Levels of processing: A framework.**
2. Anderson & Schooler, 1991: **Reflections of the environment in memory.**
1. Loftus, 1978: **On the interpretation of interactions[*].**

* Also in the “Top worst paper titles”