Artificial Neural Networks: Lecture 1

Introduction to the field + Simple Perceptrons for Classification

Wulfram Gerstner
EPFL, Lausanne, Switzerland

Objectives for today:

- supervised learning vs. reinforcement learning
- understand classification as a geometrical problem
- discriminant function of classification
- linear versus nonlinear discriminant function
- perceptron algorithm
- gradient descent for simple perceptrons

Reading for this week:

Bishop, Ch. 4.1.7 of

Pattern recognition and Machine Learning

or

Bishop, Ch. 3.1-3.5 of

Neural networks for pattern recognition

Motivational background reading:

Silver et al. 2017, Archive

Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm

Goodfellow et al., Ch. 1 of

Deep Learning

Artificial Neural Networks

Wulfram Gerstner

EPFL, Lausanne, Switzerland

1. Simple perceptrons for classification
2. Reinforcement learning1: Bellman and SARSA
3. Reinforcement learning2: variants of SARSA
4. Reinforcement learning3: Policy Gradient
5. Deep Networks1: Backprop and multilayer perceptron
6. Deep Networks2: Statistical Classification by deep networks
7. Deep Networks3: regularization and tricks of the trade *Miniproject handout*
8. Deep Networks4: Convolutional networks
9. Deep Networks5: Error landscape and optimization methods
10. Deep Reinforcement learning1
11. Deep Reinforcement learning2
12. Deep Reinforcement learning3
13. Sequence predictions and Recurrent networks

Previous 3 slides.
Every week the first two slides contain the contents and main objectives of the day.

Artificial Neural Networks

Wulfram Gerstner

EPFL, Lausanne, Switzerland

Previous slide.

Results with artificial neural networks are discussed in newspaper articles and have inspired people around the world.
These years we experience the third wave of neural networks.
The first wave happened in the 1950s with the first simple computer models of neural networks, with McCulloch and Pitt and Rosenblatt's Perceptron. There was a lot of enthusiasm, and then it died.
The second wave happened in the 1980, around the Hopfield model, the BackPropagation algorithm, and the ideas of 'parallel distributed processing'. It died in the mid-nineties when statistical methods and Support Vector Machines took over.

The third wave started around 2012 with larger neural networks trained on GPUs using data from big image data bases. These neural networks were able to beat the benchmarks of Computer vision and have been called 'deep networks'.

Artificial Neural Networks, how they work, and what they can do, will be in the focus of this lecture series.

Artificial Neural Networks
Introduction to the field

Wulfram Gerstner
EPFL, Lausanne, Switzerland

1. From Biological to Artificial Neurons

Previous slide.

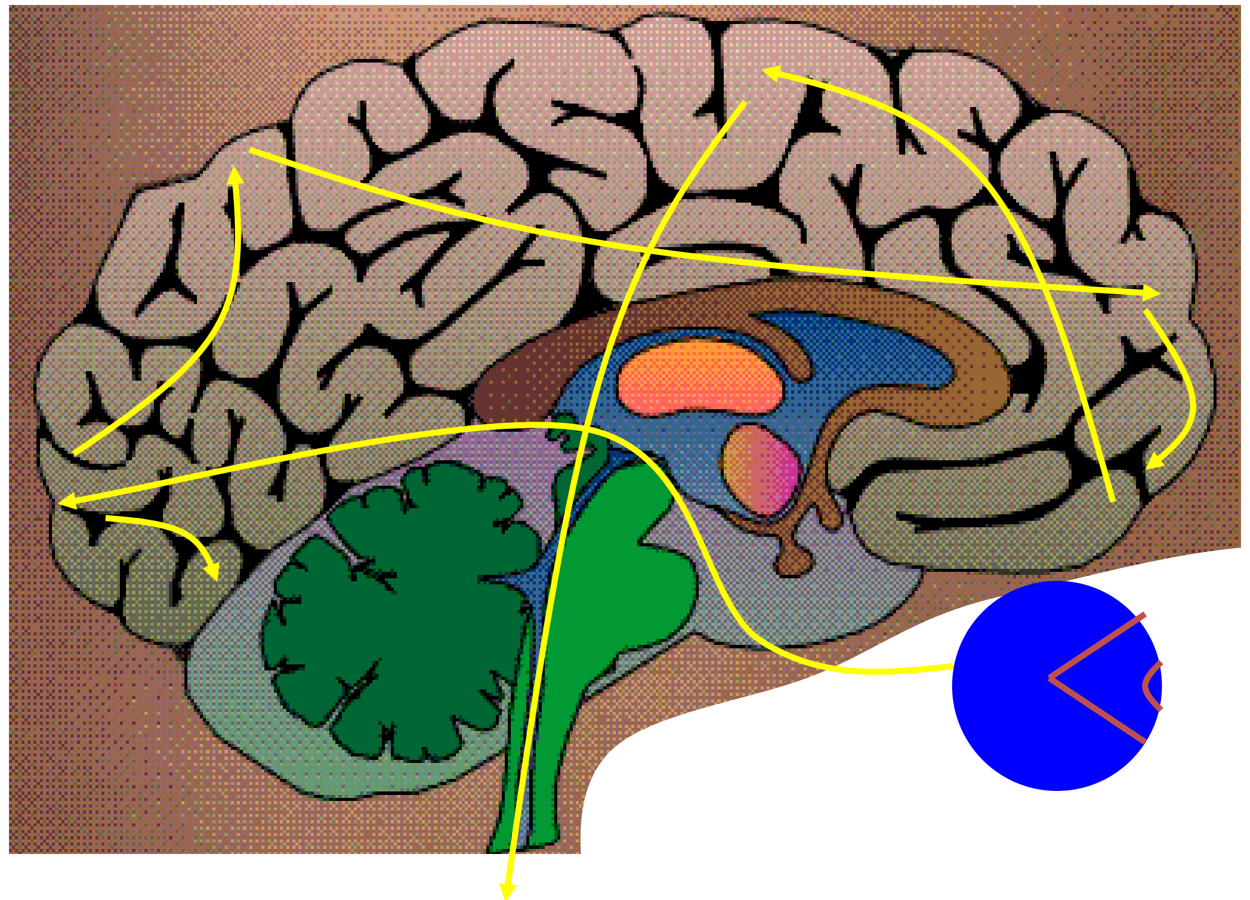
In this first lecture the focus is on two things: a general introduction into the field (with its subparts Reinforcement learning and Supervised Learning for Classification) as well as learning in a first classic model called the simple perceptron.

We start with a glimpse of the biological inspirations of the field.

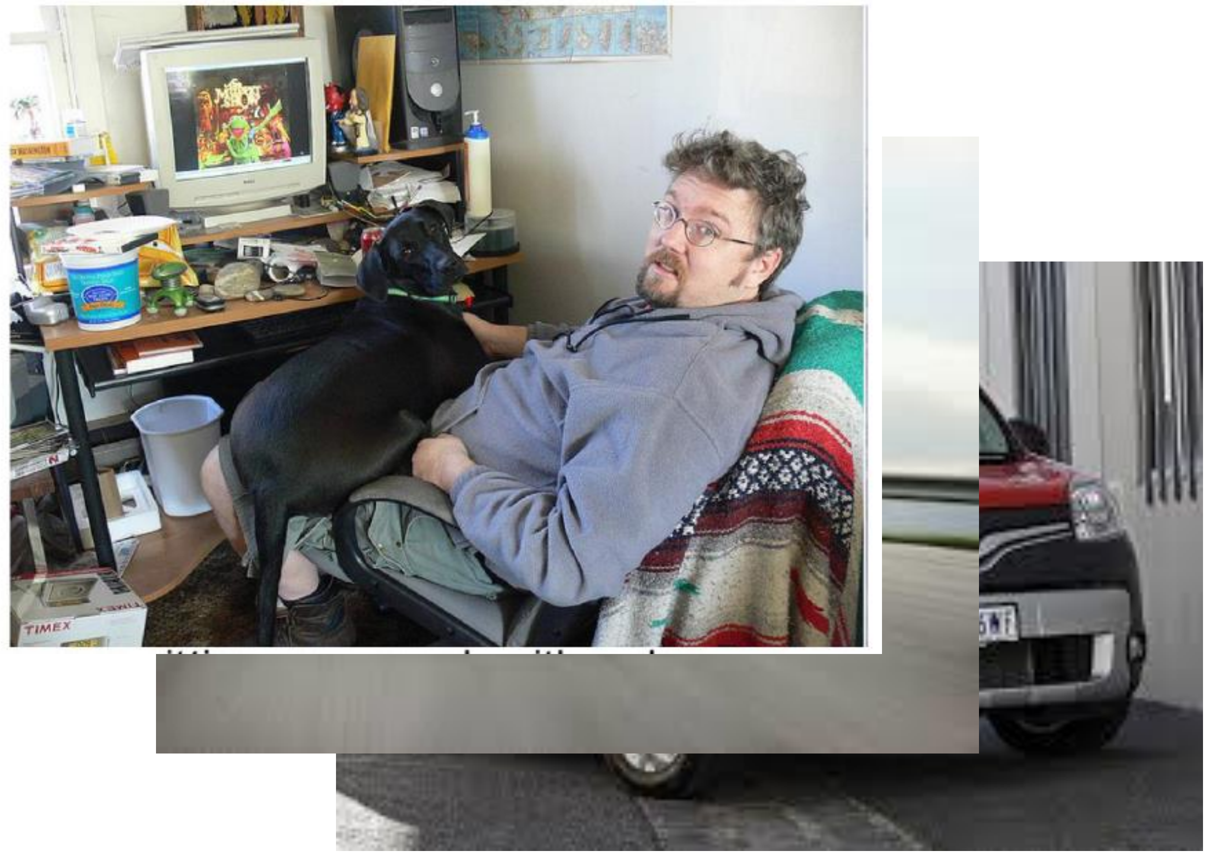
The brain: Cortical Areas

motor
cortex

visual
cortex



frontal cortex

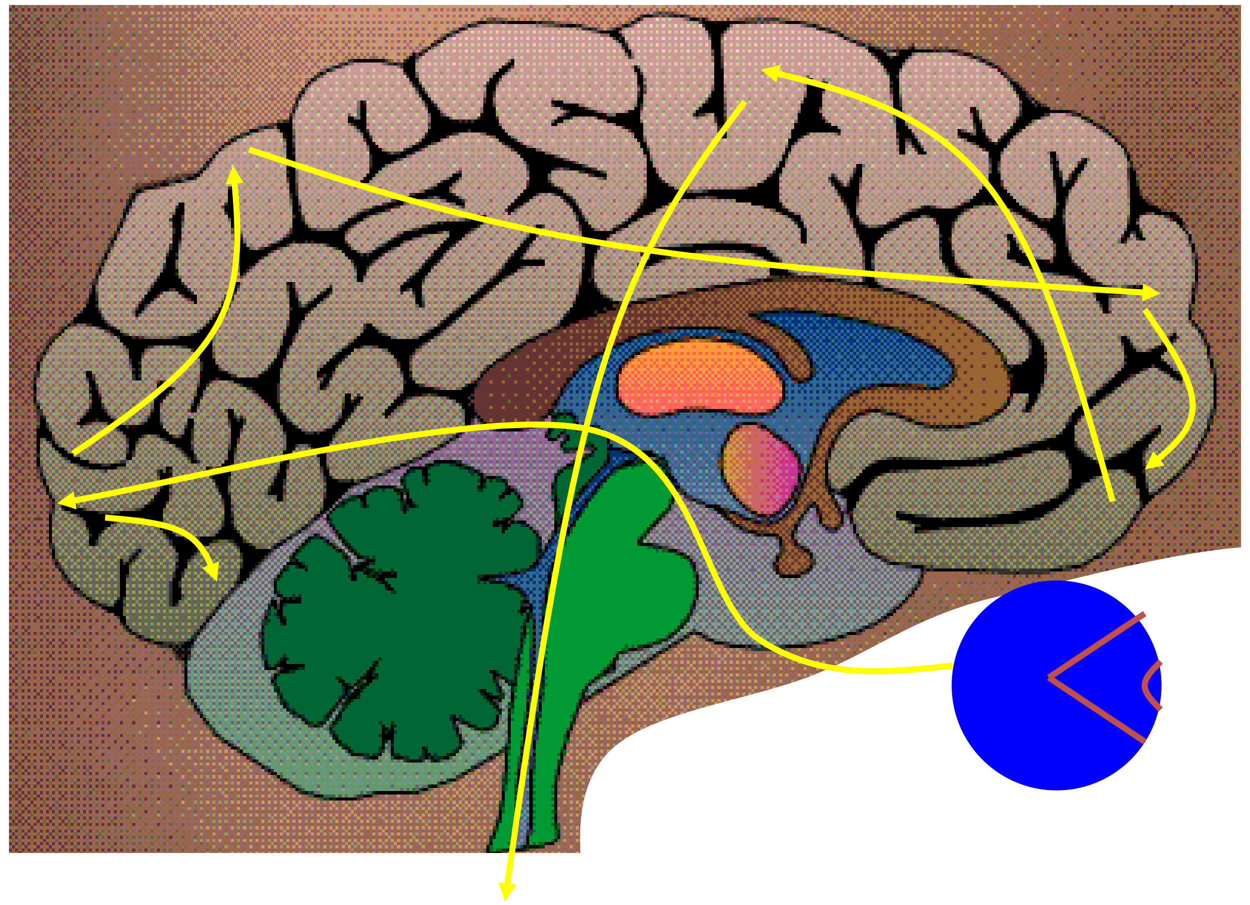


to
muscles

Previous slide.
During all these waves, during 60 years of research, artificial neural networks researchers worked on building intelligent machines that learn, the way humans learn. And for that they took inspiration from the brain.

Suppose you look at an image. Information enters through the eye and then goes to the cortex.

The brain: Cortical Areas

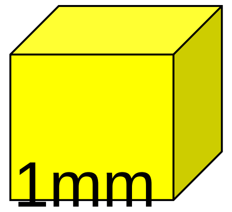


Previous slide.
Cortex is divided into different areas:
Information from the eye will first arrive at visual cortex (at the back of the head), and from there it goes on to other areas. Comparison of the input with memory is thought to happen in the frontal area (above the eyes). Movements of the arms a re controlled by motor cortex somewhere above your ears.

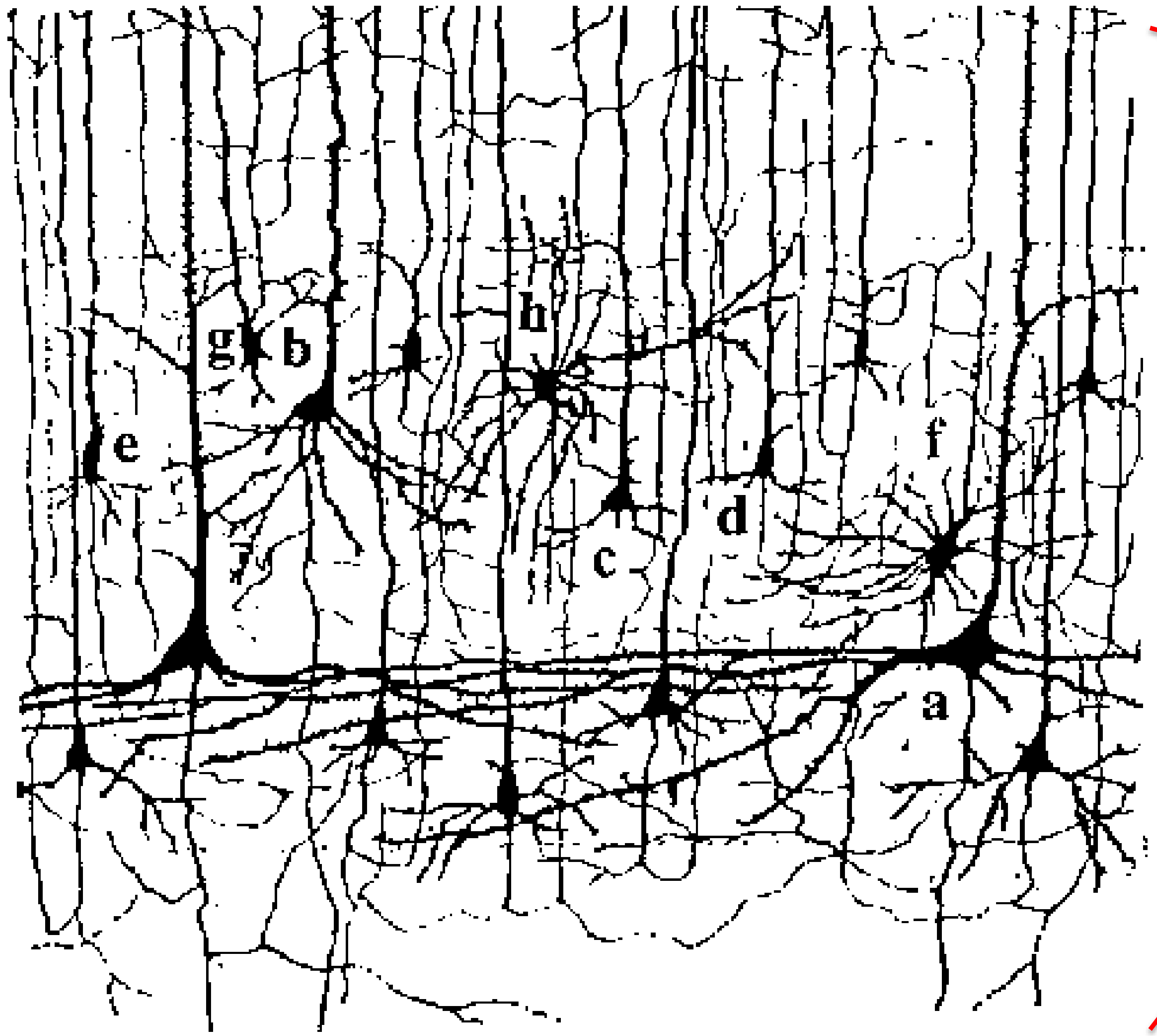
Talking about cortical areas provides a **macroscopic** view of the brain.

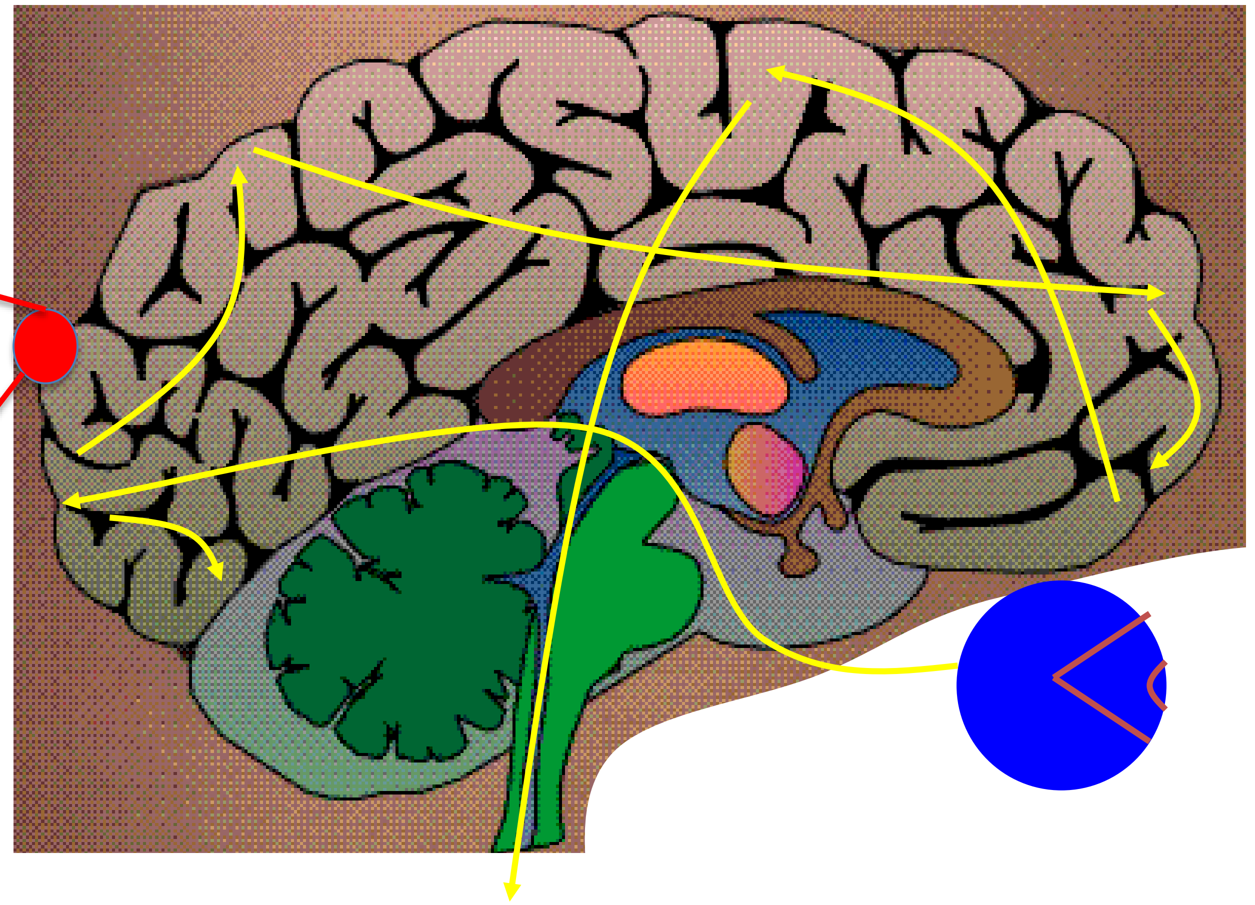
The Brain: zooming in

1mm



10 000 neurons
3 km of wire





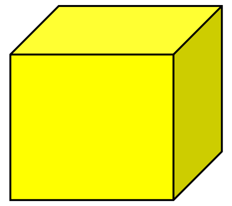
Ramon y Cajal

Previous slide.
If we zoom in and look at one cubic millimeter of cortical material under the microscope, we see a network of cells.
Each cell has long wire-like extensions.
If we counted all the cells in one cubic millimeter, we would get numbers in the range of ten thousand.
Researchers have estimated that, if you put all the wires you find in one cubic millimeter together you would find several kilometers of wire.

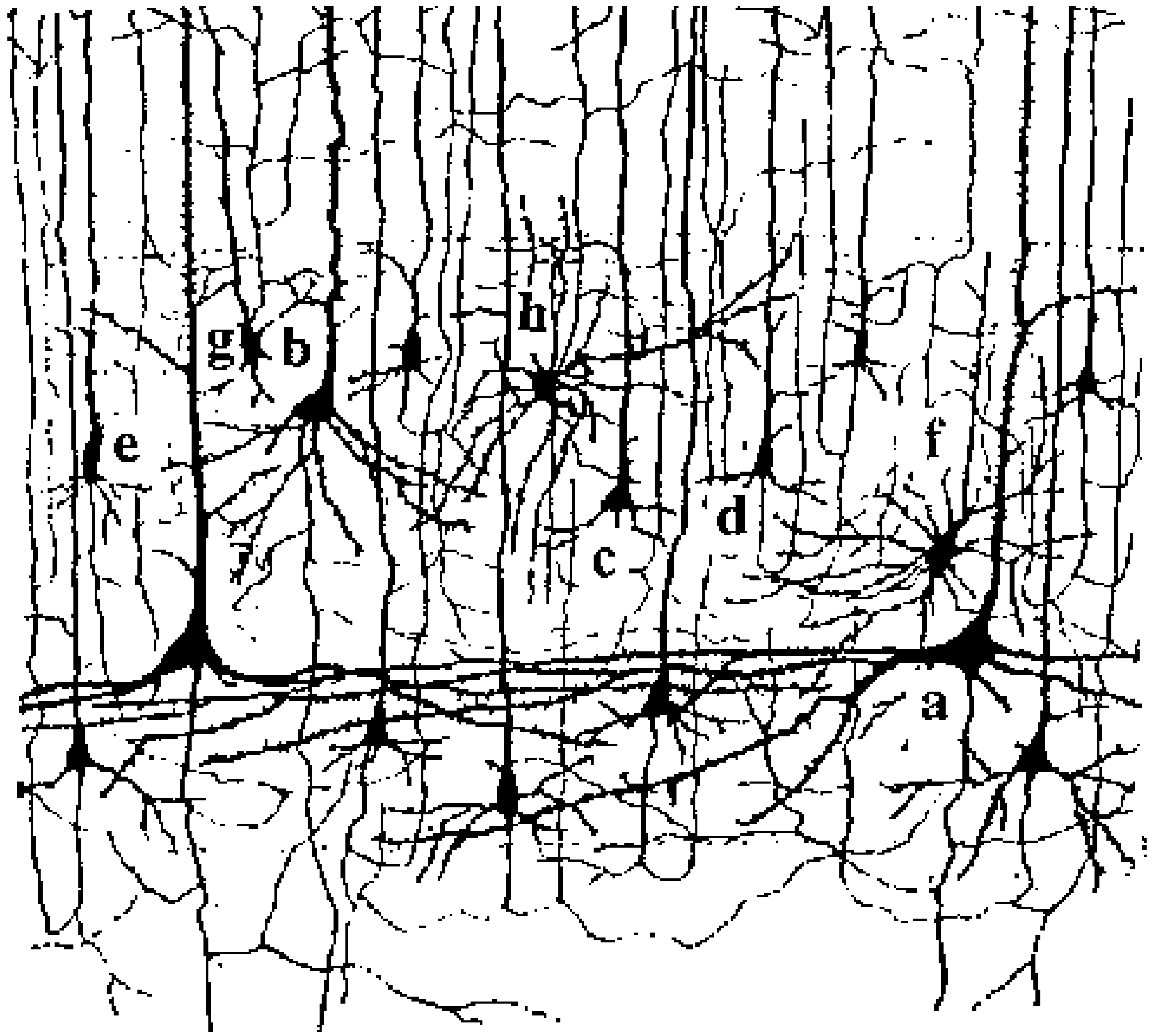
Thus, the neural network of the brain is a densely connected and densely packed network of cells.

The brain: a network of neurons

1mm

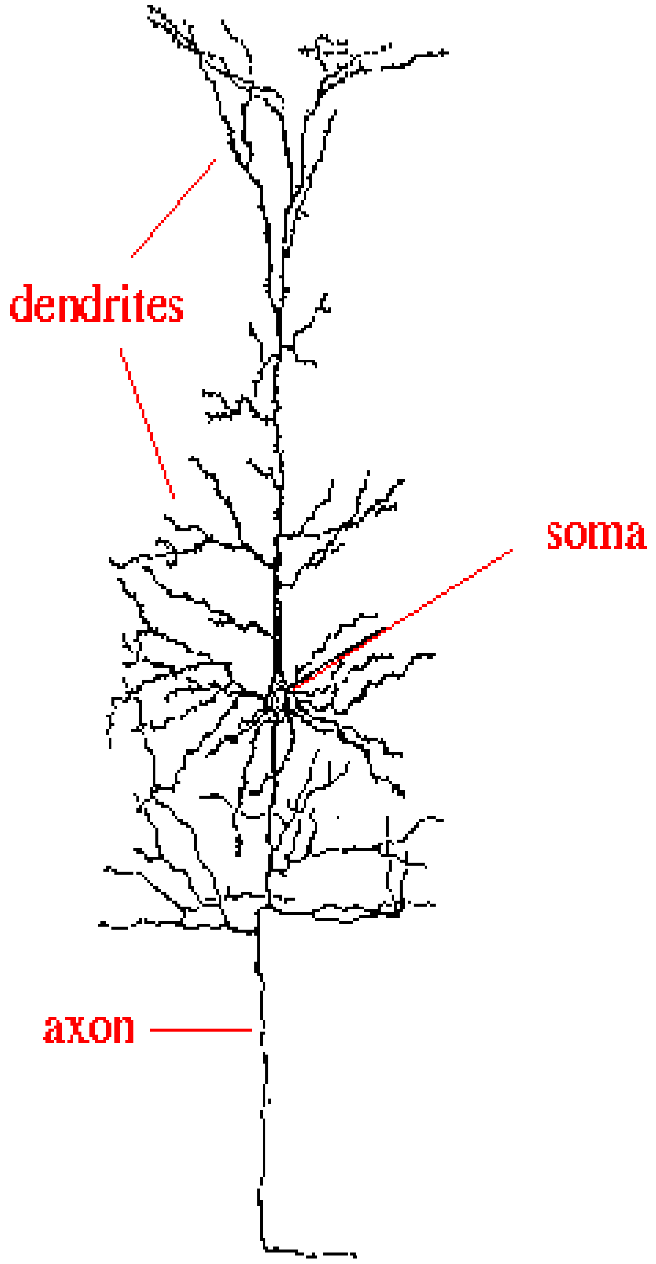


10 000 neurons
3km of wire



Ramon y Cajal

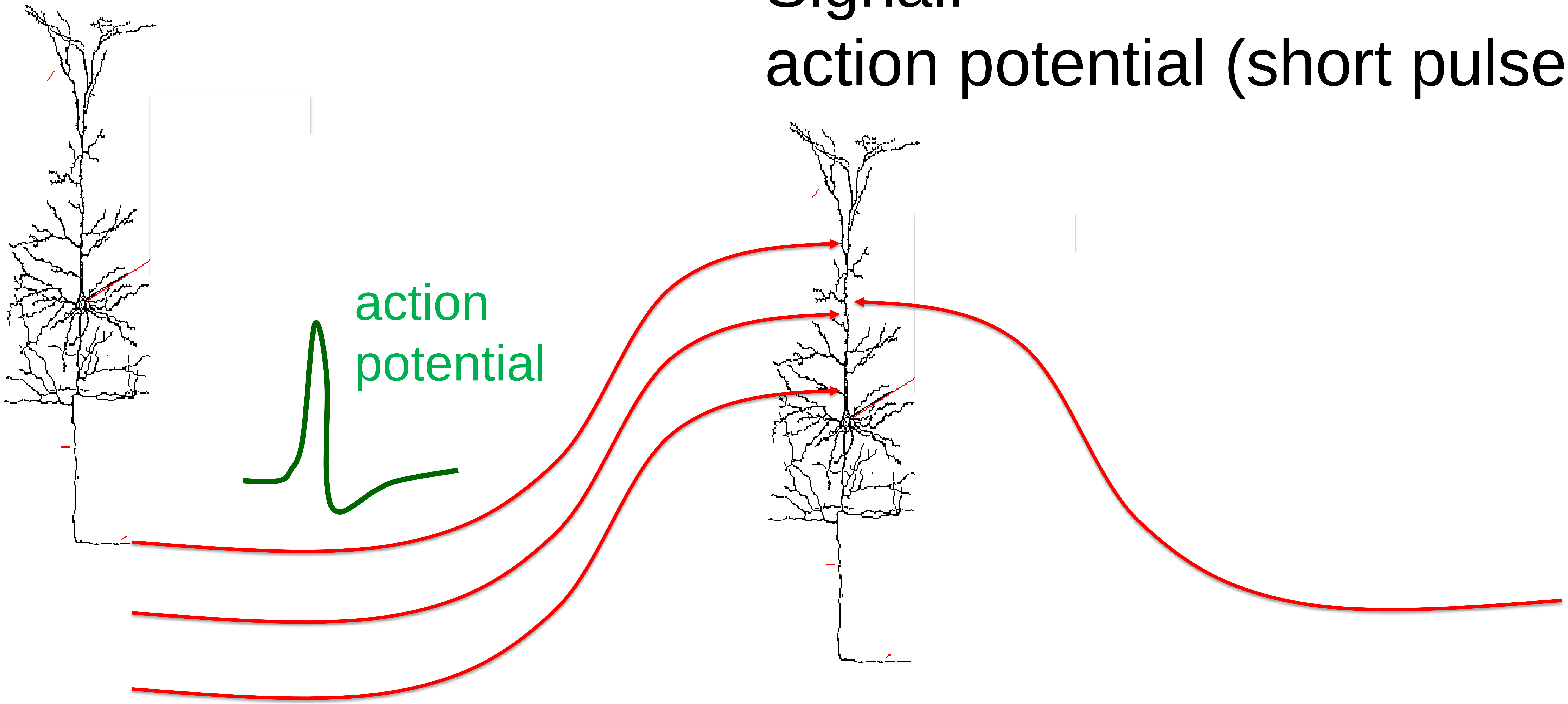
Signal:
Action potential (short pulse)



Previous slide.
These cells are called neurons and communicated by short electrical pulses, called action potentials, or ‘spikes’.

The brain: signal transmission

Signal:
action potential (short pulse)

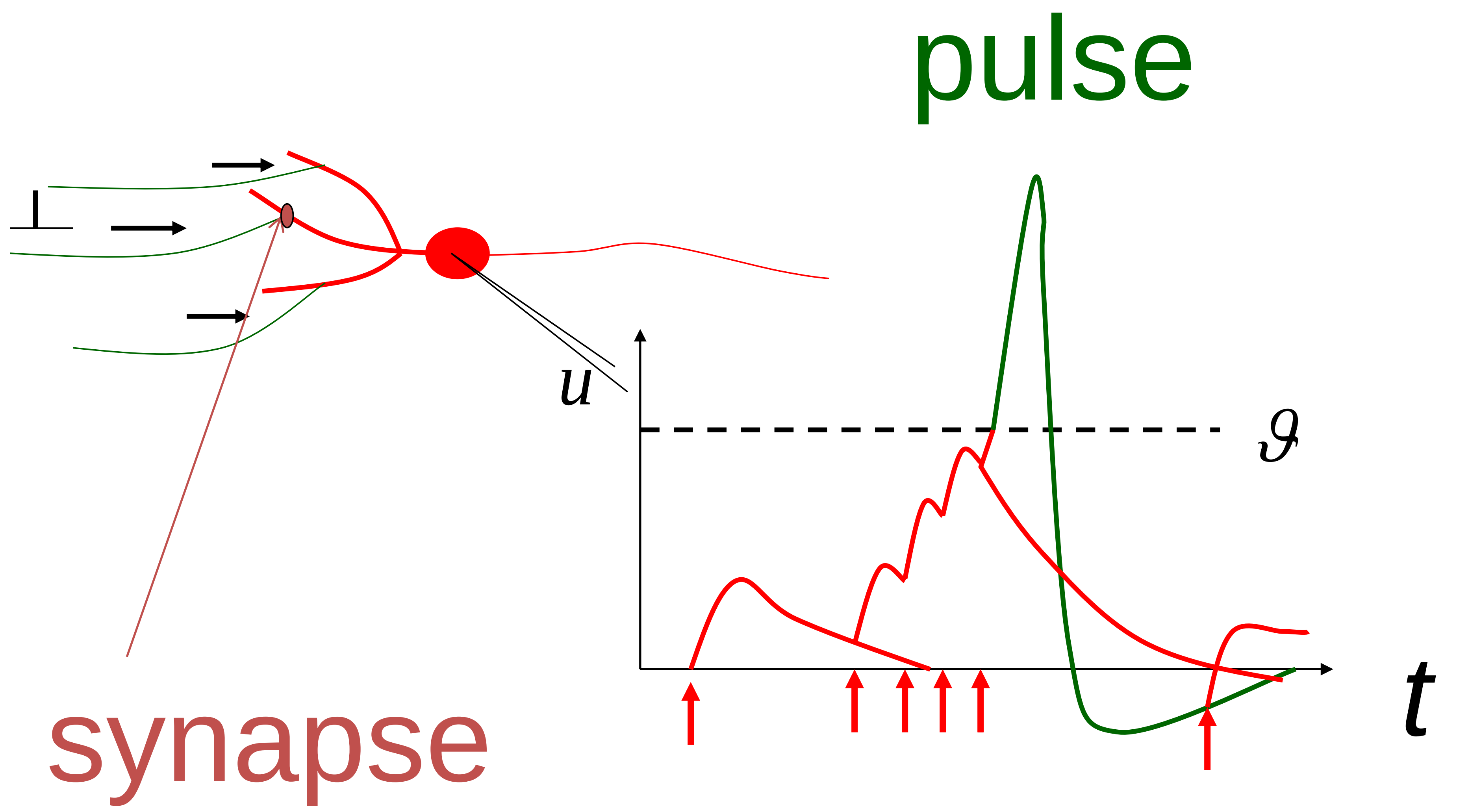


More than 1000 inputs

Previous slide.
Signals are transmitted along the wires (axons). These wires branch out to make contacts with many other neurons.

Each neuron in cortex receives several thousands of wires from other neurons that end in ‘synapses’ (contact points) on the dendritic tree.

The brain: neurons sum their inputs

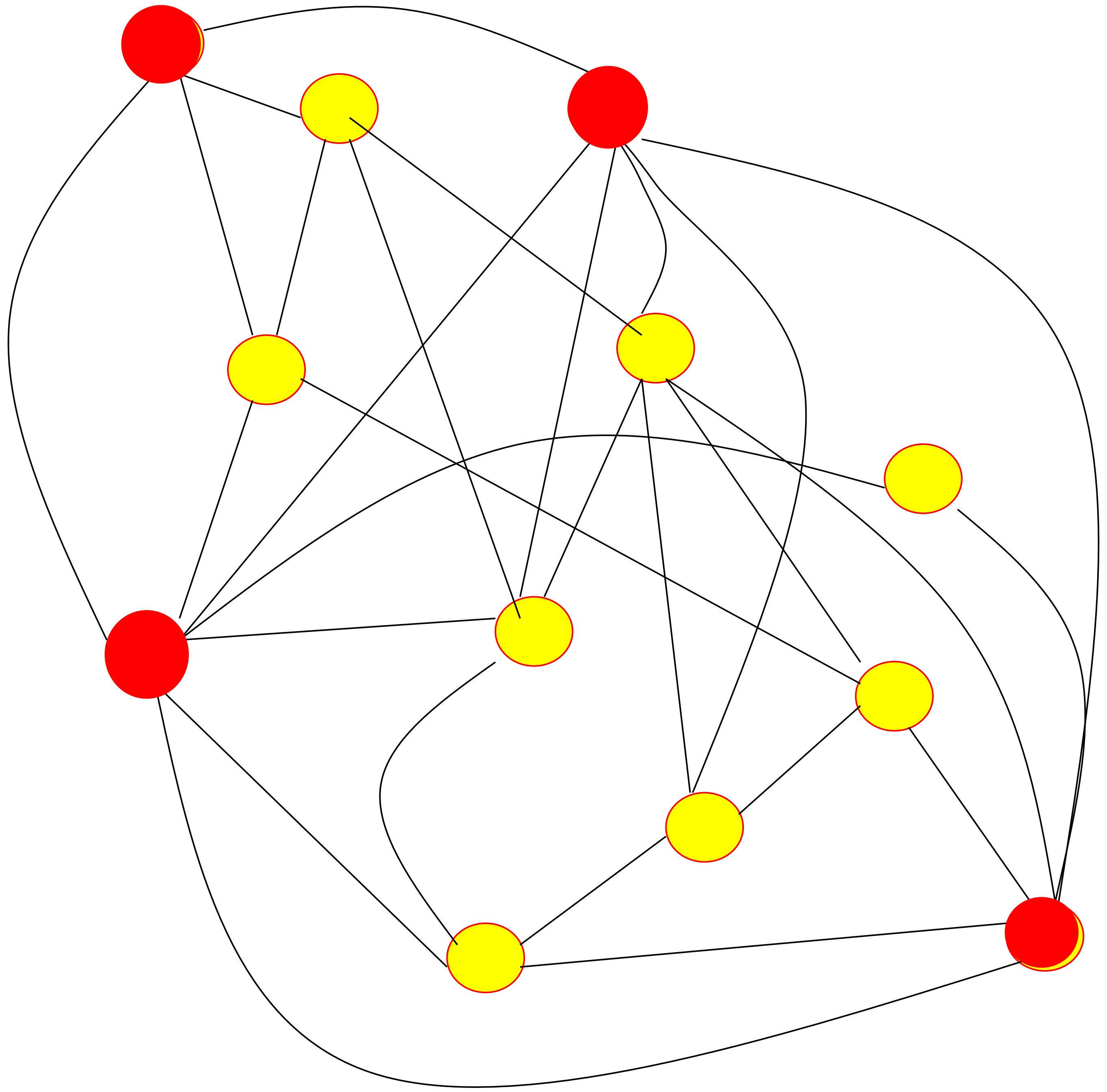


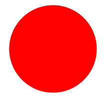
Previous slide.
If a spike arrives at one of the synapses, it causes a measurable response in the receiving neuron.

If several spikes arrive shortly after each other onto the same receiving neuron, the responses add up.

If the summed response reaches a threshold value, this neuron in turn sends out a spike to yet other neurons (and sometimes back to the neurons from which it received a spike).

Summary: the brain is a large recurrent network of neurons



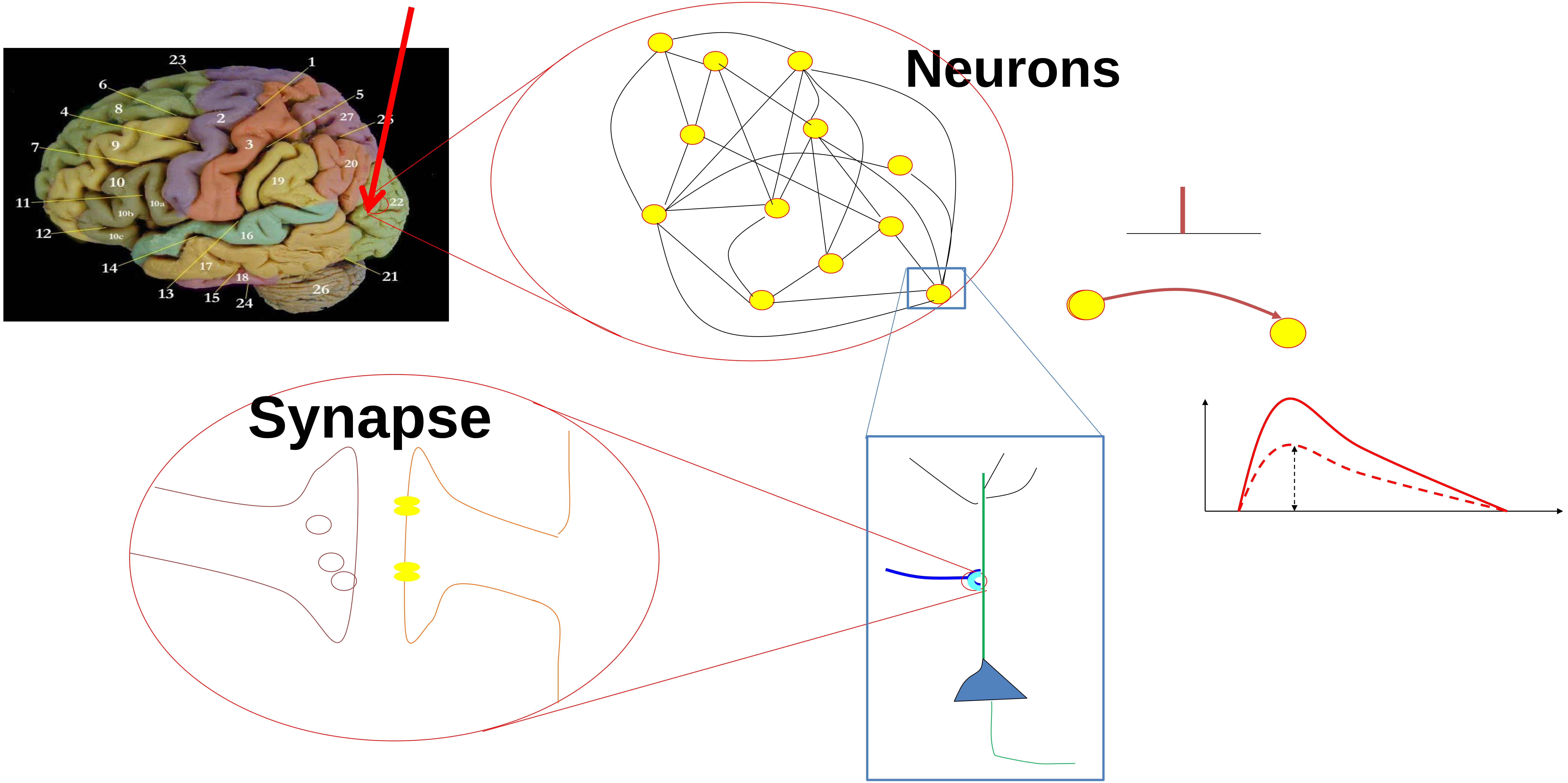

Active neuron

Previous slide.

Thus, signals travel along the connections in a densely connected network of neurons.

Sometimes I draw an active neuron (that is a neuron that currently sends out a spike) with a filled red circle, and an inactive one with a filled yellow circle.

Learning in the brain: changes between connections



learning = change of connection

Previous slide.

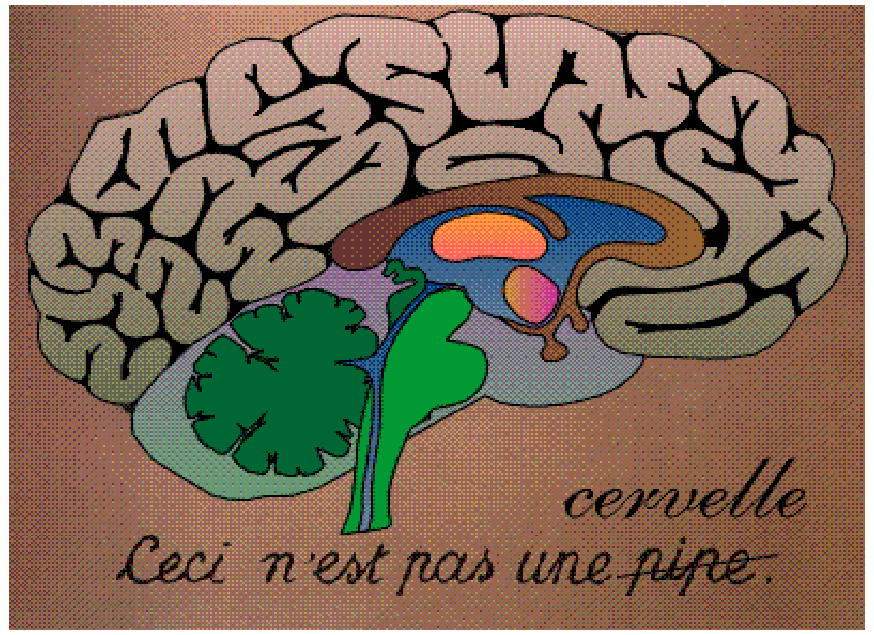
Synapses are not jut simple contact points between neurons, but they are crucial for learning.

Any change in the behavior of an animal (or a human, or an artificial neural network) is thought to be linked to a change in one or several synapses.

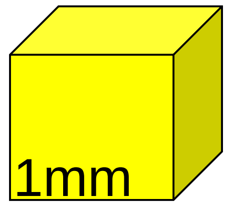
Synapses have a ‘weight’. Spike arrival at a synapse with a large weight causes a strong response; while the same spike arriving at a synapses with a small weight would cause a low-amplitude response.

All Learning corresponds to a change of synaptic weights. For example, forming new memories corresponds to a change of weights. Learning new skills such as table tennis corresponds to a change of weights.

Neurons and Synapses form a big network



Brain

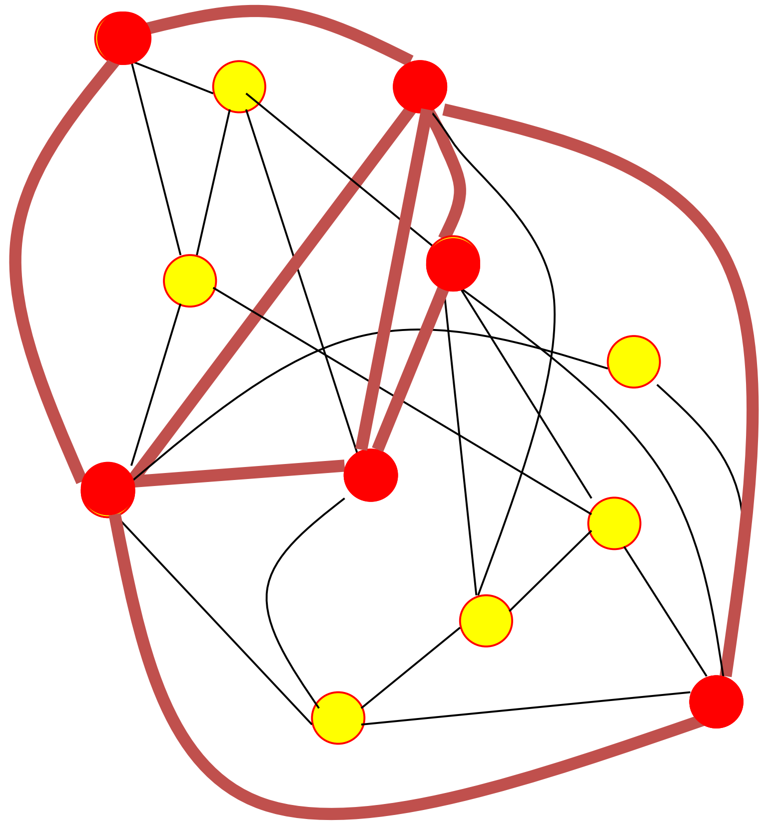


1mm

10 000 neurons
3 km of wire

10 billions neurons

10 000 connexions/neurons



Distributed Architecture

memory in the connections

No separation of processing and memory

Previous slide.

Even though we are not going to work with the Hebb rule during this class, the above example still shows that

- Memory is located in the connections
- Memory is largely distributed
- Memory is not separated from processing

(as opposed to classical computing architectures such as the van Neumann architecture or the Turing machine)

Artificial Neural Networks

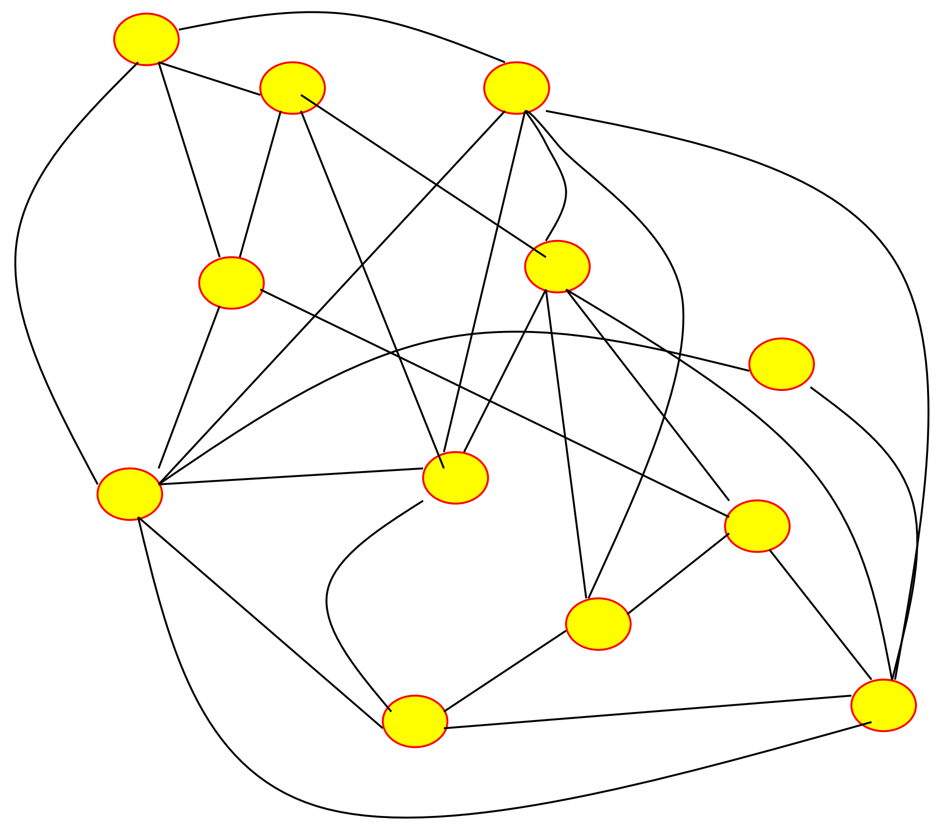
Wulfram Gerstner
EPFL, Lausanne, Switzerland

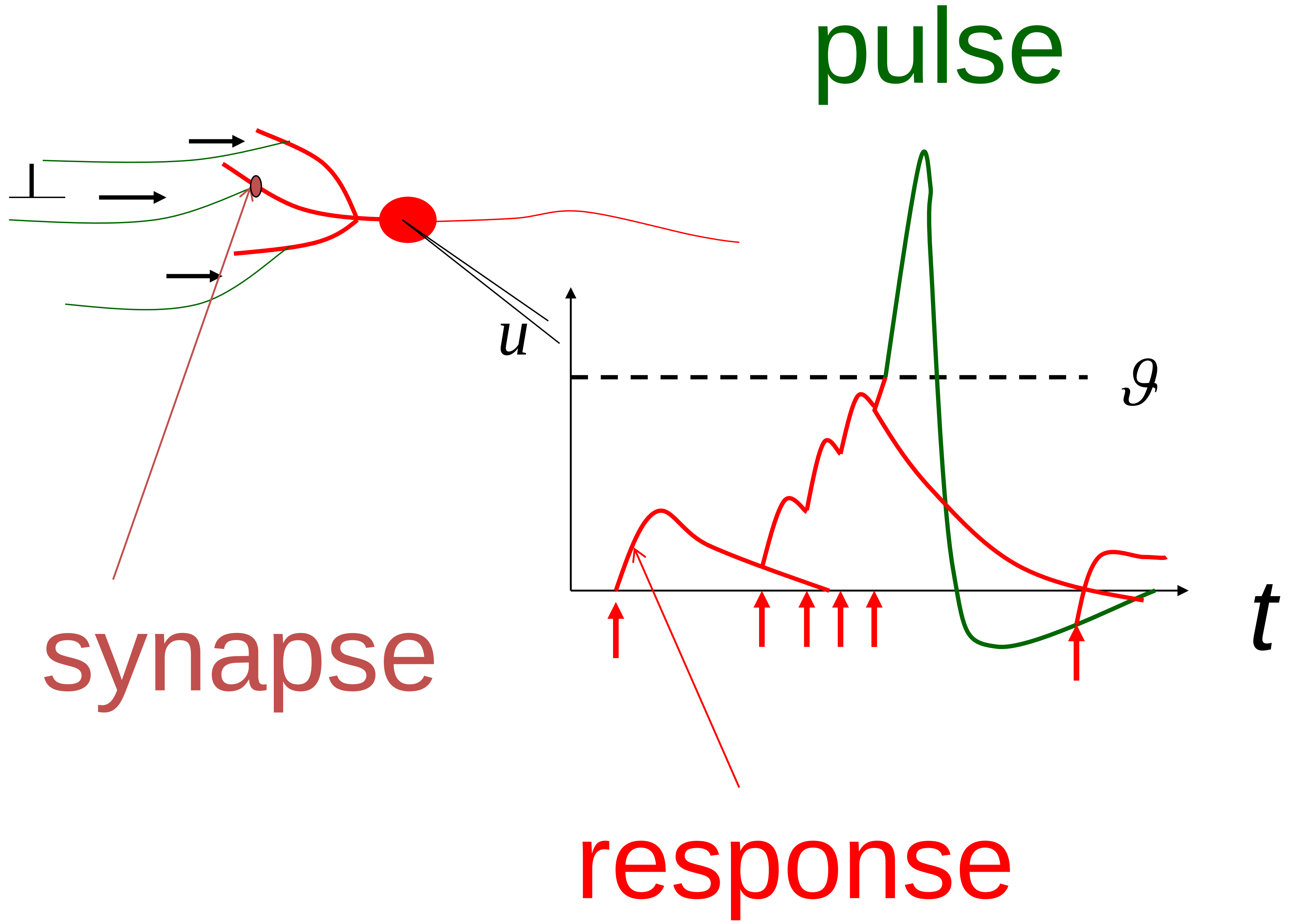
From biological neurons to artificial neurons

Previous slide.

After this super-short overview of the brain, we now turn to artificial neural networks: highly simplified models of neurons and synapses.

Modeling: artificial neurons





-responses are added
-pulses created at threshold
-transmitted to other

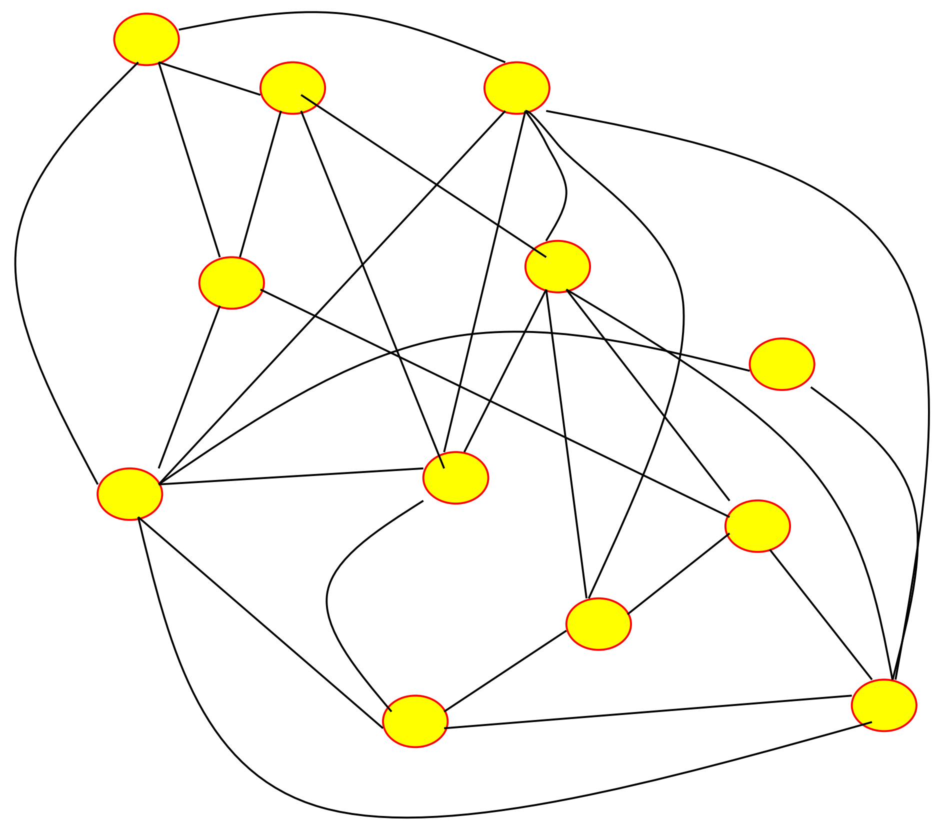
→ Mathematical description

Previous slide.
In the previous part we have seen that response are added and compared with a threshold.

This is the essential ideal that we keep for the abstract mathematical model in the following.

We drop the notion of pulses or spikes and just talk of neurons as active or inactive.

Modeling: artificial neurons

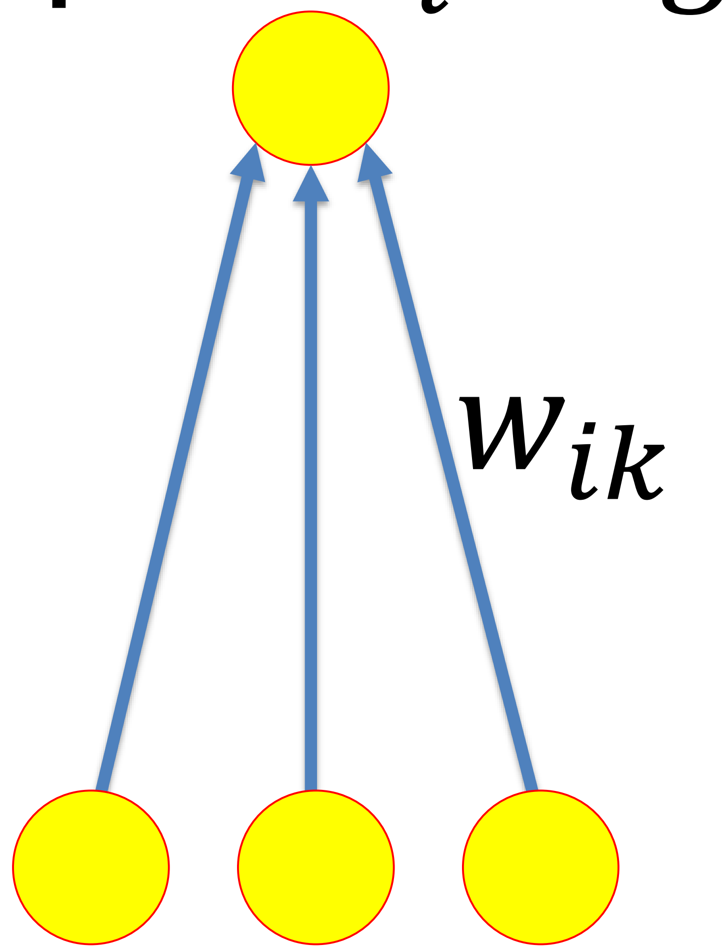


forget spikes: continuous activity x

forget time: discrete updates

nonlinearity/threshold

activity of output $x_i = g\left(\sum_k w_{ik}\, x_k\right)$



weights = adaptive parameters

activity of inputs x_k

Previous slide.

The activity of inputs (or input neurons) is denoted by x_k

The weight of a synapse is denoted by w_{ik}

The nonlinearity (or threshold function) is denoted by g

The output of the receiving neuron is given by

$$x_i = g\left(\sum_k w_{ik}\, x_k\right)$$

Quiz: biological neural networks

[ ] Neurons in the brain have a threshold.
[ ] Learning means a change in the threshold.
[ ] Learning means a change of the connection weights
[ ] The total input to a neuron is the weighted sum of individual inputs
[ ] The neuronal network in the brain is feedforward: it has no recurrent connections

Previous slide. Your notes

Artificial Neural Networks
Introduction to the field

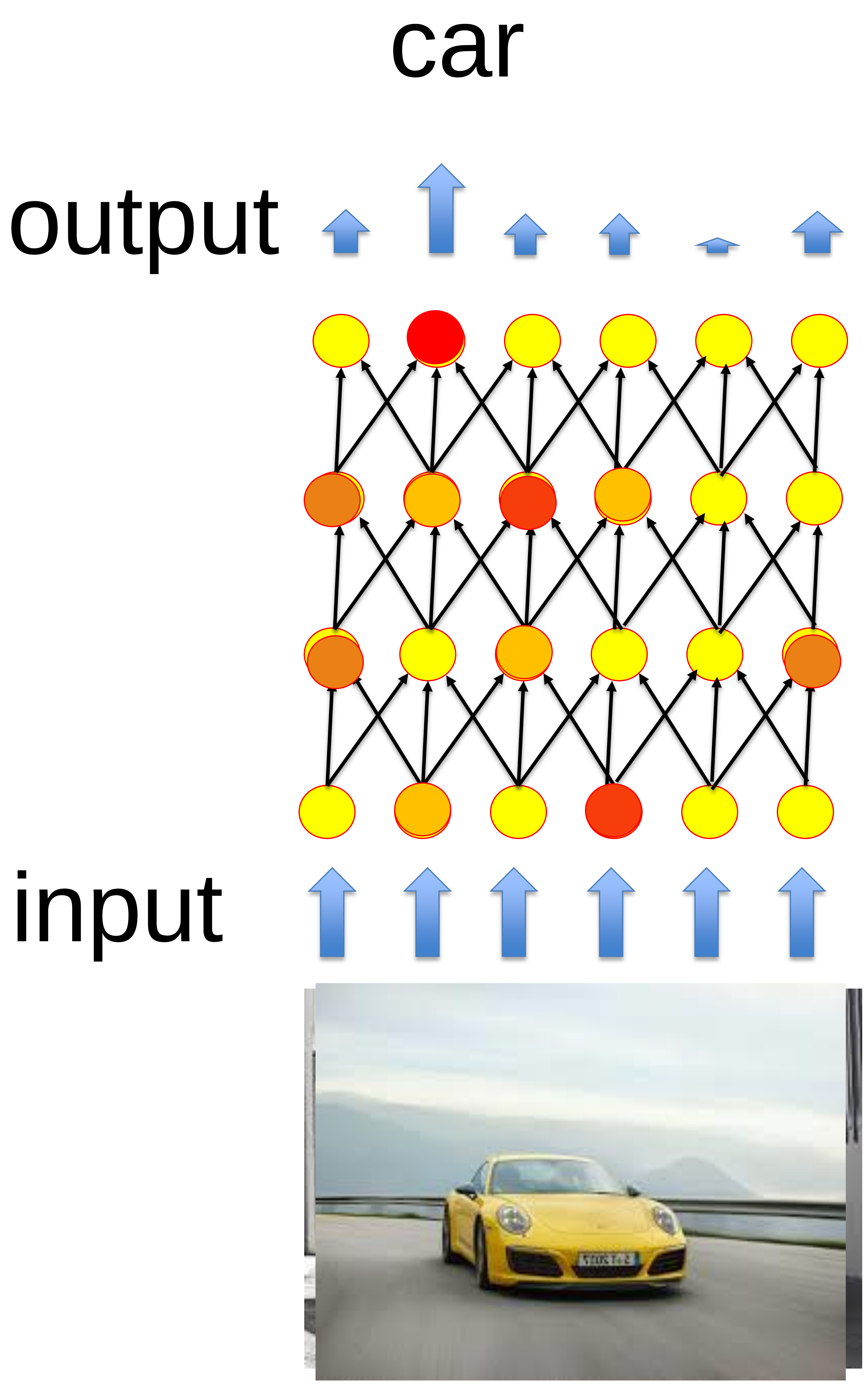
Wulfram Gerstner
EPFL, Lausanne, Switzerland

1. From Biological to Artificial Neurons
2. **Artificial Neural Networks for Classification**
   - **layered feedforward networks**
   - **recurrent networks**

Previous slide.
Now that we know about artificial neurons and synaptic weights, let us construct a useful network.
The first task we study is classification

Artificial Neural Networks for classification

feedforward network



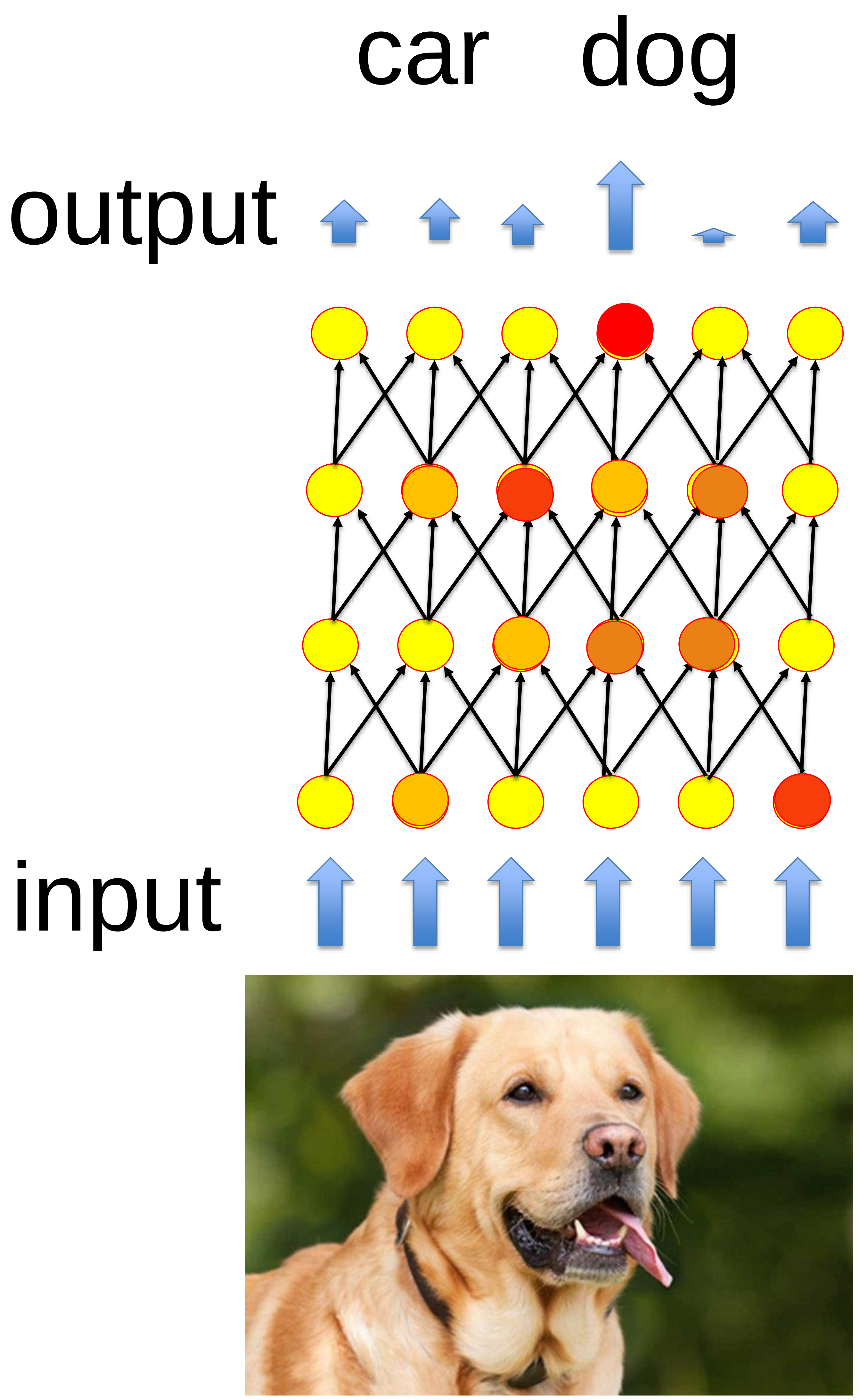
Previous slide.
An input is presented at the bottom of the network.
It passes through several layers of neurons.
All connections are directed from the bottom to the next layer further up: this architecture is called a feedforward network.

The output is a set of neurons that correspond to different ‘classes’.

An ideal network should respond with activating the neuron corresponding to ‘car’, if the input image shows a car.

Artificial Neural Networks for classification

Aim of learning:
Adjust connections such that output class is correct (for each future input)

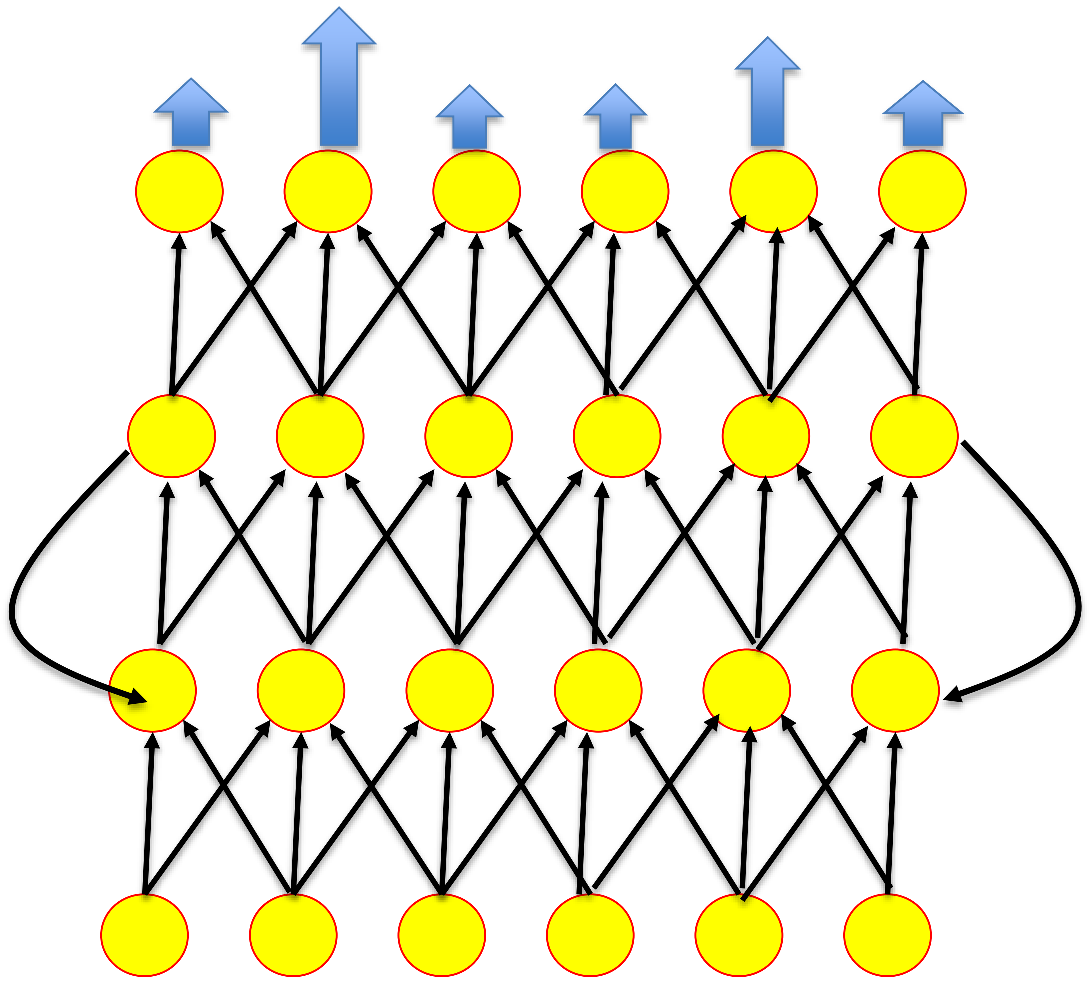


Previous slide.
The aim of learning is to adjust the connection weights such that, for each future input, the output class is correct.
If the input is a dog, the ‘dog’-neuron should respond.
If the input is a car, the ‘car’-neuron should respond.

Starting week 5 of the semester, we focus on the task of building and training artificial neural networks for classification.

Deep networks with recurrent connections

‘a man sitting on a couch with a dog’



Network desribes the image with the words:

‘a man sitting on a couch with a dog’

(Fang et al. 2015)

Previous slide.
An amazing example of sequence production with a recurrent neural network is this network which looks at a static image and outputs the spoken sentence:
‘A man is sitting on a couch with a dog’.

A sentence is a temporal sequence of words – and importantly, a sequence that follows grammatical rules.
Sequence learning requires recurrent connections (feedback connections), in contrast to the feedforward architecture that we have seen so far.
And yes, recurrent neural networks can implicitly pick up the statistical rules for sentence formation if they are trained on a sufficiently large data set containing millions of examples.

Summary

Wulfram Gerstner
EPFL, Lausanne, Switzerland

Artificial Neural Networks for Classification
- layered feedforward networks
- recurrent networks

- weights are used as adjustable parameters
  to learn a stationary classification task or sequence task
- details will follow in later videos

Previous slide.

Synaptic weights are the adjustable parameters of artificial neural networks. Artificial neural networks are mostly used in a layered feedforward structure, but recurrent neural networks are often used for sequence learning.

Artificial Neural Networks
Introduction to the field

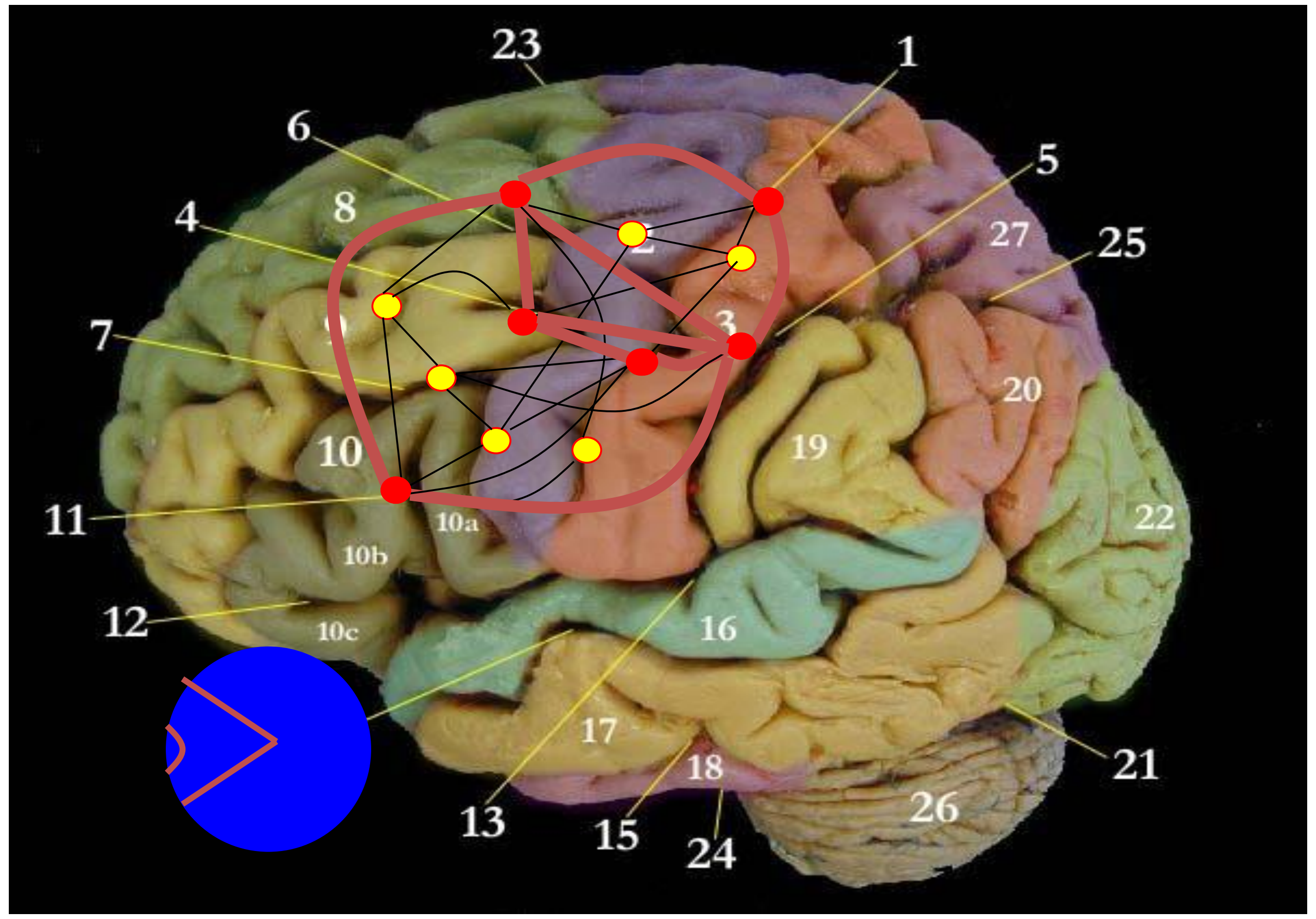
Wulfram Gerstner
EPFL, Lausanne, Switzerland

1. From Biological to Artificial Neurons
2. **Artificial Neural Networks for Classification**
   - **layered feedforward networks**
   - **recurrent networks**
3. **Artificial Neural Networks for action learning**

Previous slide.
However, classification is not the only task we are interested in.

Artificial Neural Networks for action learning

Learning from mistakes,
And from (rare!) successes



Missing:
Value of action
- ‘goodie’ for dog
- ‘success’
- ‘compliment’

Reinforcement learning = learning based on reward

Previous slide.
Let us go back for a moment to the brain, and how humans or animals learn.

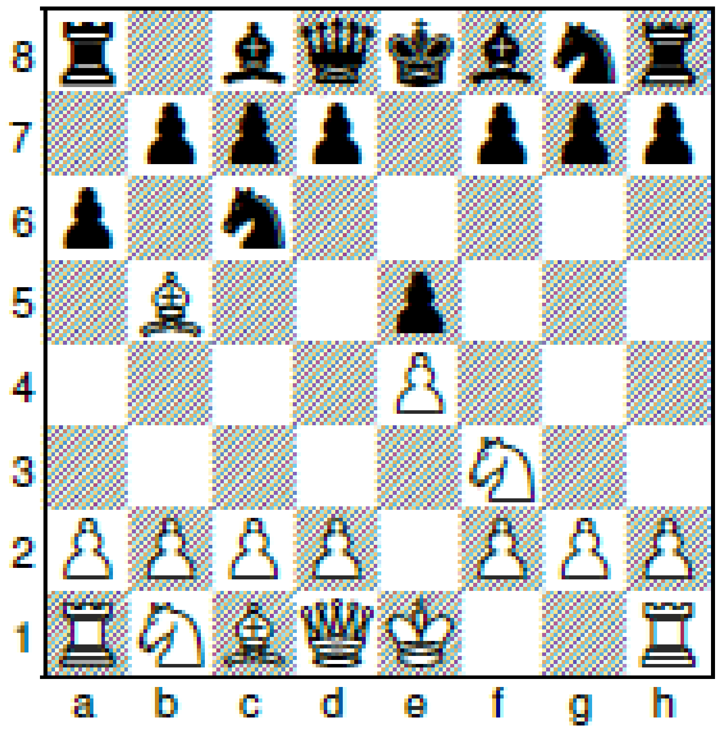
We learn actions by trial and error exploiting rather general feedback: reward or praise on one side, pleasure and pain on the other side.

Important is the notion of value of an action.

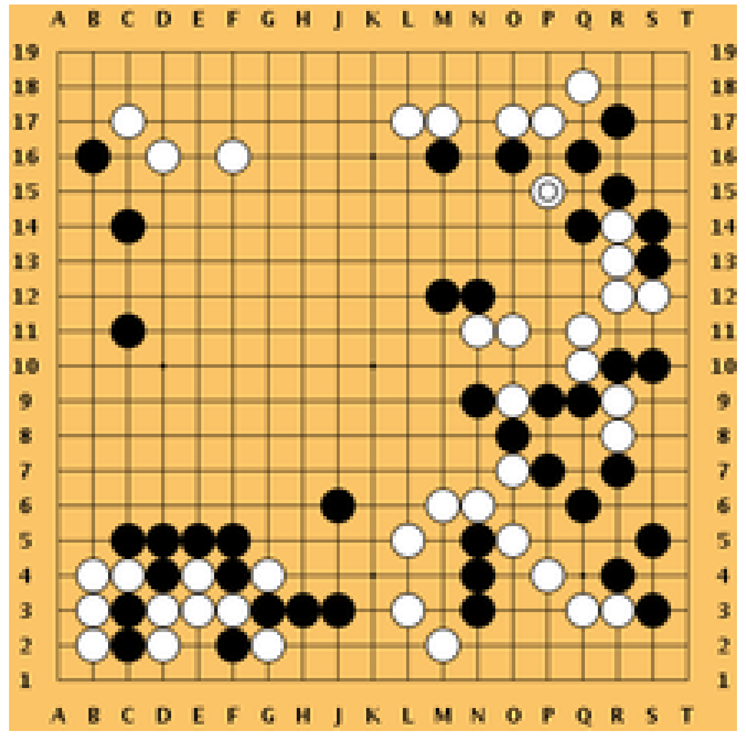
Learning actions or sequences of actions based on ‘reward’ is very different from classification: it falls in the field of ‘Reinforcement Learning’.

Deep reinforcement learning

Chess



Go



Artificial neural network (*AlphaZero*) discovers different strategies by playing against itself.

In Go, it beats Lee Sedol

Previous slide.
The same kind of ideas (learning by reward) have also been implemented in artificial neural networks that are trained by reinforcement learning.

In a game such a Chess or Go, the reward signal is only given once at the very end of the game: positive reward if the game is won, and negative reward if it is lost.

This very sparse reward information is sufficient to train an artificial neural network to a level where it can win against grand masters in chess or Go.

To improve performance, each network plays against a copy of itself. By doing so it discovers good strategies (such as openings in chess).

Deep reinforcement learning

Network for choosing action

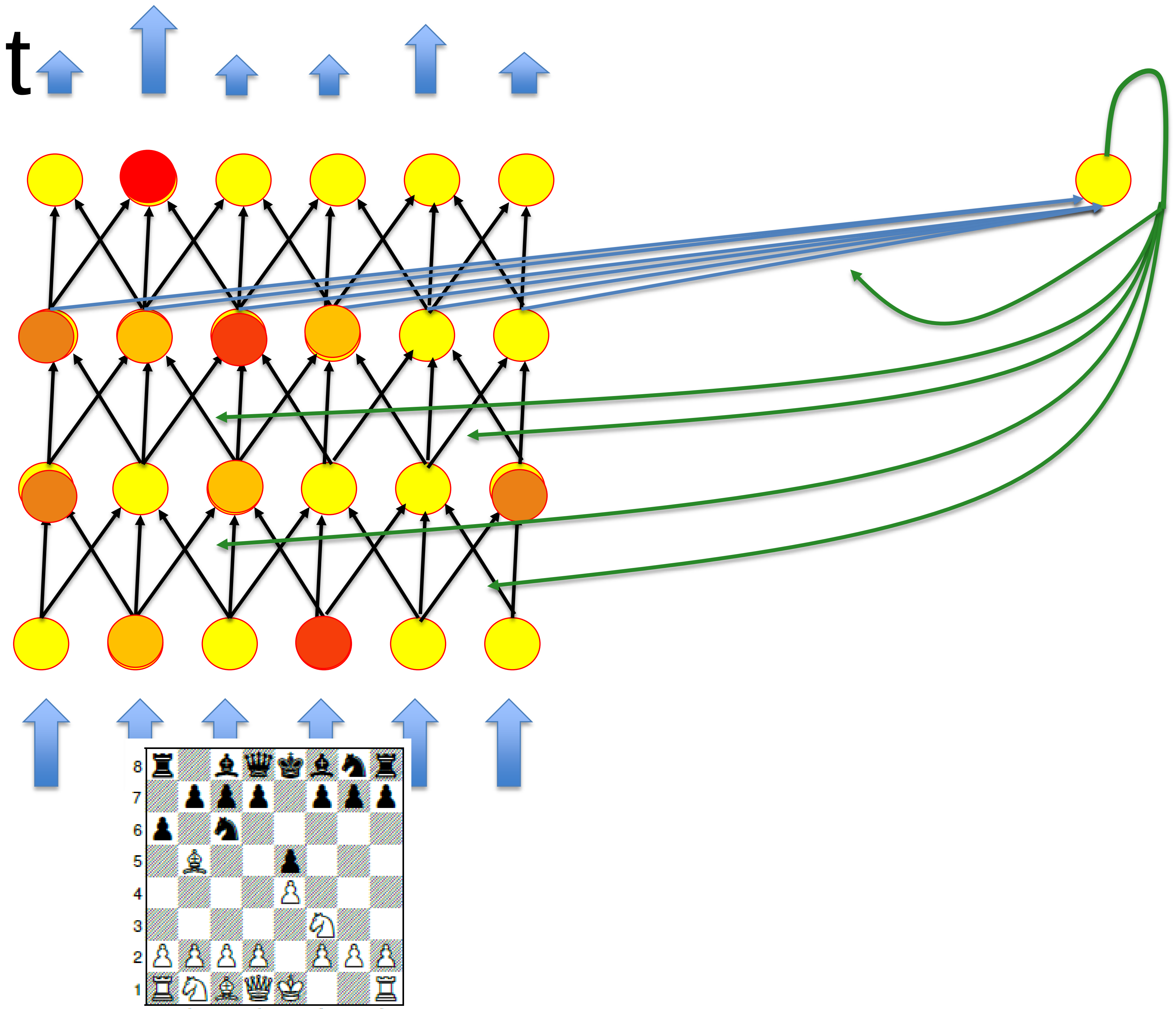
action:

Advance king

2^{nd} output for **value** of action:

probability to win

output



input

learning:

- change connections

aim:

- Predict value of position
- Choose next action to win

Previous slide.
Schematically, the artificial neural network takes the position of chess as input.
There are two types of outputs:

- The main outputs are the actions such as ‘move king to the right’
- An auxiliary output predicts the ‘value’ of each state. It can be used to explore possible next positions so as to pick the one with the highest value.
- The value can be interpreted as the probability to win (given the position)

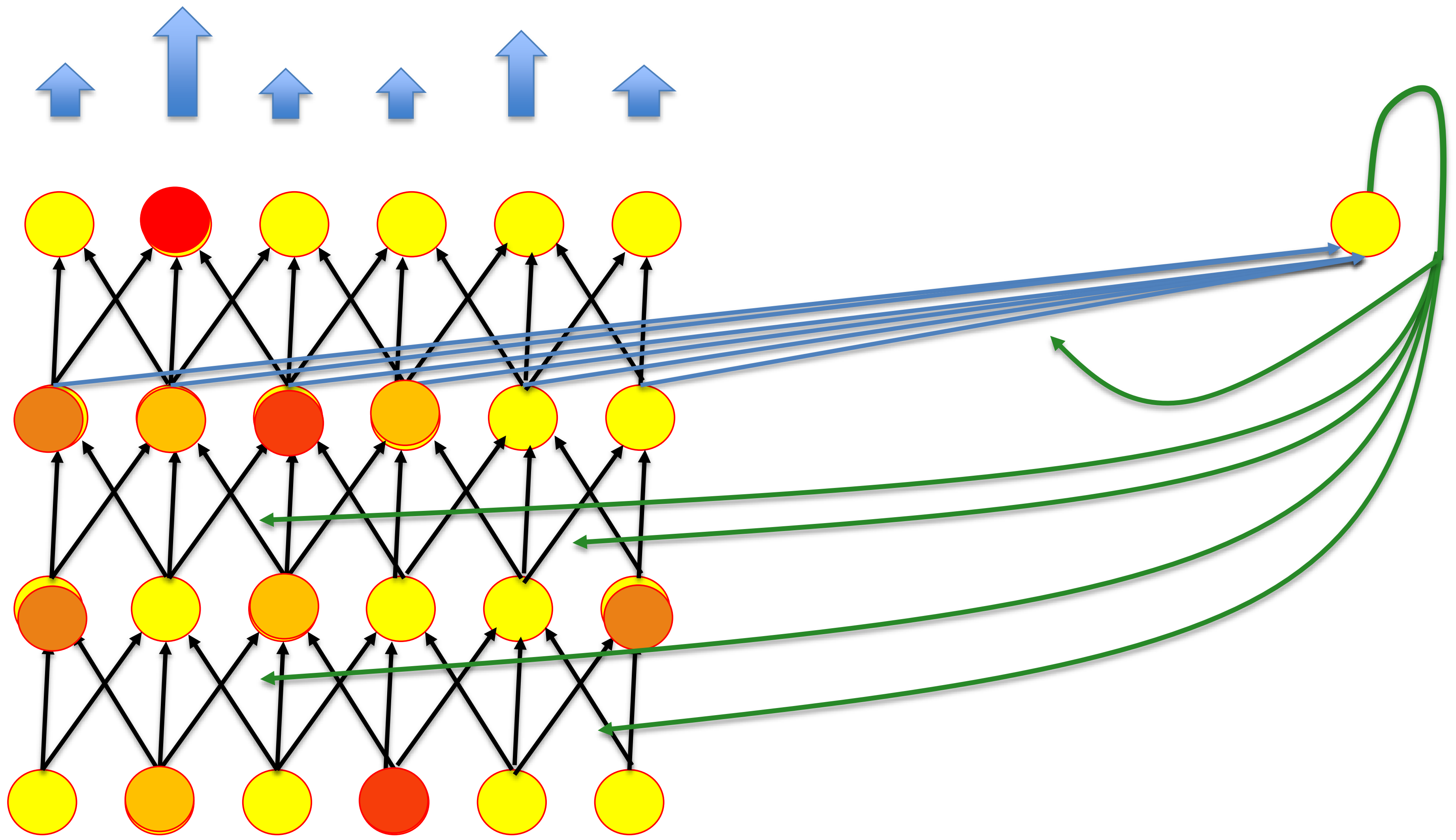
In the theory of reinforcement learning, positions are also called ‘states’.

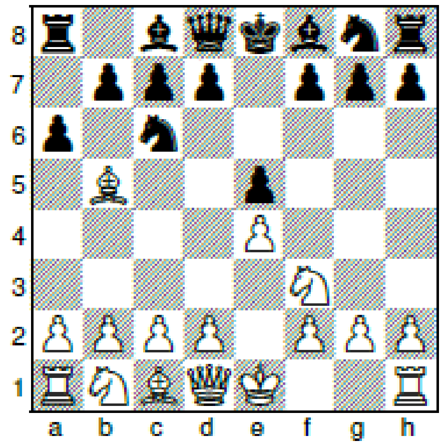
Deep reinforcement learning (alpha zero)

Silver et al. (2017) , Deep Mind

output: 4672 actions

advance king





input: 64x6x2x8 neurons
(about 10 000)

Training 44Mio games (9 hours)

Planning:
potential sequences
(during 1s before playing next action)

Previous slide.
Since there are many different positions, the number of input neurons is in the range of ten thousand:

On each of 64 positions there can be one of 6 different ‘figures’ (king, knight, bishop) of 2 different colors.
To avoid repetitions the 8 last time steps are used as input.

Training is done by playing against itself in 44 million games.

The allotted computer time for planning the next action is 1s.

Deep reinforcement learning (alpha zero)

Silver et al. (2017)

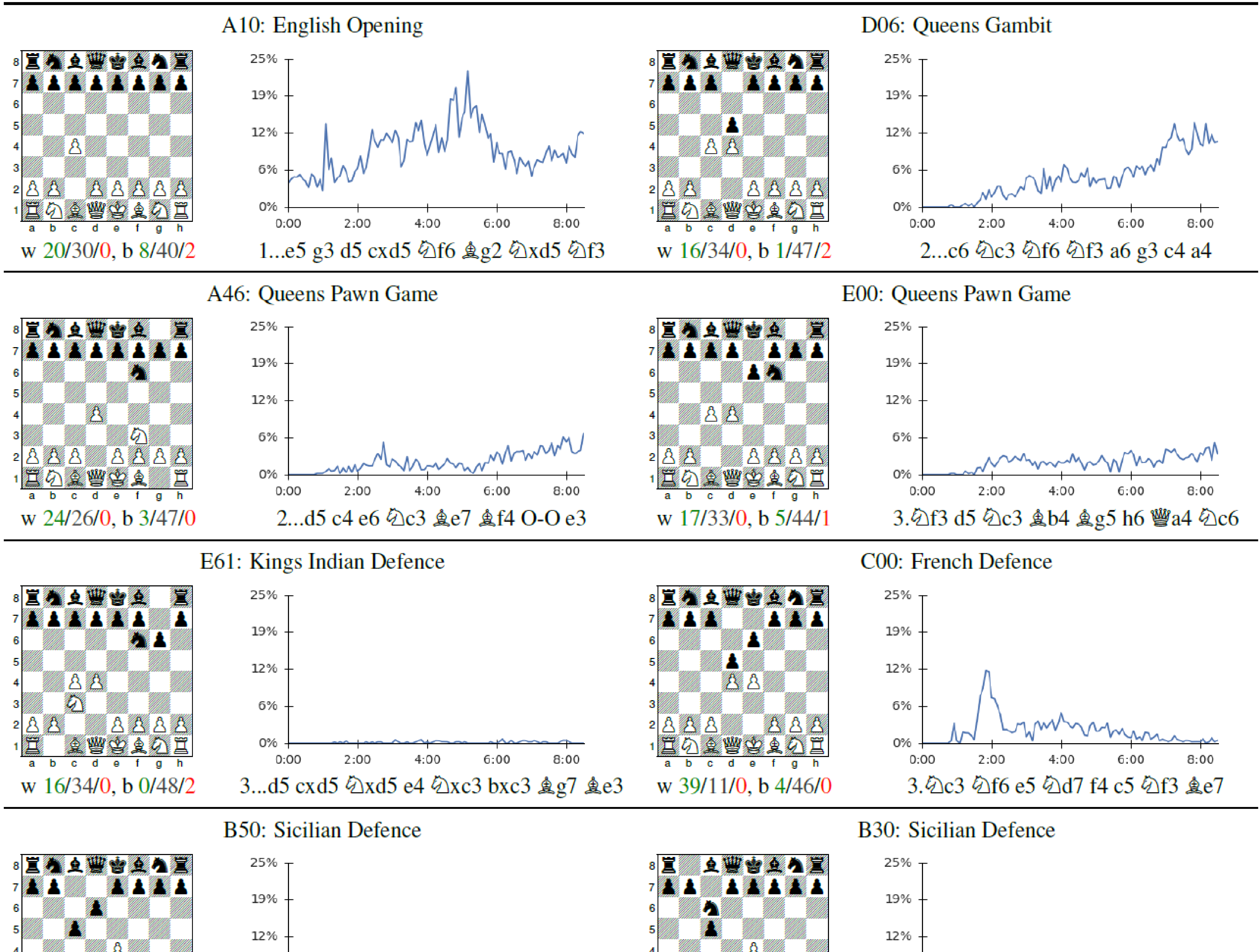
Chess:

- Trained for 44 Mio

-discovers classic openings

-beats best human players

-beats best classic AI algorithms



Previous slide.
After training for 44 Milllion self-play games, the algorithm matches or beats classical AI algorithms for chess.

Interestingly, it ‘discovers’ well-known strategies for openings, corresponding closely to well known openings in textbooks on chess.

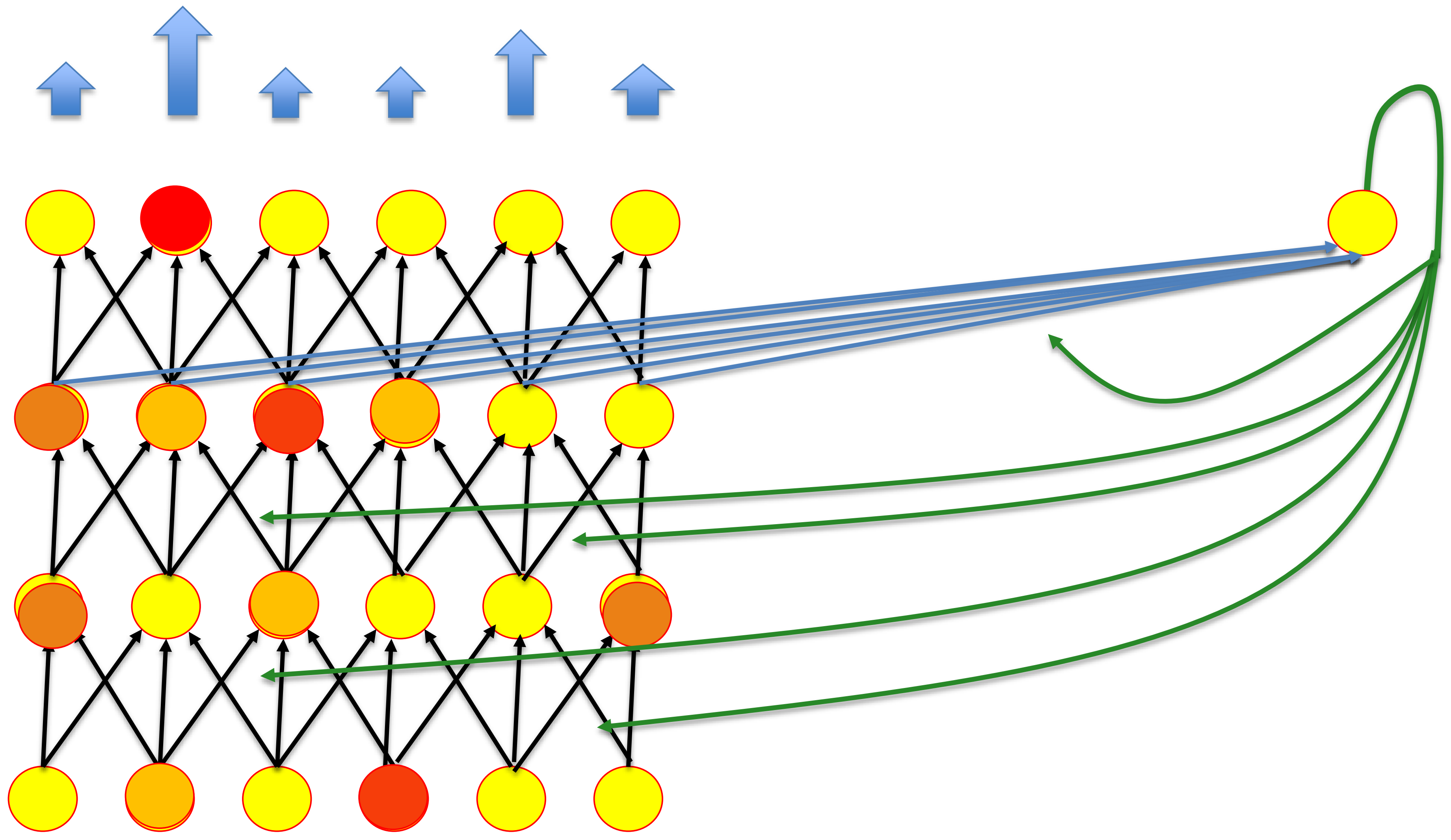
When trained on Go it beats the world champions.

Self-driving cars

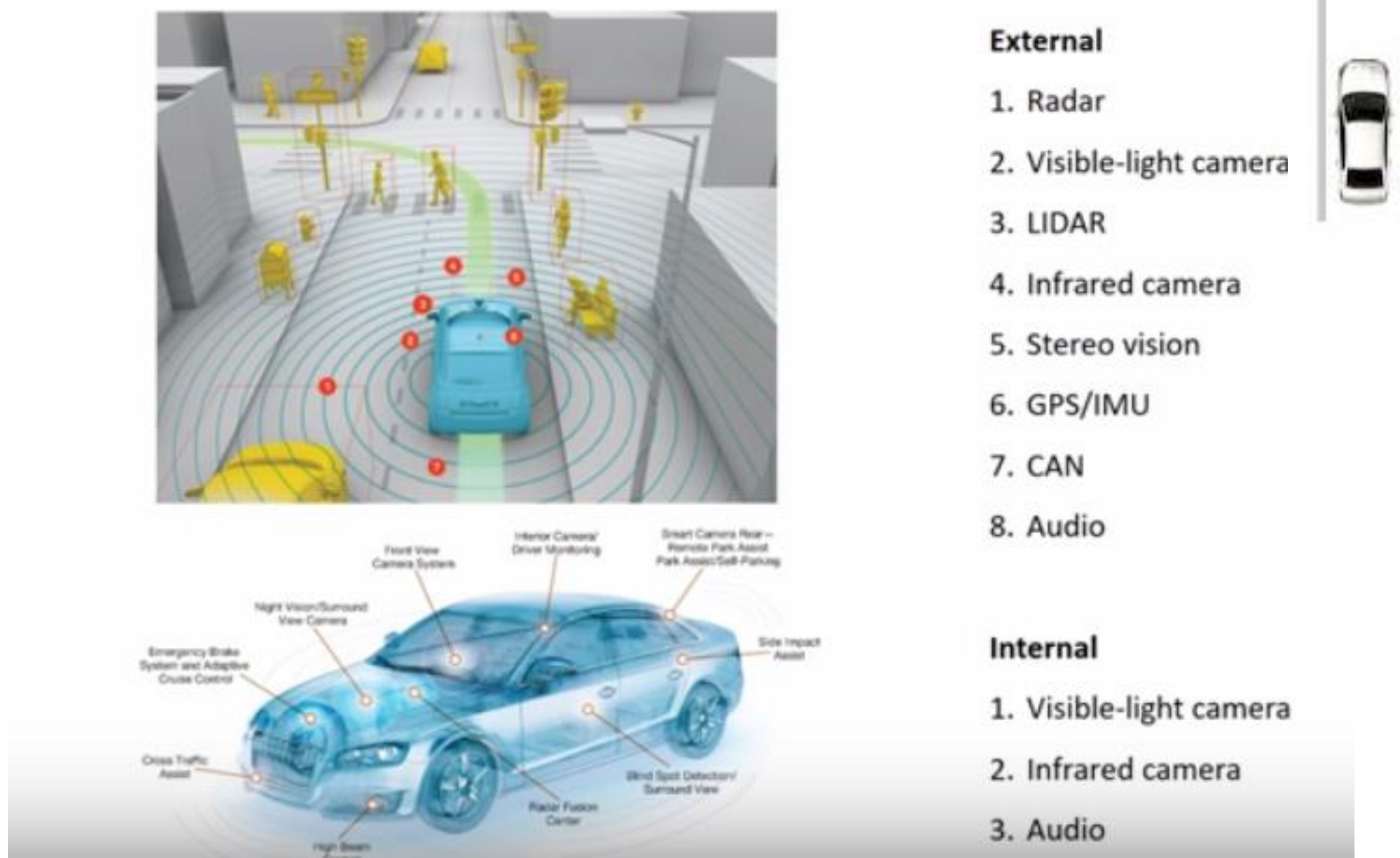
https://selfdrivingcars.mit.edu/

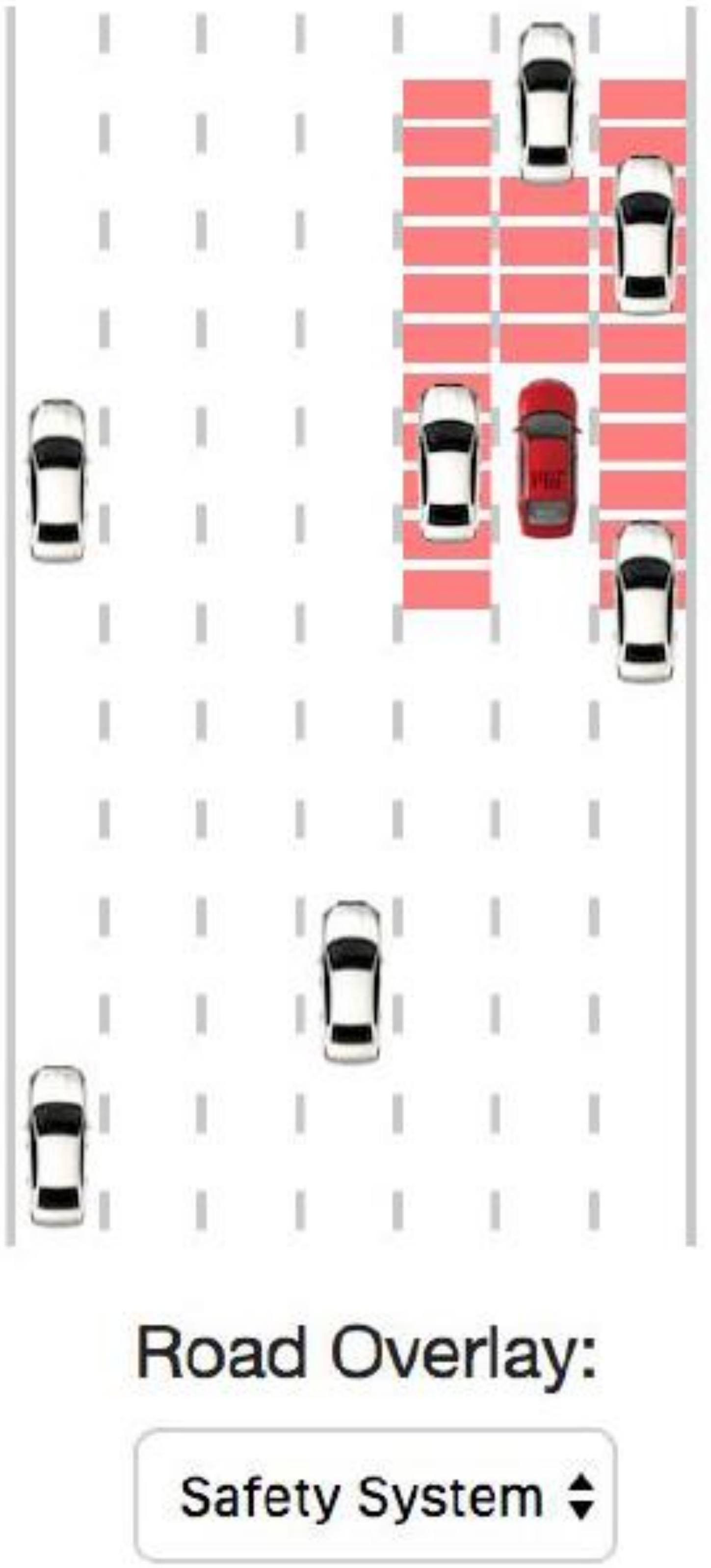
Lex Friedman, MIT

advance and accerate



Value: security, duration of travel





Previous slide.
Similar reinforcement learning algorithms are also used to train selfdriving cars.
There is a nice series of video lectures by Lex Friedman on the WEB.
Inputs are video images as well as distance sensors.

The value is security (top priority) combined with duration of travel.

The main focus of the class ANN is on reinforcement learning.

Supervised Classification and Reinforcement Learning

This class has two main parts:

1. Classification by Supervised learning
- simple perceptrons
- deep learning
- convolutional networks

2. Reinforcement learning/Reward-based learning
- Q-learning, SARSA,
- policy gradient
- Deep Reinforcement learning

Previous slide.
This class has two main parts:

Classification by Supervised learning.

- simple perceptron, geometry of classification (this week)
- deep learning in multilayer networks (later)
- convolutional networks for image classification (later)

Reinforcement learning/action learning driven by sparse rewards

- Q-learning, SARSA (next week)
- Policy gradient methods (in two weeks)
- Deep Reinforcement learning (later)
- Games and model-based reinforcement learning

Quiz: Classification versus Reinforcement Learning

[ ] Classification aims at predicting the correct category such as ‘car’ or ‘dog’
[ ] Classification by supervised learning is based on rewards
[ ] Reinforcement learning is based on rewards
[ ] Reinforcement learning aims at optimal action choices

Your notes.