AI in Culture and Arts – Tech Crash Course

Introduction to Artificial Intelligence and Machine Learning

Benedikt Zönnchen
5th of November 2025

MUC.DAI
Munich Center for
Digital Sciences and AI

HM

myt Hochschule
für Musik und Theater
München

1. How to Model Intelligence?

2. When is Learning Possible?

3. How Do Machines Learn?

4. How Do Humans Train Machines?

5. How to Interact with Leaning Machines?

1. How to Model Intelligence?

2. When is Learning Possible?

3. How Do Machines Learn?

4. How Do Humans Train Machines?

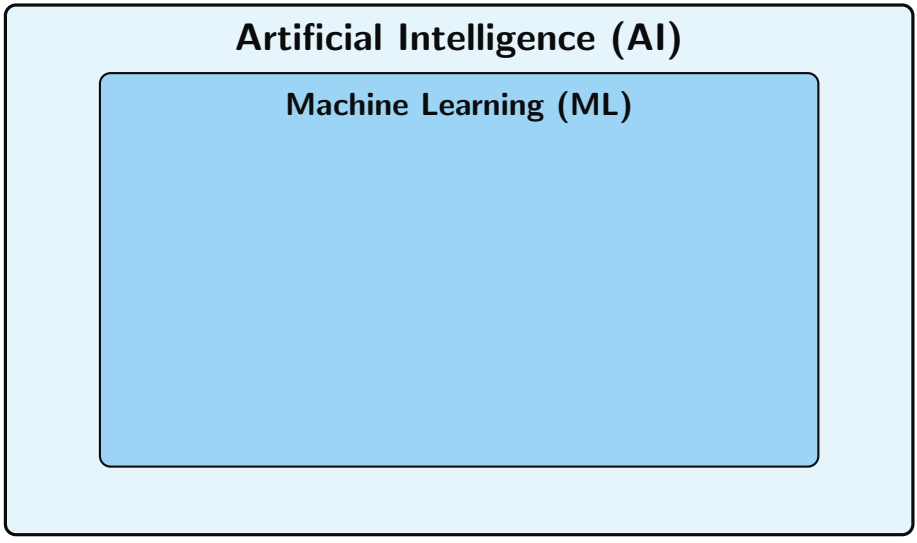
5. How to Interact with Leaning Machines?

How to Model Intelligence?

"The ability of an agent to achieve goals in a wide range of environments."
– Stuart Russell

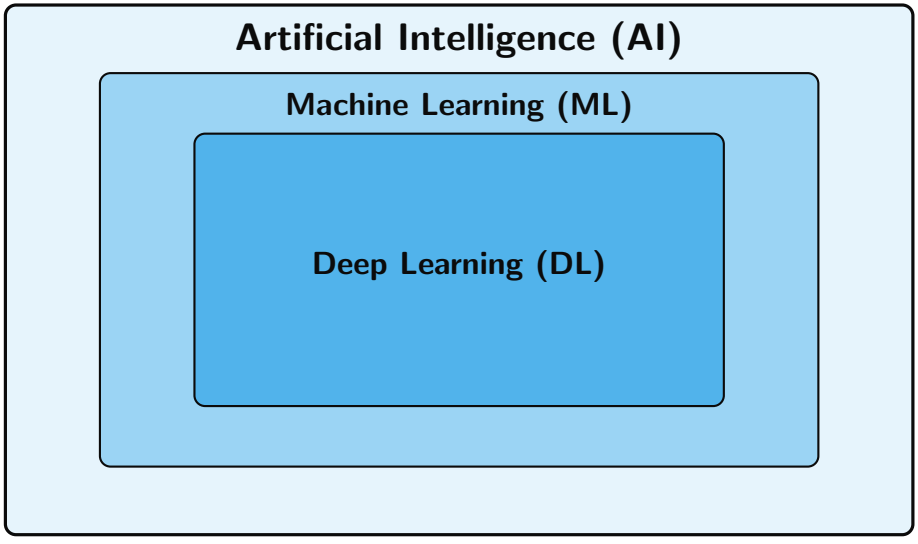
Artificial Intelligence (AI)

Most of the hype is based on **Deep Learning**.

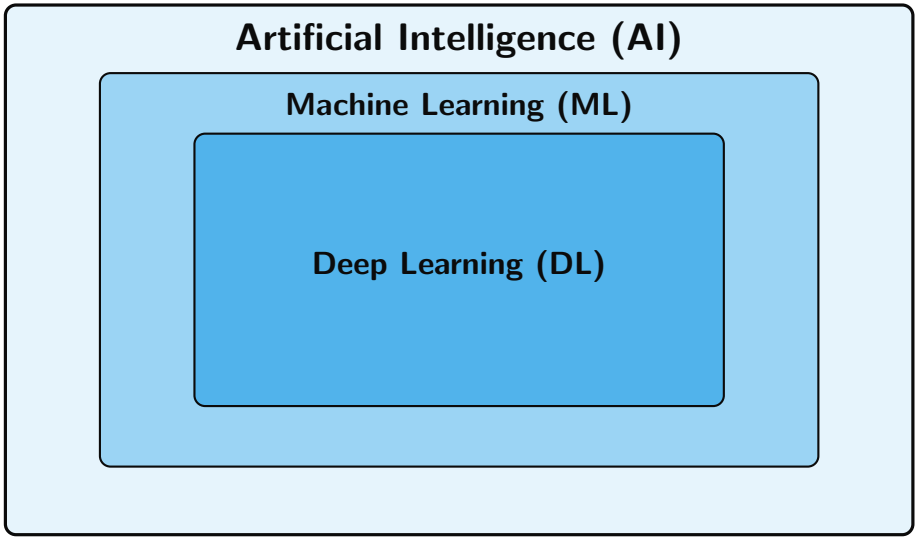


Most of the hype is based on **Deep Learning**.

How to Model Intelligence?



Most of the hype is based on **Deep Learning**.



Most of the hype is based on **Deep Learning**.

Symbolism

Exploits explicit, rule-based symbolic manipulation, logic, and structured reasoning to represent knowledge and solve problems.

- **Assumption:** Intelligence uses high-level, human-readable symbols to represent problems and logic to solve them.
- **Motivation:** Model the **mind**!

Connectionism

Exploits artificial neural networks & statistics, emphasizing learning from patterns, distributed representations, and emergent behaviors.

- **Assumption:** Intelligence emerges from the interaction of simple and low-level units, i. e. biological neurons.
- **Motivation:** Model the **brain**!

How to Model Intelligence?

Symbolism

Exploits explicit, rule-based symbolic manipulation, logic, and structured reasoning to represent knowledge and solve problems.

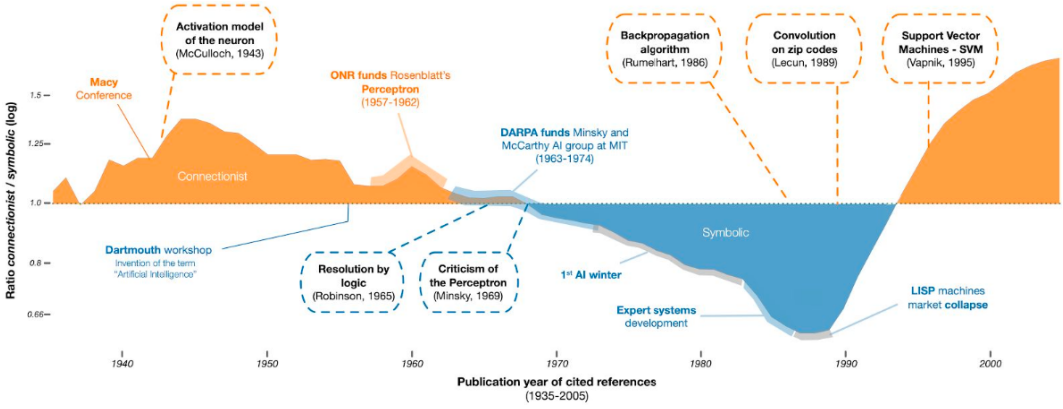
- **Assumption:** Intelligence uses high-level, human-readable symbols to represent problems and logic to solve them.
- **Motivation:** Model the **mind**!

Connectionism

Exploits artificial neural networks & statistics, emphasizing learning from patterns, distributed representations, and emergent behaviors.

- **Assumption:** Intelligence emerges from the interaction of simple and low-level units, i. e. biological neurons.
- **Motivation:** Model the **brain**!

How to Model Intelligence?



Source: (Cardon et al., 2018)

1. How to Model Intelligence?

2. When is Learning Possible?

3. How Do Machines Learn?

4. How Do Humans Train Machines?

5. How to Interact with Leaning Machines?

"The organism feeds on negative entropy." – Erwin Schrödinger

01010101010101010101010101010101

11001101010111010001110010111 1

Let's say we have the following two 30-bit information:

$$x_1 = 01010101010101010101010101$$
$$x_2 = 11001101010111010001110010111$$

Question: Which carries "more" information?

Answer: We say the **entropy** (Shannon, 1948) of x_2 is higher than the entropy of x_1. If we read the bits from left to right we are more **suprised** when reading x_2.

Let's say we have the following two 30-bit information:

$$x_1 = 010101010101010101010101010101$$
$$x_2 = 110011010101110100011100101111$$

Question: Which carries "more" information?

Answer: We say the **entropy** (Shannon, 1948) of x_2 is higher than the entropy of x_1. If we read the bits from left to right we are more **suprised** when reading x_2.

Information entropy measures uncertainty / surprise.

- High entropy: highly unpredictable, many possible outcomes, each similarly likely
- Low entropy: predictable, only a few likely outcomes

Lerning requires

$$\textbf{entropy} + \textbf{structure} \approx \textbf{complexity}.$$

We have to be surprised but also be able to exploit a structure to compress observations.

Information entropy measures uncertainty / surprise.

- High entropy: highly unpredictable, many possible outcomes, each similarly likely
- Low entropy: predictable, only a few likely outcomes

Lerning requires

$$\textbf{entropy} + \textbf{structure} \approx \textbf{complexity}.$$

We have to be surprised but also be able to exploit a structure to compress observations.

A physical formula like

$$F = m \cdot a$$

can be thought of a **highly compressed representation** of some aspects of physical reality!

In other words, $F = m \cdot a$ reveals the structure of motion, thus the informational entropy cannot be too high.

Some neuroscientists (K. Friston, Kilner, & Harrison, 2006; K. J. Friston, 2011) think that organisms try to **minimize surprise** by

1. adjusting expectations (perception, learning)
2. realize expectations (acting)

According to this school of neuroscience,

organisms are their own existence proof.

By acting to keep themselves alive, they continuously generate sensory inputs that confirm their continued existence.

1. How to Model Intelligence?

2. When is Learning Possible?

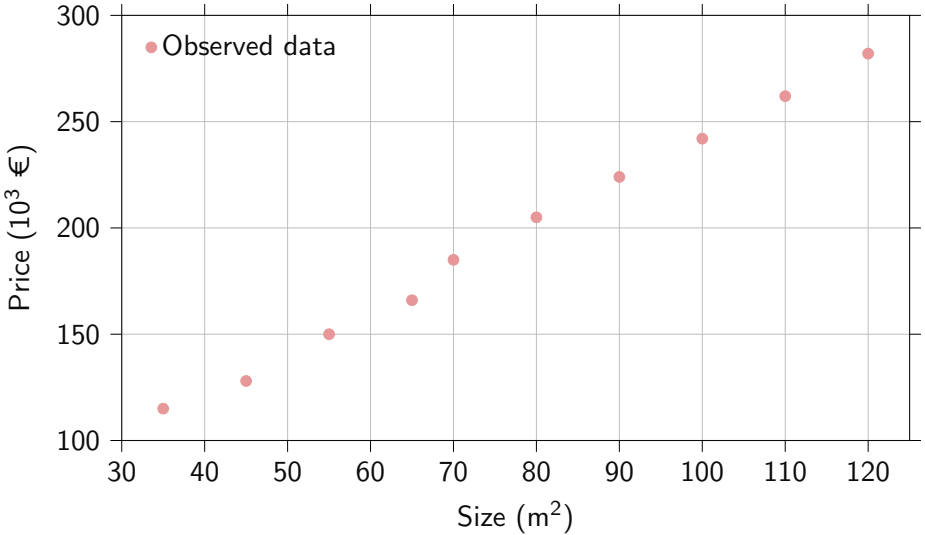
3. How Do Machines Learn?

4. How Do Humans Train Machines?

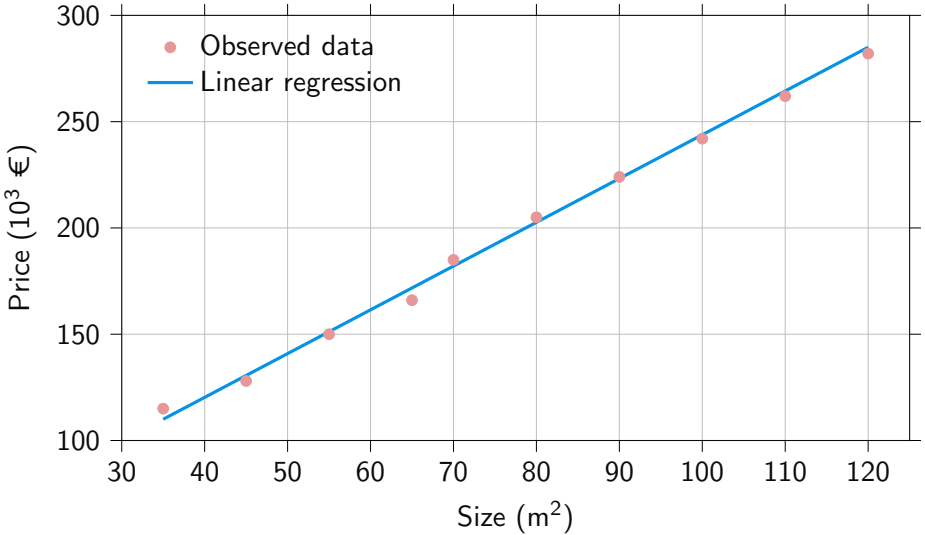
5. How to Interact with Leaning Machines?

"We do not learn from experience [...] we learn from reflecting on experience." – John Dewey

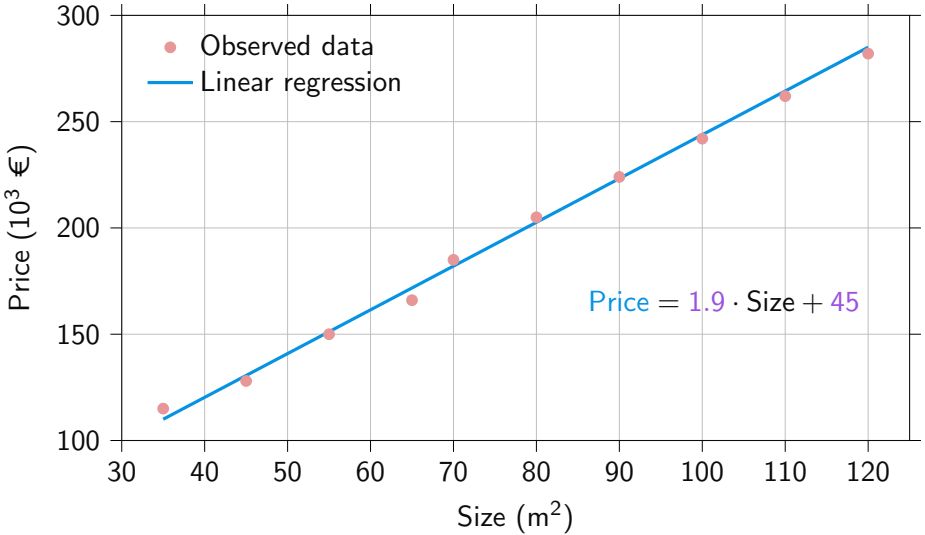
Learning a model of "the world"



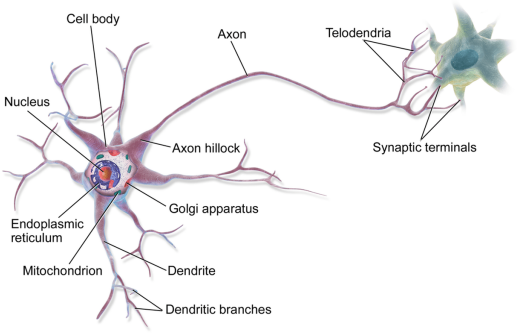
Learning a model of "the world"



Learning a model of "the world"

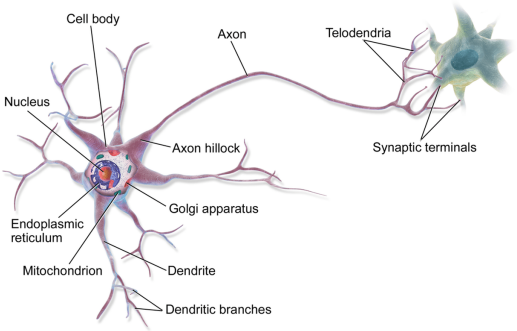


Neuron

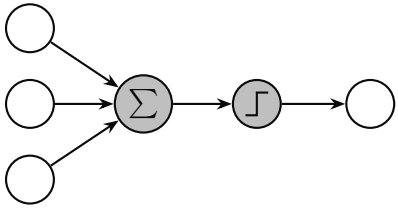


Artificial neuron

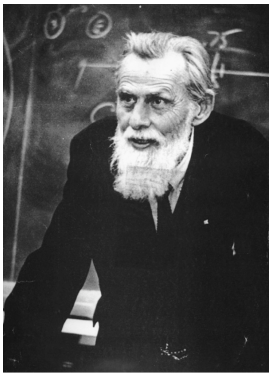
Neuron



Artificial neuron

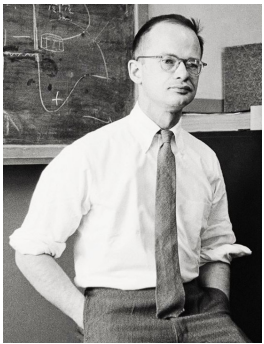


The Mathematical Neuron



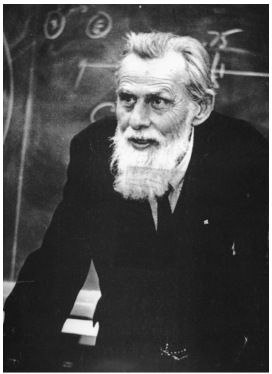
Warren S. McCulloch

- Communication takes place when an electro-chemical signal gets transferred from one neuron to another.
- All the processing then happens in the cell body.
- The processed information then travels to the synaptic terminal which "decides" whether the information should be passed to the next neuron or not.
- The same information passes to millions of neurons.



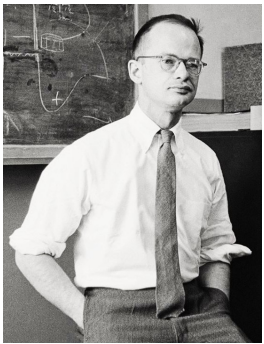
Walter H. Pitts Jr

The Mathematical Neuron



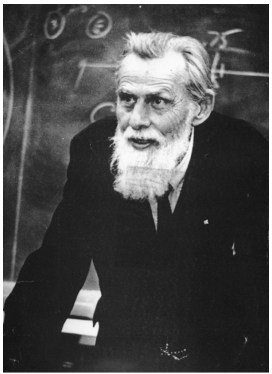
Warren S. McCulloch

- Communication takes place when an electro-chemical signal gets transferred from one neuron to another.
- All the processing then happens in the cell body.
- The processed information then travels to the synaptic terminal which "decides" whether the information should be passed to the next neuron or not.
- The same information passes to millions of neurons.



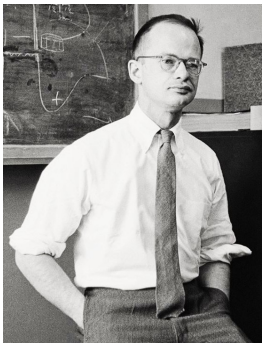
Walter H. Pitts Jr

The Mathematical Neuron



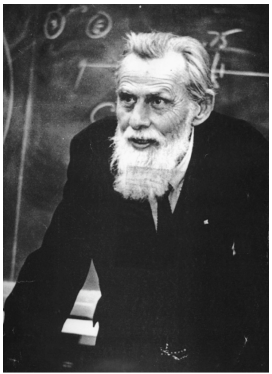
Warren S. McCulloch

- Communication takes place when an electro-chemical signal gets transferred from one neuron to another.
- All the processing then happens in the cell body.
- The processed information then travels to the synaptic terminal which "decides" whether the information should be passed to the next neuron or not.
- The same information passes to millions of neurons.



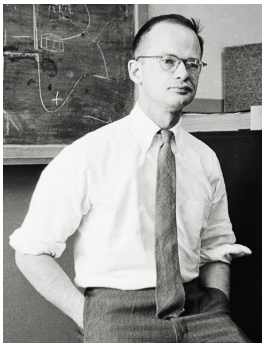
Walter H. Pitts Jr

The Mathematical Neuron



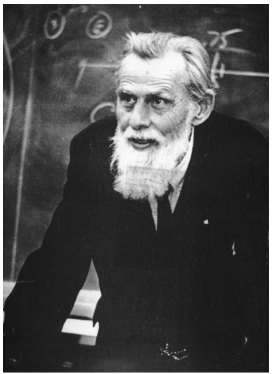
Warren S. McCulloch

- Communication takes place when an electro-chemical signal gets transferred from one neuron to another.
- All the processing then happens in the cell body.
- The processed information then travels to the synaptic terminal which "decides" whether the information should be passed to the next neuron or not.
- The same information passes to millions of neurons.



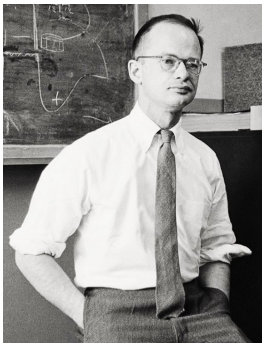
Walter H. Pitts Jr

The Mathematical Neuron



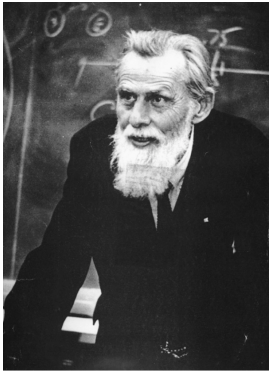
Warren S. McCulloch

- Communication takes place when an electro-chemical signal gets transferred from one neuron to another.
- All the processing then happens in the cell body.
- The processed information then travels to the synaptic terminal which "decides" whether the information should be passed to the next neuron or not.
- The same information passes to millions of neurons.



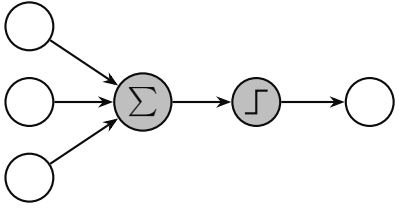
Walter H. Pitts Jr

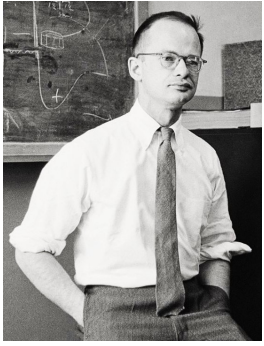
The Mathematical Neuron



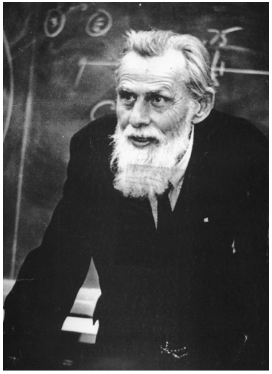
Warren S. McCulloch

"Because of the 'all-or-none' character of nervous activity, neural events and the relations among them can be treated by means of propositional logic."





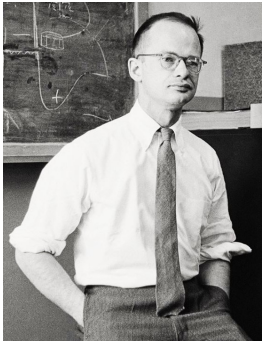
Walter H. Pitts Jr



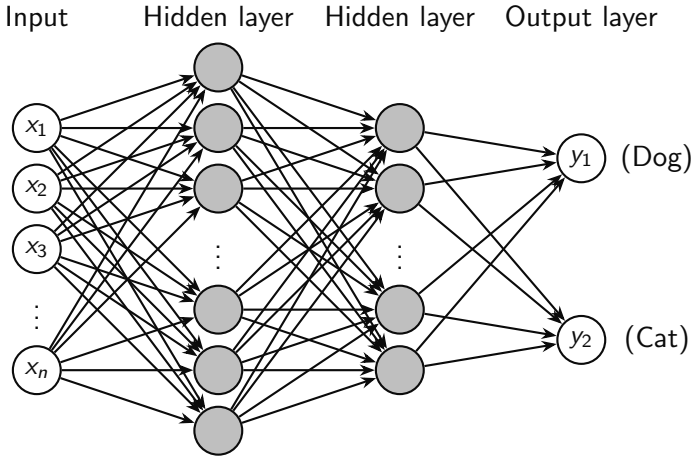
Warren S. McCulloch

"Because of the 'all-or-none' character of nervous activity, neural events and the relations among them can be treated by means of propositional logic."

$$f(x_1, \ldots, x_n) = \begin{cases} 1 & \text{if } \sum_{k=1}^{n} x_k > 1 \\ 0 & \text{otherwise.} \end{cases}$$

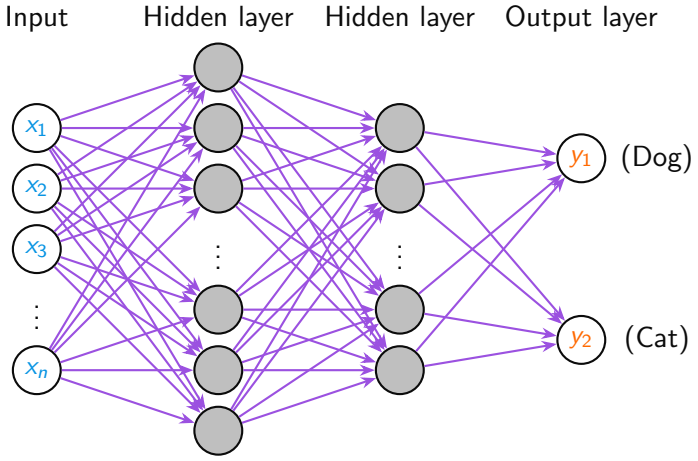


Walter H. Pitts Jr



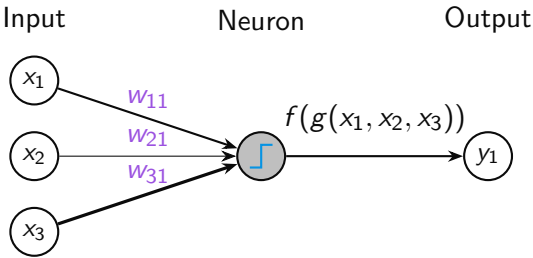
$$h_\theta(\mathbf{x}) = \mathbf{y},$$

where $\mathbf{x} = (x_1, \ldots x_n)$ and $\mathbf{y} = (y_1, \ldots, y_k)$



$$h_\theta(\mathbf{x}) = \mathbf{y},$$

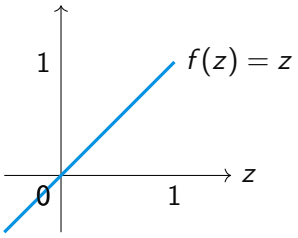
where $\mathbf{x} = (x_1, \ldots x_n)$ and $\mathbf{y} = (y_1, \ldots, y_k)$



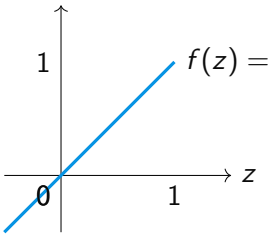
Parameters determine how strong neurons are wired together:

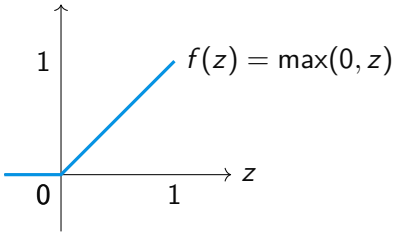
$$g(x_1, x_2, x_3) = x_1 \cdot w_{11} + x_2 \cdot w_{21} + x_3 \cdot w_{31}$$

Activation Functions

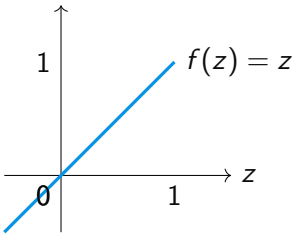


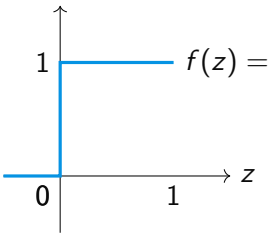
Activation Functions

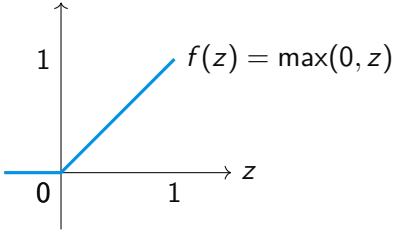




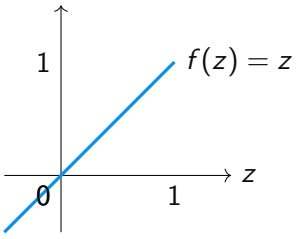
Activation Functions

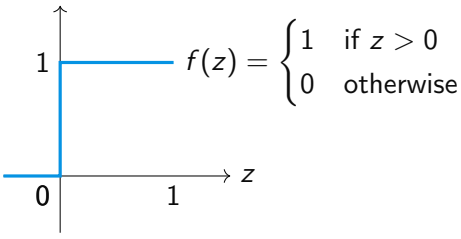


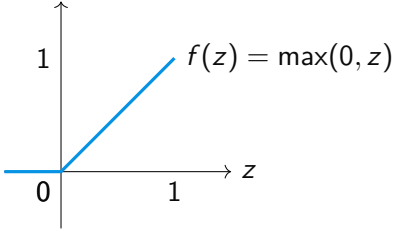


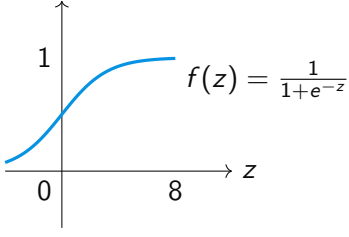


Activation Functions

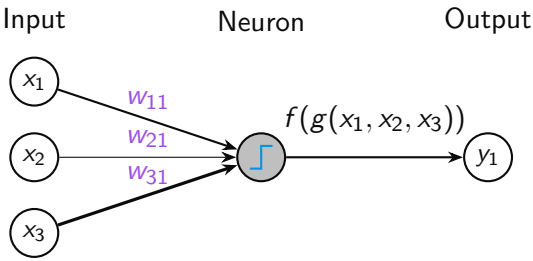






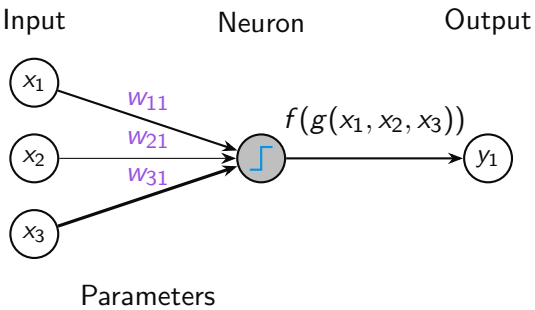


Synaptic Plasticity



"Neurons that fire together, wire together."

$$w_{ij} = w_{ij} - \eta \cdot x_i \cdot y_j$$



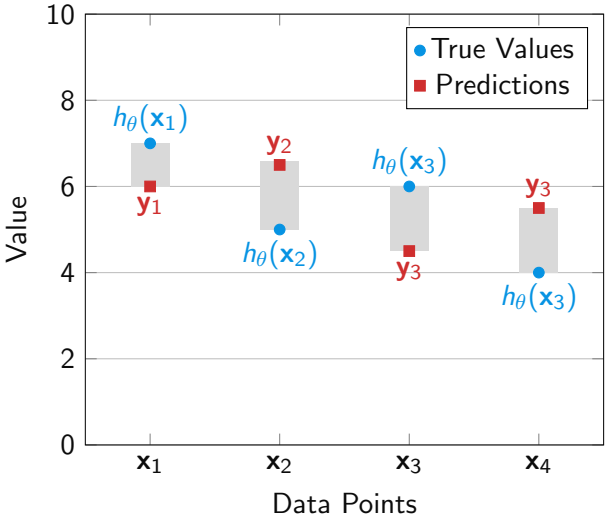
"Neurons that fire together, wire together."

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla J(\theta_t)$$

In this case $\theta_t = (w_{11}, w_{21}, w_{31})$.

Cost Function (Regression)

Mean Squared Error (MSE):



Mean Squared Error (MSE):

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{y}_i - h_\theta(\mathbf{x}_i))^2$$

where $\mathbf{y}_i$ is the correct label of a data point $\mathbf{x}_i = (x_1, \ldots, x_n)$ in our training data.

Example: Let us suppose $h_\theta(x_1, x_2) = w_{11} \cdot x_1 + w_{21} \cdot x_2$ and let us assume $w_{11} = 2, w_{21} = 0.5$ and we have two data points $\mathbf{x}_1 = (1, 1), y_1 = 1$ and $\mathbf{x}_2 = (-1, -2), y_2 = -3$. Then our mean squared error is:

$$\begin{aligned} J(w_{11}, w_{21}) &= \frac{1}{2}\left[(1 - (w_{11} \cdot 1 + w_{21} \cdot 1))^2 + (-3 - (w_{11} \cdot (-1) + w_{21} \cdot (-2)))^2\right] \\ &= \frac{1}{2}\left[(1 - (2 \cdot 1 + 0.5 \cdot 1))^2 + (-3 - (2 \cdot (-1) + 0.5 \cdot (-2)))^2\right] \end{aligned}$$

The gradient would be:

$$\nabla J(w_{11}, w_{21}) = \begin{bmatrix} -4 + 2w_{11} + 3w_{21} \\ -7 + 3w_{11} + 5w_{21} \end{bmatrix}$$

Cost Function (Classification)

Idea: Let's say our preduction clasifies our i-th image, that is $\mathbf{x}_i$, as 0.3 dog and 0.7 cat:

$$h_\theta(\mathbf{x}_i) = (0.3, 0.7)$$

but in reality it is most certainly a dog, that is, $(0.95, 0.05)$. A good error would be:

$$-[0.95 \cdot 0.3 \cdot (1 - 0.95) \cdot (1 - 0.3)] \cdot [0.05 \cdot 0.7 \cdot (1 - 0.05) \cdot (1 - 0.7)]$$

This term is minimal for $\mathbf{x}_i = (0.95, 0.05)$

Cost Function (Classification)

Idea: Let's say our preduction clasifies our i-th image, that is $\mathbf{x}_i$, as 0.3 dog and 0.7 cat:

$$h_\theta(\mathbf{x}_i) = (0.3, 0.7)$$

but in reality it is most certainly a dog, that is, $(0.95, 0.05)$. A good error would be:

$$-\left[0.95 \cdot 0.3 \cdot (1 - 0.95) \cdot (1 - 0.3)\right] \cdot \left[0.05 \cdot 0.7 \cdot (1 - 0.05) \cdot (1 - 0.7)\right]$$

This term is minimal for $\mathbf{x}_i = (0.95, 0.05)$

Idea: Let's say our preduction clasifies our i-th image, that is $\mathbf{x}_i$, as 0.3 dog and 0.7 cat:

$$h_\theta(\mathbf{x}_i) = (0.3, 0.7)$$

but in reality it is most certainly a dog, that is, $(0.95, 0.05)$. A good error would be:

$$-\left[0.95 \cdot 0.3 \cdot (1 - 0.95) \cdot (1 - 0.3)\right] \cdot \left[0.05 \cdot 0.7 \cdot (1 - 0.05) \cdot (1 - 0.7)\right]$$

This term is minimal for $\mathbf{x}_i = (0.95, 0.05)$

Categorical Cross Entropy Cost:

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} [\mathbf{y}_i \cdot \log(h_\theta(\mathbf{x}_i)) + (\mathbf{1} - \mathbf{y}_i) \cdot (\mathbf{1} - \log(h_\theta(\mathbf{x}_i)))]$$

where $\mathbf{y}_i$ is interpreted as the probability distribution of categories for $\mathbf{x}_i = (x_1, \ldots, x_n)$, i. e. a data point.

To improve the model's prediction, we try to minimize the cost function. One way to do this is **gradient decent**:

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla J(\theta_t)$$

Condition: $\nabla J(\theta_t)$ exits!

Interactive Tutorial

To improve the model's prediction, we try to minimize the cost function. One way to do this is **gradient decent**:

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla J(\theta_t)$$

Condition: $\nabla J(\theta_t)$ exits!

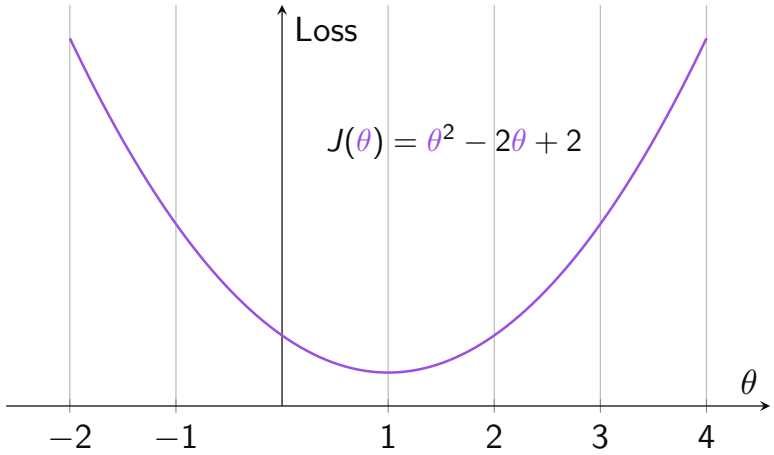
Interactive Tutorial

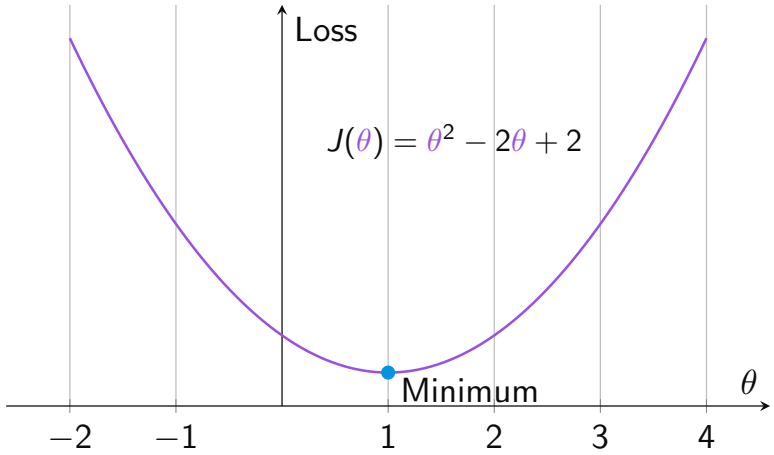
To improve the model's prediction, we try to minimize the cost function. One way to do this is **gradient decent**:

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla J(\theta_t)$$

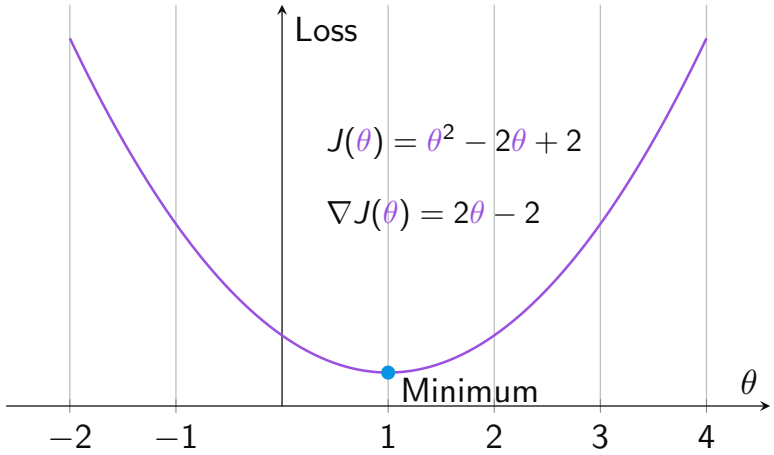
Condition: $\nabla J(\theta_t)$ exits!

Interactive Tutorial

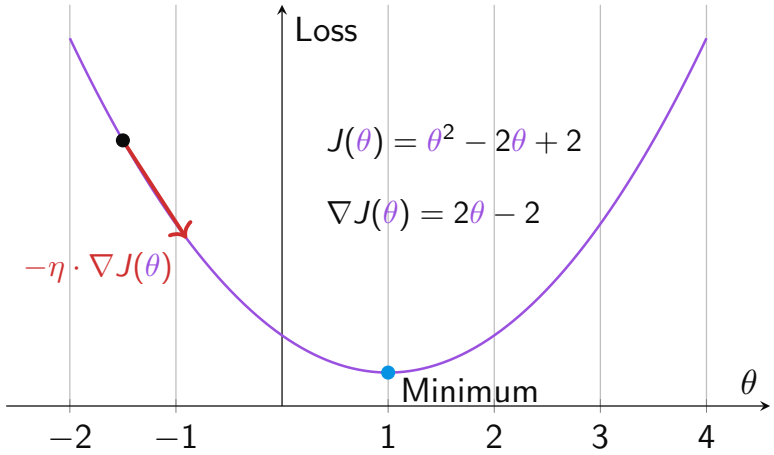

Loss
$J(\theta) = \theta^2 - 2\theta + 2$
θ
-2
-1
1
2
3
4


Loss
$J(\theta) = \theta^2 - 2\theta + 2$
Minimum
θ
-2
-1
1
2
3
4

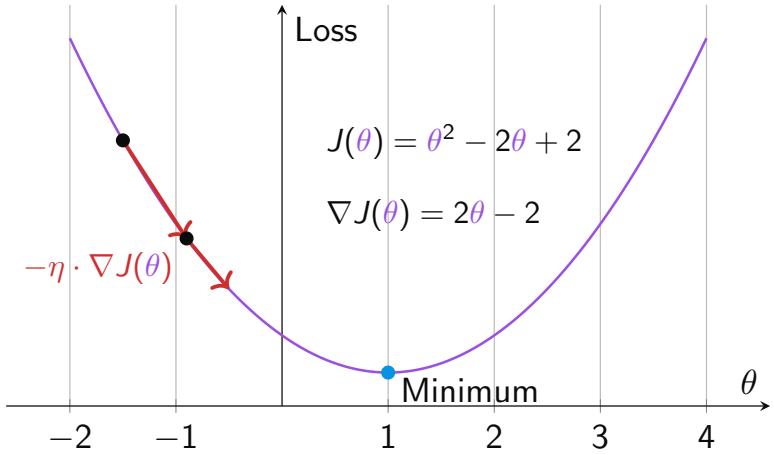
Gradient Decent



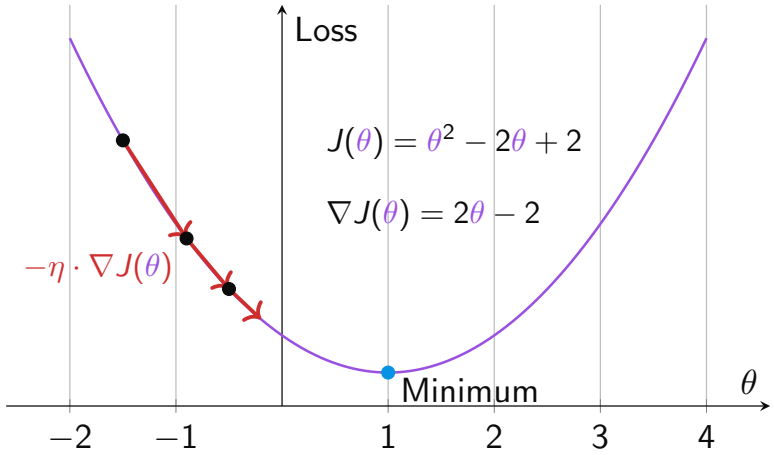
Gradient Decent



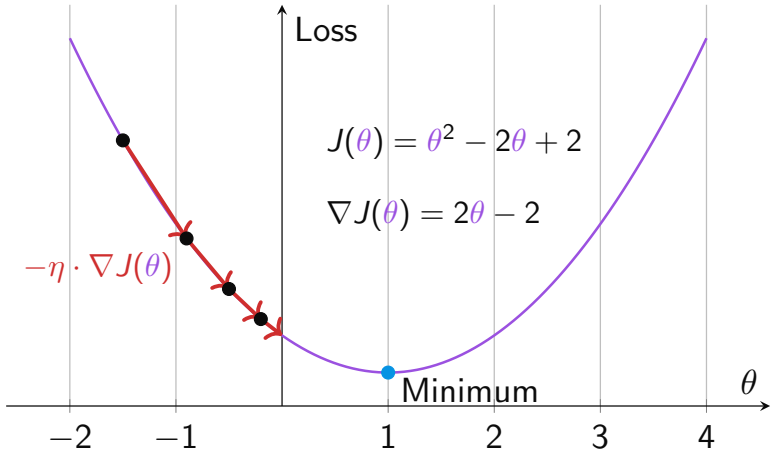
Gradient Decent



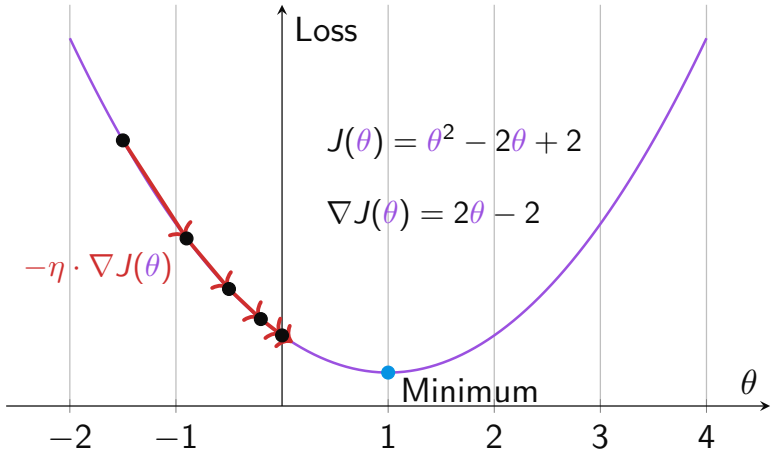
Gradient Decent



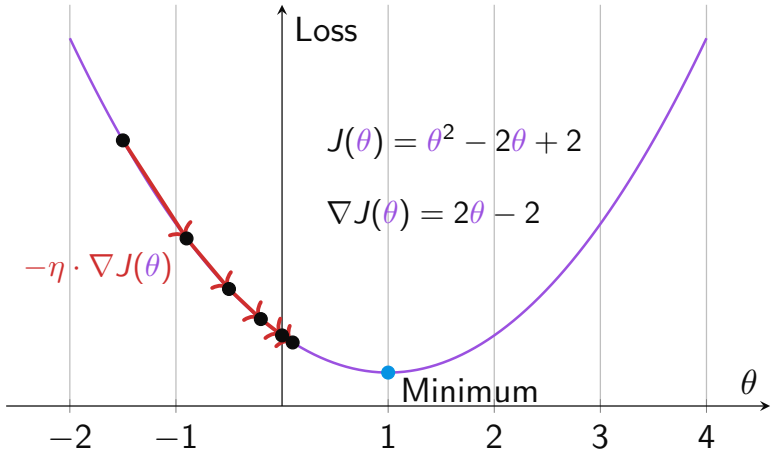
Gradient Decent



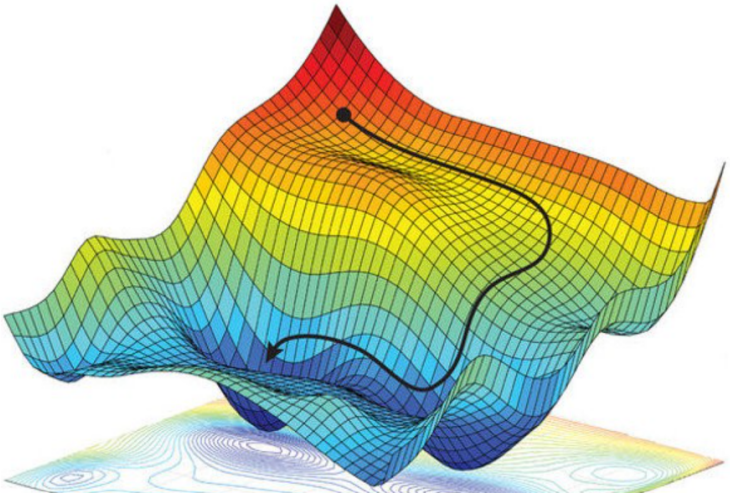
Gradient Decent



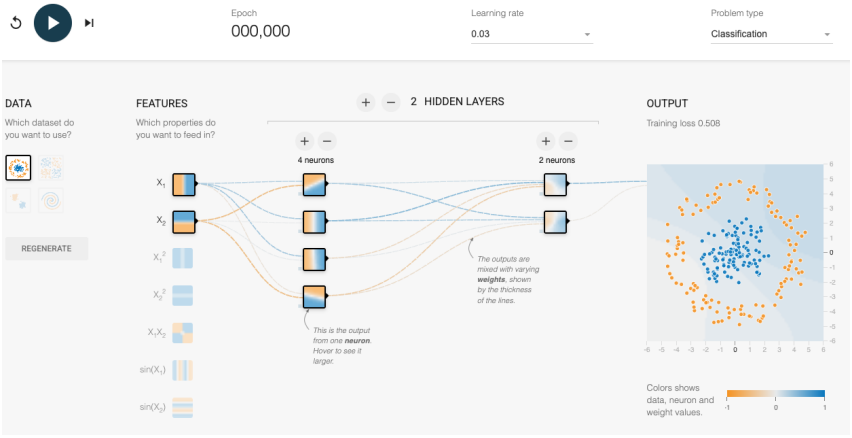
Gradient Decent



Gradient Decent



Design and Try Your Perceptron

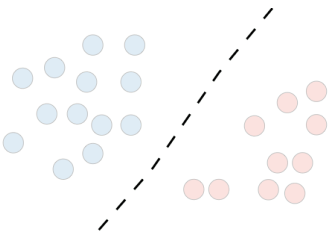


Simplified Tensorflow Playground

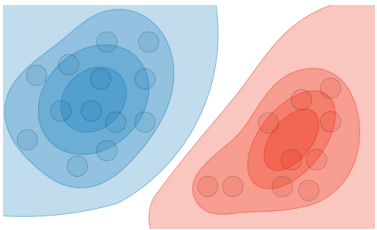
Extended Tensorflow Playground

Modeltypes

- **Discriminative models:** Learn the boundaries of decisions.
- **Generative models:** Learn the whole distribution of the data.



Discriminative modelling



Generative modelling

1. How to Model Intelligence?

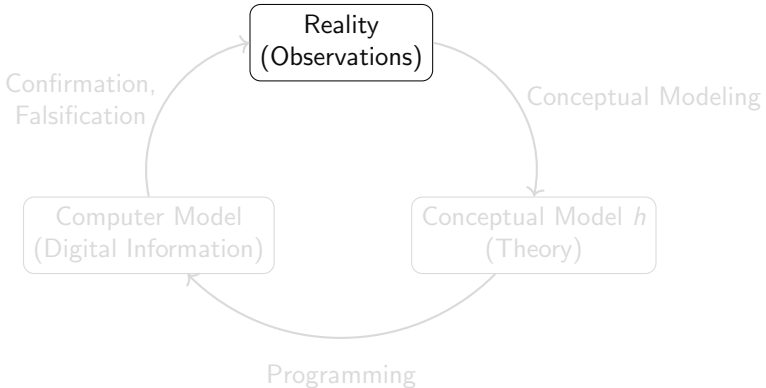
2. When is Learning Possible?

3. How Do Machines Learn?

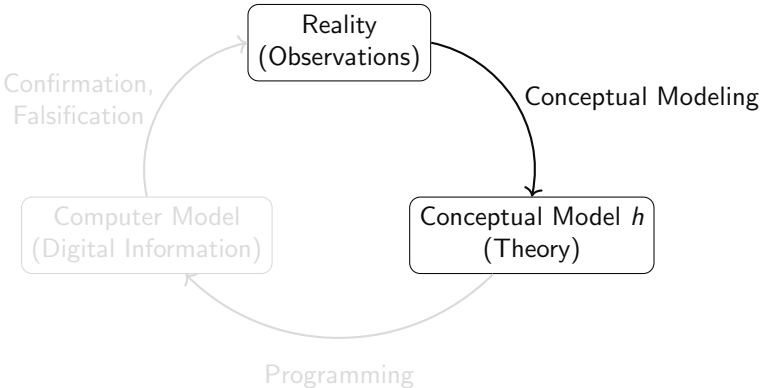
4. How Do Humans Train Machines?

5. How to Interact with Leaning Machines?

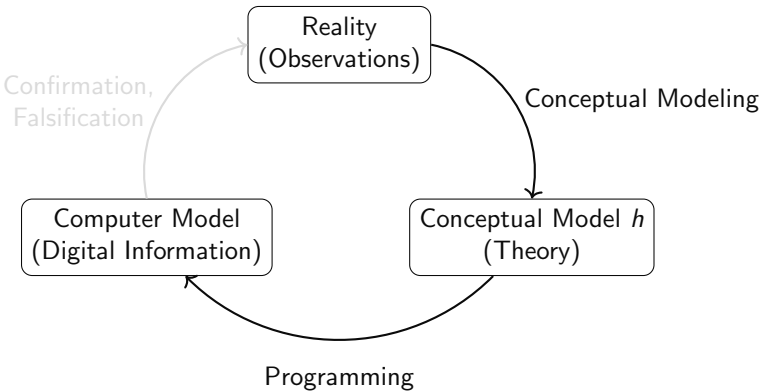
Theory-driven Modeling



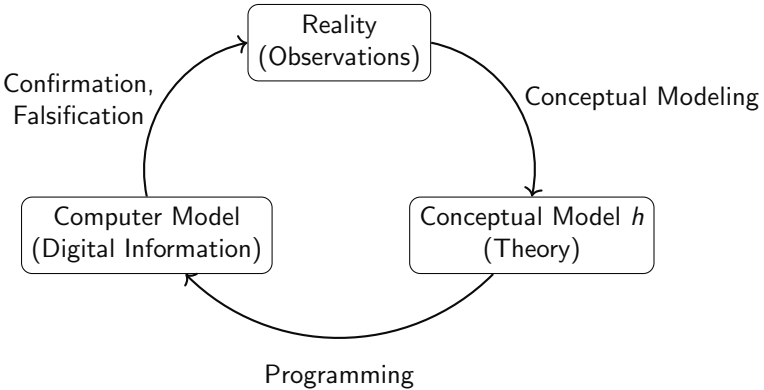
Minds contructs a (falsifiable) theory or hypothesis about reality to test against.



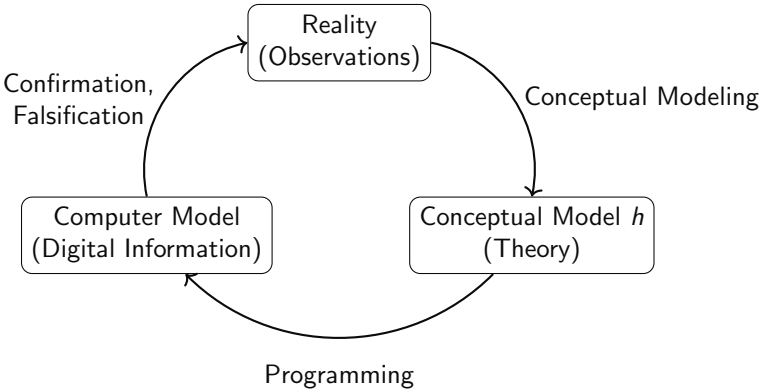
Minds contructs a (falsifiable) theory or hypothesis about reality to test against.



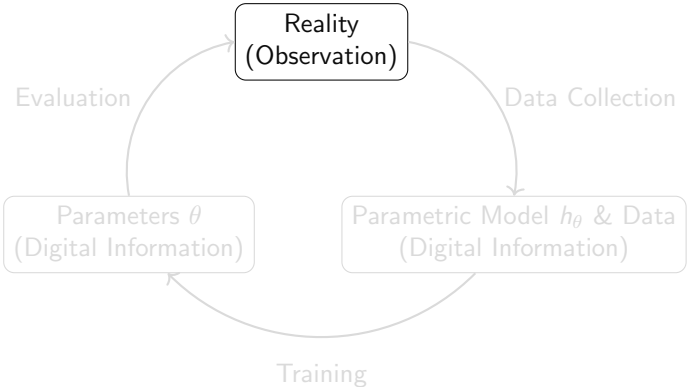
Minds contructs a (falsifiable) theory or hypothesis about reality to test against.



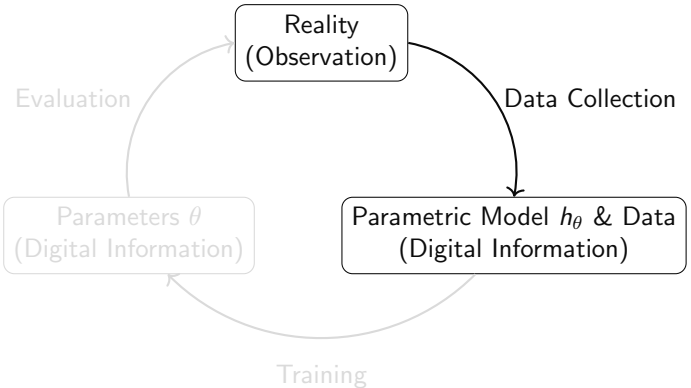
Minds contructs a (falsifiable) theory or hypothesis about reality to test against.



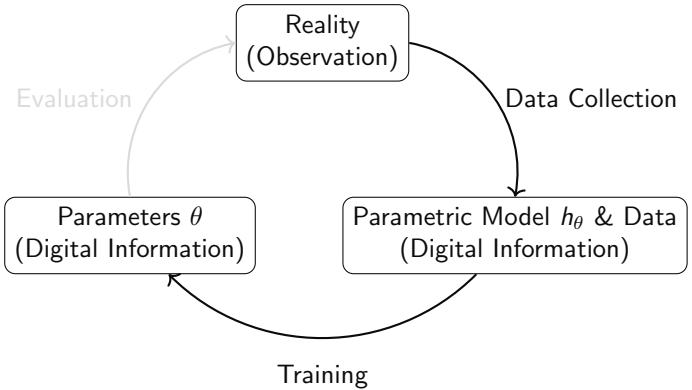
Minds contructs a (falsifiable) theory or hypothesis about reality to test against.



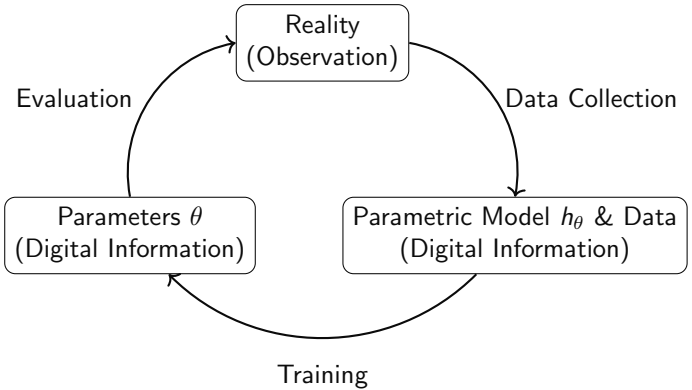
Algorithms (directly) fit a parametric model to the data. **Minds** are usually unable to conceptualize the trained model.



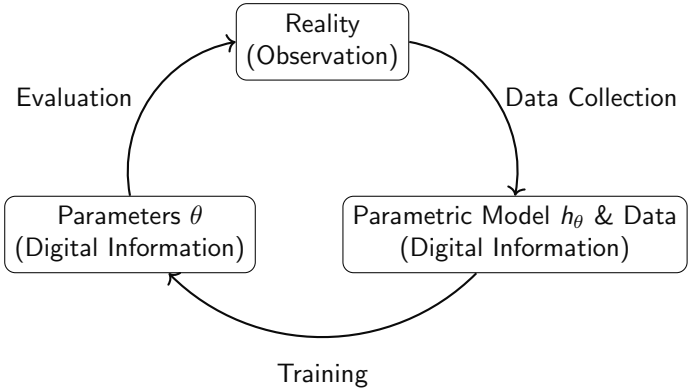
Algorithms (directly) fit a parametric model to the data. **Minds** are usually unable to conceptualize the trained model.



Algorithms (directly) fit a parametric model to the data. **Minds** are usually unable to conceptualize the trained model.

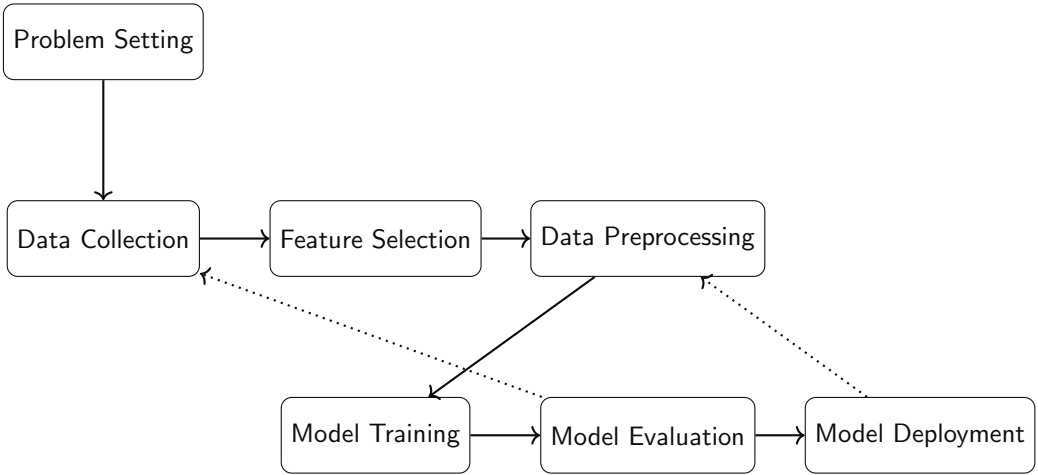


Algorithms (directly) fit a parametric model to the data. **Minds** are usually unable to conceptualize the trained model.



Algorithms (directly) fit a parametric model to the data. **Minds** are usually unable to conceptualize the trained model.

Development Cycle



```
100  class fast_qlinear(torch.autograd.Function):
101      def forward(ctx, a, b, scales, zeros):
102
103          m, k = a.shape
104          _, n = b.shape
105
106          quant_groupsize = 128
107          block_size_m = 16
108          block_size_n = 32 # [N = 4096 // 32] = 128 blocks
109          block_size_k = 256
110          group_size_m = 8
111          num_warps = 4
112          num_stages = 8
113          total_blocks_m = triton.cdiv(m, block_size_m)
114          total_blocks_n = triton.cdiv(n, block_size_n)
```

Python and ML libraries (PyTorch, tensorflow, JAX etc.)

Train a Model with Python

1. How to Model Intelligence?

2. When is Learning Possible?

3. How Do Machines Learn?

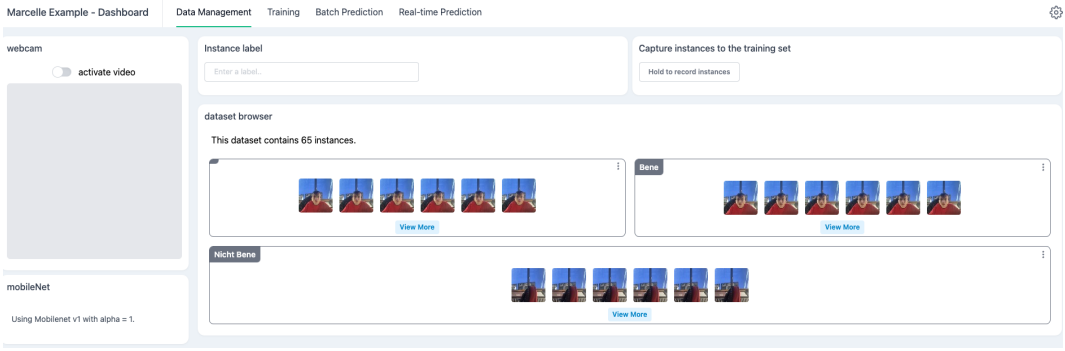
4. How Do Humans Train Machines?

5. How to Interact with Leaning Machines?

Marcelle: composing interactive machine learning workflows and interfaces (Françoise, Caramiaux, & Sanchez, 2021).

https://marcelle.dev/

The Marcelle Toolkit



1. How to Model Intelligence?
2. When is Learning Possible?
3. How Do Machines Learn?
4. How Do Humans Train Machines?
5. How to Interact with Leaning Machines?

Any questions?

References I

Cardon, D., Cointet, J.-P., & Mazieres, A. (2018). Neurons spike back: The invention of inductive machines and the artificial intelligence controversy. *Reseaux*, *36*, 173–220.

Françoise, J., Caramiaux, B., & Sanchez, T. (2021). Marcelle: Composing interactive machine learning workflows and interfaces. In *The 34th annual acm symposium on user interface software and technology* (pp. 39–53). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/3472749.3474734

Friston, K., Kilner, J., & Harrison, L. (2006). A free energy principle for the brain. *Journal of Physiology-Paris*, *100*(1), 70–87. doi: 10.1016/j.jphysparis.2006.10.001

Friston, K. J. (2011). Embodied inference: or "I think therefore I am, if I am what I think". In W. Tschacher & C. Bergomi (Eds.), *The implications of embodiment (cognition and communication)* (pp. 89–125). Exeter, UK: Imprint Academic.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell Syst. Tech. J.*, *27*(3), 379-423.