
Computation
and the Brain
2019
welcome
to Week 4
PASAJE GALVEZ

First: What happened last Wednesday

How does SGD avoid overfitting?

project alert!

"Understanding deep learning requires rethinking generalization" ICLR 2017; also, *cf* T. Poggio I, II, III 2017

Learning Theory

Easy case: we need data points $|S| = \log(|H| / \delta) / \boldsymbol{\varepsilon}$

(δ = level of certainty, $\boldsymbol{\varepsilon}$ = allowed classification error)

Learning reduces to an optimization problem:

$$\mathbf{min}_{h \text{ in } H}\ \mathbf{LOSS(S, h)}$$

(Important desideratum: LOSS differentiable wrt h)

If no realizability → an extra $\boldsymbol{\varepsilon}$ in the denominator

If H is not finite, then log |H| → VCdim(H)

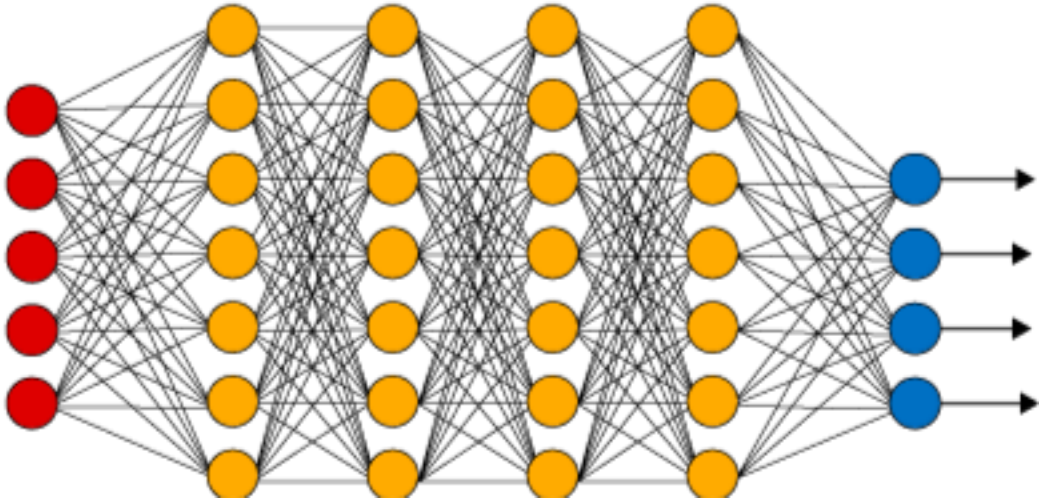
Deep Neural Nets

H = all DNNs of this shape and any possible weight combination (VCdim ~ number of links/parameters!)

You have to

minimize_W LOSS(H, S)=

Σ_S (classification error)^2

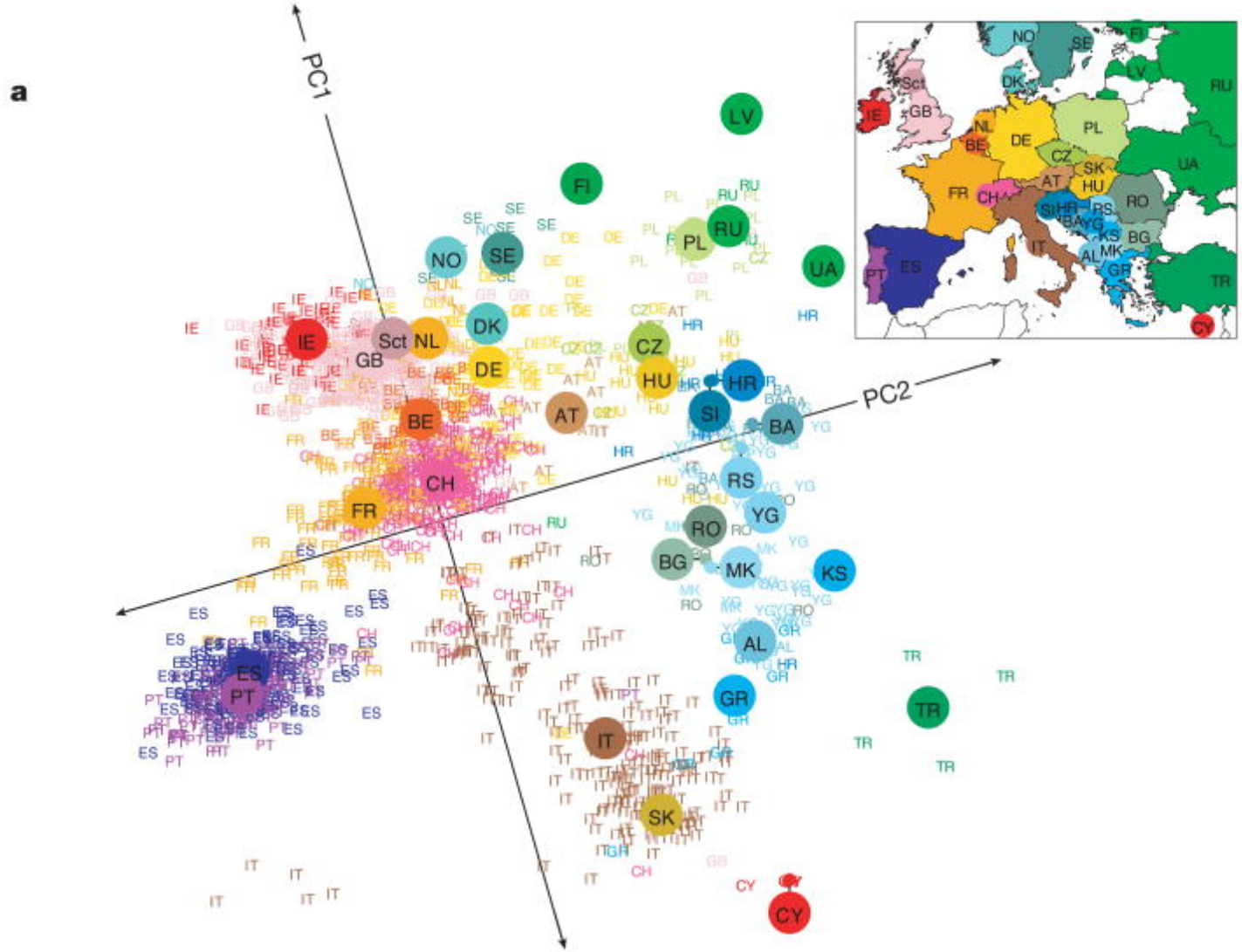


by ***stochastic gradient descent***

The Johnson – Lindenstrauss Lemma (JLL)

- for any data set
- if the dimensions have no special meaning for you
- and you are mainly interested in the distances between the data
- and you can tolerate an error of $\boldsymbol{\varepsilon}$
- then your dimension should not exceed $8 \ln n/\boldsymbol{\varepsilon}^2$

PCA: planar insights for a high-dim data set



To find the k largest eigenvectors of symmetric matrix A

- Starting from a random vector x compute the sequence of (normalized to 1) vectors Ax, A^2x, A^3x, etc, until it converges to v_1
- v_1 is the first eigenvector (the one corresponding to the largest eigenvalue)
- Now project the matrix to the space orthogonal to v_1 and repeat to find the second eigenvector. And so on, k times.
- Fast matrix multiplication version (A A^2 A^4 A^8 …)
- Works as long as the eigenvalues are well separated. Otherwise, just a little more complicated…

Sparse coding: Algorithms

- Gradient descent [Olshausen Fields 1998]
- Maximum likelihood, assuming a generative model and noise [Lewicki and Sejnowski 200]
- Alternating minimization:

Start with a basis B

Repeat: Sparse decode the data on B to get x

Gradient step on B

- Performance guarantees, see [Arora et al 2015]

- What is the **intuition behind choice of activation function** (such as logistic or arctangent), if several different choices satisfy the conditions required to approximate a function?
- Are there real-world scenarios in which **RL suffices without self-supervised learning**? What are some examples? computational neuroscience level and how could this be limited in specific context (say vision)?
- What function do the **multiple time scales neurons** operate at perform and **why would this be useful** for artificial neural networks?
- How **can symbolic AI continue to be helpful** in other tasks we are solving? (e.g language)
- How **does the brain approach self-supervised learning** on a Lecun claims **prediction is the essence of intelligence**. Is it true?
- Can we use the **biologically inspired** approach to create a **symbolic approach** to a problem?
- Hinton said: "**The brain is a device for getting gradients**." What experimental backing for this assertion do you think he has in mind?
- What are **current results on the "fast weights"** mentioned by Yann LeCun? Are there any indications that they are actually able to model logic and recursion?

- **What evidence** is there of computational operations in the brain which **use a radial basis function**?
- Is there much **evidence that human brains explicitly implement logical reasoning functions**? ... maybe our explanations are more analogous to an image captioning task, where we exploit statistical relationships...?
- In the spirit of the universal approximation capabilities of neural networks, **does an "inverse" statement hold for autoencoders**, suggesting that any embedding function from a high dimensional space to a low-dimensional manifold (if one exists) can be represented with a 2 certain function class?
- What are **some tasks that require a lot of hours and data for humans** to learn but may comparatively be **easier for a machine** to learn through RL?
- 9re Khaligh-Razavi paper in Kriegskorte review) If using biologically inspired processes could improve neural networks, do we think that there's merit in **teaching more about biological processes in computer science classes**?
- On a more practical level, **how much further into CS theory** do we need to delve into to have a better practical understanding of solving these problems?
- When comes to **computational neuroscience,** are people more interested in and **excited about theory or practical results?** Is it different for other related fields (in computer science) and why?

- IWhy is **symbolic AI considered ‘intelligent**’?
- When self-supervised learning is put into **practice** is it used in the general **‘learn to model the world’** sense?
- s the **Helmholtz machine** used in training today’s vision models? How could we test its validity against how **dreaming actually functions** in the human brain?
- Why is it so ”certain” that supervised learning is not the future of machine learning?
- What are the steps being taken to understand and **visualize what exactly is happening within neural nets**? What oversight could there be on the internals of it?
- **How** the hell do they **measure “similarity of representations” between neural networks** and parts of the VC and IF? (Kriegeskorte paper)
- Is there any reason to think the **brain is Bayesian**?

Today

- Synaptic connectivity
- Plasticity

The Brain: a *Uuuuuuge* directed graph

- About 10^{11} neurons
- Between $10^{14\text{-}15}$ synapses
- Notice: this means that every neuron has on average 1000 – 10,000 presynaptic neurons, and again as many postsynaptic ones
- *A random graph?*

A well known random graph

- $G_{n,p}$ or the Erdős – Renyi random graph [1959]
- n nodes
- **Prob** [edge I → j] = p independently of other pairs
- hence, in-degree or out-degree concentrated at pn
- ***Sharp thresholds:***
- p < 1/n: dwarf components ~ln n
- 1/n < p < ln n/n: a giant component, still disconnected
- p > ln n/n: connected.
- Clique: 2ln n/ln(1/p)

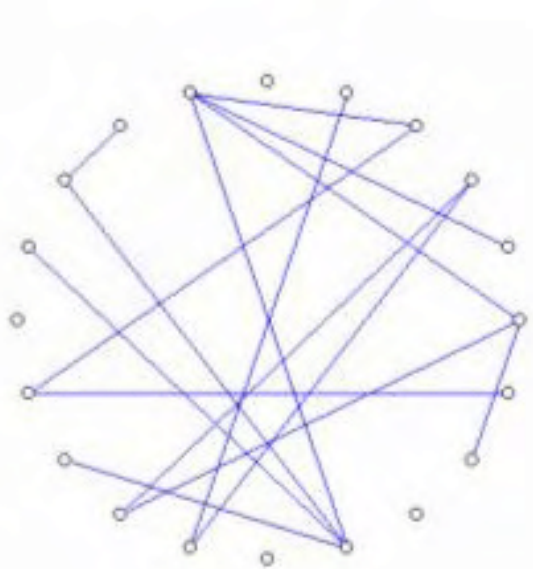
Another well known random graph

- P_n^{α} power-law (or Internet-like) graph
- Undirected
- **Prob**[there is a node of degree d] ~ $n^{-\alpha d}$
- Internet, www: α ~ 2.6 - 2.8
- Or: generated by some preferential attachment process (“the rich get richer”)

Erdős – Renyi graphs

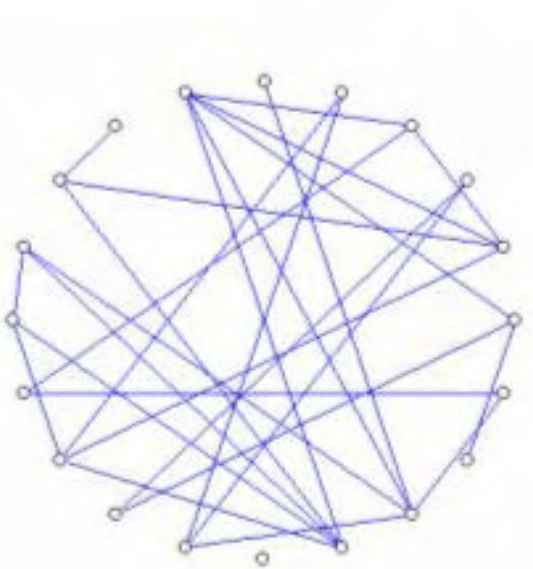
$p = 0$

(a)



$p = 0.1$

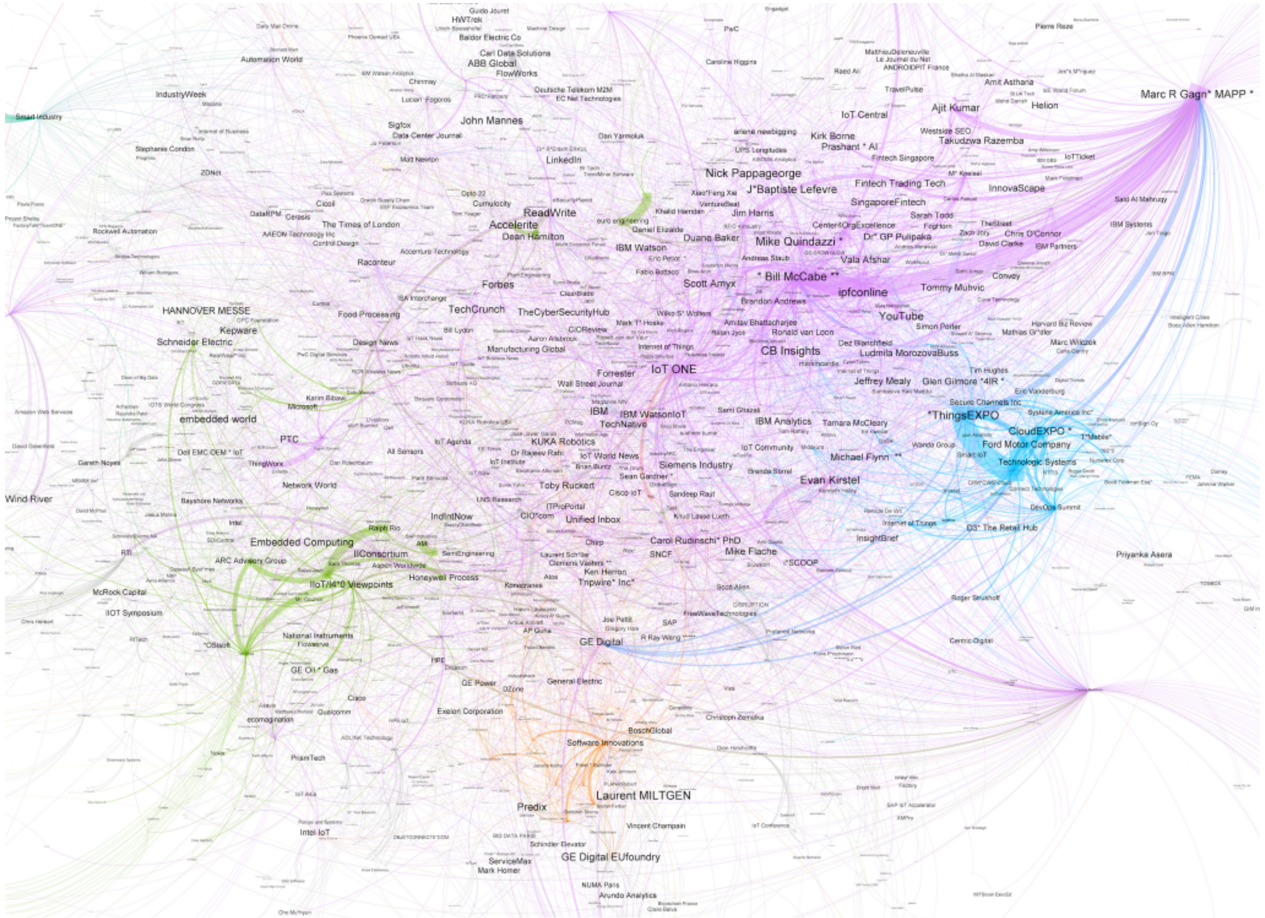
(b)



$p = 0.2$

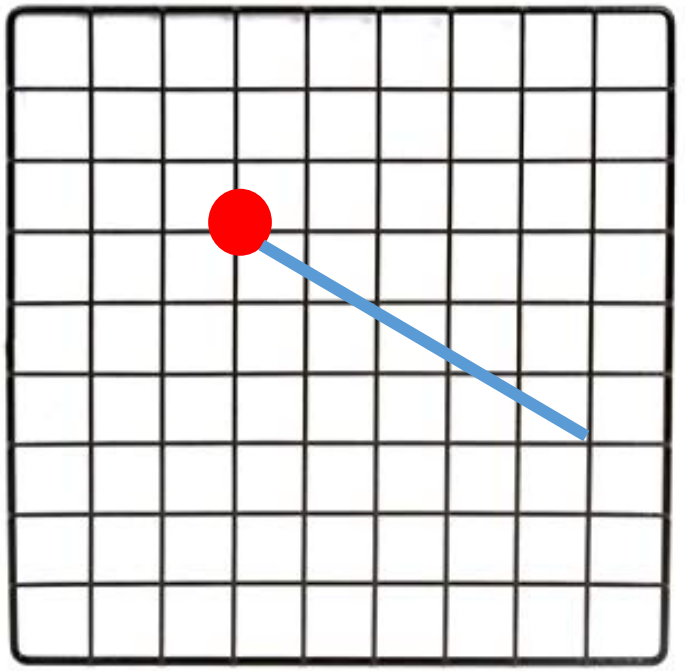
(c)

Power law / Internet like graph



A third kind of brain-relevant graph: The small world graph [Kleinberg 2000]

- A grid (2D geometry!)
- Plus from each node
  very few random edges
- Going distance d away
  with probability ~ d^{-2}



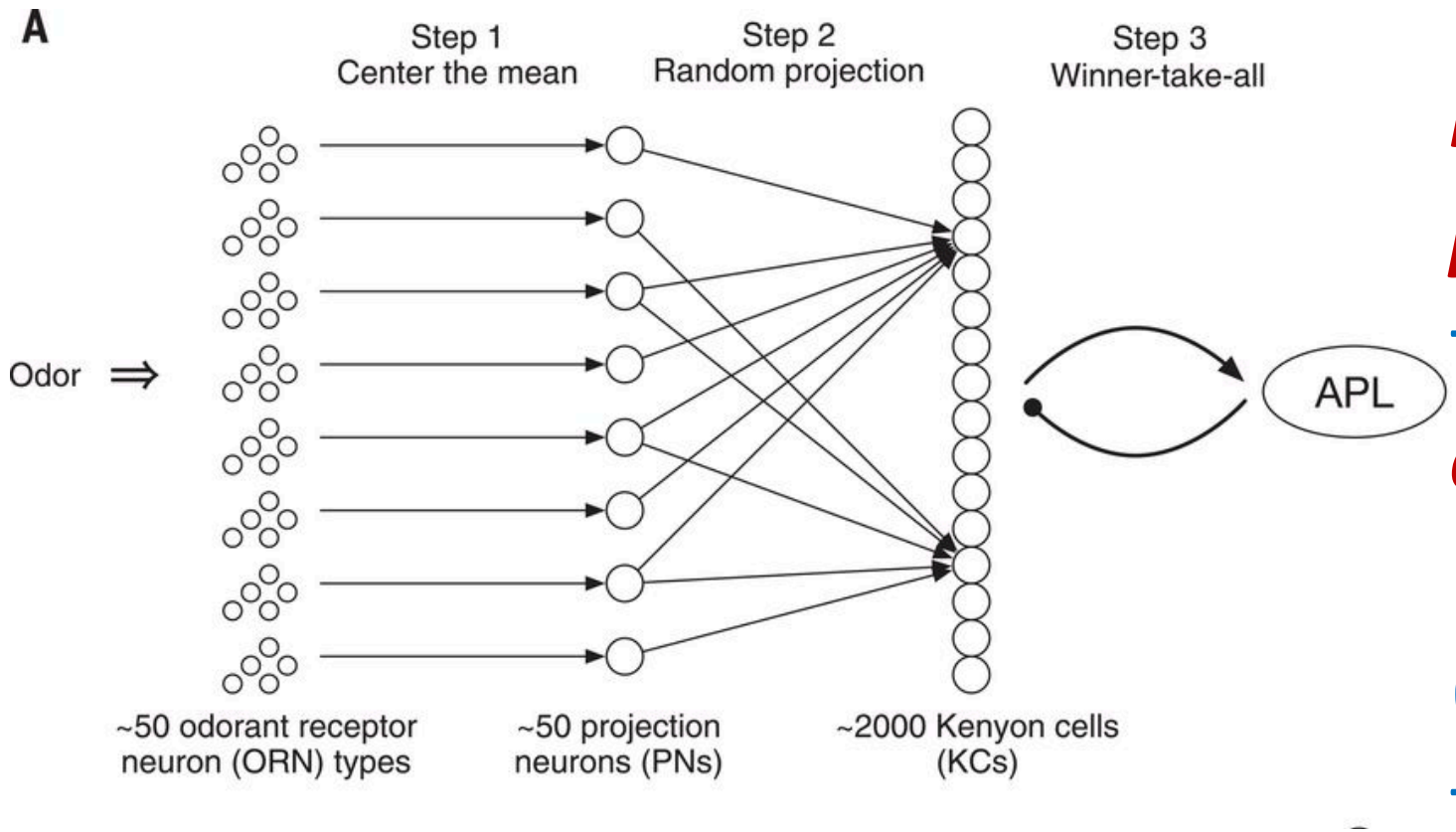
- **Theorem**: Greedy algorithm routes in very few steps

Which of these kinds of random graphs is the brain more like?

To start with, in what sense is it random? Can brain connections be random?

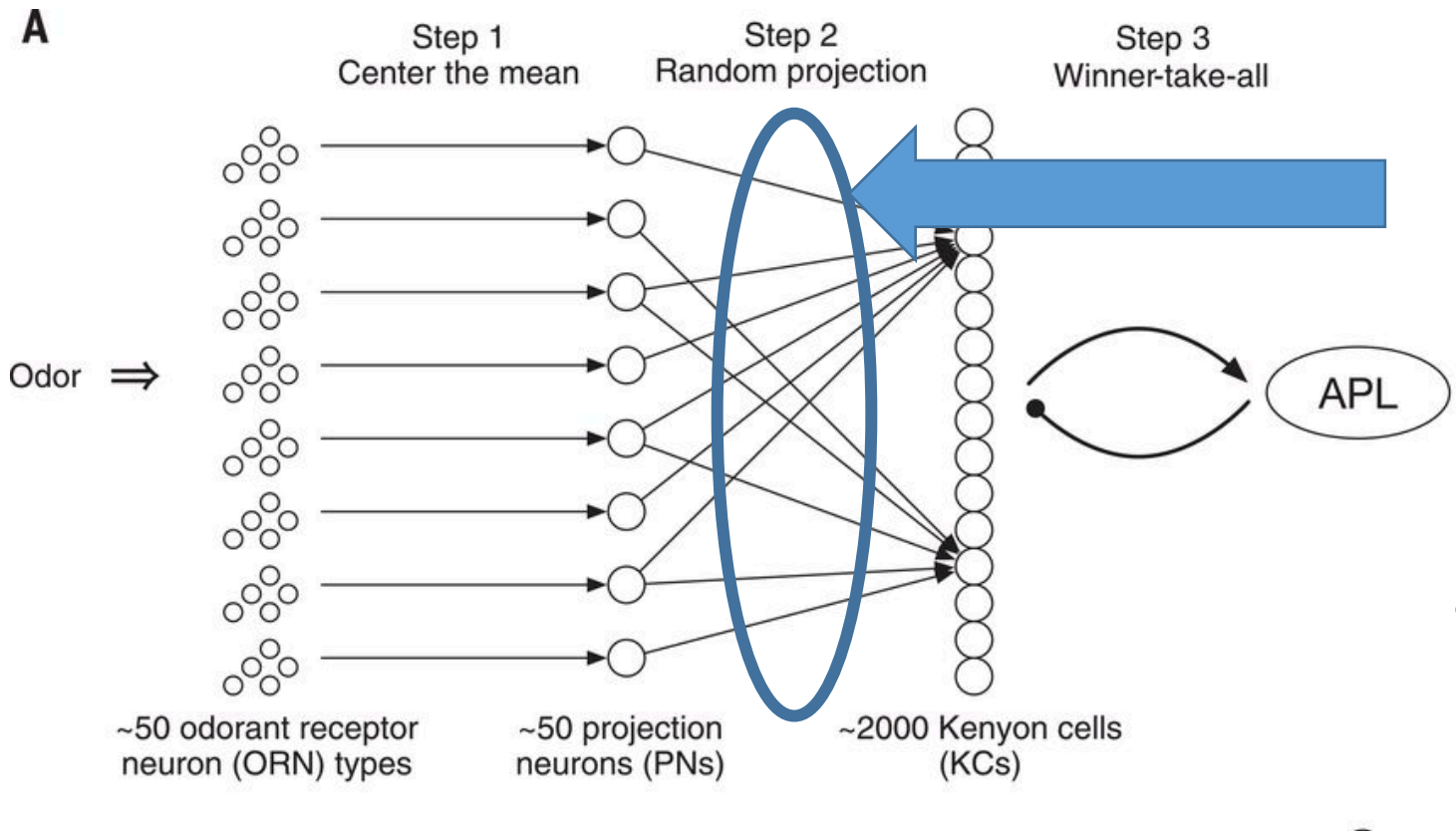
- An answer from a place very far in the animal phylogenetic tree

How fruit flies remember smells



random projection followed by ***cap (RP&C):*** 100 winners (out of 2000) take all

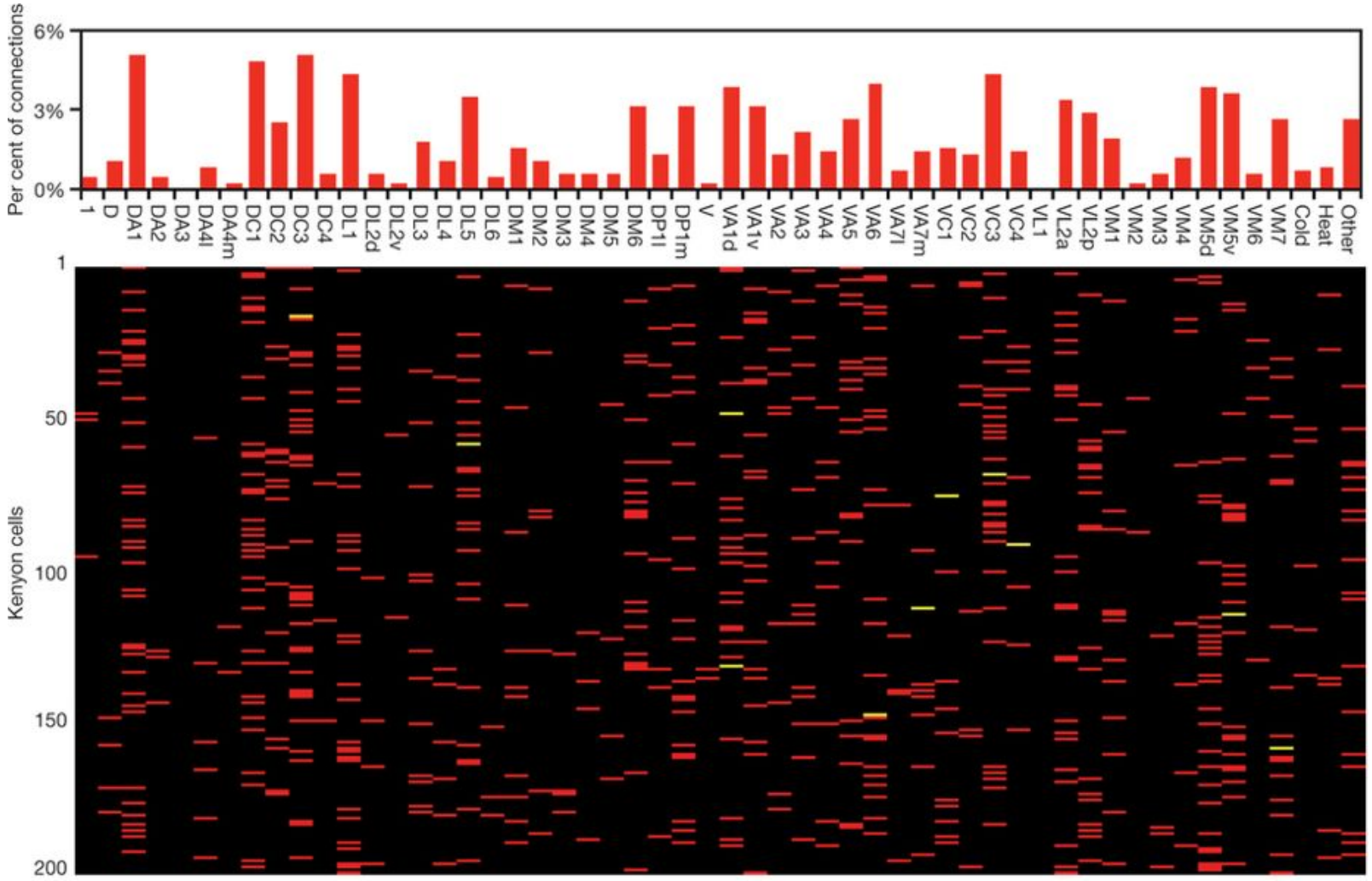
How fruit flies remember smells



Q: but wait a minute! Is this a random bipartite graph?

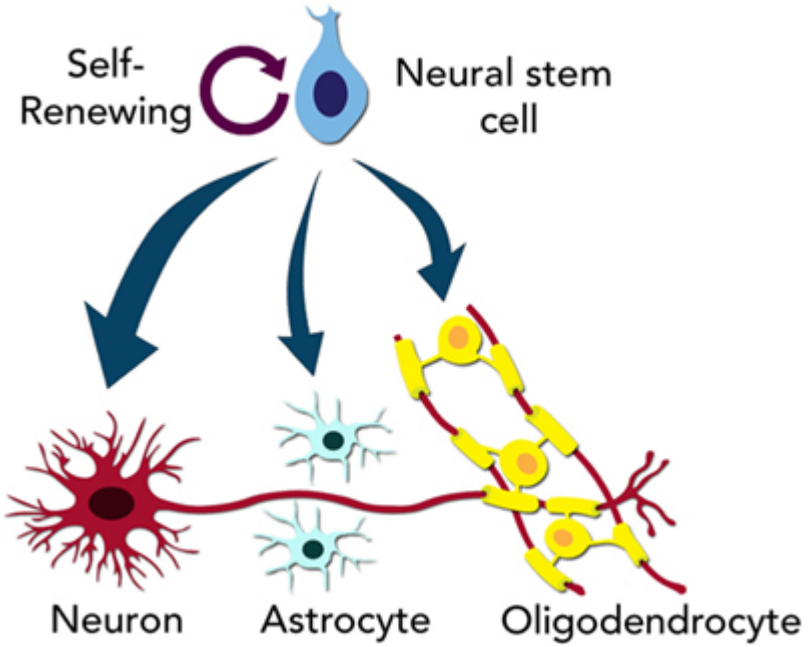
A: ***Random convergence of olfactory input in the Drosophila mushroom body*** by S. Caron, V. Ruta, L. Abbott, R. Axel,2013

Bottom line: looks like a random bipartite graph, except that the degree distribution of the LHS is not uniform



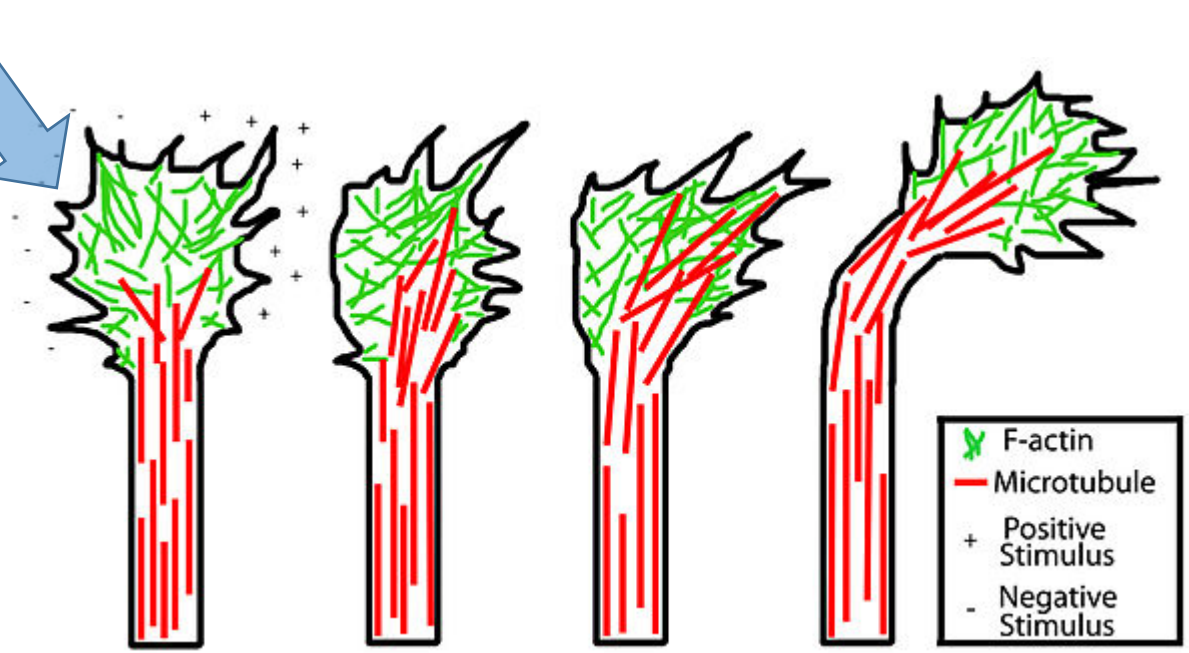
What is the generative model of the Human Connectome?

1. Neurons are created from stem cells
2. They sprout dendrites and an axon hillock (basis)



What is the generative model of the Human Connectome?

3. Then the axon's growth cone takes over



- A sensory – motor organ

- Navigates the brain following chemical cues (advance, turn, stop)

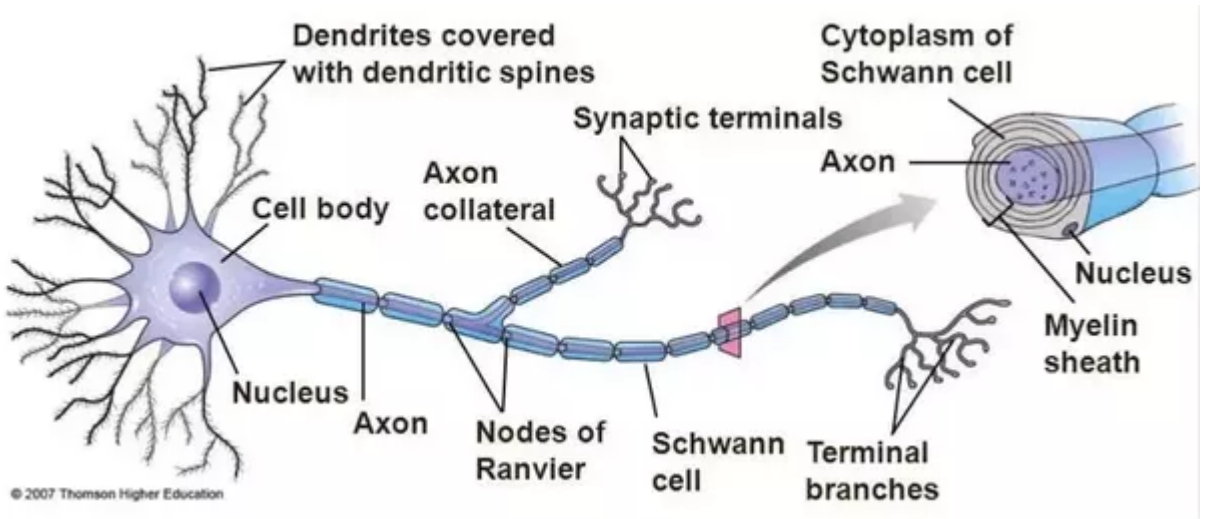
How long does the growth cone travel?

- Many yards in the giraffe
- Some µm in local circuits, or for most inhibitory neurons (exponentially concentrated to zero)
- About 4mm on average in the cortex

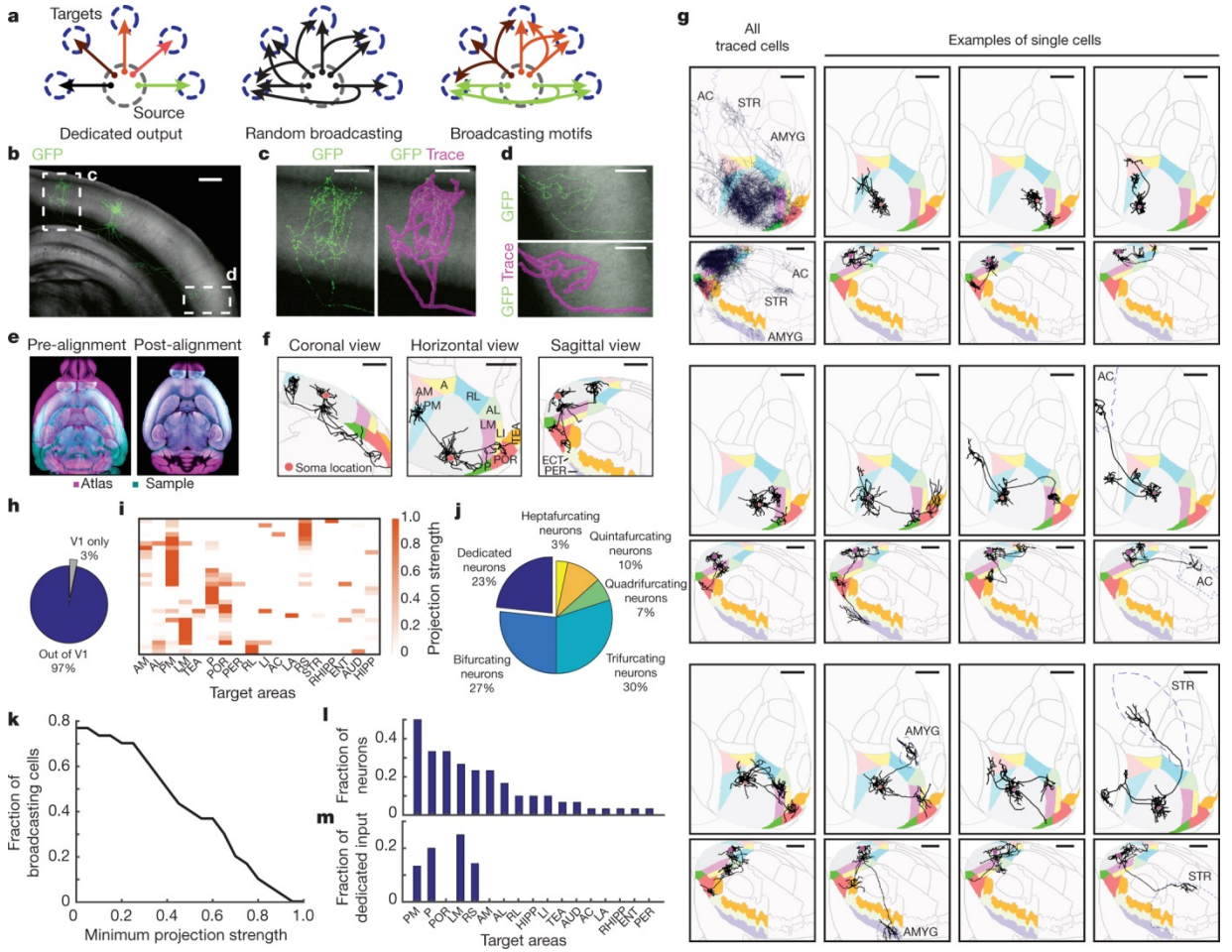
Many navigational strategies

- sometimes there are *early scaffold* neurons
- other times there are *signpost* neurons
- advance, turn, stop: Q: is there a ***split***?

- axon collateral



Brain-wide single-cell tracing reveals the diversity of axonal projection patterns of layer-2/3 V1 neurons,with most cells projecting to more than one target area

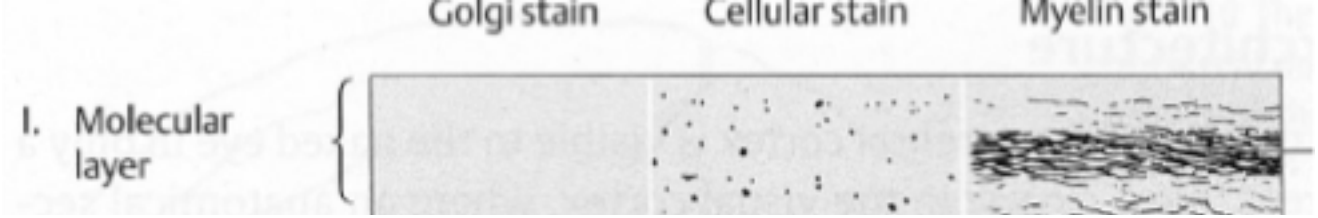


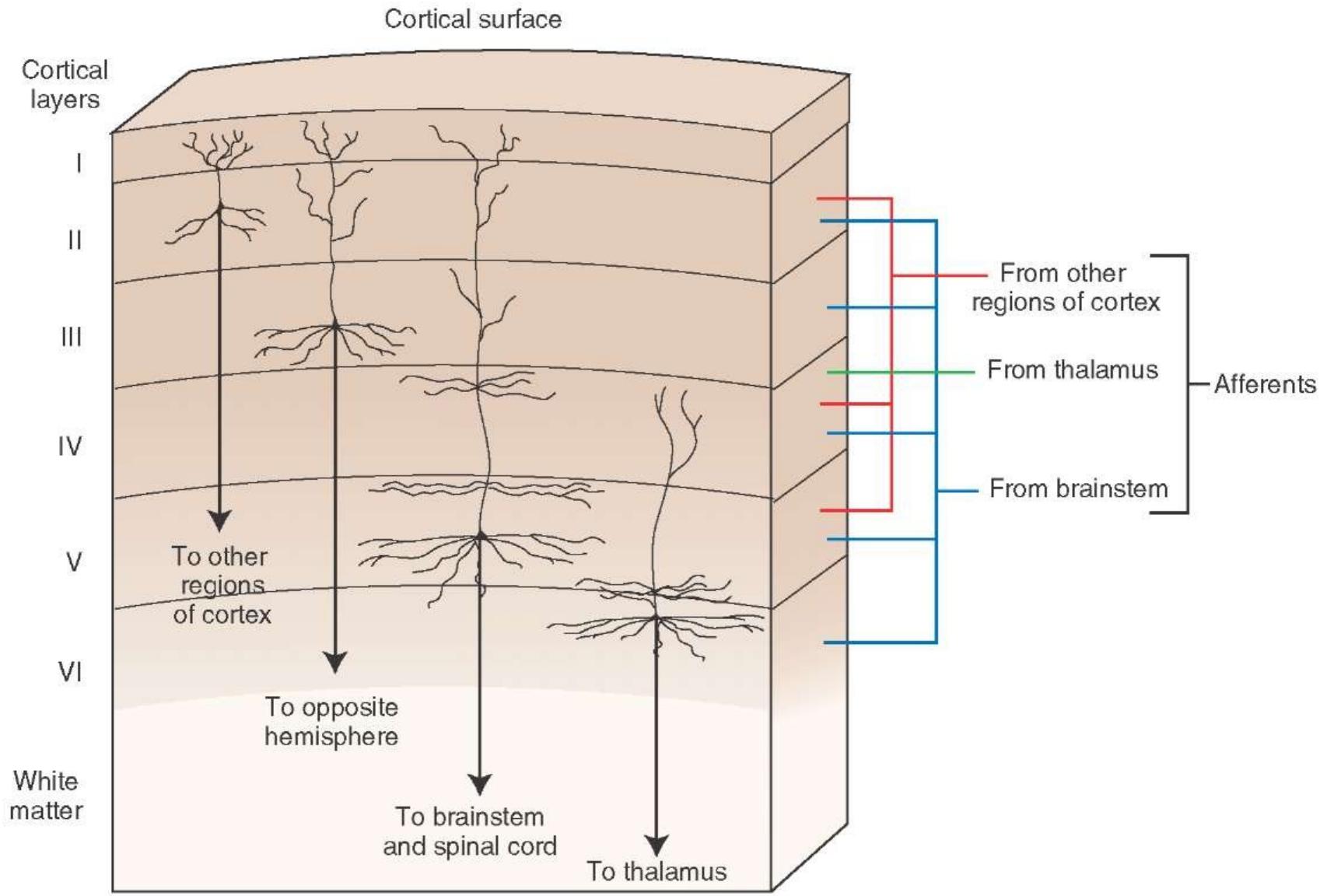
Y Han *et al. Nature* **556**, 51–56 (2018)
doi:10.1038/nature26159

nature

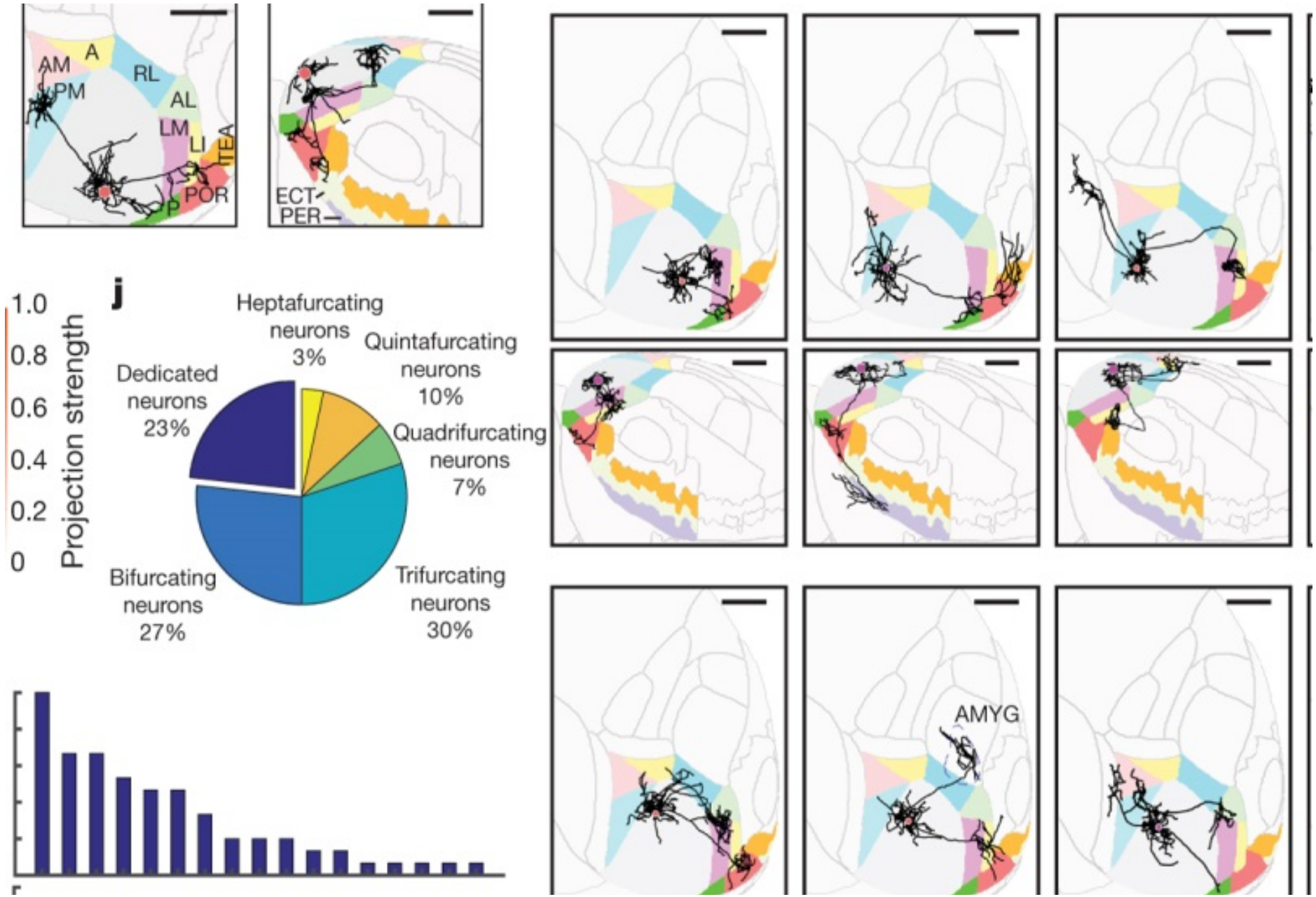
(Cerebral) Cortex: the engine

- 1.5 mm
- 2500 cm^2
- six layers
- 20B neurons



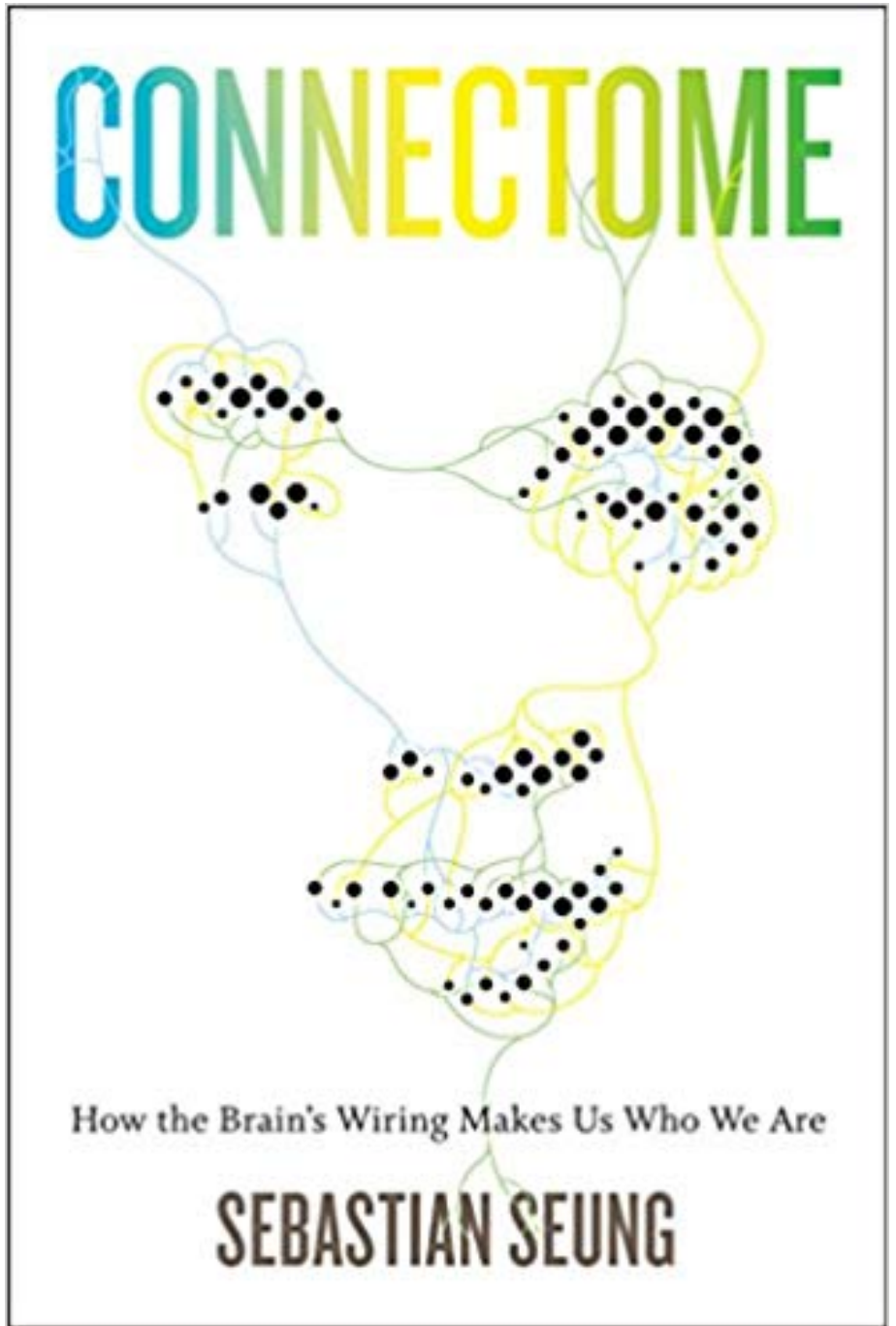


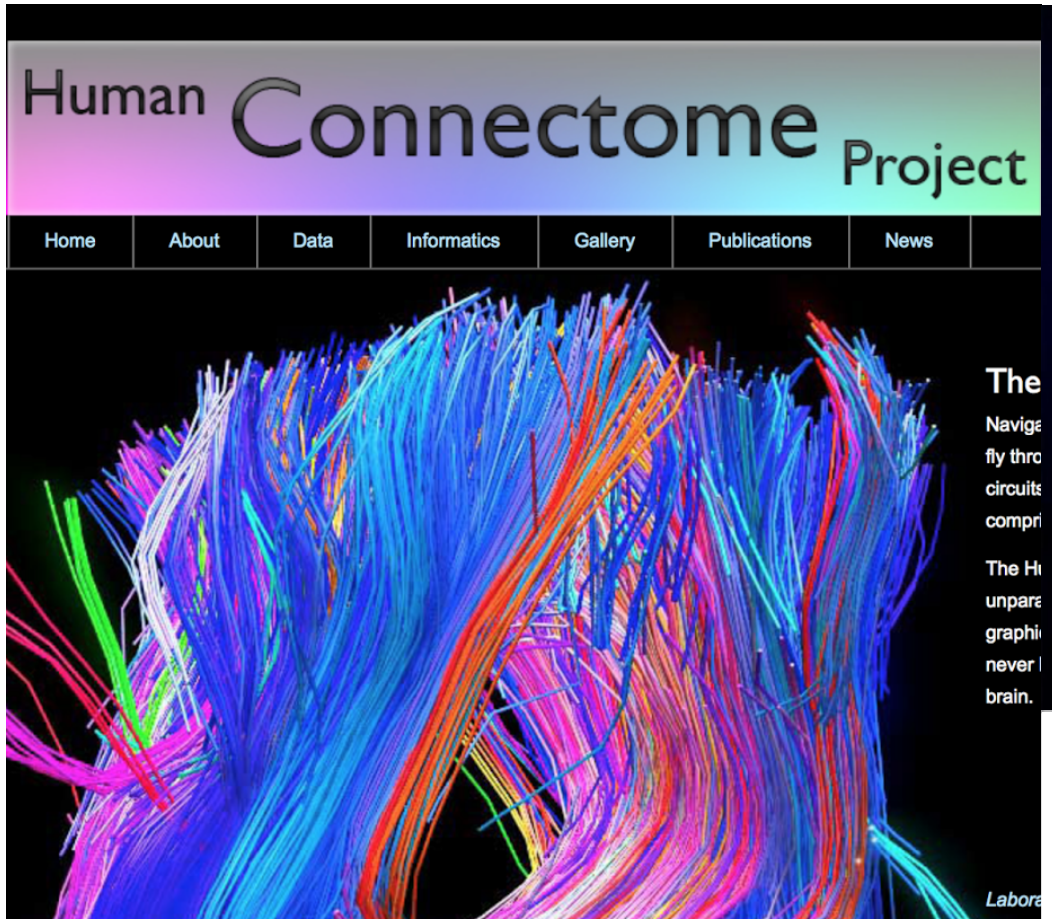
So, there is a split… and it seems to be important



Connectome

(as in “you are your _”)

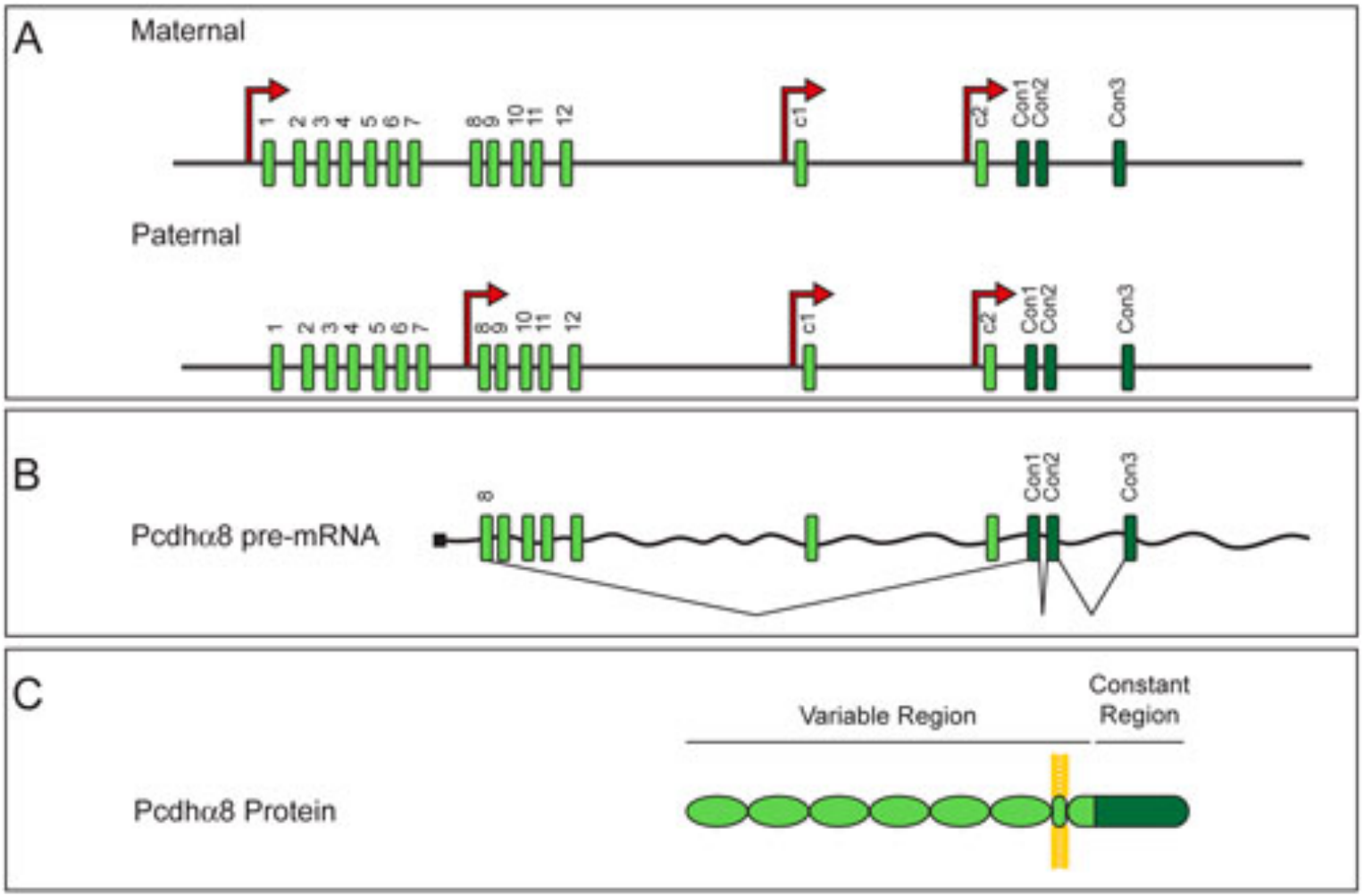


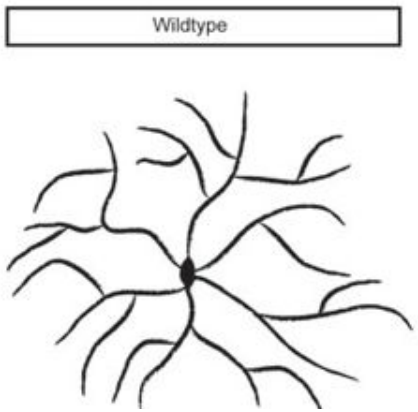

Human Connectome Project
Home
About
Data
Informatics
Gallery
Publications
News
The
Naviga
fly thro
circuits
compri
The Hu
unpara
graphi
never
brain.
Labora

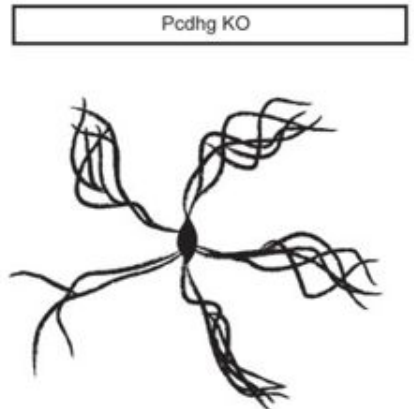

BLUE BRAIN
BY:- VIJAY THAKKAR


GGAA
ATCG
ACCT
CTAG
TTTC
GTGA
AAGC
ATCG

Speaking of bar codes…
Protocadherin proteins and molecular self-avoidance







Question

- Suppose we have the human connectome
- Will it help us understand how the brain works?

So, what kind of graph is the connectome?

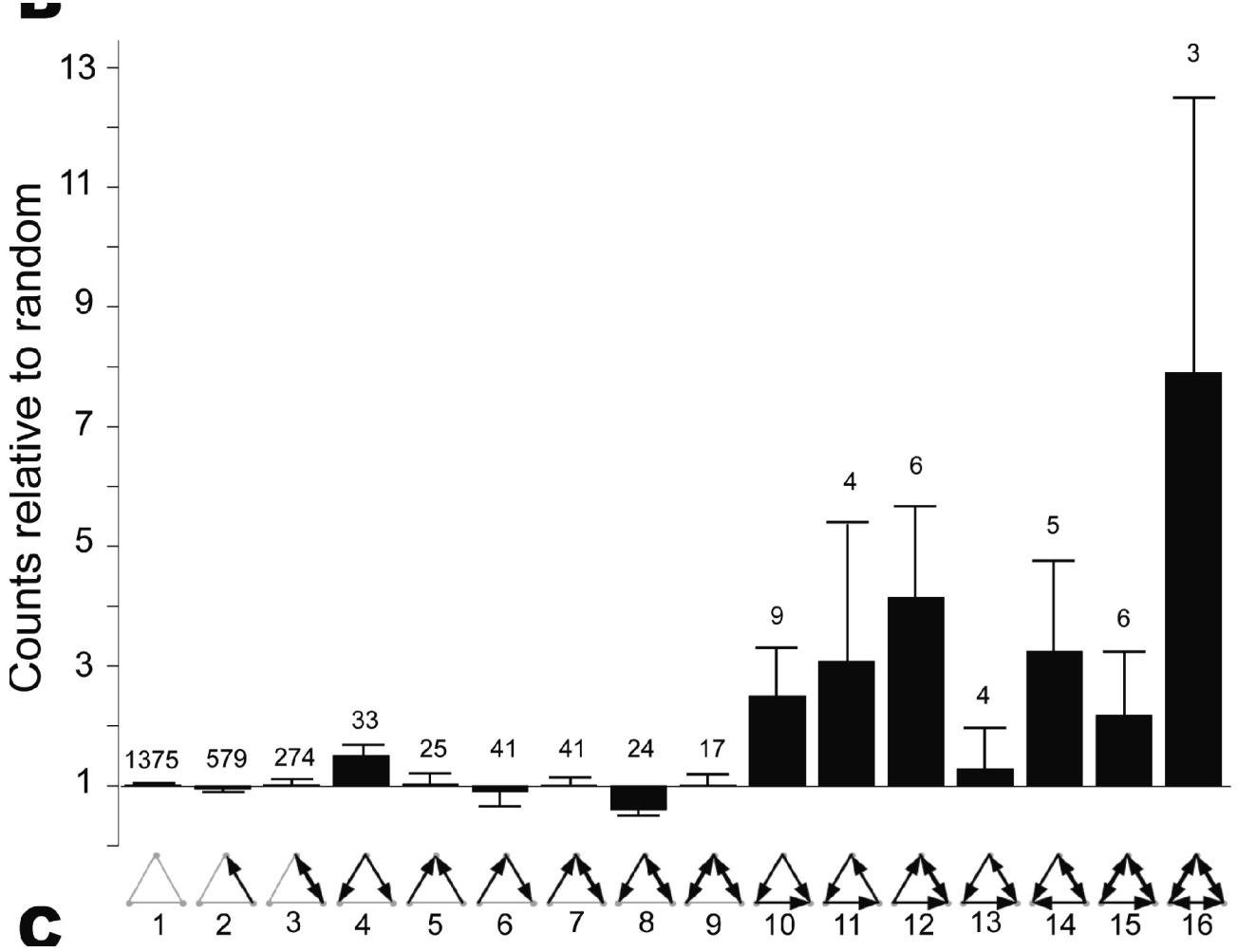
Hypothesis: Locally Erdős – Renyi and globally Internet-like (or small world?)

So, what kind of graph is the connectome? Two-way connected neurons

Two-way connection between two neurons is about ten times more likely than Erdős – Renyi predicts

“Highly Nonrandom Features of Synaptic Connectivity in Local Cortical Circuits”
S. Song et al., PLOS Bio 2005

So, what kind of graph is the connectome? Three neuron connectivity



“Highly Nonrandom Features of Synaptic Connectivity in Local Cortical Circuits”
S. Song et al., PLOS Bio 2005

Why is that?

- Geometry?

"Independently Outgrowing Neurons and Geometry-Based Synapse Formation Produce Networks with Realistic Synaptic Connectivity"

van Ooyen et al. PLOS 1 2014

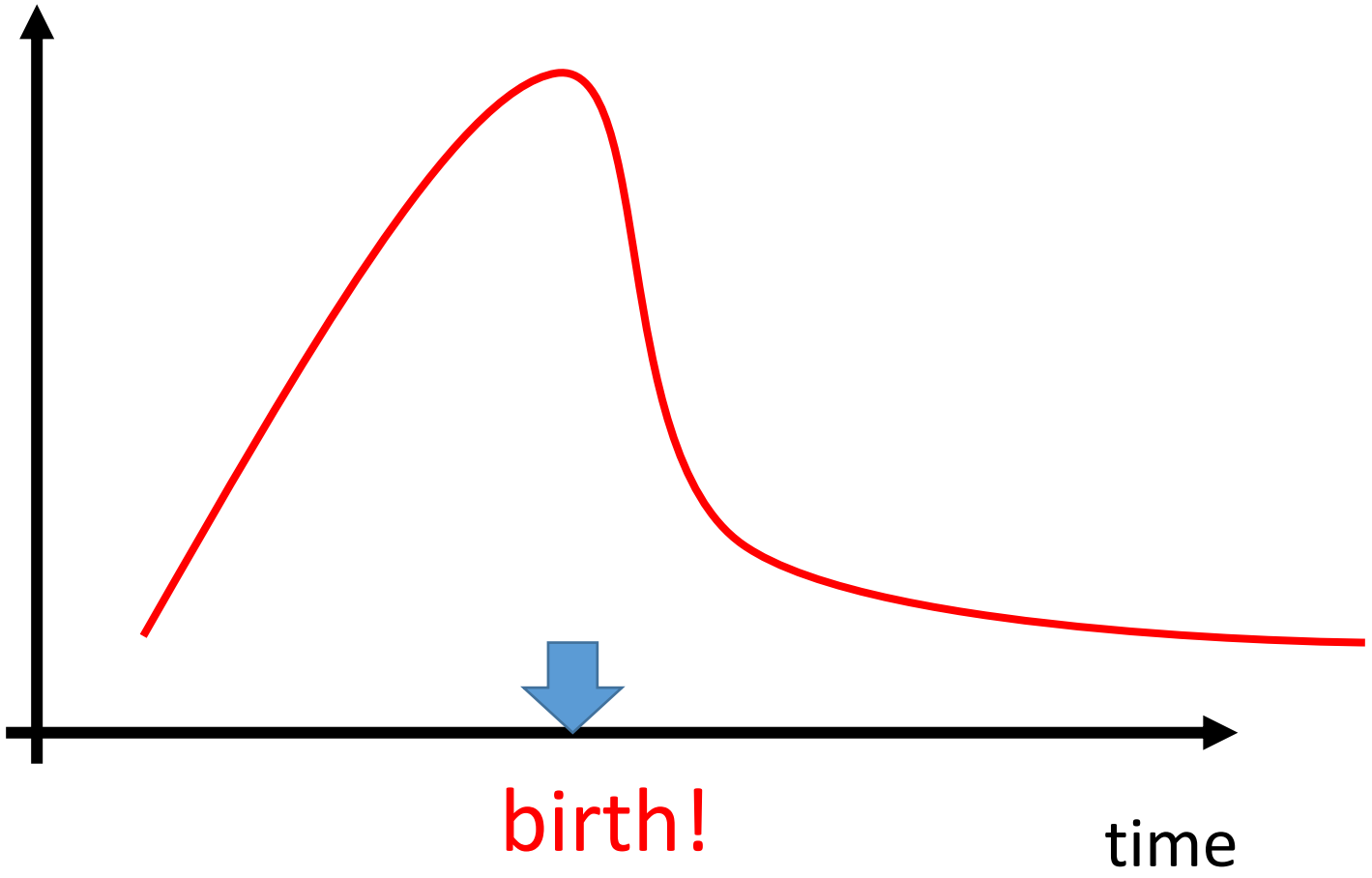
- Or plasticity?

cf H. Markram V2 cells responding to the same edge direction are much more likely to be connected

Your connectome changes
(…but so do you…)

- New connections are formed
  - In the embryo
  - In the baby
  - In the adult
- Existing ones are destroyed

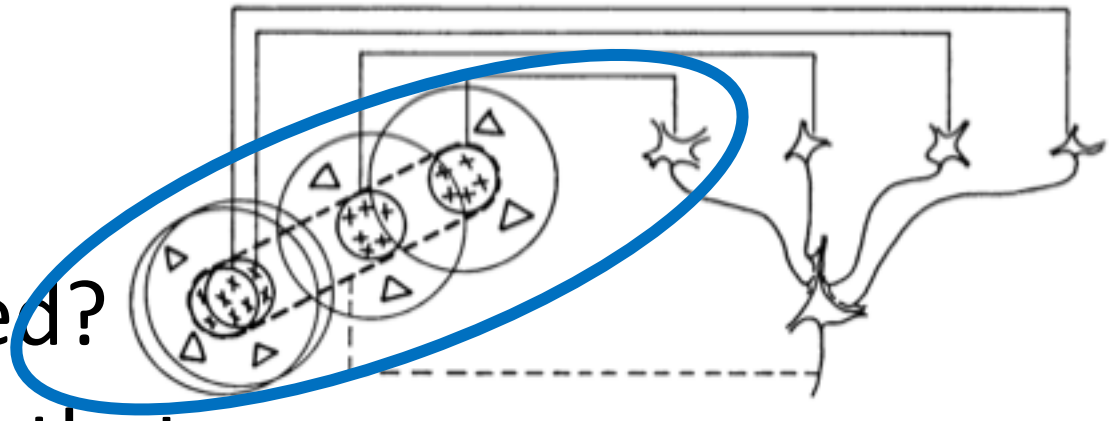
Number of synapses (into muscular fibers)



How is this deletion done?

- cf algorithmic work on graph sparsification, [Spielman and Teng 2004] and others 

Incidentally, remember Hubel and Wiezel's simple cells?



- How are these synapses formed?
- How do all these ganglia know that they are on a straight line in the retina?
- Was it evolution?
- Is it done during development?
- Or is it learning and synapse deletion?

Btw: yet another thing to consider

- An axon can touch the postsynaptic dendrite at many places
- At many “edges of the tree”
- How do these add up to effect the postsynaptic neuron?
- Sum of products? Products of sums? Other?
- Much work is done on this front
- Some say, ***that*** is where the brain computes…

Connections between neurons are not permanent

- New ones are formed
  - In the embryo
  - In the baby
  - In the adult
- Existing ones are destroyed
- Existing ones become stronger or weaker at a large range of time scales (1ms to months)
- ***Plasticity:* the way brains learn**

Pre-synaptic activity affects the magnitude of post-synaptic response

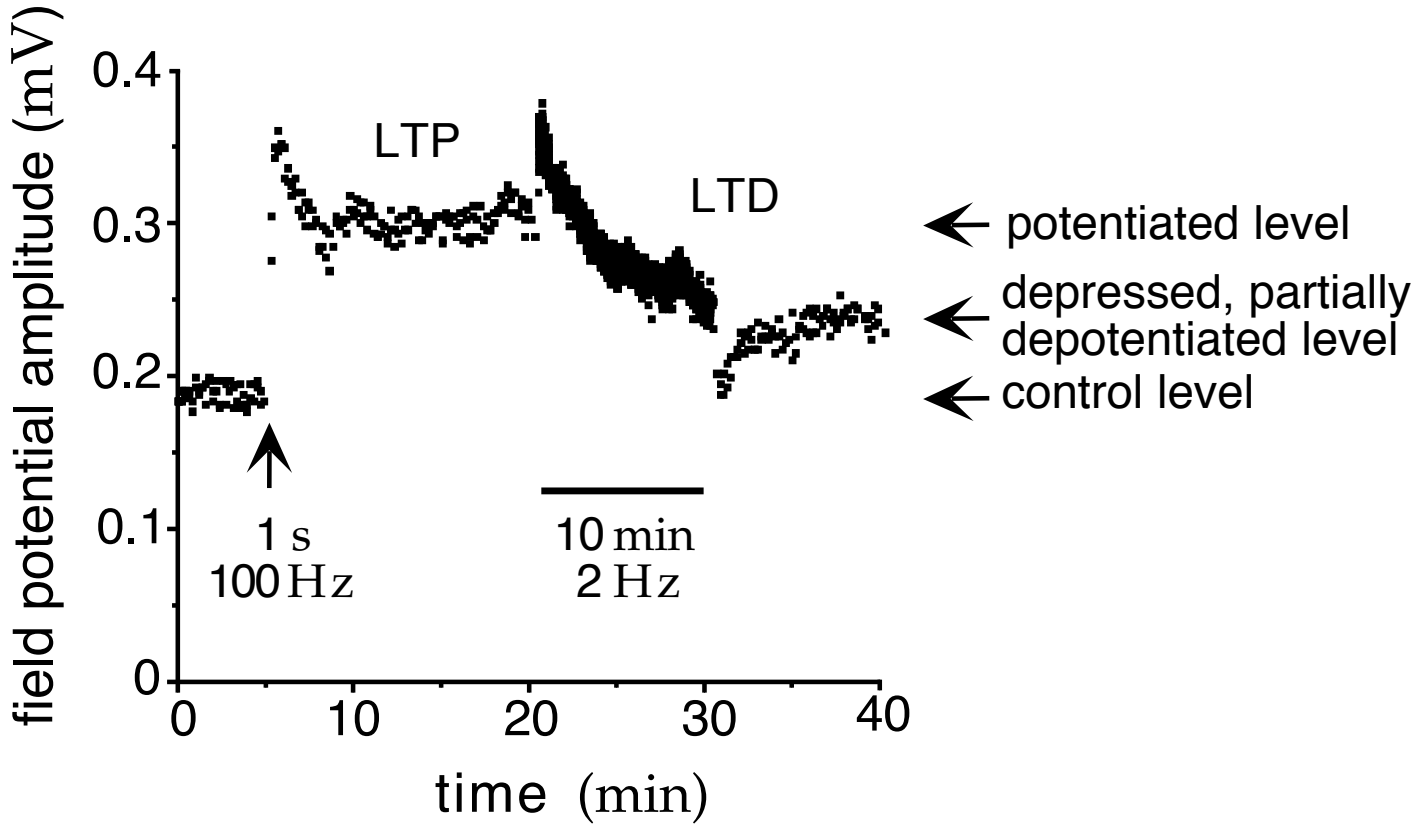
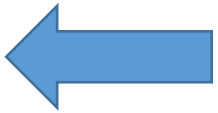


Figure 8.1 LTP and LTD at the Schaffer collateral inputs to the CA1 region of a r

History of plasticity

- 1890: Ramon y Cajal hypothesizes that synapses between neurons are the agents of learning
- 1949: Hebb's rule: "fire together wire together"
- 1966: Lomo observes long term potentiation (LPT) in the rabbit brain
- 1970s: Long term depression (LTD) is observed
- 1980s: The biochemical basis of plasticity: Ca^{++} and NMDA receptors
- 1986: Backprop: software plasticity through SGD
- 1995: Spike timing dependent plasticity (STDP)

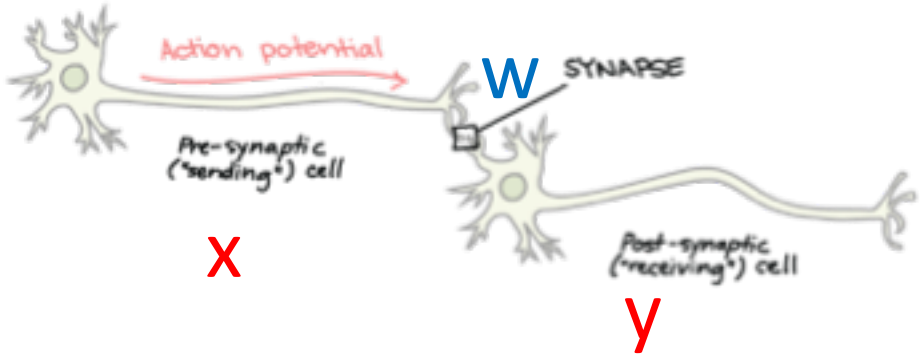


where does this fit?

Models of plasticity

- Mathematical, black box models
- Spike timing models
- Biochemical/physiological models of plasticity
  - Plasticity seems to have to do with Ca^{++} and NMDA receptors
  - Spine shape change, as well as spinogenesis, may affect plasticity
  - Latest: MNDA receptors ***move*** to enhance plasticity ***(at the 1ms scale…)***

Models of plasticity



Hebb: $\Delta w \sim x\, y$

Vector Hebb: $\Delta \mathbf{W} \sim \mathbf{X}\, y$ ($\mathbf{X}$ vector of presynaptic neurons)

Covariance form of Hebb (assuming y is the average input times $\mathbf{W}$): $\Delta \mathbf{W} \sim \mathrm{cov}[\mathbf{X}]\, \mathbf{W}$

Hebb with LTD: $\Delta \mathbf{W} \sim \mathbf{X}\, (y - \vartheta)$ – but then LTD if y = 0

BCM rule corrects this: $\Delta \mathbf{W} \sim \mathbf{X}\, y\, (y - \vartheta)$

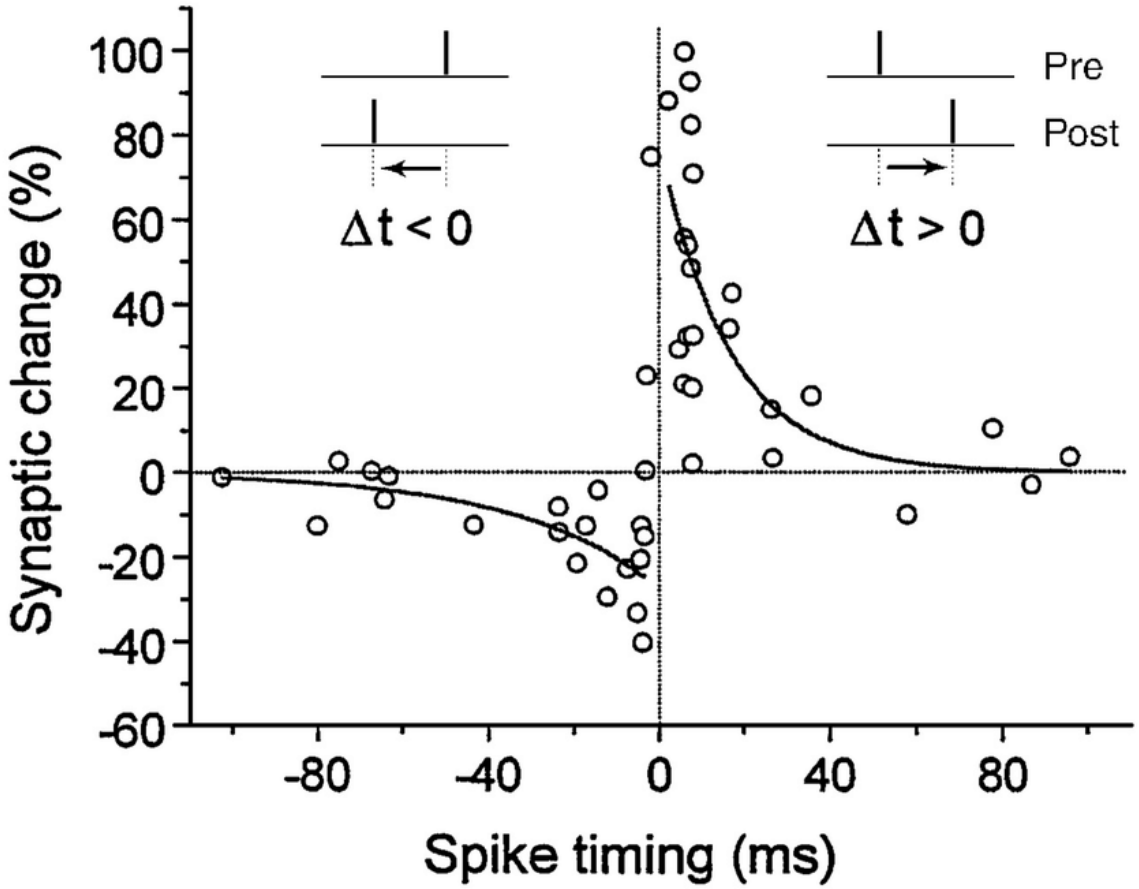
Homeostasis, normalization and competition: at a slower time scale, the sum of all presynaptic weights is renormalized to 1

Models of plasticity: Plus: Homeostasis, normalization and competition

- Homeostasis means that the organism is stable, its parameters stay in check and its resources are not depleated.
- For example, inhibition makes sure that not all neurons fire at the same time
- Q: So, what mechanism makes sure plasticity does not cause synaptic weights to become infinite?
- A: at a slower time scale, the sum of all presynaptic weights is gradually renormalized to one
- Note that this begets competition between presynaptic cells

Spike timing-dependent plasticity (STDP)

If spike arrives in time, some gain. Just in time, big gain.
If it just misses it, some loss.
Just misses it, big loss.



Btw: Hebb's exact words

Rhyming soundbite: "fire together, wire together"

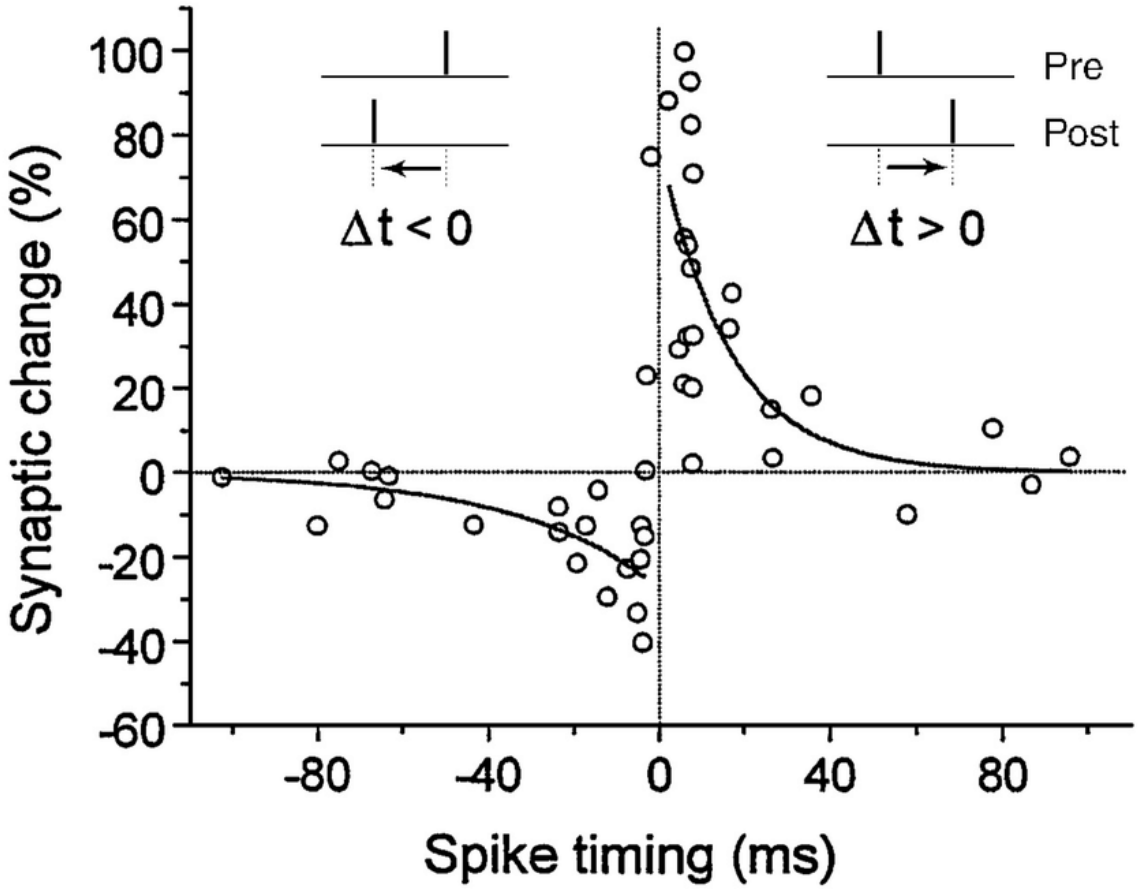
Actual quote from [Hebb 1949 *The Organization of Behaviour*]:

"When an axon of cell A is near enough to excite cell B or repeatedly or persistently ***takes part in firing it***, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased."

NB: "or both cells" is an impossibility

Spike timing-dependent plasticity (STDP)

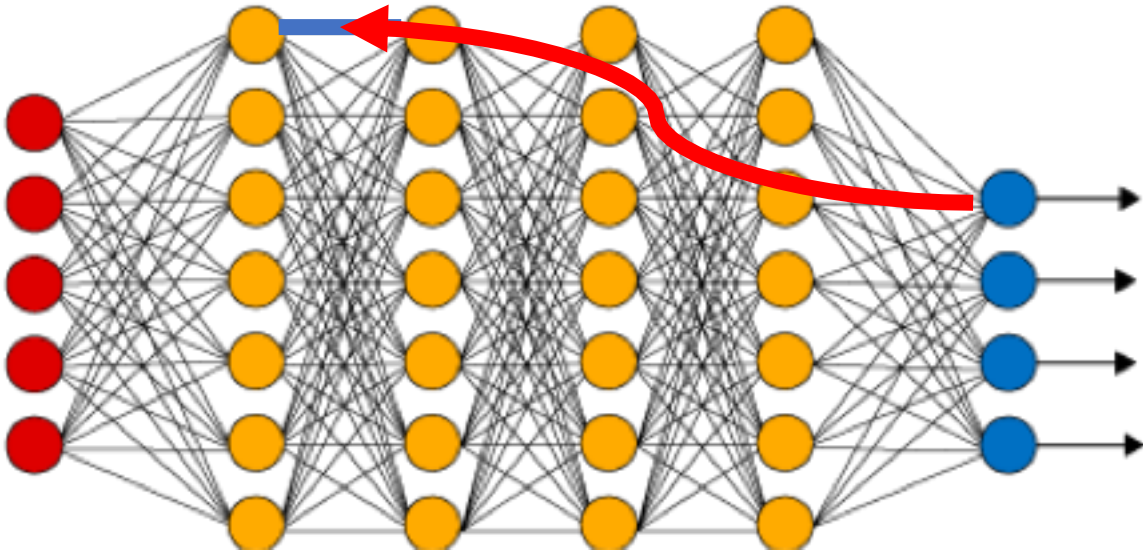
If spike arrives in time, some gain. Just in time, big gain.
If it just misses it, some loss.
Just misses it, big loss.



Wow! What does this mean?

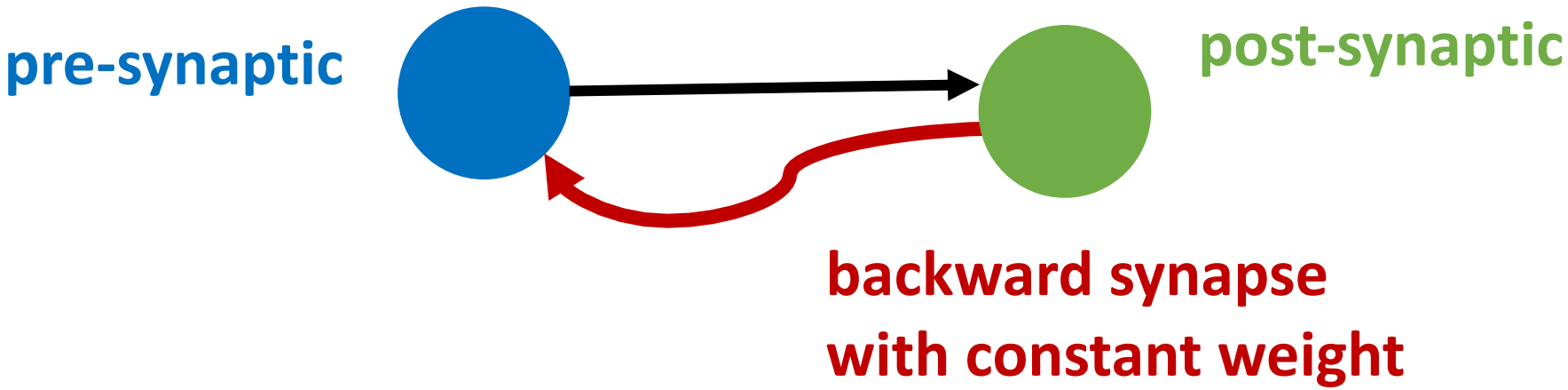
Plasticity in the Brain vs in ANNs

- The Brain apparently learns through the plasticity of the synapses
- DNNs learn through back propagation, a kind of synaptic plasticity
- ***Big*** difference: in back propagation you need information ***from downstream activity***



Biologically plausible ANNs

- [Lillicrap et al. 1914]: constant (non-plastic) random synaptic weights in a **backwards** synapse suffice for some learning!



Bengio et al. 2016 ``Towards biologically plausible deep learning”

- Gradient descent: $\Delta x^t = \alpha\, \delta(t)\, \nabla f(x^t)$ update happens at time t
- SDTP: $\Delta w^t = \beta\, \delta(t) \nabla V(w^t)$ t is the time the spike arrives at the synapse
- Some similarity, huh?
- Idea: What if we use an STDP feedforward net to optimize some objective function whose “local derivative” is ∇V?
- This idea is pursued in the paper; some learning can be done, but there are catches and different kinds of biological implausibility…

Biologically Plausible ANNs

- Several other versions of these ideas
- Also [Hopfield et al 2019 PNAS] *“Learning through plasticity and competition between neurons”*

Biologically Plausible ANNs: Some new ideas

- Dopaminergic NNs, see Yagishita et al, "A critical time window for dopamine actions on the structural plasticity of dendritic spines," *Science* 2014.
- What if every link of the ANN that fired is increased by, say α (1/4 – error^2)

Biologically Plausible ANNs: NNevolution

- G genes
- Each gene has two alleles, 0-1, iid binomial p_i
- Every genotype is a bitstring in $\{0, 1\}^G$
- The weight of each link L is a sparse linear function of the genes
- $\Sigma\ a_{Lj}\ x_j$ where each a_{Lj} is 0 with probability 1- ε and otherwise random in [-1, +1]

Biologically Plausible ANNs: NNevolution, the experiment

- Repeat for T generations:
  - Generate P genotypes and the corresponding ANNs
  - Evaluate each on a minibatch
  - Find the performance f(A) of each allele A (1/4 minus the average square error over all genotypes that have it)
  - Update the gene probabilities $p_A \rightarrow \sim p_A (1 + \varepsilon f(A))$