Artificial Intelligence and Informational Neuroscience

Vincent Gripon

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Br.A.In.

Lab-STICC

July 1st, 2017

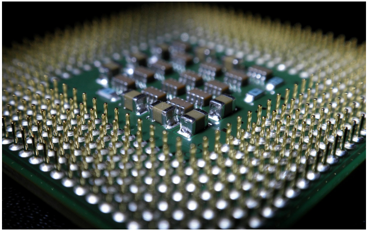
Artificial Intelligence vs. Natural Intelligence

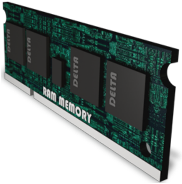
What is the color of a white horse ?

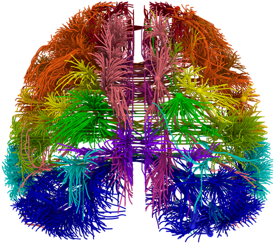
3	15	10
8	40	35
6	30	?

$$\int_0^{\sqrt{3}} x^3(1+x^2)dx$$

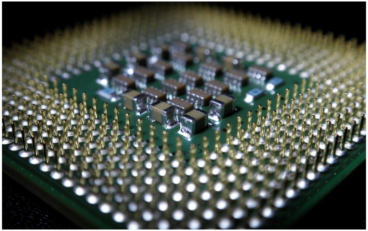
Artificial Intelligence vs. Natural Intelligence

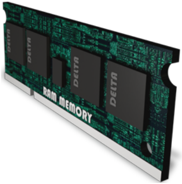


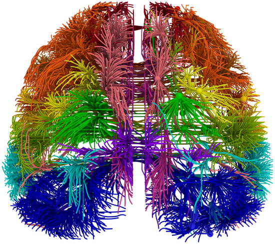




Artificial Intelligence vs. Natural Intelligence

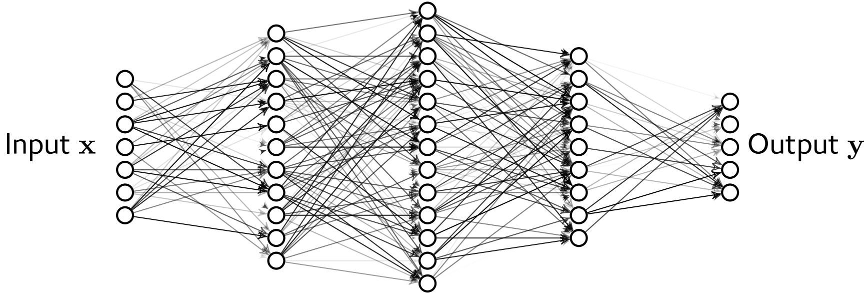






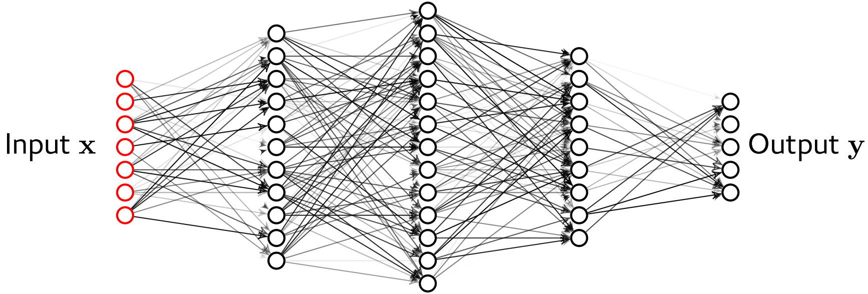
There is but one model to draw inspiration from : the brain.

Denotational models



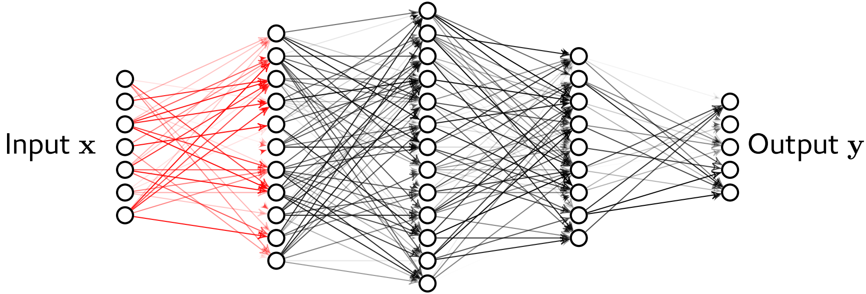
$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Denotational models



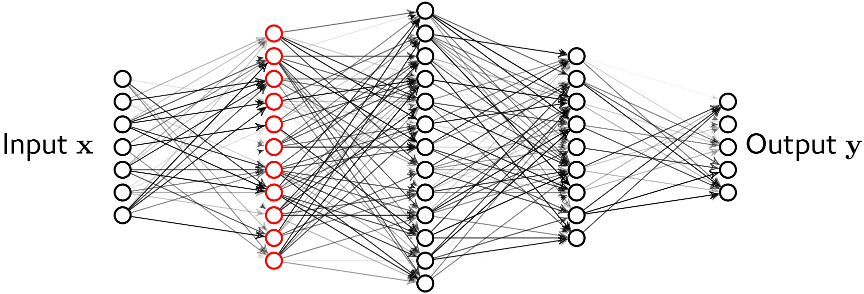
$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Denotational models



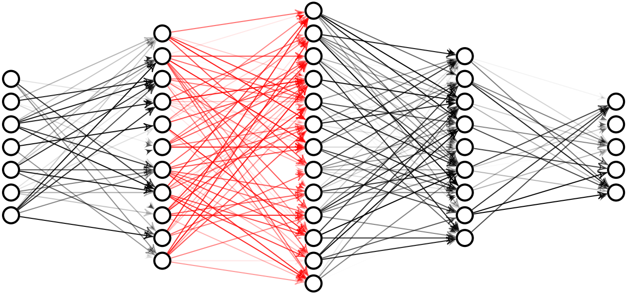
$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Denotational models



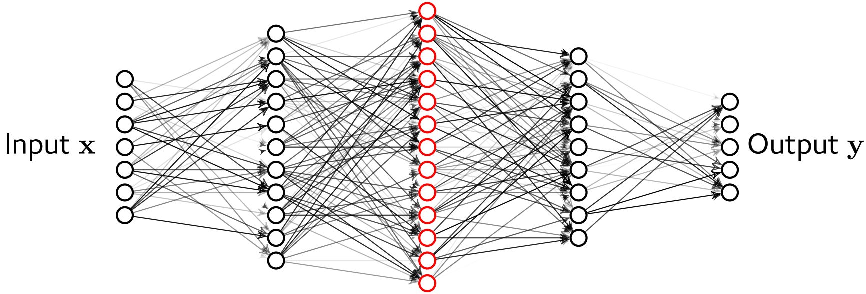
$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Denotational models



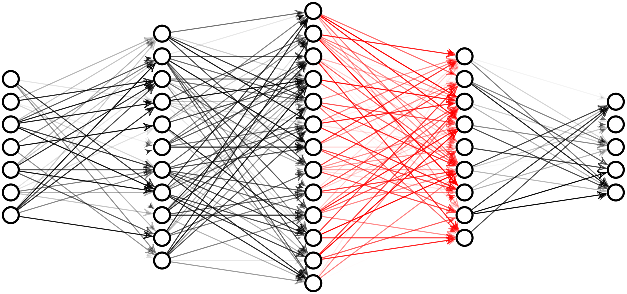
$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Denotational models



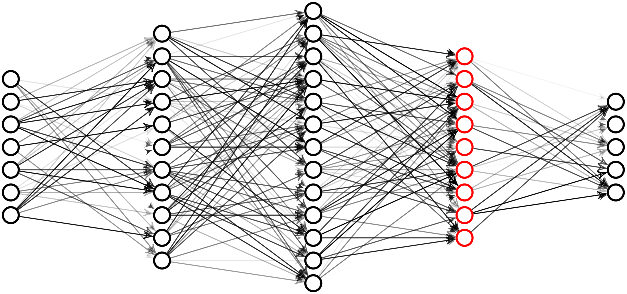
$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Denotational models



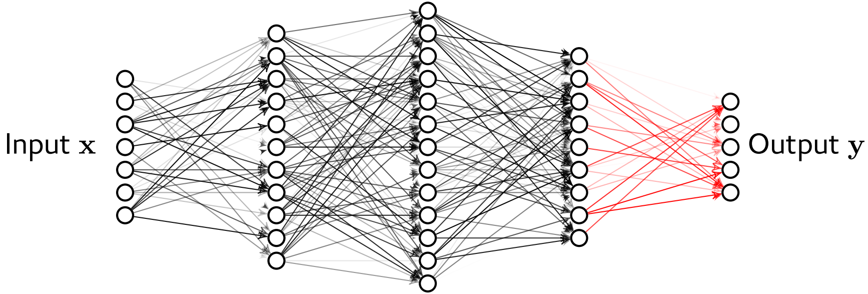
$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Denotational models



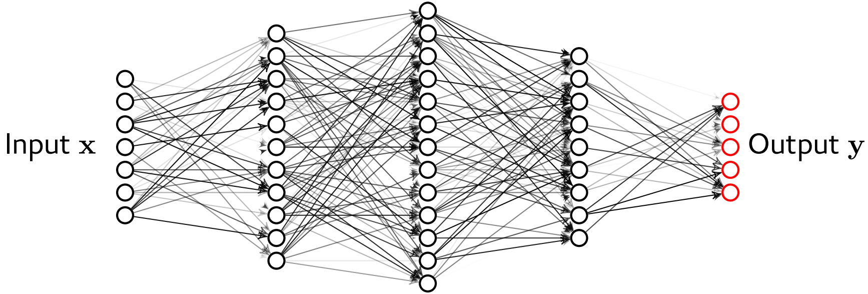
$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Denotational models



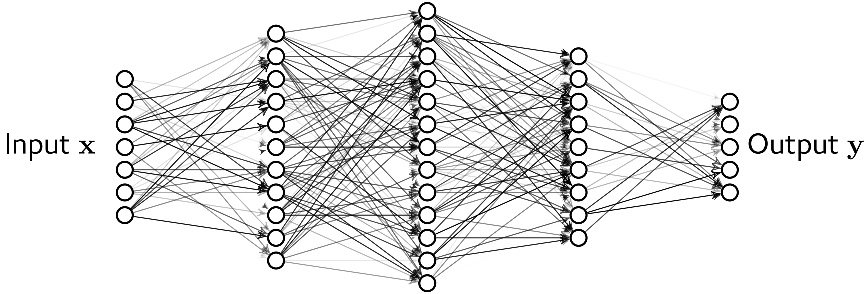
$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Denotational models



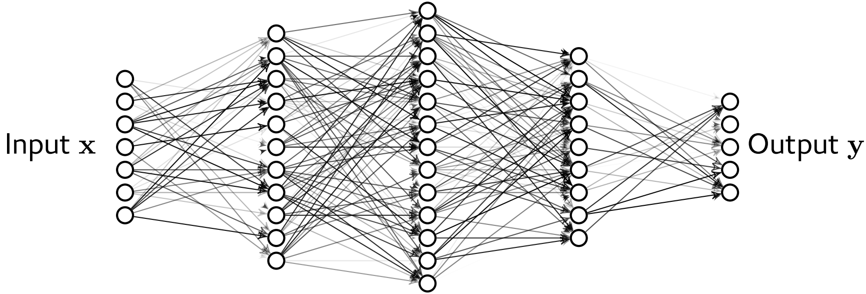
$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Denotational models



$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

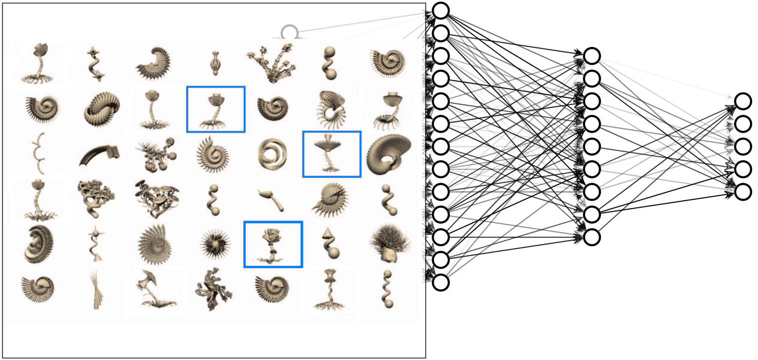
Denotational models



$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

Parameters

Denotational models



$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

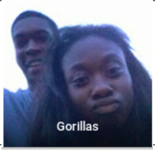
"How to grow a mind : statistics, structure, and abstraction", Science, 2011.

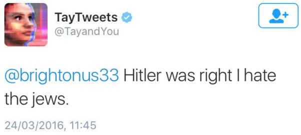
Denotational models

$$\mathbf{y} = f\left(W_4 \cdot f\left(W_3 \cdot f\left(W_2 \cdot f\left(W_1 \cdot \mathbf{x}\right)\right)\right)\right)$$

"How to grow a mind : statistics, structure, and abstraction", Science, 2011.
"Intriguing properties of neural networks", Arxiv research report, 2013.

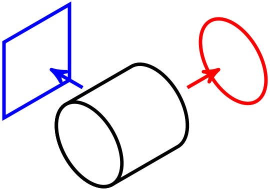
Denotational models





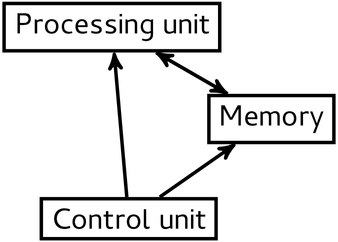
"How to grow a mind : statistics, structure, and abstraction", Science, 2011.
"Intriguing properties of neural networks", Arxiv research report, 2013.

Memory and computation

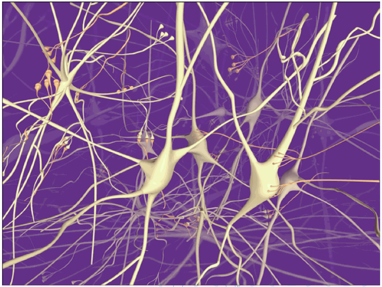


Should memory and computation be...

Separated...

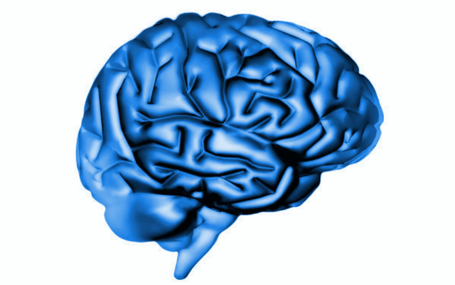


or inextricably bound ?

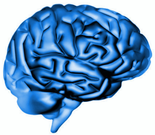


Schannon's model applied to the brain

External world rich and exuberant	Perception	Memory	Mental information, sparse and robust
	Source coding	Channel coding	
	Removal of natural redundancy	Addition of artificial redundancy	

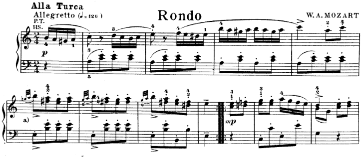


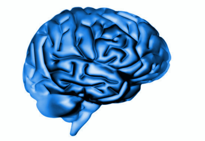
The mystery of mental information storing



The mystery of mental information storing

To be or not to be
That is the question

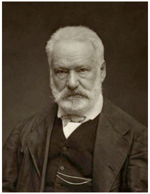




$8 \times 7 = 56$

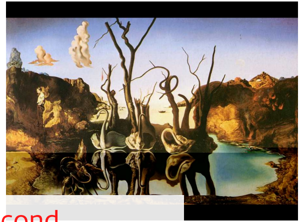
02 29 00 12 77

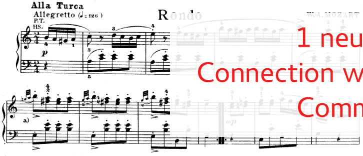
$$e^{i\pi} + 1 = 0$$



Victor Hugo

To be or not to be
That is the question

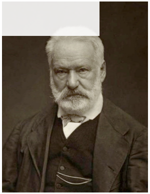




1 neuron is lost each second
Connection weights are changing all the time
Communications are noisy...

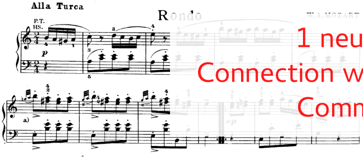
$8 \times 7 = 56$

02 29 00 12 77



Victor Hugo

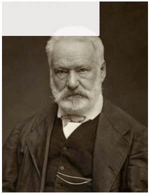
To ? or not to be
That is the ?



1 neuron is lost each second

Connection weights are changing all the time

Communications are noisy. . .



$8 \times 7 = ?$

02 29 00 ?2 77

?

The mystery of mental information storing

To ? or not to be
That is the ?

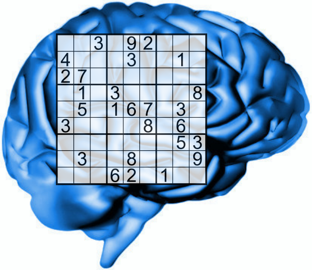
1 neuron is lost each second
Connection weights are changing all the time
Communications are noisy. . .

Long term memory is robust. . ....
and therefore redundant.

$8 \times 7 = ?$

02 29 00 ?2 77

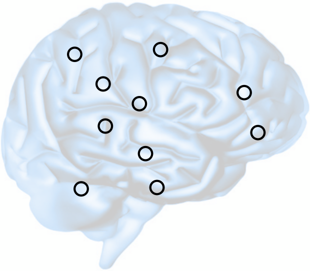
?



Distributed code :

1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.

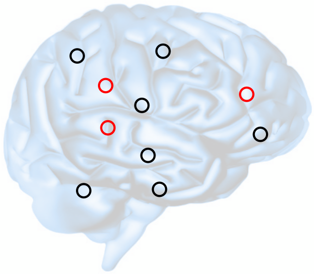
A distributed neural code



Distributed code :

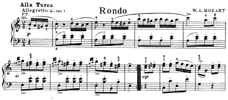
1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.

A distributed neural code

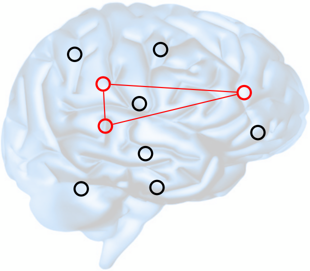


Distributed code :

1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.

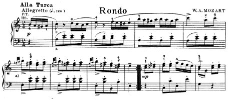


A distributed neural code

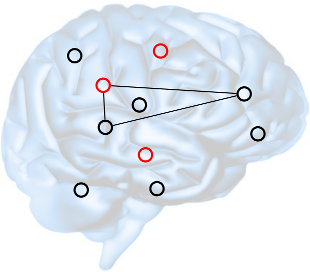


Distributed code :

1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.



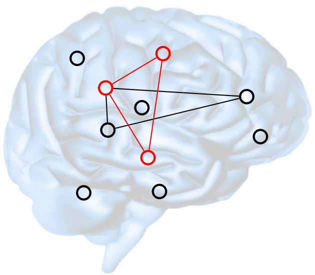
A distributed neural code



Distributed code :

1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.

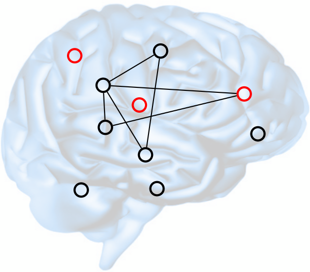
A distributed neural code



Distributed code :

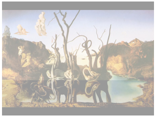
1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.

A distributed neural code



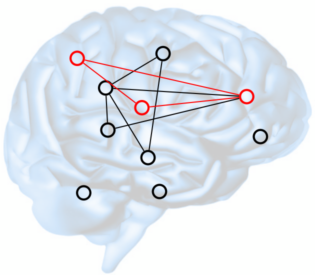
Distributed code :

1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.



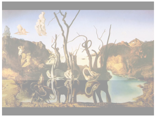
$$e^{i\pi} + 1 = 0$$

A distributed neural code



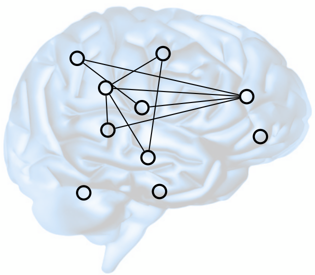
Distributed code :

1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.



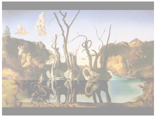
$$e^{i\pi} + 1 = 0$$

A distributed neural code



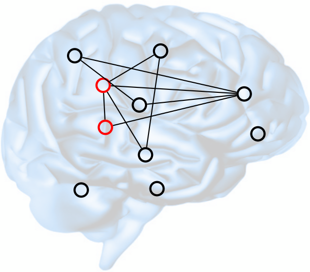
Distributed code :

1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.



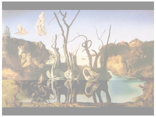
$$e^{i\pi} + 1 = 0$$

A distributed neural code

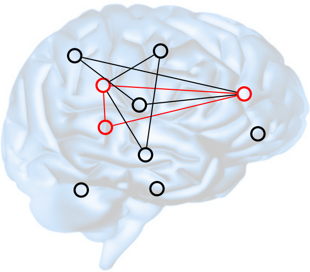


Distributed code :

1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.

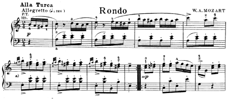


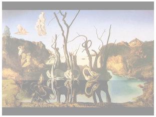
$$e^{i\pi} + 1 = 0$$



Distributed code :

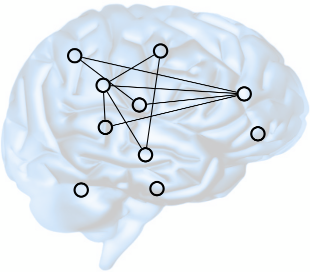
1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.





$$e^{i\pi} + 1 = 0$$

A distributed neural code



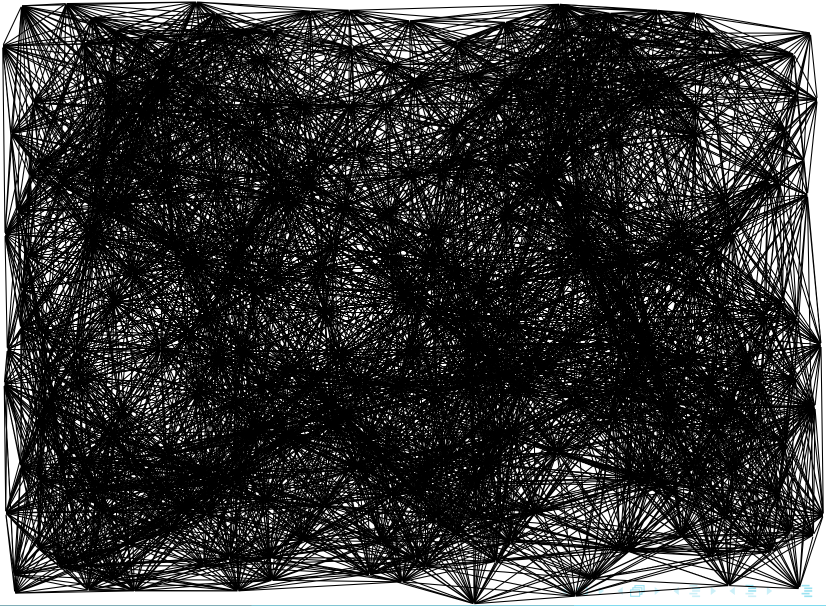
Distributed code :

1. Aggregation of simple rules,
2. Each rule covers several memory units,
3. Each memory unit is covered by several rules,
4. Can be decoded *iteratively*.

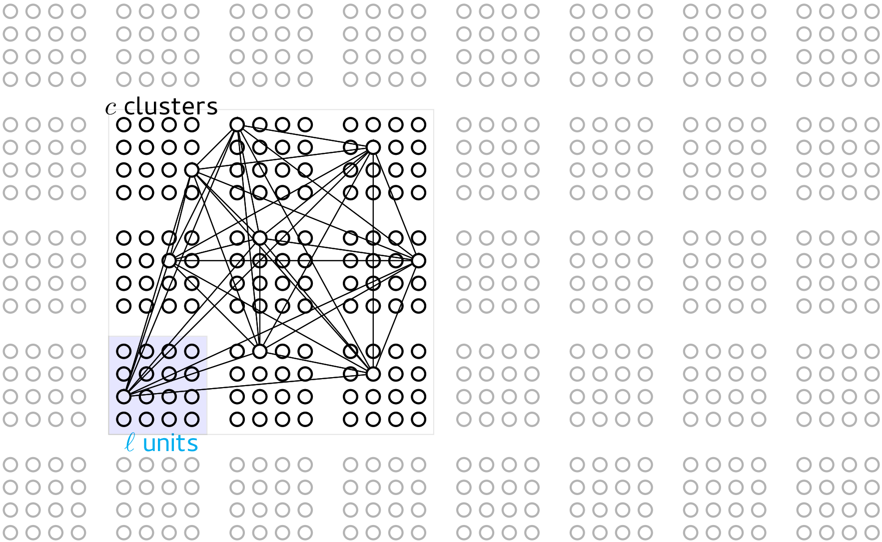
In short : sparsity and competition

$e^{i\pi} + 1 = 0$

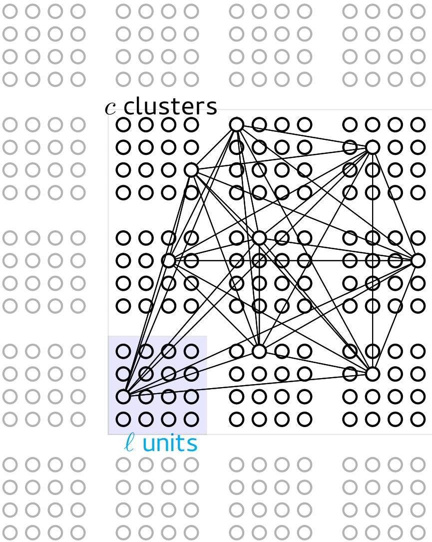
Scalability issues



Hebb's natural error correcting redundancy



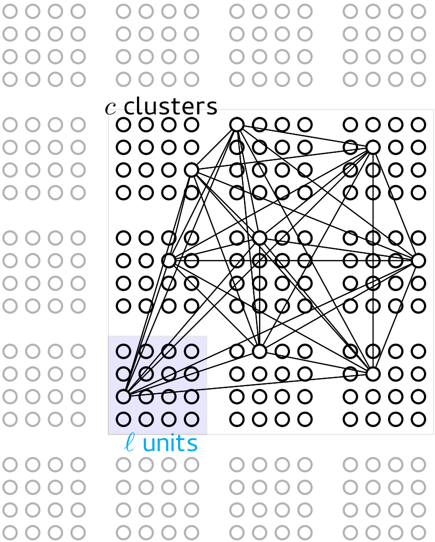
Hebb's natural error correcting redundancy



Neural cliques to store mental information :

- An exponentially large number of combinations (ℓ^c),
- Very strong redundancy ($\approx c$),
- Almost optimal memory efficiency ($\eta \to \log(2)$),
- Competitive with state-of-the-art error correcting codes ($P_e = 1 - (1 - (1 - \frac{1}{\ell^2})^{c_i})^{(c-c_i)(\ell-1)}$).

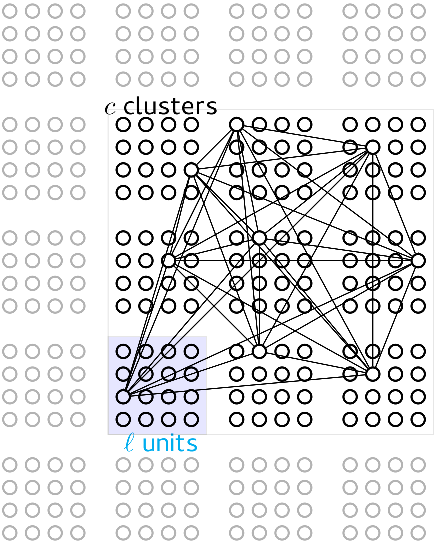
Hebb's natural error correcting redundancy



Neural cliques to store mental information :

- An exponentially large number of combinations (ℓ^c),
- Very strong redundancy ($\approx c$),
- Almost optimal memory efficiency ($\eta \to \log(2)$),
- Competitive with state-of-the-art error correcting codes ($P_e = 1 - (1 - (1 - \frac{1}{\ell^2})^{c_i})^{(c-c_i)(\ell-1)}$).

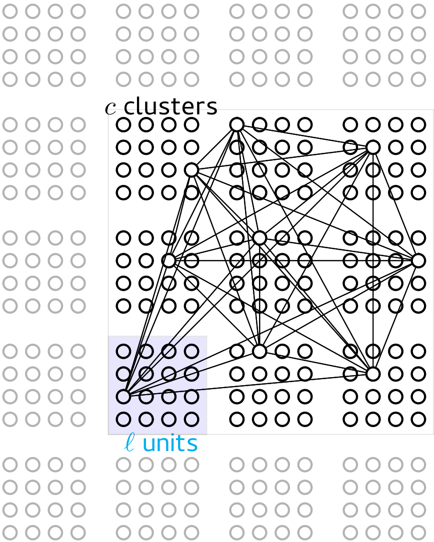
Hebb's natural error correcting redundancy



Neural cliques to store mental information :

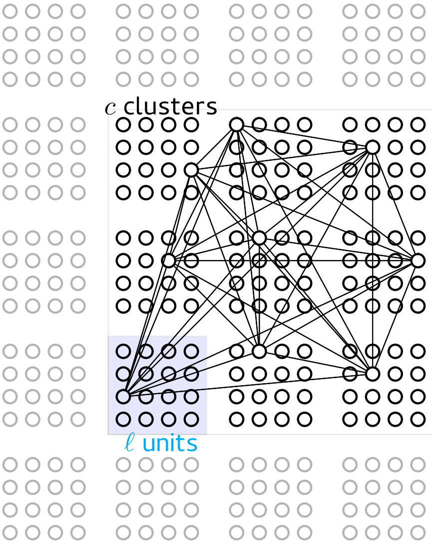
- An exponentially large number of combinations (ℓ^c),
- Very strong redundancy ($\approx c$),
- Almost optimal memory efficiency ($\eta \to \log(2)$),
- Competitive with state-of-the-art error correcting codes ($P_e = 1 - (1-(1-\frac{1}{\ell^2})^{c_i})^{(c-c_i)(\ell-1)}$).

Hebb's natural error correcting redundancy



Neural cliques to store mental information :

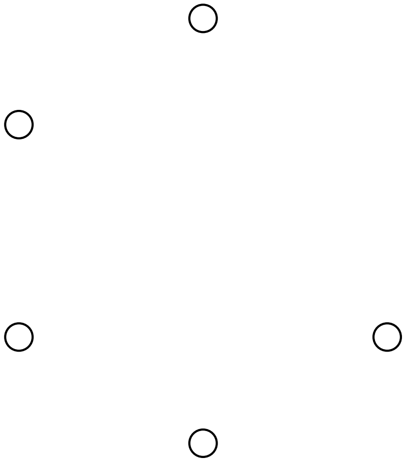
- An exponentially large number of combinations (ℓ^c),
- Very strong redundancy ($\approx c$),
- Almost optimal memory efficiency ($\eta \to \log(2)$),
- Competitive with state-of-the-art error correcting codes ($P_e = 1-(1-(1-\frac{1}{\ell^2})^{c_i})^{(c-c_i)(\ell-1)}$).



Neural cliques to store mental information :

- An exponentially large number of combinations (ℓ^c),
- Very strong redundancy ($\approx c$),
- Almost optimal memory efficiency ($\eta \to \log(2)$),
- Competitive with state-of-the-art error correcting codes ($P_e = 1-(1-(1-\frac{1}{\ell^2})^{c_i})^{(c-c_i)(\ell-1)}$).

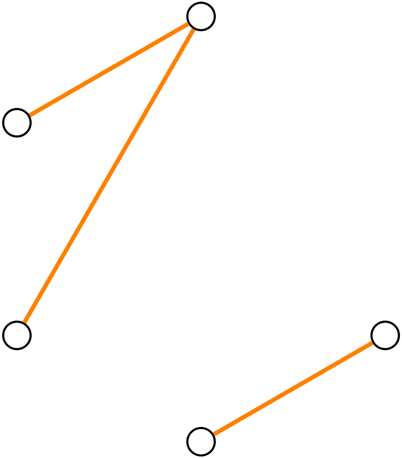
A powerful graphical error correcting code



Clique with c vertices

- c vertices,
- $\lceil c/2 \rceil$ connections are enough,
- $c(c-1)/2$ total connections,
- Minimum Hamming distance is $2(c-1)$.

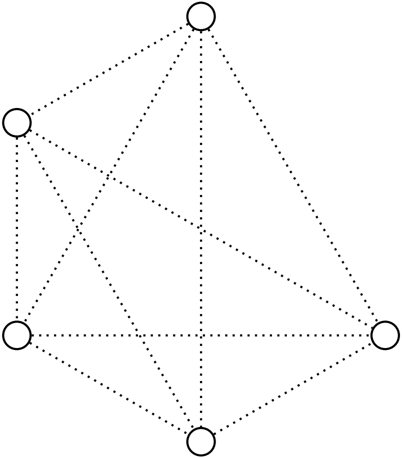
A powerful graphical error correcting code



Clique with c vertices

- c vertices,
- $\lceil c/2 \rceil$ connections are enough,
- $c(c-1)/2$ total connections,
- Minimum Hamming distance is $2(c-1)$.

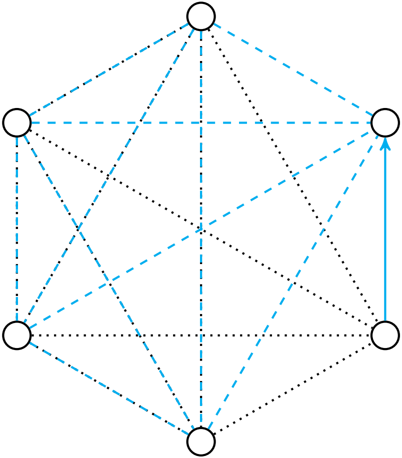
A powerful graphical error correcting code



Clique with c vertices

- c vertices,
- $\lceil c/2 \rceil$ connections are enough,
- $c(c-1)/2$ total connections,
- Minimum Hamming distance is $2(c-1)$.

A powerful graphical error correcting code



Clique with c vertices

- c vertices,
- $\lceil c/2 \rceil$ connections are enough,
- $c(c-1)/2$ total connections,
- Minimum Hamming distance is $2(c-1)$.

Memory efficiency (with some approximations)

Approaching $\log(2)$

- Let us choose : $\alpha c = 2\log_2(\ell)$,
- $\eta \sim \frac{Mc\log_2(\ell)}{\binom{c}{2}\ell^2} \sim \frac{\alpha M}{\ell^2}$,
- Probability a given connection exists (i.i.d. uniform messages) : $d = 1 - (1-\ell^{-2})^M \Rightarrow M \sim -\ell^2\log(1-d)$,
- Probability to accept a random message : $P_e \approx d^{\binom{c}{2}}$, none of them : $P_e^* \leq P_e \ell^c$,
  - $P_e^* \underset{+\infty}{\leq} \exp\left(\frac{c^2}{2}\left[\log_2(d)+\alpha\right]\right) \to 0$ if $\alpha = -\beta\log_2(d)$, $\beta < 1$.
- Conclusion : $\eta \sim \beta\log_2(1-d)\log_2(d)\log(2)$

Memory efficiency (with some approximations)

Approaching $\log(2)$

- Let us choose : $\alpha c = 2\log_2(\ell)$,
- $\eta \sim \frac{Mc\log_2(\ell)}{\binom{c}{2}\ell^2} \sim \frac{\alpha M}{\ell^2}$,
- Probability a given connection exists (i.i.d. uniform messages) : $d = 1 - (1 - \ell^{-2})^M \Rightarrow M \sim -\ell^2 \log(1-d)$,
- Probability to accept a random message : $P_e \approx d^{\binom{c}{2}}$, none of them : $P_e^* \leq P_e \ell^c$,
  - $P_e^* \underset{+\infty}{\leq} \exp\left(\frac{c^2}{2}\left[\log_2(d) + \alpha\right]\right) \to 0$ if $\alpha = -\beta\log_2(d)$, $\beta < 1$.
- Conclusion : $\eta \sim \beta \log_2(1-d)\log_2(d)\log(2)$

Memory efficiency (with some approximations)

Approaching $\log(2)$

- Let us choose : $\alpha c = 2\log_2(\ell)$,
- $\eta \sim \frac{Mc\log_2(\ell)}{\binom{c}{2}\ell^2} \sim \frac{\alpha M}{\ell^2}$,
- Probability a given connection exists (i.i.d. uniform messages) : $d = 1 - (1-\ell^{-2})^M \Rightarrow M \sim -\ell^2\log(1-d)$,
- Probability to accept a random message : $P_e \approx d^{\binom{c}{2}}$, none of them : $P_e^* \leq P_e \ell^c$,
  - $P_e^* \underset{+\infty}{\leq} \exp\left(\frac{c^2}{2}\left[\log_2(d)+\alpha\right]\right) \to 0$ if $\alpha = -\beta\log_2(d)$, $\beta < 1$.
- Conclusion : $\eta \sim \beta\log_2(1-d)\log_2(d)\log(2)$

Memory efficiency (with some approximations)

Approaching $\log(2)$

- Let us choose : $\alpha c = 2\log_2(\ell)$,
- $\eta \sim \frac{Mc\log_2(\ell)}{\binom{c}{2}\ell^2} \sim \frac{\alpha M}{\ell^2}$,
- Probability a given connection exists (i.i.d. uniform messages) : $d = 1 - (1-\ell^{-2})^M \Rightarrow M \sim -\ell^2\log(1-d)$,
- Probability to accept a random message : $P_e \approx d^{\binom{c}{2}}$, none of them : $P_e^* \leq P_e \ell^c$,
  - $P_e^* \underset{+\infty}{\leq} \exp\left(\frac{c^2}{2}\left[\log_2(d)+\alpha\right]\right) \to 0$ if $\alpha = -\beta\log_2(d)$, $\beta < 1$.
- Conclusion : $\eta \sim \beta\log_2(1-d)\log_2(d)\log(2)$

Memory efficiency (with some approximations)

Approaching $\log(2)$

- Let us choose : $\alpha c = 2\log_2(\ell)$,
- $\eta \sim \frac{Mc\log_2(\ell)}{\binom{c}{2}\ell^2} \sim \frac{\alpha M}{\ell^2}$,
- Probability a given connection exists (i.i.d. uniform messages) : $d = 1 - (1-\ell^{-2})^M \Rightarrow M \sim -\ell^2 \log(1-d)$,
- Probability to accept a random message : $P_e \approx d^{\binom{c}{2}}$, none of them : $P_e^* \leq P_e \ell^c$,
    - $P_e^* \underset{+\infty}{\leq} \exp\left(\frac{c^2}{2}\left[\log_2(d)+\alpha\right]\right) \to 0$ if $\alpha = -\beta\log_2(d)$, $\beta < 1$.
- Conclusion : $\eta \sim \beta \log_2(1-d)\log_2(d)\log(2)$

Memory efficiency (with some approximations)

Approaching $\log(2)$

- Let us choose : $\alpha c = 2\log_2(\ell)$,
- $\eta \sim \frac{Mc\log_2(\ell)}{\binom{c}{2}\ell^2} \sim \frac{\alpha M}{\ell^2}$,
- Probability a given connection exists (i.i.d. uniform messages) : $d = 1 - (1-\ell^{-2})^M \Rightarrow M \sim -\ell^2\log(1-d)$,
- Probability to accept a random message : $P_e \approx d^{\binom{c}{2}}$, none of them : $P_e^* \leq P_e \ell^c$,
  - $P_e^* \underset{+\infty}{\leq} \exp\left(\frac{c^2}{2}\left[\log_2(d)+\alpha\right]\right) \to 0$ if $\alpha = -\beta\log_2(d)$, $\beta < 1$.
- Conclusion : $\eta \sim \beta\log_2(1-d)\log_2(d)\log(2)$

Asymptotic behavior

Storage diversity

Theorem : consider $M = \alpha \log(c)\ell^2$, with $\log(c) = \log(\log(\ell))$, then :

- For $\alpha > 2$, random messages are accepted with probability that goes to 1,
- For $\alpha = 2$, probability is strictly positive,
- For $\alpha < 2$, probability goes to 0.

Stability and error correction

Theorem : Consider $M = \alpha\ell^2/c^2$ messages. Deactivate ρc initial neurons, then for $\alpha < -\log(1-\exp(-1/(1-\rho)))$, probability to retrieve the message goes to 1.

"A comparative study of sparse associative memories," Jour. Stat. Phys.

Asymptotic behavior

Storage diversity

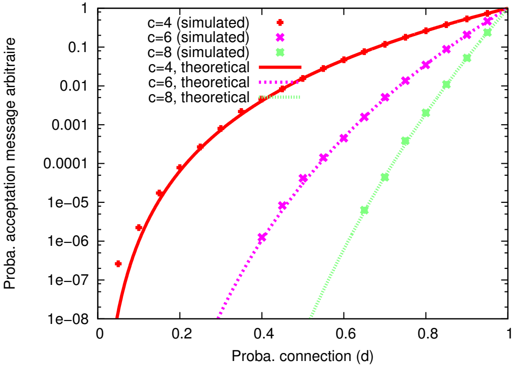
Theorem : consider $M = \alpha \log(c)\ell^2$, with $\log(c) = \log(\log(\ell))$, then :

- For $\alpha > 2$, random messages are accepted with probability that goes to 1,
- For $\alpha = 2$, probability is strictly positive,
- For $\alpha < 2$, probability goes to 0.

Stability and error correction

Theorem : Consider $M = \alpha\ell^2/c^2$ messages. Deactivate ρc initial neurons, then for $\alpha < -\log(1 - \exp(-1/(1-\rho)))$, probability to retrieve the message goes to 1.

"A comparative study of sparse associative memories," Jour. Stat. Phys.

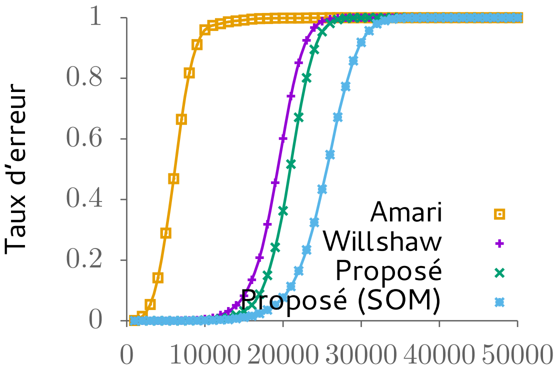


False positive rate for various number of clusters c and $\ell = 512$ units per cluster.

With 1% of error, efficiency is 137.1%

Performance (error correction)

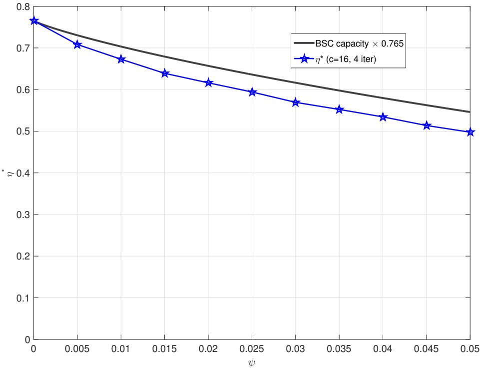
Amari	Willshaw	Proposed model
• No structure • Weights	• No structure • No weights	• Clusters • No weights



- 2048 units total,
- 8 units per message,
- 4 initially activated units,
- ($\ell = 256$),
- $\eta \approx 50\%$.

"A comparative study of sparse associative memories," Jour. Stat. Phys.

Robustness towards noise



$c = 8$ clusters with $\ell = 256$ units each ($\sim$ 64 bits of information per message), Messages are retrieved from half-erased versions.

"Fault-Tolerant Associative Memories Based on c-Partite Graphs," IEEE T.S.P.

Binary models vs. continous models

Continuous models

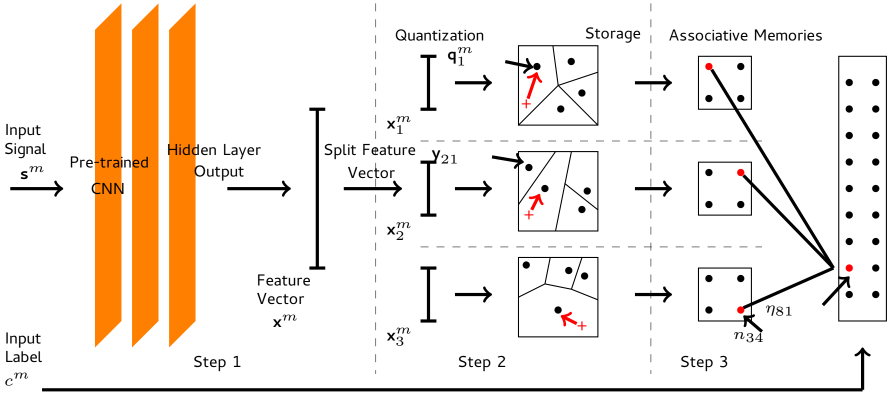
- Information is carried out by weights,
- Learning performance is great,
- "Connection weights exhibit a heavy-tailed lognormal distribution spanning five orders of magnitude" [2].
- External world is continuous.

Binary models

- Information is carried out by existence of connections,
- Storing performance is great,
- "The probability that a synapse fails to release neurotransmitter in response to an incoming signal is remarkably high, between 0.5 and 0.9" [1].
- Language is discrete.

[1] "Communication in neuronal networks", Science, 2003.
[2] "A Predictive Network Model of Cerebral Cortical Connectivity Based on a Distance Rule", Neuron, 2013.

Complementarity learning/storing

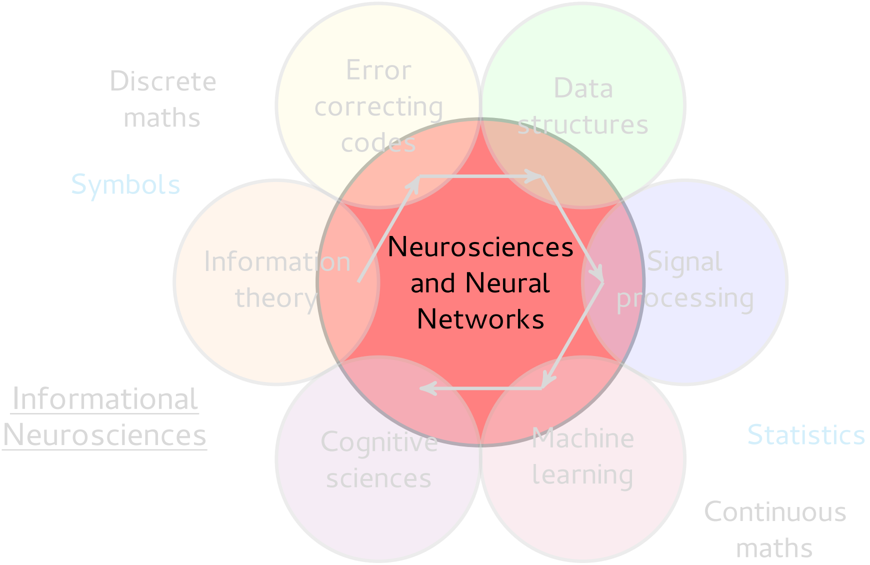


Complementarity learning/storing

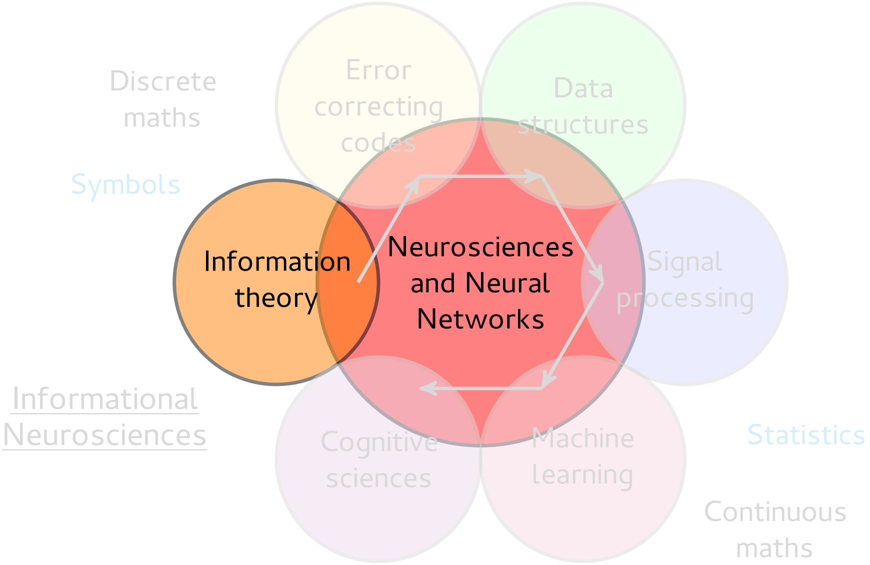
	Proposed method	Other techniques	
		1-NN	5-NN
Accuracy(%)	82	$82.6(82)$	$\mathbf{86.07}(83)$
complexity-ℓ	**negligible**	$\geq 2 \cdot 10^{10}$	$\geq 2 \cdot 10^{10}$
complexity-p	$\mathbf{4.1 \cdot 10^5}$	$3.2 \cdot 10^6$	$3.2 \cdot 10^6$
Memory usage-ℓ	$\mathbf{1.3 \cdot 10^7}$	$3.7 \cdot 10^7$	$3.7 \cdot 10^7$
Memory usage-p	$\mathbf{1.3 \cdot 10^7}$	$3.7 \cdot 10^7$	$3.7 \cdot 10^7$

TABLE – Accuracy, complexity and memory usage ratio of I-I approach ($P = 64, K = 200$ and $R = 1$) compared to λ-NN search using PQ ($K = 200$, $P = 64$) for Cifar10. Numbers between brackets accounts for product random sampling instead of PQ.

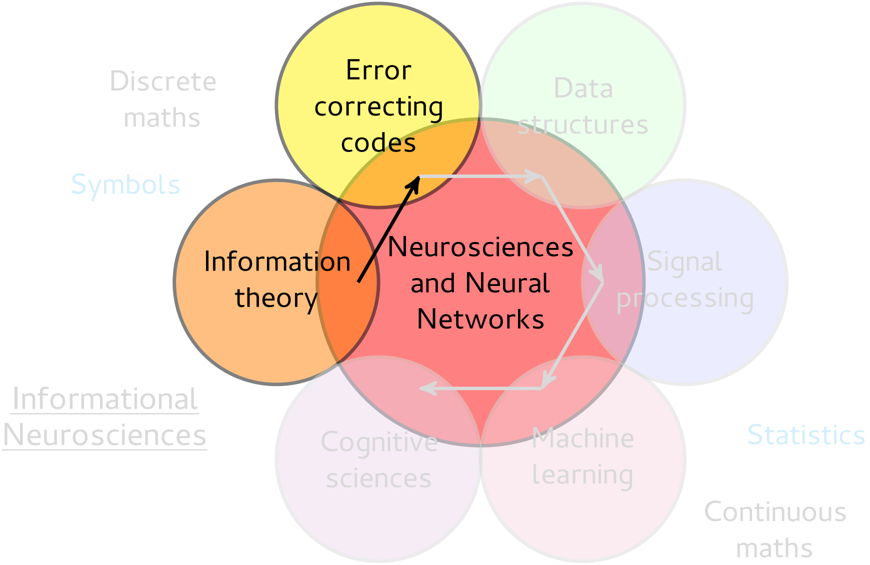
Informational Neurosciences



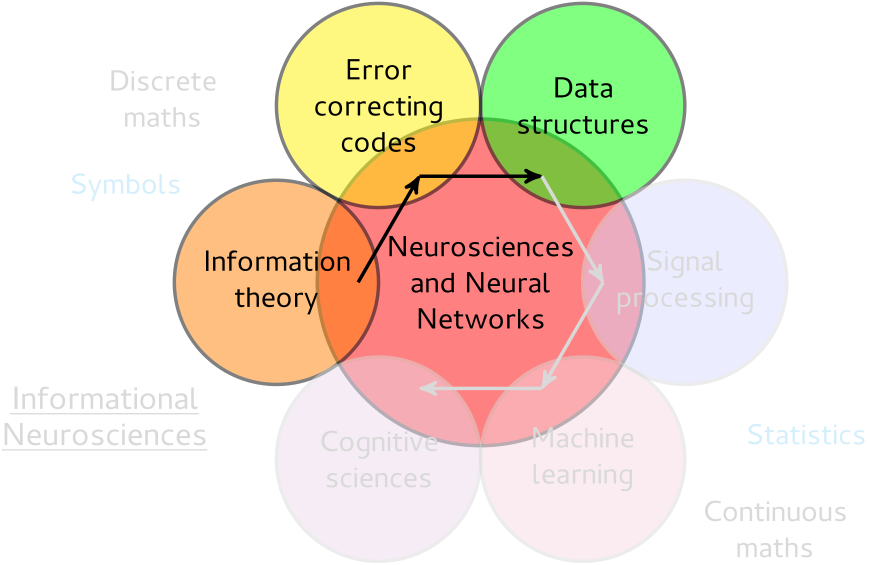
Informational Neurosciences



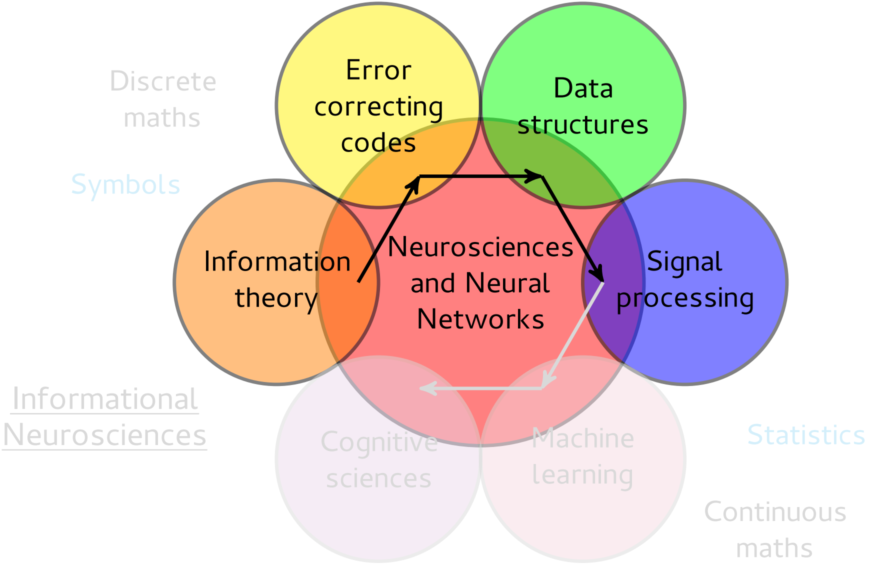
Informational Neurosciences



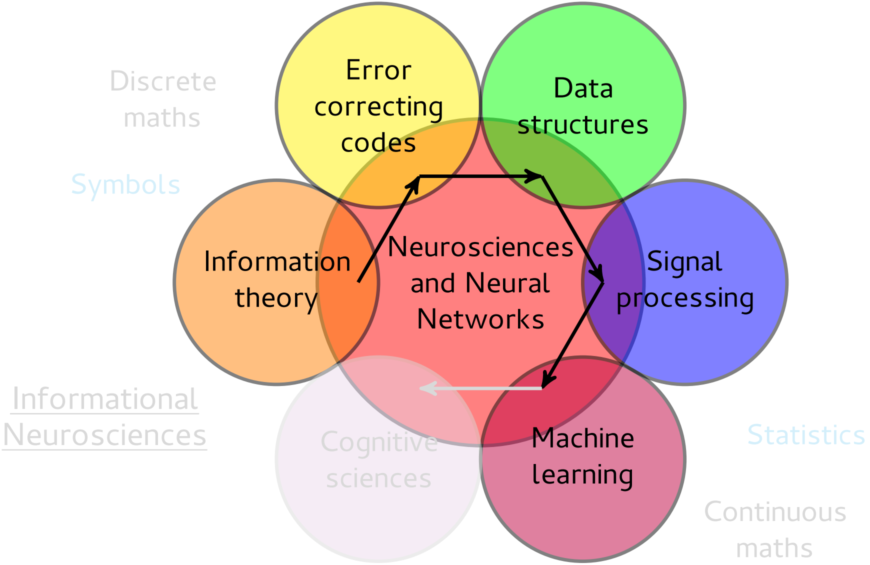
Informational Neurosciences



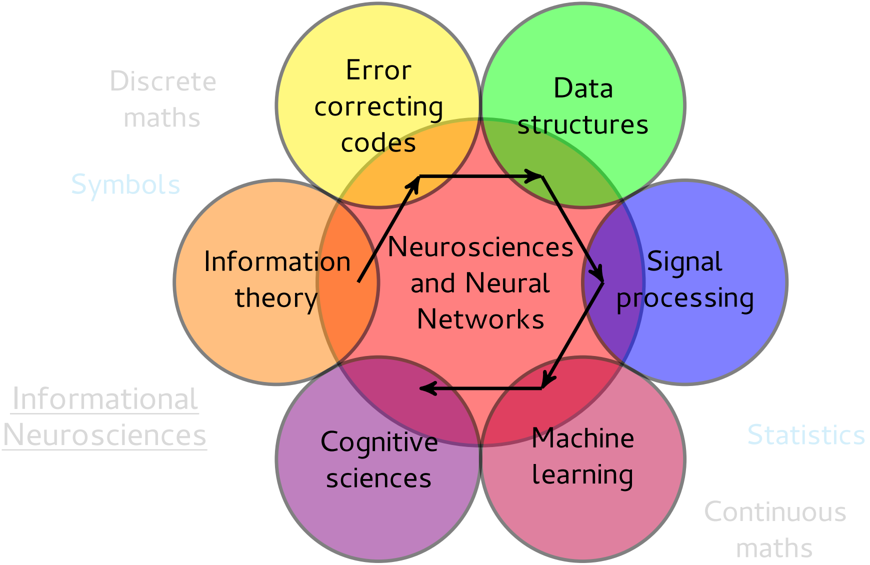
Informational Neurosciences



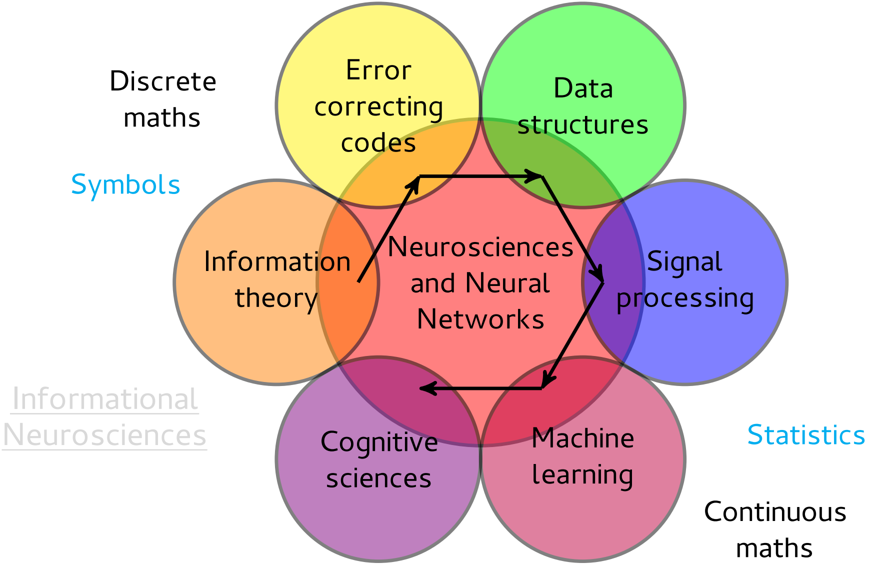
Informational Neurosciences



Informational Neurosciences



Informational Neurosciences



Informational Neurosciences