

Computational Neuroscience

How Your Brain Works

Prof. Jan Schnupp

wschnupp@cityu.edu.hk

HowYourBrainWorks.net



Two Sides to Neural Computation

- Use computers to model the brain.
(Part 1 of this lecture)
- Use the brain to inspire new types of computers.
(Part 2 of this lecture)

Part 1:

Using computers to model the brain:

- Simple versus complex models
- Very detailed models:
 - “NEURON” and the NMDL programming language
 - Henry Markram’s “Blue brain”
- Less detailed models
 - The “Izhikevich neuron”
 - Linear and LN receptive field models.

“Models” in Science

- **ALL scientific activity revolves around** the creation, testing, refinement, and debate of “**Scientific Models**”.
- From the Oxford English Dictionary, meaning #8a of the word ‘model’:

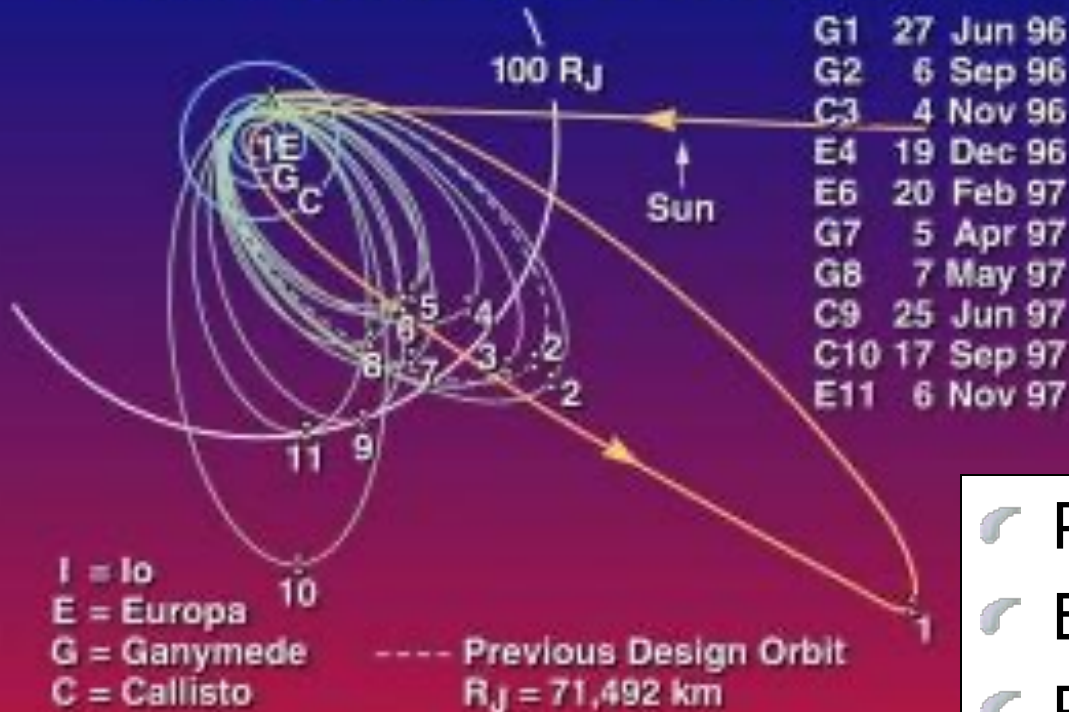
“A simplified or idealized description or conception of a particular system, situation, or process, often in mathematical terms, that is put forward as a basis for theoretical or empirical understanding, or for calculations, predictions, etc.;" (my emphasis)

- Scientific models are the fruit of creative thought of scientists. Therefore they are all **ultimately works of fiction**, and it is not always useful to think of them as “*true*” or “*false*”.
- However, unlike other works of fiction, scientific models are **expected to describe essential aspects of observable natural phenomena in a reliable manner**. Some models will live up to this expectation better than others.

Newtonian Mechanics

Galileo Space Probe

Orbital Tour of the Jupiter System



- Inertia:
 $F = m a$
- Gravity:
 $F = G(m_1 m_2)/r^2$

- ☞ Parsimony ++
- ☞ Explanatory Power ++
- ☞ Predictive Power +++
- ☞ Falsifiability +++

Newtonian Mechanics

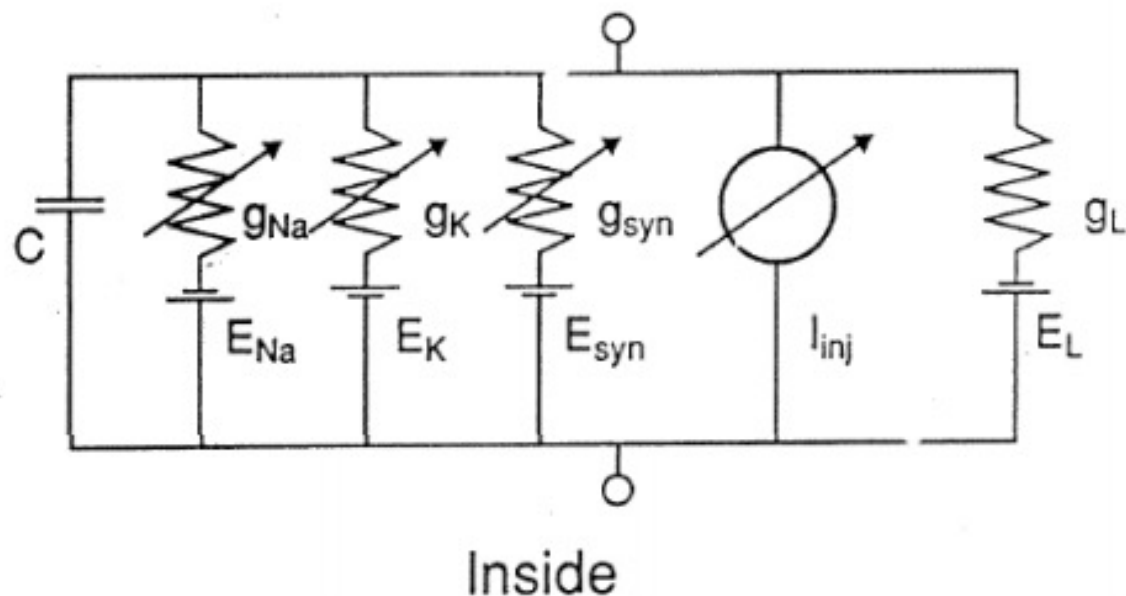
- Was able to use a small number of very simple mathematical equations to describe the movement of planets with remarkable accuracy.
- To do so, a lot of detail of the planets is ignored – they are just “point masses” (**parsimony**).
- The equations give good description of past movement of planets (**explanatory power**) and allows precise predictions of future movements (**predictive power** and hence **falsifiability**).
- Newton’s laws remain the most successful scientific model ever, even though, they are wrong. (Einstein’s relativity tells us that Newton's equations are only approximations that hold pretty well if speeds are small compared to speed of light).

Models of the Brain

- Like models in other scientific disciplines, models of the brain have to strike the right balance between simplicity (or parsimony) and accuracy of the explanations or predictions they can make.
- What level of detail is appropriate for a model depends on the question one is studying. There is no single correct answer.

Equivalent Electrical Circuit

- A patch of membrane is equivalent to an electrical circuit

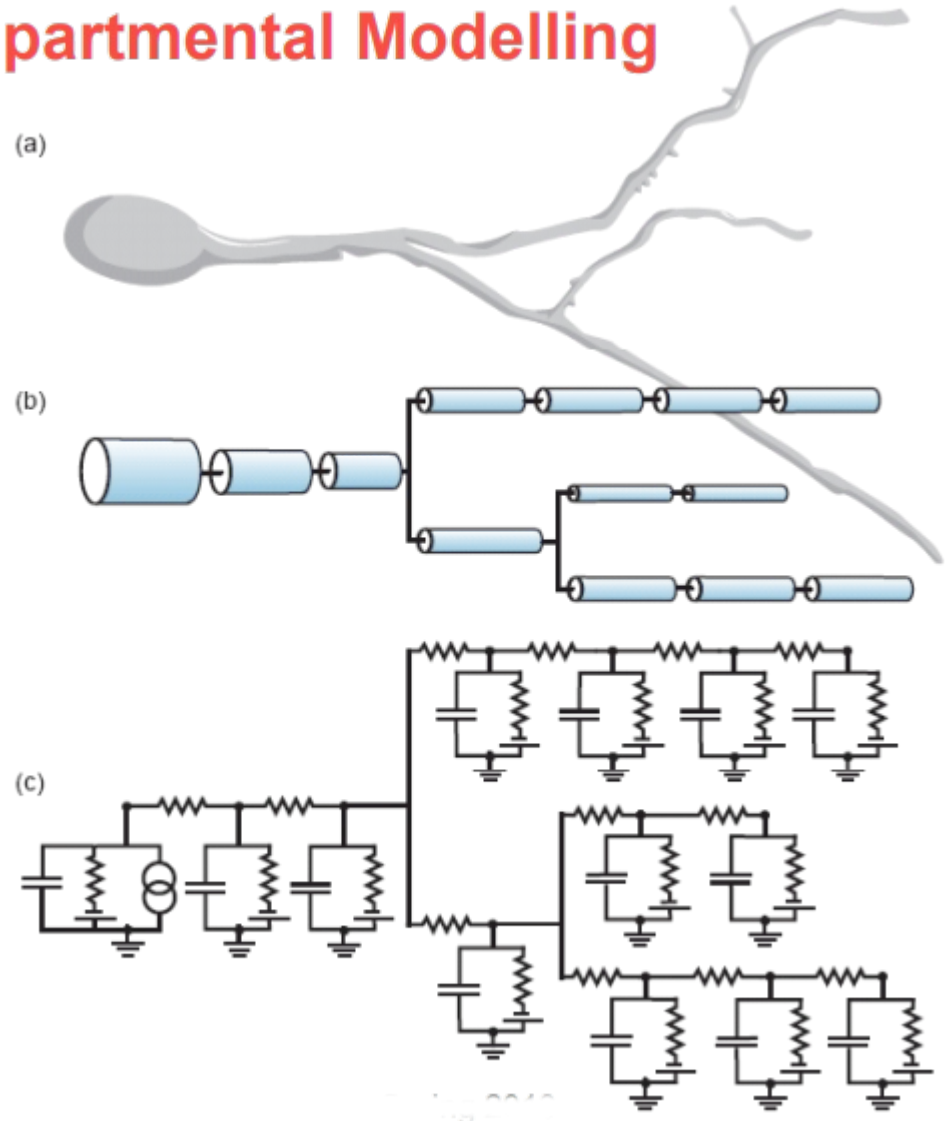


- This circuit can be described by a differential equation:

$$i_m(V, t) = (V - E_{Na})g_{Na} + (V - E_K)g_K + (V - E_{syn})g_{syn} + (V - E_L)g_L + I_{inj} + C \frac{dV}{dt}$$

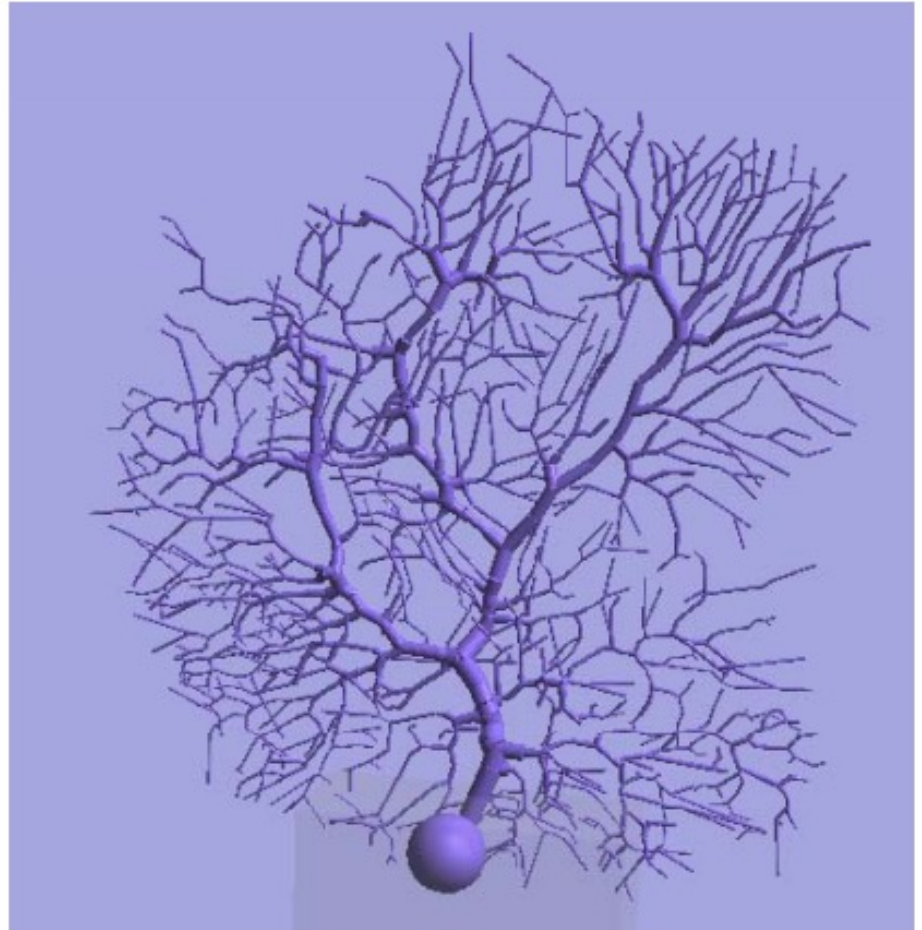
Compartmental Modelling

- **Compartmental models** simulate individual neurons as lots (sometimes thousands) of connected tubelets, and each has its equivalent circuit.



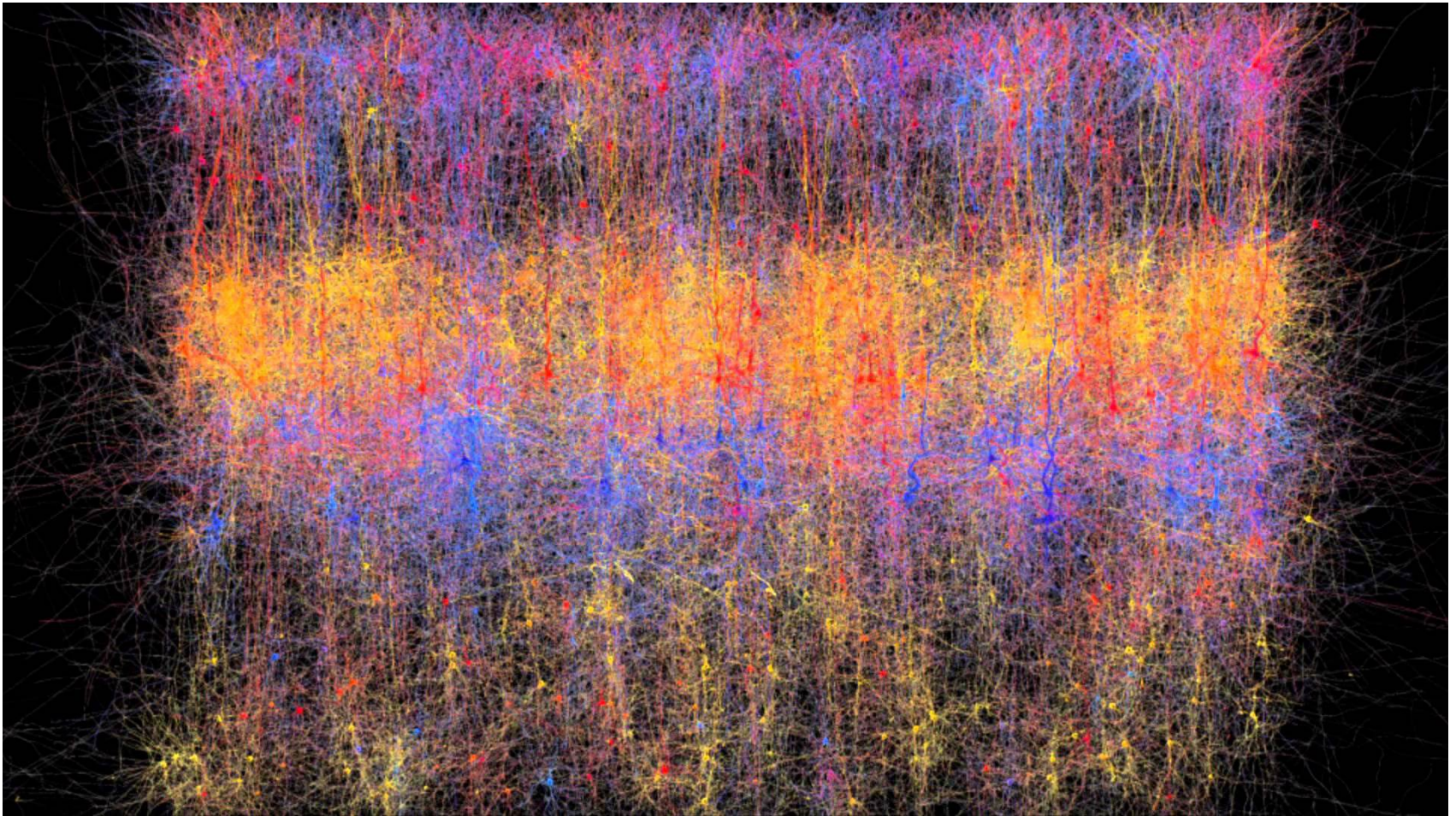
Large-scale Example

- Purkinje cell from cerebellum
- 4550 compartments
- 8021 ion channels
- 10 different voltage-dependent channels distributed across compartments
- 3500 synapses
- Reproduces electrical current injection and synaptic responses



(De Schutter and Bower, 1994)

A Very Large Scale Example: the “Blue Brain”



The Blue Brain Project

- Run at the Ecole Federale Polytechnique de Lausanne under the supervision of Prof Markram tries to model parts of the neocortex (columns) by running vast compartmental models on massive super computers.

Blue Brain Progress (From Wikipedia):

- By 2005, the first single cellular model was completed. The first artificial cellular neocortical column of 10,000 cells was built by 2008. By July 2011, a cellular mesocircuit of 100 neocortical columns with a million cells in total was built. A cellular rat brain is planned^[needs update] for 2014 with 100 mesocircuits totalling a hundred million cells. Finally a cellular human brain is predicted possible by 2023 equivalent to 1000 rat brains with a total of a hundred billion cells.^{[8][9]}
- Now that the column is finished, the project is currently busying itself with the publishing of initial results in scientific literature, and pursuing two separate goals:
- construction of a simulation on the *molecular level*,^[1] which is desirable since it allows studying the effects of [gene expression](#);
- simplification of the column simulation to allow for parallel simulation of large numbers of connected columns, with the ultimate goal of simulating a whole neocortex (which in humans consists of about 1 million cortical columns).
- In 2015, scientists at École Polytechnique Fédérale de Lausanne (EPFL) developed a quantitative model of the previously unknown relationship between the glial cell astrocytes and neurons. This model describes the energy management of the brain through the function of the neuro-glial vascular unit (NGV). The additional layer of neuron-glial cells is being added to Blue Brain Project models to improve functionality of the system.^[10]

Blue Brain Criticism

- The Blue Brain project appears to be a modelling effort on a vast scale which completely ignores key insights into successful models: make the model “parsimonious” by leaving out as much irrelevant detail as possible.
- There isn't even a method for trying to decide what details are relevant and which ones are not because it is not clear what aspects of brain function the blue brain is supposed to model.
- It is unsurprising that the Blue Brain has yielded very poor returns on investment.

Vastly Simpler (and Arguably More Successful) Models

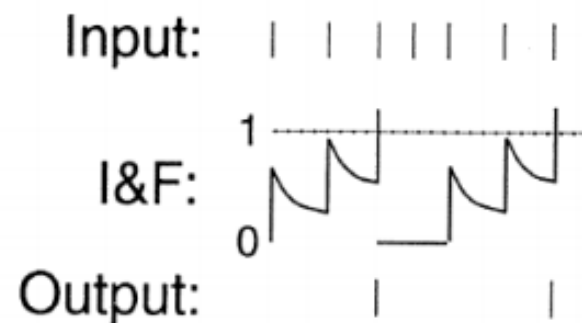
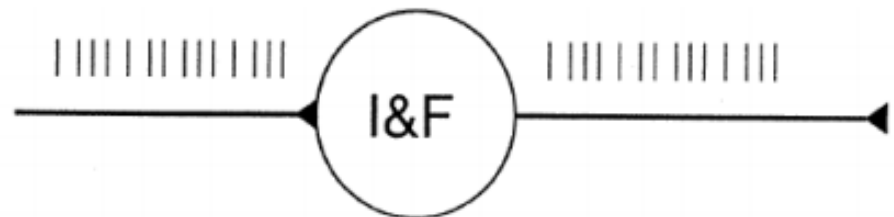
- Spiking Models:
 - Integrate and Fire
 - The “Izhikevich” Neuron
- Non-Spiking (Rate Based) Models:
 - Linear Neurons
 - Linear Threshold or Linear-Non Linear Neurons.

What sort of “computers” are neurons?

- Neurons are cells which, if you look carefully, look staggeringly complex, with hundreds of thousands of ion channels which may or may not be gated by dozens of different mechanisms, spread over far-flung dendritic and axonal arbours.
- Nevertheless, it can be very useful to think of neurons “in a first order approximation” as carrying out a very simple “computation”:
 - Take all your synaptic inputs (many of which may be zero most of the time).
 - Multiply each input by its “synaptic weight”.
 - Sum.
 - Use a threshold criterion or a linear or sigmoidal function to decide how much output to send given the weighted sum of inputs.

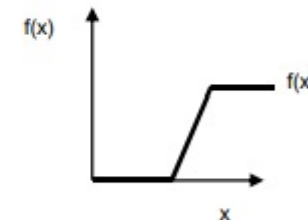
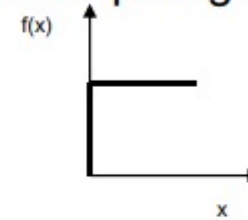
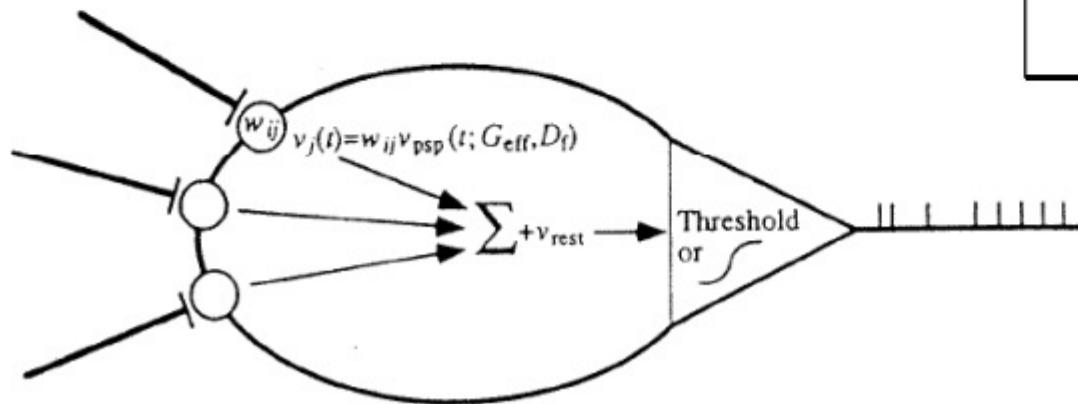
Simplified Models of Spiking Neurons

- Major property of biological neurons is firing of action potentials in response to synaptic input
- Essential features of this include:
 - Action potential – shape is same each time, so can just treat the occurrence of an AP (spike) as a binary 1
 - All EPSPs and IPSPs eventually get summed in the cell body, so do not need to model dendrites
- “Integrate-and-fire” model
 - Point neuron
 - Resistance + capacitance
 - Synaptic input



Rate-based Neuron Models

- Computing units used in ANNs can be regarded as models of biological neurons
- Activation function captures essence of a neuron's spiking response
 - Binary (0 or 1): neuron is silent or firing
 - McCulloch-Pitts threshold logic unit
 - Analog: number represents neuron firing rate
 - Piece-wise linear or sigmoid



- Inputs are weighted and summed to calculate the “activation”.
Output = $f(\text{activation})$ where ***f*** could be a ***sigmoid*** or a ***threshold*** or a threshold linear function ***or just a linear function***.

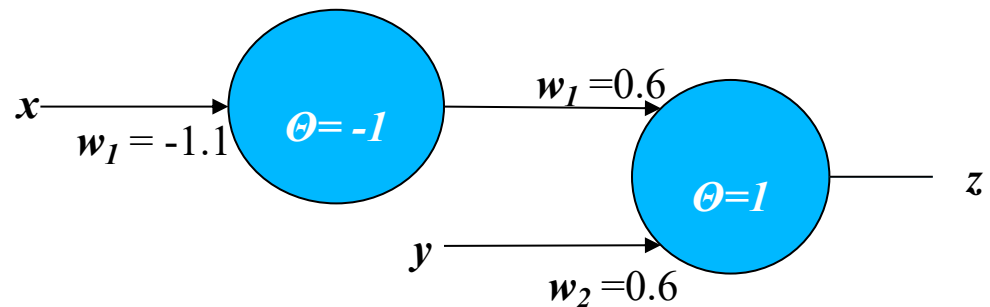
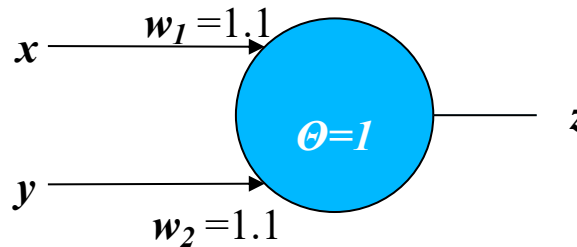
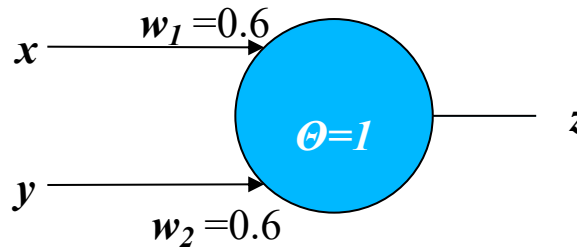
Binary Threshold (McCullough-Pitts) Neurons

- Perhaps the simplest possible neuron model.
- Receive inputs from synapses that have positive or negative weights w_i (excitatory or inhibitory synapses).
- Their “activation” is the weighted sum of inputs.
- Produce a binary output of 0 or 1 depending on whether the activation is greater than a given threshold θ .
- This is roughly equivalent to real, biological neurons, which summate synaptic currents in their dendrites and soma, and fire an action potential or not depending on whether the depolarization is enough to open voltage gated sodium channels in axons.

$$\text{output} = \begin{cases} 0 & \text{if } \sum_j w_j x_j \leq \text{threshold} \\ 1 & \text{if } \sum_j w_j x_j > \text{threshold} \end{cases}$$

Binary Threshold Neurons as “Logic Gates”

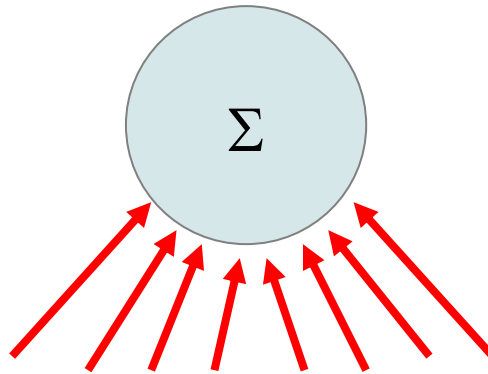
- A binary threshold neuron computing the logic function
“ $z = x \text{ AND } y$ ”
- A binary threshold neuron computing the logic function
“ $z = x \text{ OR } y$ ”
- A network computing the function
“ $z = (\text{NOT } x) \text{ AND } y$ ”



By setting the synaptic weights on networks of binary threshold neurons appropriately it is in principle possible to make these networks compute any function you like. Large numbers of binary threshold units could be used to build a “universal computer”.

“Linear” and “Linear-NonLinear” (LN) Neurons

- Weight and sum inputs in the same way binary threshold neurons do to compute activation.
- Produce a **real valued output** (their “firing rate”) which is **either just proportional to the activation (linear) or is a non-linear, often sigmoidal, function of the activation.**

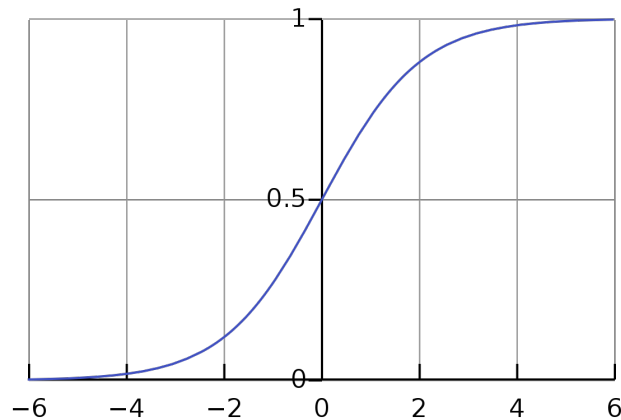


$$r = f(i_1 w_1 + i_2 w_2 + i_3 w_3 \dots)$$

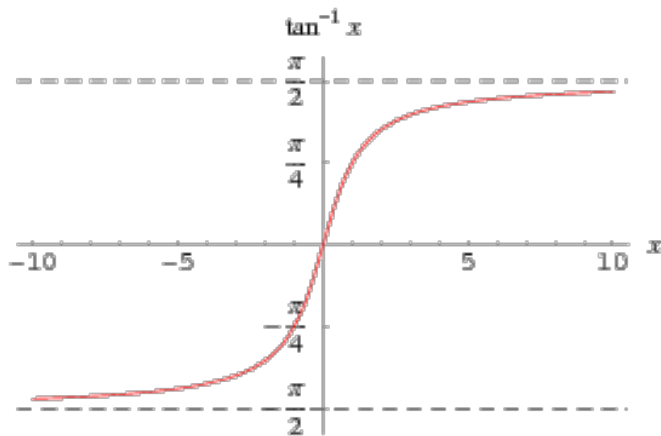
$$r = f(\mathbf{i} \cdot \mathbf{w}^T)$$

Sigmoids

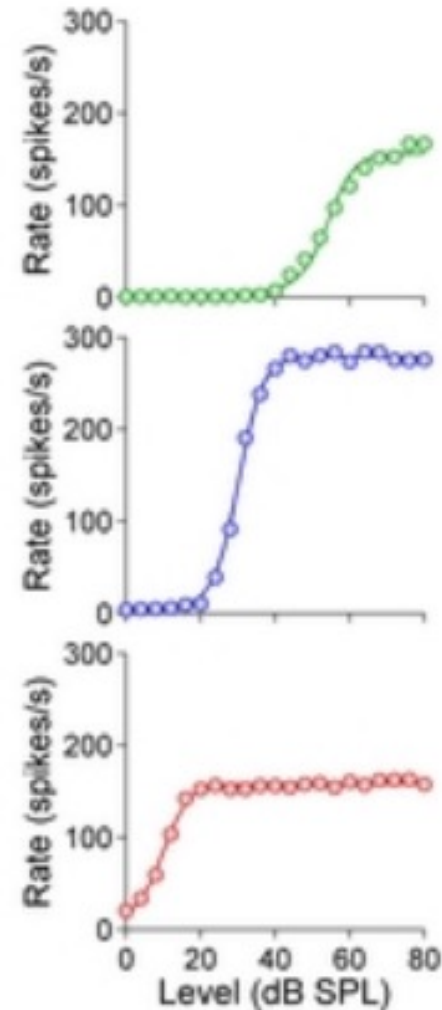
logistic function $\sigma(z) \equiv \frac{1}{1 + e^{-z}}$.



arctan function



Auditory nerve fiber
rate-level functions



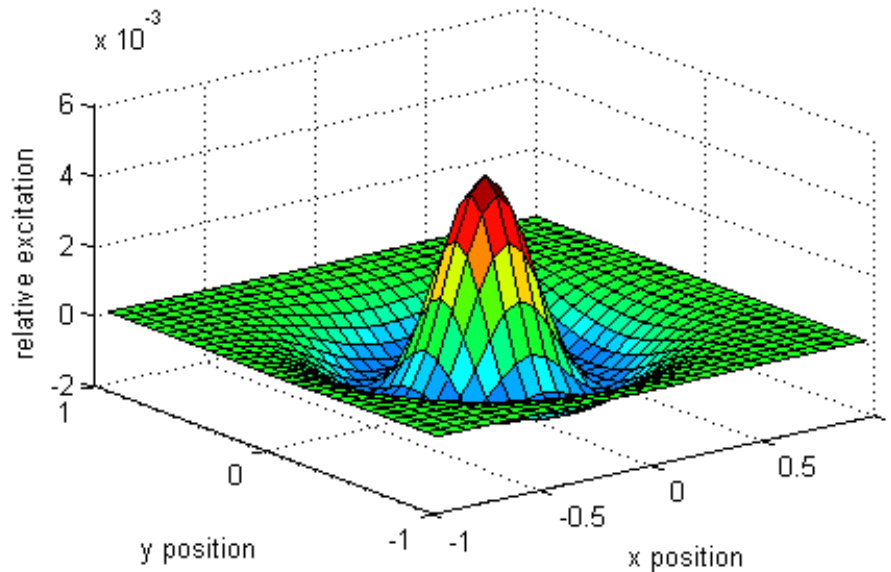
Neurons vs. Statistical Modelling

- For the statistics geeks among you, you might find it useful to think of a linear neuron as a device for computing a linear multiple regression, and an LN neuron as a device for calculating a logistic regression.
- If you aren't a statistics geek, don't worry about it.

Receptive Field Models

- Sometimes one can model the input-output relationships of sensory neurons simply as a “weight matrix” which says what the synaptic weight of each given “spatial” or “auditory frequency channel” would be.
- Examples:
 - center-surround receptive fields in retina or LGN
 - Simple cell receptive fields in V1
 - Spectro-temporal receptive fields in auditory structures
- Such linear or LN models of sensory neurons make it very easy to predict what the response to any arbitrary stimulus should be (-> predictive power and falsifiability).

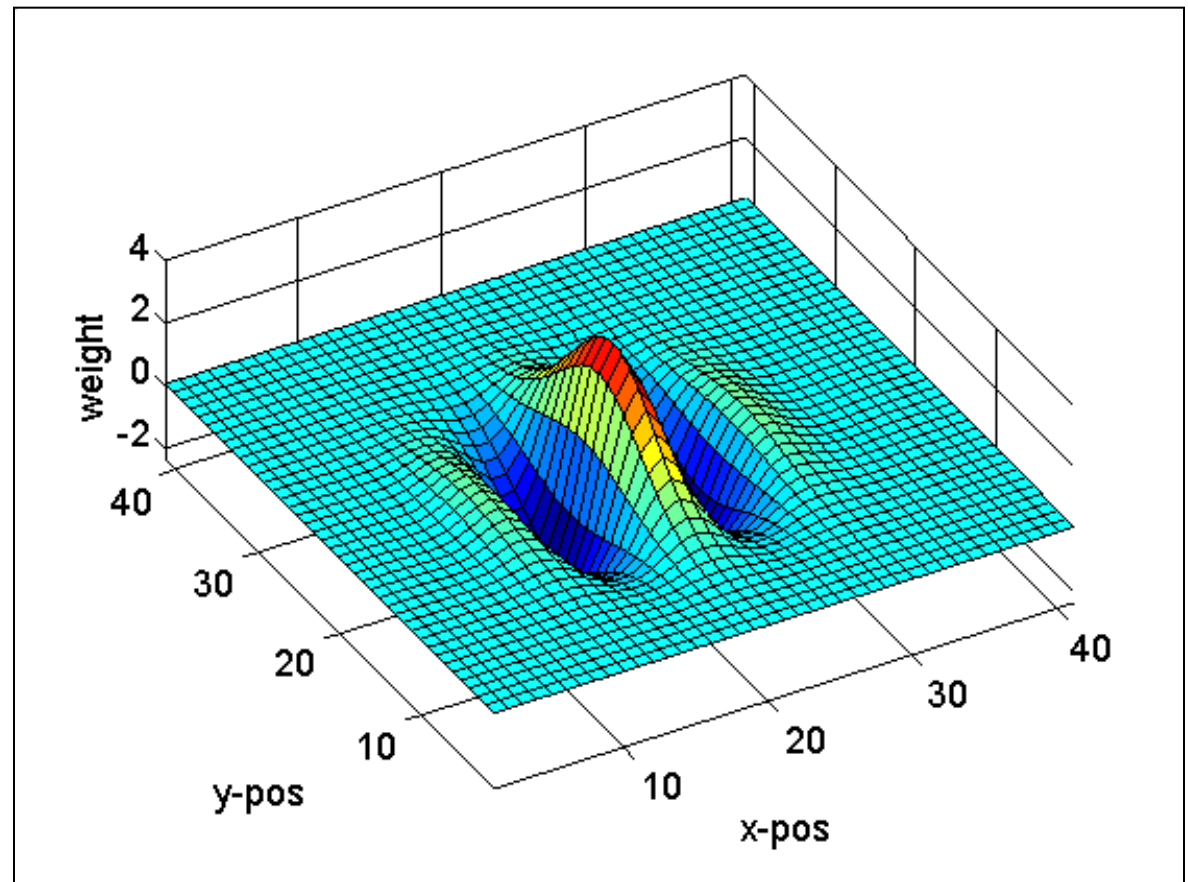
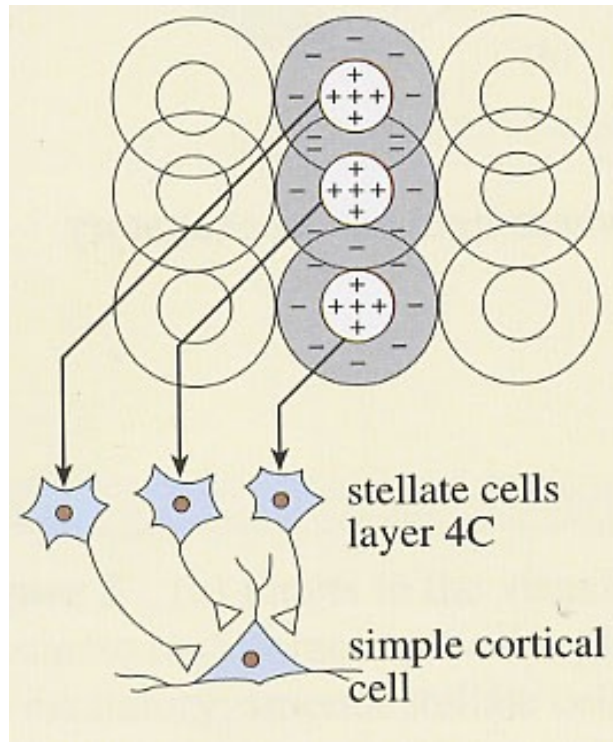
Difference of Gaussians (DoG) Model of Retinal Ganglion Cell Receptive Fields



- DoG models posit that the “weight” of a spot of light on the retina as a function of distance from the receptive field center is given by a broad inhibitory (negative, surround) bell curve and a sharp excitatory (positive, center) bell curve.
- The centre-surround structure of Retinal Ganglion Cells turns them into “spatial frequency filters”. Larger RGC receptive fields are tuned to “coarsely grained” structure in the visual scene, while smaller RFs are tuned to fine grain structure.

The Gabor Filter Model of V1 Simple Cells

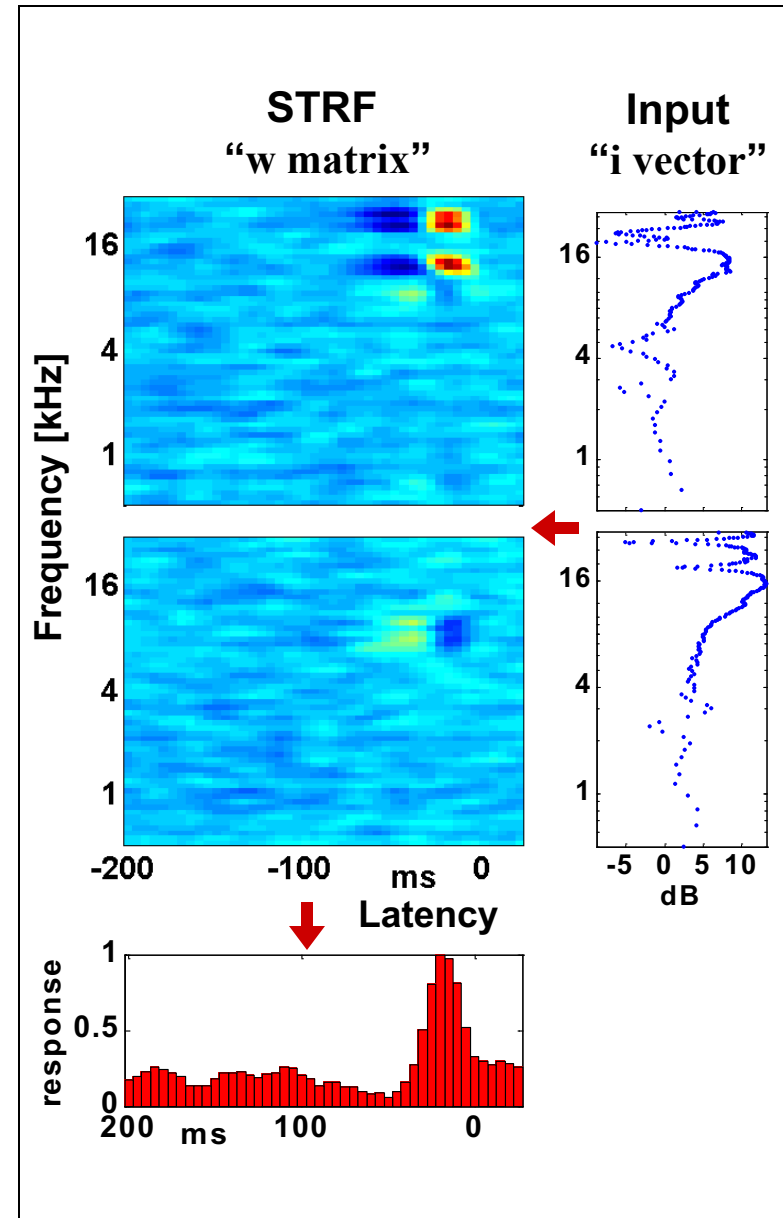
Retina->LGN->V1 simple cell: linear



Binaural STRFs in A1

$$\begin{aligned}
 r(t) = & i_1(t-\tau_1) \cdot w_1(\tau_1) + i_1(t-\tau_2) \cdot w_1(\tau_2) + \dots \\
 & + i_2(t-\tau_1) \cdot w_2(\tau_1) + i_2(t-\tau_2) \cdot w_2(\tau_2) + \dots \\
 & + i_3(t-\tau_1) \cdot w_3(\tau_1) + i_2(t-\tau_2) \cdot w_3(\tau_2) + \dots
 \end{aligned}$$

Schnupp et al, Nature (2001)



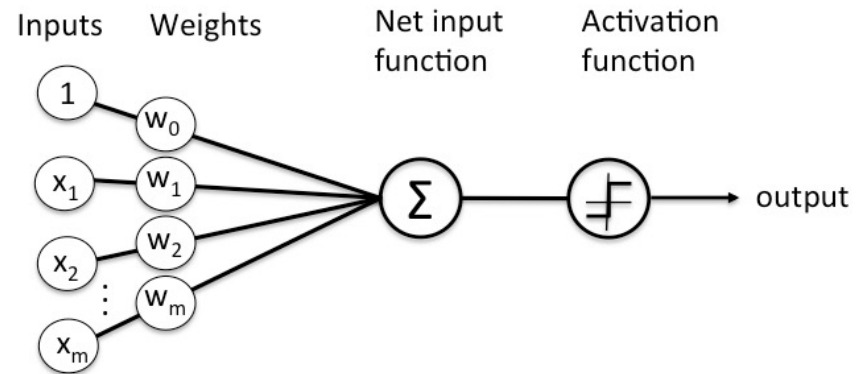
Break

Part 2:

Using the brain to inspire new types of computers:

- The perceptron
- Multilayer perceptrons
- “Deep” neural networks
- The perceptron learning rule and error backpropagation
- Unsupervised learning
- Reinforcement learning, TD Gammon
- Recurrent artificial neural networks.

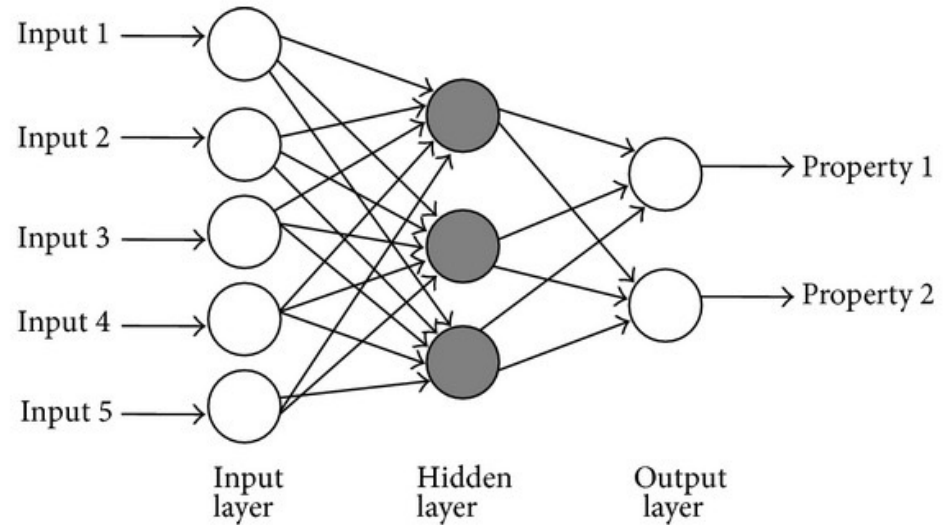
Rosenblatt's Perceptron



Schematic of Rosenblatt's perceptron.

- The first “artificial neural net”, invented in 1958, basically just a single binary threshold neuron.
- Used as a “classifier” (output goes high if inputs belong to class one, low for class two)
- Weights are set through training by (thousands of) examples.
 - If the desired output for a given input set is high, then increase the weight of those inputs that were also high and decrease those that were low.
 - Do the converse for low inputs.
 - A form of “Associative Learning”
- A single layer perceptron cannot learn functions like “exclusive or” (XOR). But multilayer perceptrons can in principle compute any arbitrary function.

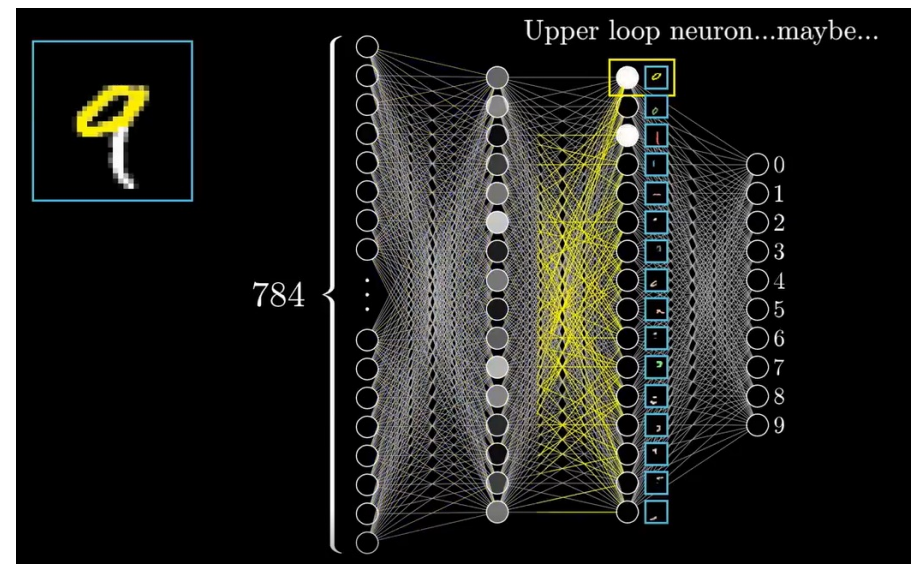
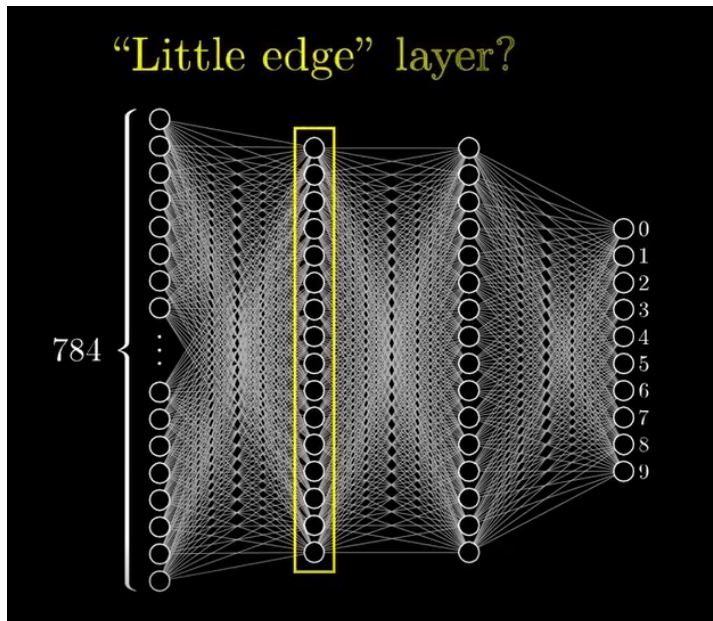
Multilayer Perceptrons



- Have one or more “hidden layers” of artificial neurons between input and output.
- Rosenblatt’s perceptron uses a threshold function to compute binary outputs. Multilayer perceptrons use sigmoids (arctan or logistic).
- Weights are set still set through training and associative learning, but errors during training stages must be “backpropagated” to adjust the weights of the hidden layer units.
- Training is often slow. Some neuroscientists bemoan that there is no biological evidence that real brains use anything like backpropagation. But there is in principle no input-output mapping that a sufficiently big multilayer perceptron could not eventually learn.

“Deep” Neural Networks

- “Deep” neural networks are often just multilayer perceptrons with more than one hidden layer.
- After training, neurons in subsequent layers may become sensitive to “features” of ever greater complexity and abstraction.
- ThreeBlueOneBrown example:
 - First hidden layer ~ V1: sensitive to little edges
 - Second hidden layer ~ V2: sensitive to more complex shapes
 - In both real and artificial neural nets, the features the hidden neurons are actually sensitive to may be obscure.

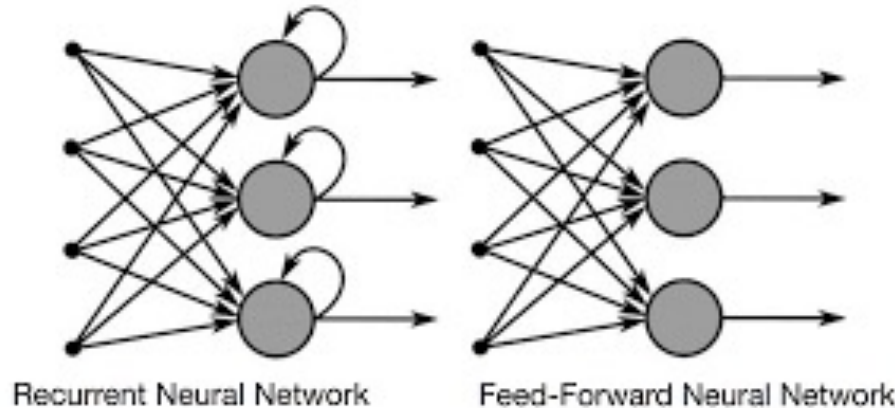


“Artistic Style Transfer”



- To allow techniques such as image search on large internet image databases, scientists have developed large deep neural networks for image classification. These often have a hierarchy of layers sensitive to high level features. Very roughly, you can think some of these as being roughly equivalent to various parts of the ventral “what” stream and the dorsal “where” stream or color areas of extrastriate visual areas.
- Combine activity in “what” and “where” parts of these networks in response to image 1 with the activity in the “color” parts in response to image 2, and you get the network “hallucinating” image 1 in the color palette and style of image 2.
- Trace back from the high levels all the way to the input layer to reconstruct the “hallucinated” image.
- <https://medium.com/data-science-group-iitr/artistic-style-transfer-with-convolutional-neural-network-7ce2476039fd>

Recurrent Artificial Neural Networks



- Blog: “The Unreasonable Effectiveness of Recurrent Neural Networks” by Andrej Karpathy
- <https://karpathy.github.io/2015/05/21/rnn-effectiveness>

Shakespeare

“Measure for Measure”

(written 1604)

Act 2 Scene 2

ANGELO:

You're welcome: what's your will?

ISABELLA:

I am a woeful suitor to your honour,
Please but your honour hear me.

ANGELO:

Well; what's your suit?

ISABELLA:

There is a vice that most I do abhor,
And most desire should meet the blow of
justice;
For which I would not plead, but that I must;
For which I must not plead, but that I am
At war 'twixt will and will not.

ANGELO:

Well; the matter?

ISABELLA:

I have a brother is condemn'd to die:
I do beseech you, let it be his fault,
And not my brother.

Provost[Aside]:

Heaven give thee moving graces!

Artificial Shakespeare

A 3 layer RNN
with only 512
hidden neurons
trained on a
4.4MB text file
containing all the
collected works
of Shakespeare
produced this
output.

PANDARUS:

Alas, I think he shall be come approached and the day
When little strain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:

They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:

Well, your wit is in the care of side and that.

Second Lord:

They would be ruled after this chamber, and
my fair nudes begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:

Come, sir, I will make did behold your worship.

VIOLA:

I'll drink it.

How do Artificial RNNs process speech?

- Hard to know for sure, but neurons within the “hidden” layer appear to develop a sensitivity to “features”:
- In an RNN trained on Wikipedia articles, one neuron would fire more and more as sentences got longer.
- Another neuron would fire inside a quote, but not outside.
- The “features” that most hidden neurons are sensitive to are decidedly unclear though.

Training Multilayer Perceptrons: “Supervised learning”

- Multilayer perceptrons need to be trained on very large datasets, comprising tens of thousands to millions of examples, and for each example the desired, “correct” answer must be known.
- The synaptic weights of the network are adjusted gradually to make the network output more similar to the correct answers in the training set.
- “Errors” (differences between current and correct output) must “back-propagate” through the layers of the network.
- One criticism of this type of artificial neural networks is that real brains do not learn like this. There is no known biological mechanism for backprop nor do our teachers give us tens of thousands of examples.

Unsupervised Learning

- It is possible to set up neural networks so that they can learn some types of things even without a “teacher” input which tells them “correct” answers. For example, such networks might discover natural “clustering” of certain features. Imagine a network shown thousands of pictures of either forest scenes or seaside scenes. Just by associative, Hebbian learning it might discover that, in that dataset, images come in two types, some with lots of green and brown and some with lots of blue horizons.
- Unsupervised learning is perhaps more biological.

Reinforcement learning

- In reinforcement learning, a network is not told what the output for a certain input should be, but it is simply “rewarded” for “good” outputs or “punished” for “bad outputs”.
- Such networks will try to “predict” which outputs (“behaviours”) lead to maximum reward and minimum punishment.
- Some parts of the brain, which we will study in later lectures, are thought to implement reinforcement learning, with levels of the neurotransmitter dopamine representing “reward prediction error”

Reinforcement learning example: TD Gammon

[Tesauro, 1995]

Learn to play Backgammon.

Immediate reward:

- +100 if win
- -100 if lose
- 0 for all other states

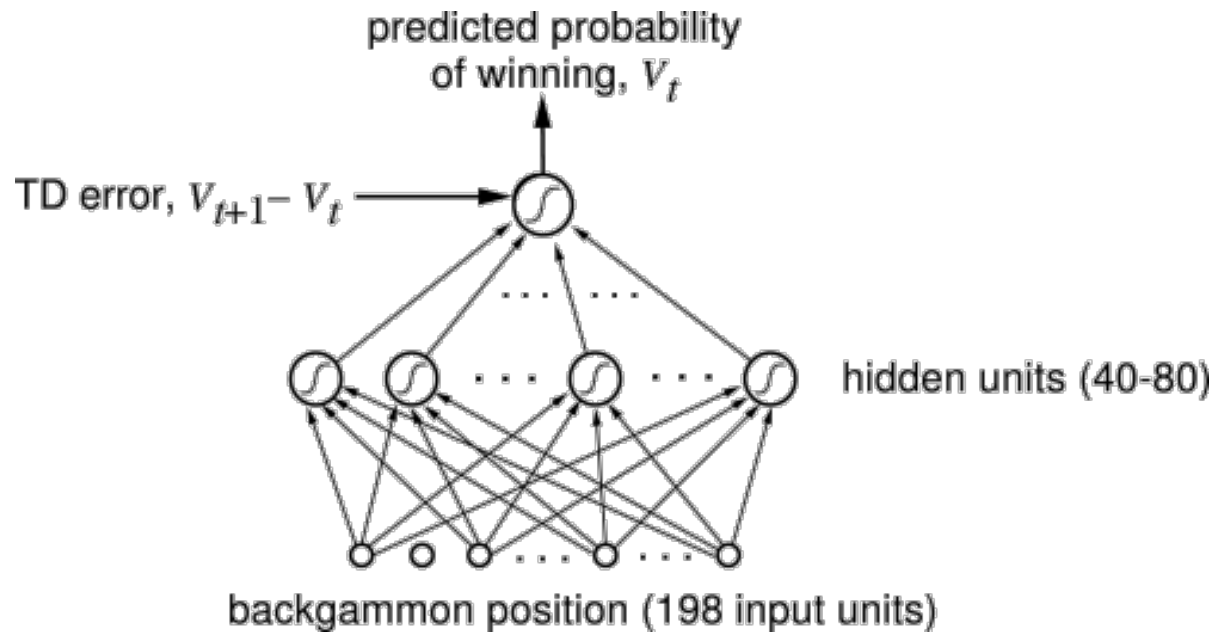


Trained by playing 1.5 million games against itself.

Now approximately equal to best human player.

- TD stands for “temporal difference” learning.

TD Gammon network architecture



- TD gammon is a very modestly sized network of less than 300 artificial neurons.
- Nevertheless it can beat most human players at backgammon.
- Much larger networks, such as AlphaGo, can now beat humans at much more complex games, such as Go.

Summary

- Neural Networks, whether biological or artificial, compute by combining weighted inputs to calculate outputs.
- Networks must often be “trained” by adjusting the weights to produce useful outputs. Different types of training schemes: supervised, unsupervised, reinforcement.
- The trained weight matrix is a form of “long term memory”.
- Recurrent neural networks also have “short term memory” through activity running in loops.
- Subsequent layers of the network tend to become sensitive to increasingly higher order features of the inputs.
- Artificial NNs can be used to remarkable effect. EG: <https://www.youtube.com/watch?v=DgPaCWJL7XI>