Introduction & Deep Learning Basics

SS 2020 – Machine Perception

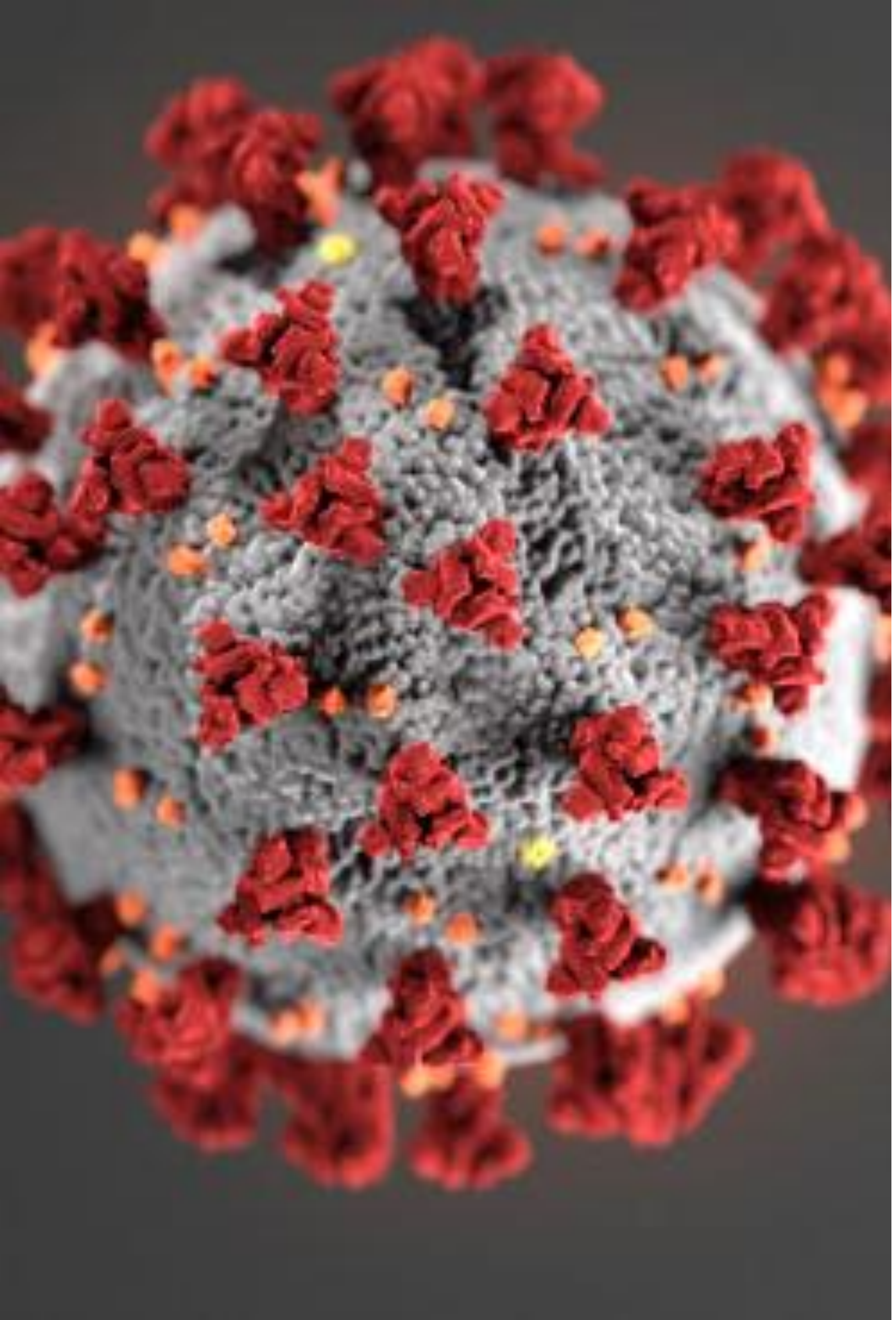
Otmar Hilliges

6 March 2020

Advanced Interactive Technologies

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich


R



Coronavirus

Guidance from the rector:

- Teaching will continue
- Anyone with flu-like symptoms (fever, coughing or breathing difficulties)
- And travellers to/from high-risk areas (China, Italy, South Korea, Iran), are ***not allowed to*** attend class or enter ETH premises

What we do:

- Lectures will continue and are recorded
- Use own judgement and stay home if in doubt

Course Logistics

Lecture
- Thursday 10:15 – 11:55 [CAB G 11 »]

Exercise:
- Thursday 13:15 – 15:00 [CAB G 11 »] or
- Friday 13:15 – 15:00 [CAB G 11 »]
- 4 pen & paper assignments (not graded)
- 4 interactive tutorials
- 2 lecture-style classes with practical tips
- 1 multi-week project **(40% of grade)**

Machine Perception (263-3710-00L)
2L + 1U + 1A
5 ECTS Credits

Audience:
- Computer Science Master | RSC

The lecture will be recorded

https://www.video.ethz.ch/lectures/d-infk/2020/spring/263-3710-00L

user: hil-20s
pw: 3Ewk6Fv

Course Logistics

Machine Perception (263-3710-00L)
2L + 1U + 1A
5 ECTS Credits

Audience:

- Computer Science Master | RSC

Slides will be available the day of the lecture:

https://ait.ethz.ch/teaching/courses/2020-SS-Machine-Perception/

user: mp-2020
pw: Perception20

Write this down now!

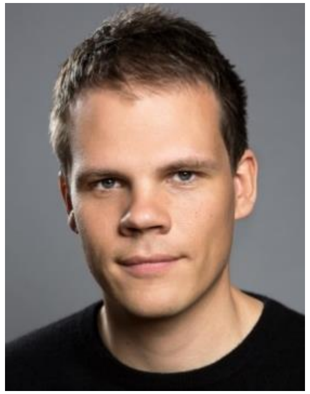
Lecture

- Thursday 10:15 – 11:55 CAB G 11 »

Exercise:

- Thursday 13:15 – 15:00 CAB G 11 » or
- Friday 13:15 – 15:00 CAB G 11 »
- 4 pen & paper assignments (not graded)
- 4 interactive tutorials
- 2 lecture-style classes with practical tips
- 1 multi-week project **(40% of grade)**

Lecturer:



Otmar Hilliges
STD H24
hilliges@inf.ethz.ch

Assistants:

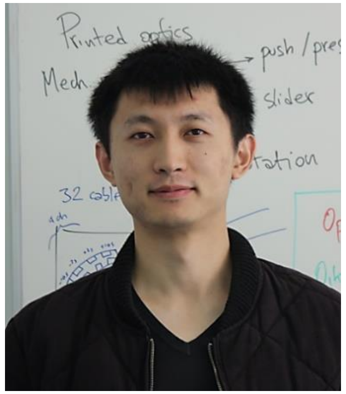
Emre Aksan
STD H27
eaksan@inf.ethz.ch

Seonwook Park
STD H27
spark@inf.ethz.ch

Adrian Spurr
STD H27
spurra@inf.ethz.ch

Manuel Kaufmann
STD H 29.1
kamanuel@inf.ethz.ch

Siyu Tang
CNB G104
siyu.tang@inf.ethz.ch



Jie Song
STD G23
jsong@inf.ethz.ch

Xucong Zhang
STD H29.2
zhangxuc@inf.ethz.ch

Xu Chen
STD G23
xuchen@inf.ethz.ch

Sammy Christen
STD G28
sammyc@inf.ethz.ch

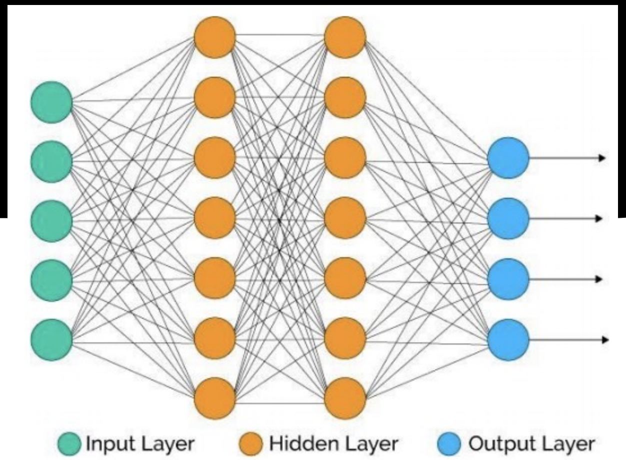
Machine Perception

Machine perception is the *capability* of a computer system to *interpret data* in a manner that is *similar to the way humans use their senses* to *relate to the world around them*. Computing systems perceive their environment through some form of sensing hardware.

Machine perception - Motivation

+

MAGIC BLACKBOX

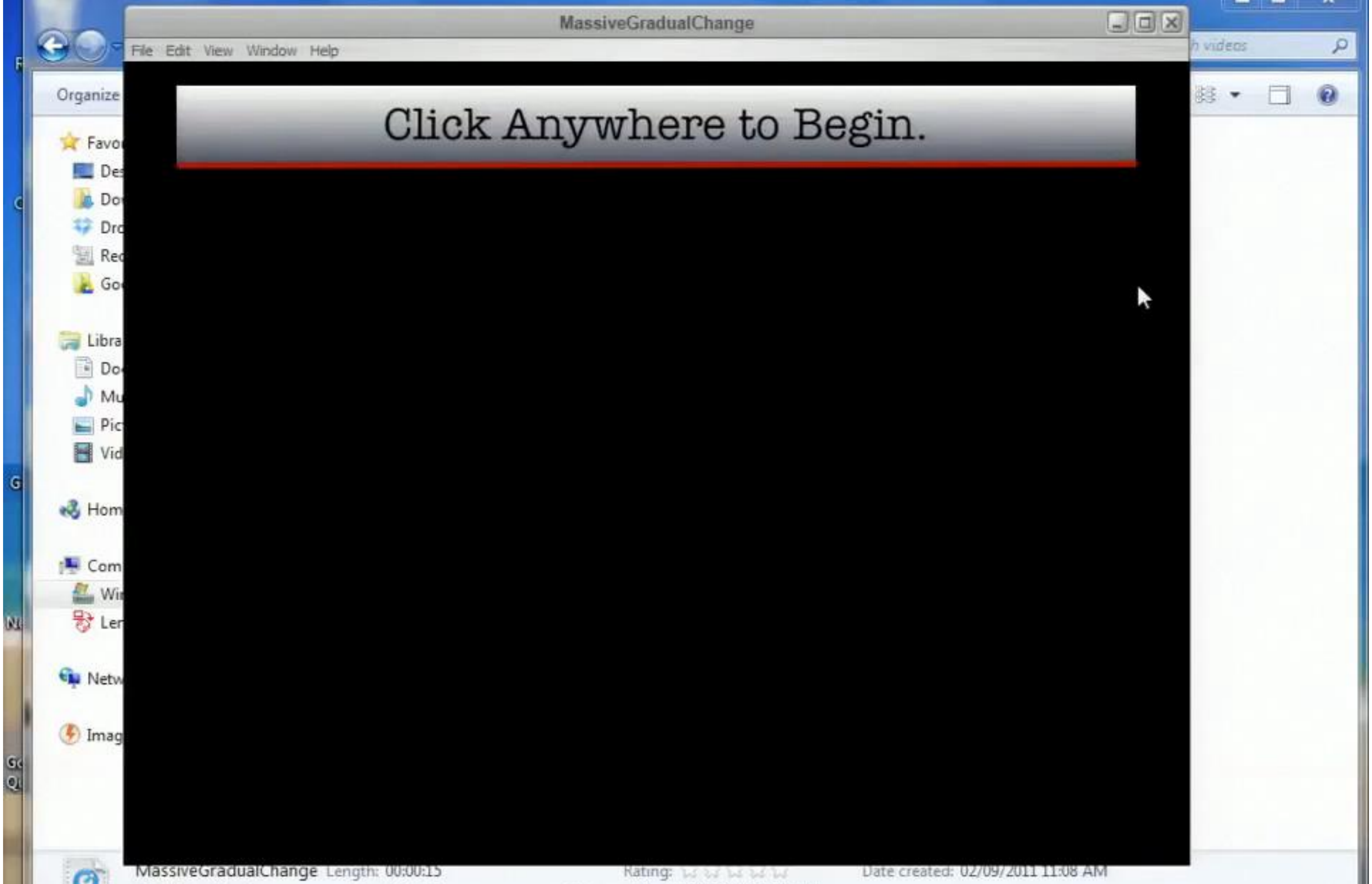


=

Learning Objectives

After completing the course students will have acquired *theoretical and practical knowledge* about the most important problems in *machine understanding (of human behavior).*

Students will be able to *understand and reason* about machine perception algorithms and will be able to *design novel* architectures for hard problems in machine perception.


MassiveGradualChange
File Edit View Window Help
Click Anywhere to Begin.
Organize
MassiveGradualChange Length: 00:00:15
Rating:
Date created: 02/09/2011 11:08 AM


MassiveGradualChange
File Edit View Window Help
What Changed? Click to See Original.
LA MERE MITRON
©CVCL Lab, MIT

Question of the Day:

If you don’t know how you see

–

How are you going to program a machine to do it?

Learning Objectives

After completing the course students will have acquired *theoretical and practical knowledge* about the most important problems in *machine understanding (of human behavior).*

Students will be able to *understand and reason* about Deep Learning architectures and will be able to *design novel* architectures for hard problems in machine perception.

Our philosophy

- Thorough and detailed:
  - Understand how to write from scratch, debug and train (convolutional) neural networks.
  - Understand underlying Math.
- Practical:
  - In addition to deriving and understanding algorithms students will learn how to solve non-trivial problems with DL approaches via exercises and multi-week project
- Research Oriented:
  - Most materials are new from research world in the past 1-3 years. Very exciting stuff!
  - Introduction to open research problems in learning-based computer vision, robotics and HCI

Course Requirements

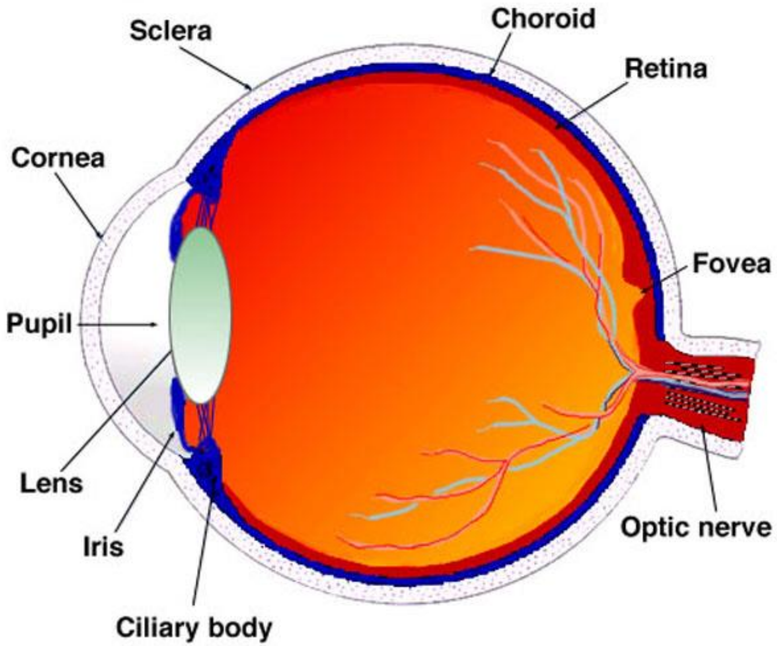
Required:

- Solid programming skills (practical part in Python)
- Solid understanding of linear algebra, vector calculus and (some) continuous optimization
- Basic understanding of statistics (probabilities and distributions, Bayes)
- ‘Traditional’ machine learning concepts (ML tasks, loss functions, regularization, etc.)

Beneficial:

- Basic understanding of image and signal processing
- Visual Computing class
- Computer Vision class

Motivation – How Do I See?



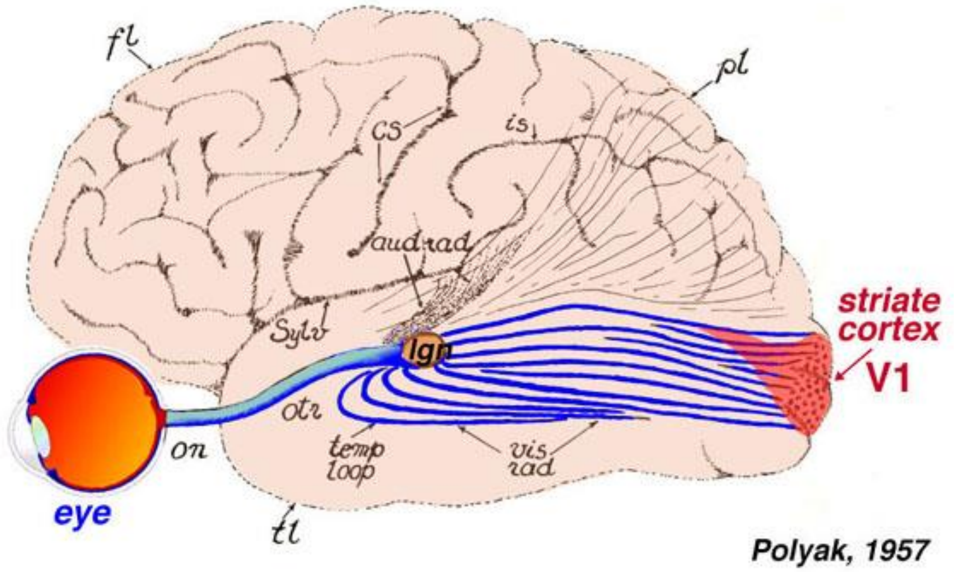
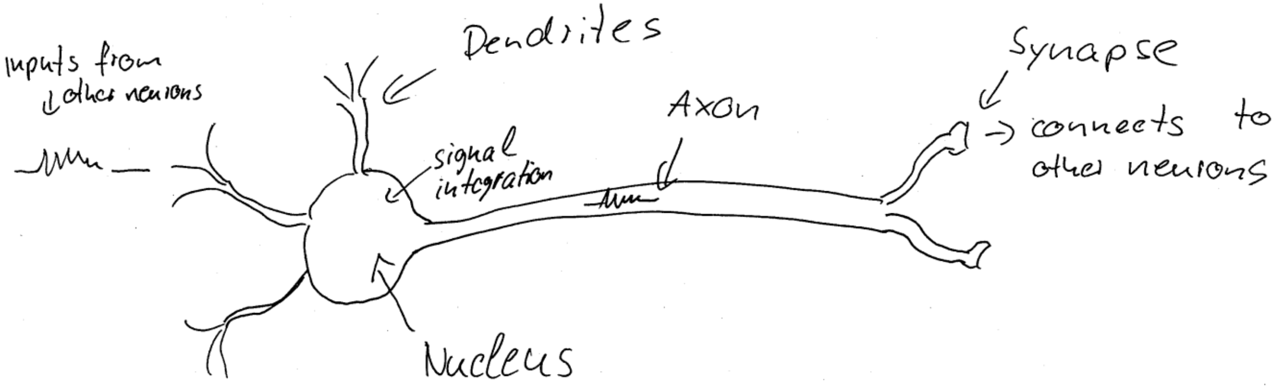
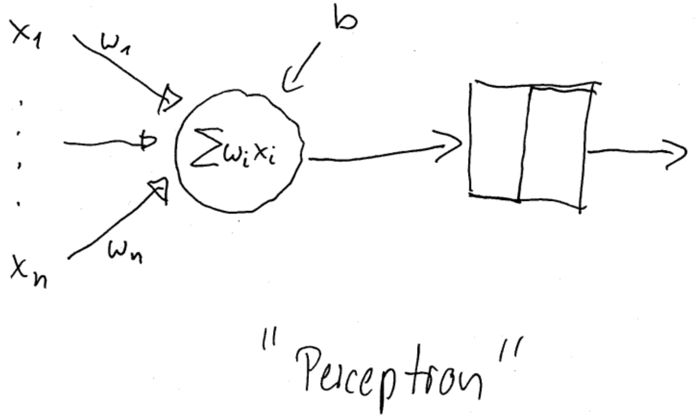


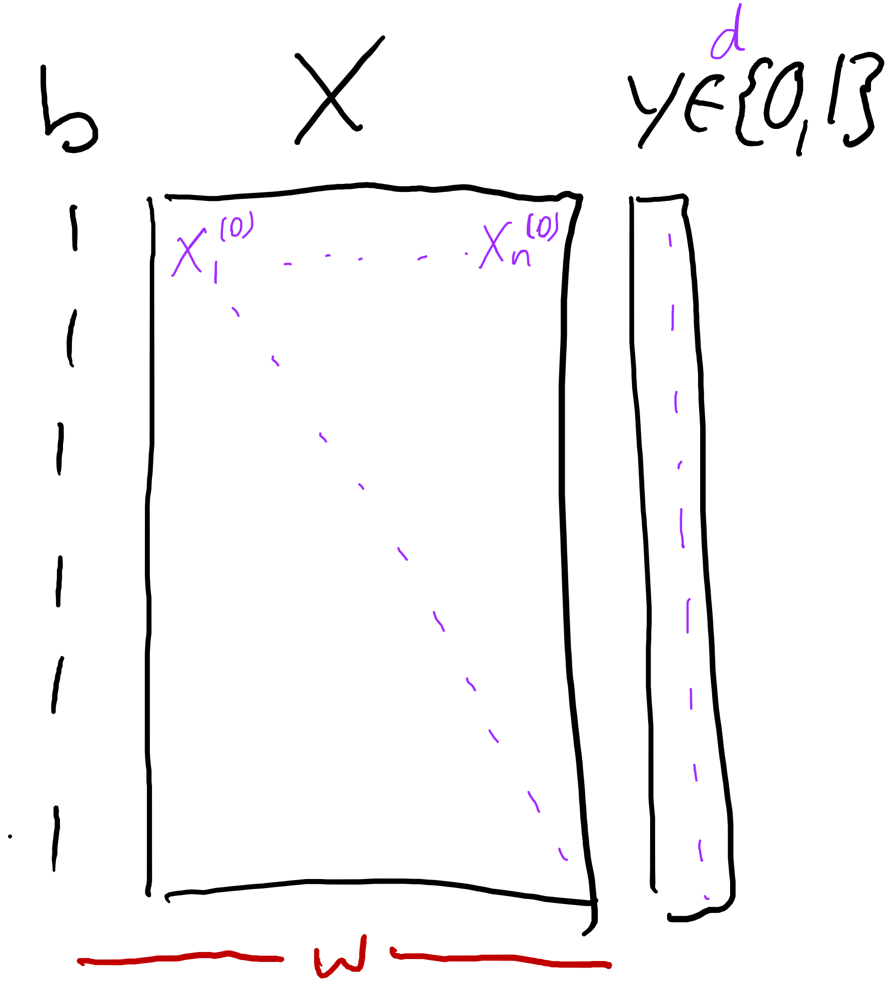
Figure 8. Visual input to the brain goes from eye to LGN and then to primary visual cortex, or area V1, which is located in the posterior of the occipital lobe.
Adapted from Polyak (1957).

Neuron – An engineering blueprint





Perceptron learning



init: $w^0 = 0$

while: $\exists\, i_k$ s.t. $({w^k}^T x^{i_k} \geq 0) \neq y^{i_k}$

do: $w^{k+1} = w^k + \eta \underbrace{(y - \hat{y})}_{\text{error}} x^{i_k}$

where: $\hat{y} = (w^T x \geq 0)$

y	$\hat{y}$	$(y - \hat{y})$
0	0	0
0	1	-1
1	0	1
1	1	0

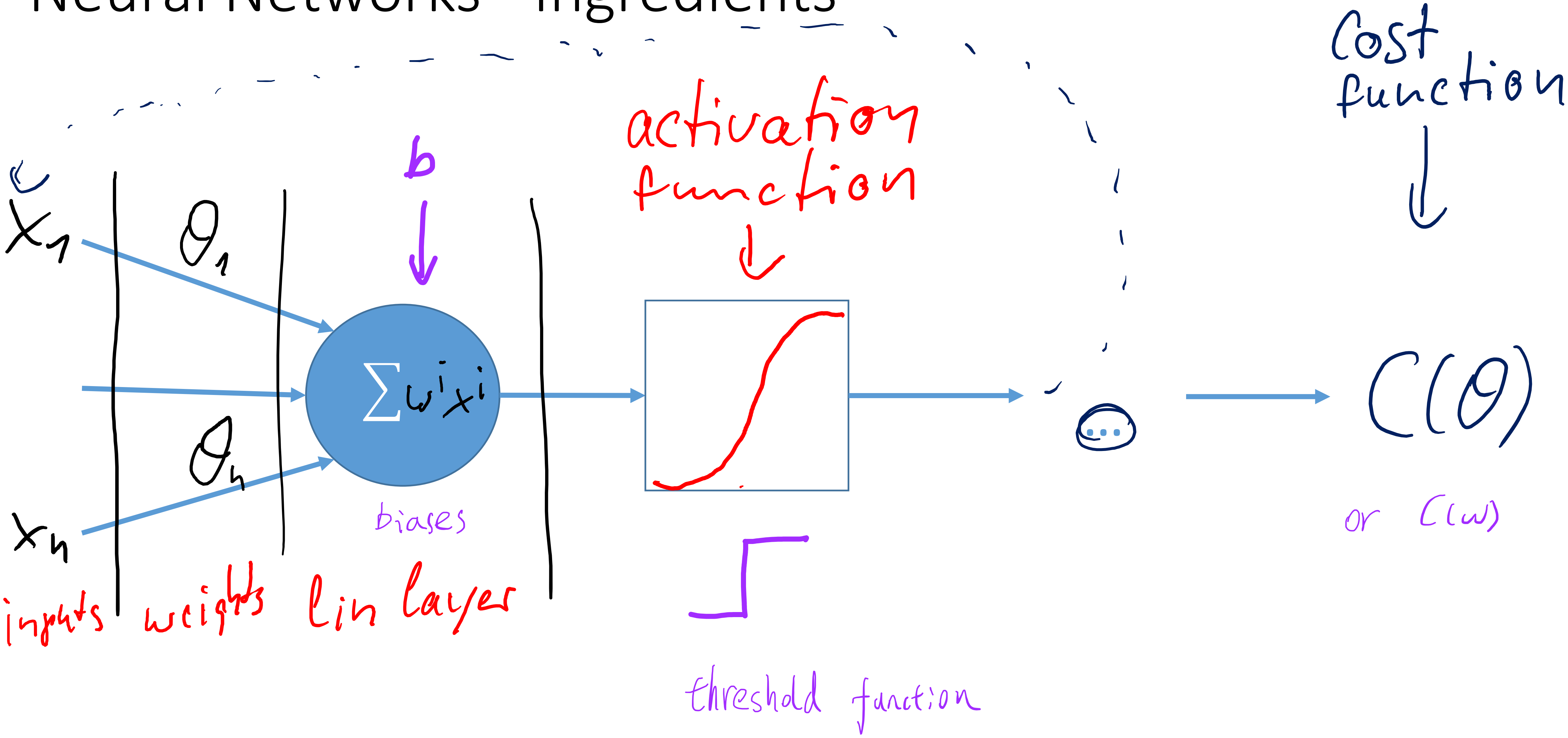
data is iff linearly sep

$\Rightarrow$ converges in finite time

Sample code & visualization

https://colab.research.google.com/drive/1AfVZpBwbOfMqN5wkOmAmhKj3pmLOjPfc#scrollTo=4oylaFh3OMPs

Neural Networks - Ingredients

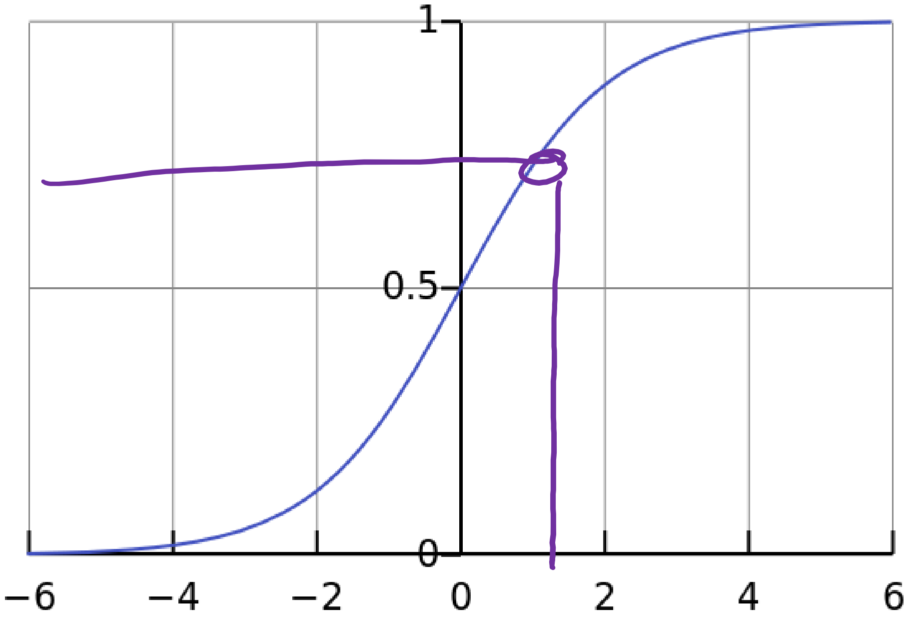


Sigmoid function

$sigm(\eta)$ is the sigmoid function also known as logistic function

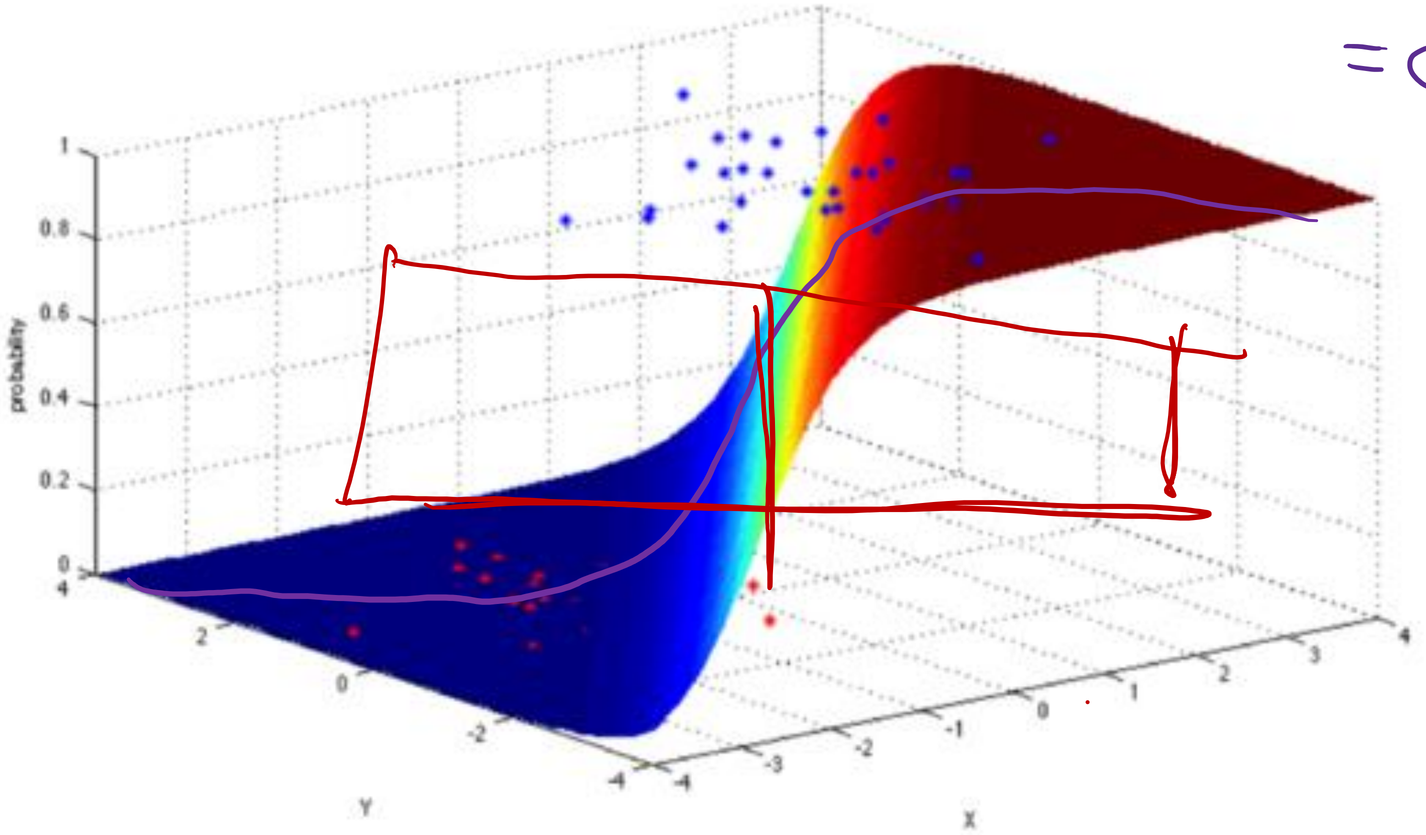
$$sigm(\eta) = \frac{1}{1+e^{-\eta}} = \frac{e^{\eta}}{e^{\eta}+1}$$

$\eta = \bar{w}^T \bar{x}$



$P(y^{(i)}=1 | X, w)$

Aside: Logistic regression



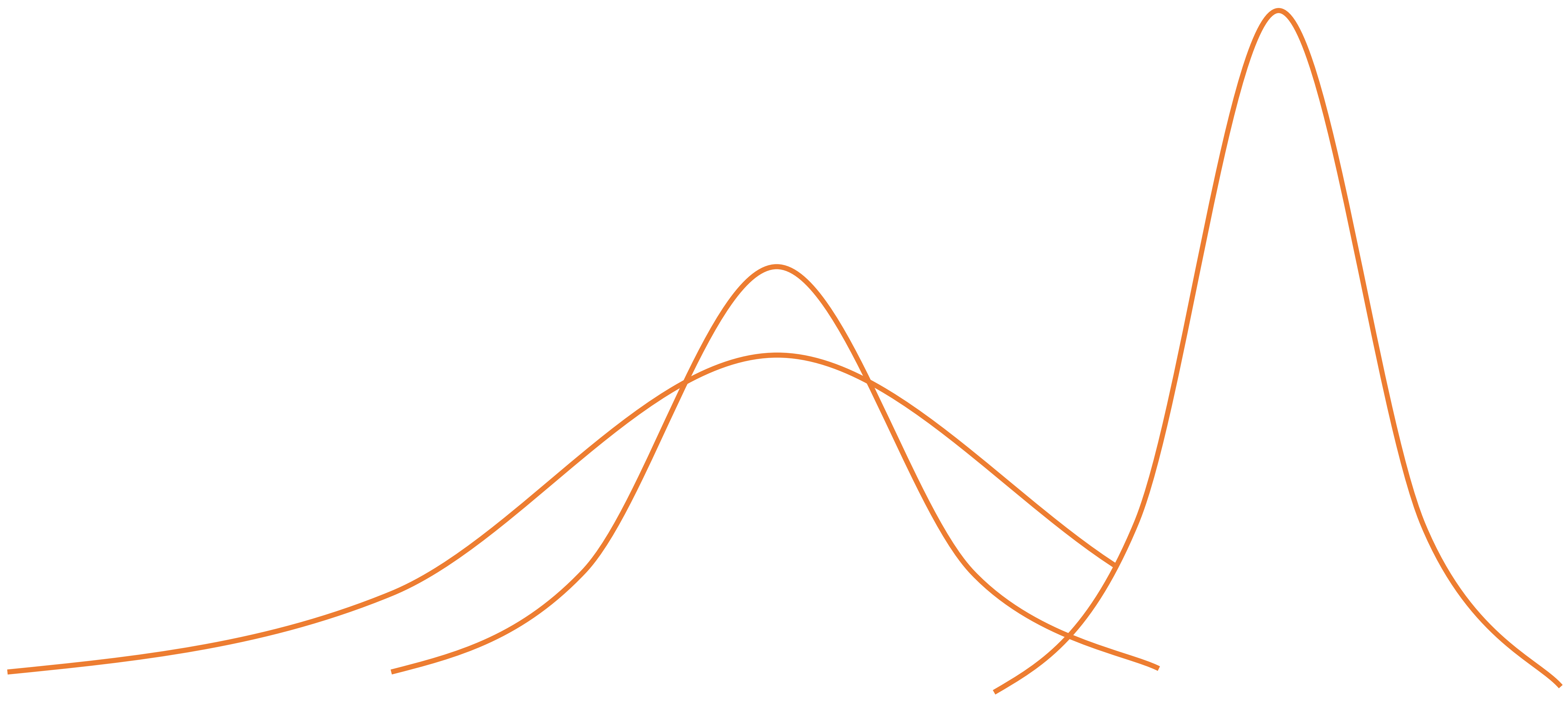
$$p(y^{(i)} = 1 \mid x^{(i)}, w)$$
$$= \sigma(\bar{w}^T \bar{x}) = \frac{1}{2}$$
$$\Updownarrow$$
$$\bar{w}^T \bar{x} = 0$$
$$\frac{1}{1 + e^{(0)}} = \frac{1}{2}$$

Likelihood and Maximum Likelihood (MLE)

[Fisher, 1929]

- Likelihood is a function of the model parameters
- MLE extremely successful:
    - e.g. least squares and cross-entropy are MLE estimators
- Direct link to the Kullback Leibler divergence

$$\arg\max_{w} p(y \mid x, w)$$

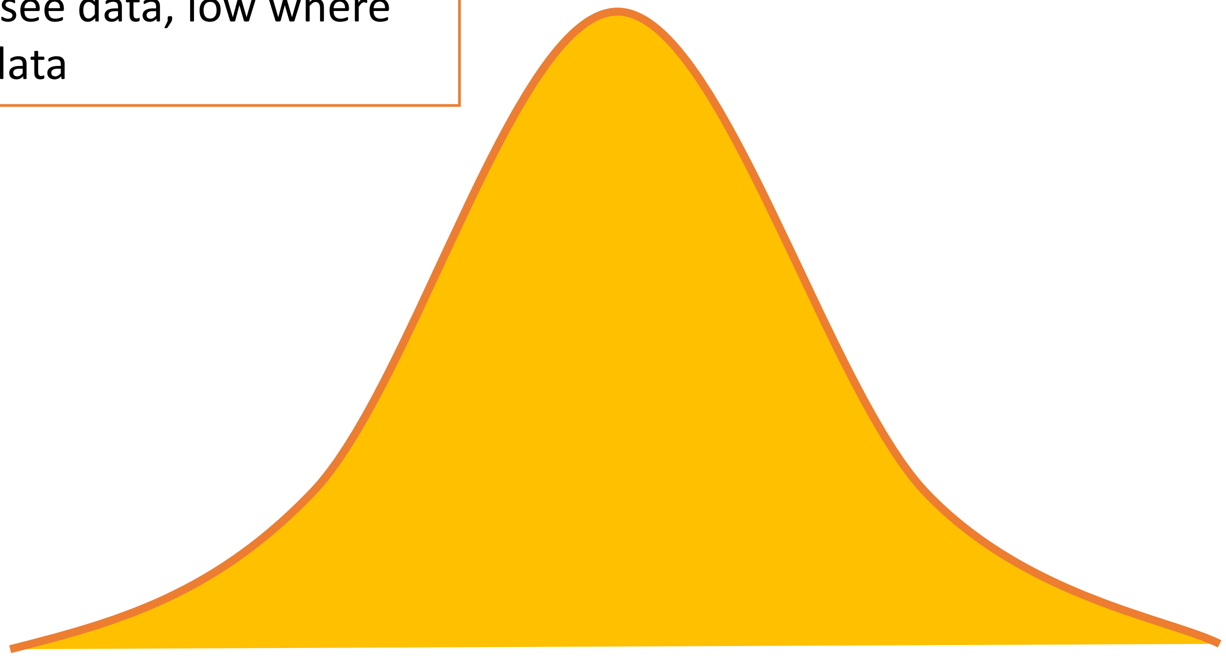


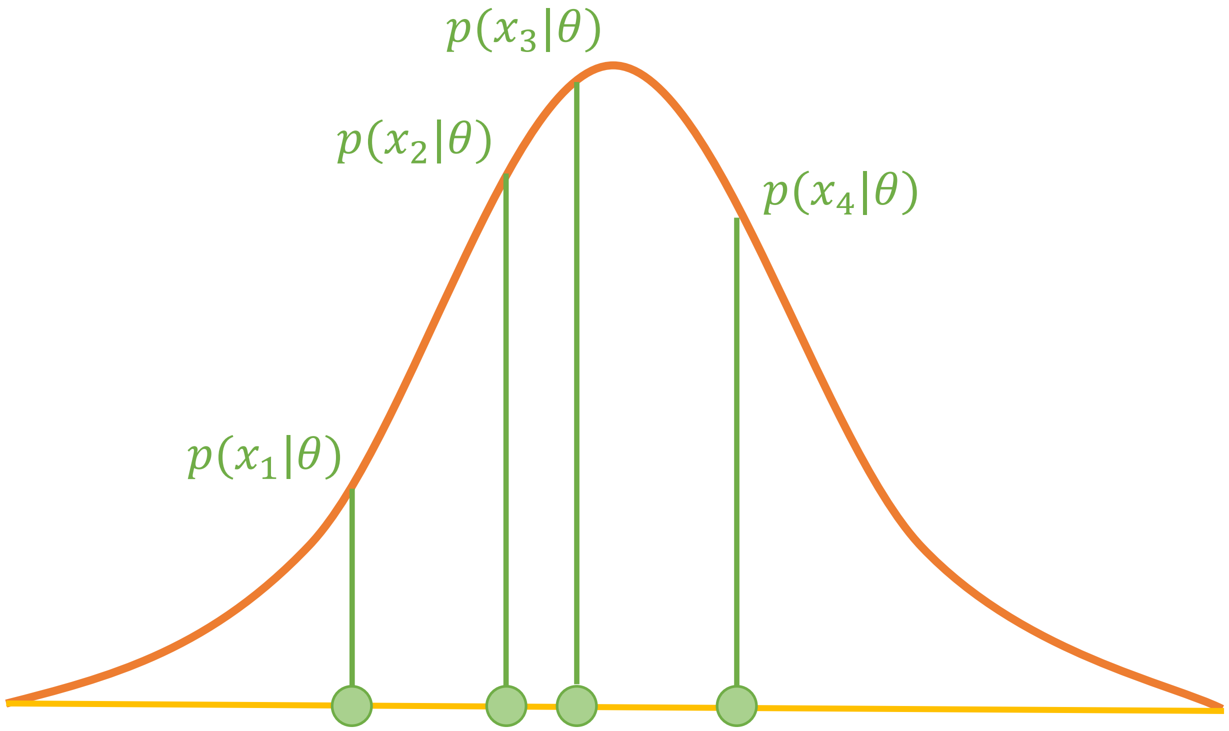
Under changing model parameters:

- Volume is always conserved
- Integral under the density function is always = 1

Intuition:

- High where we expect to see data, low where we do not expect to see data





X's iid

Goal: maximize likelihood function

$$L(\theta) = \prod_i p(x_i|\theta)$$

$$\log L(\theta) = \log \prod_i p(x_i|\theta)$$

$$\log L(\theta) = \sum_i \log p(x_i|\theta)$$

Cross-entropy: a maximum likelihood estimator

Logistic Regression (MLE framework):

Given: $\mathbb{D} = \{ (x^{(1)}, y^{(1)}), \dots, (x^{(n)}, y^{(n)}) \}$

Model: $y^{(i)} \sim \text{Bernoulli}(\sigma(\bar{w}^T \bar{x}^{(i)}))$

- 其概率質量函數為:
- $$f_X(x) = p^x (1-p)^{1-x} = \begin{cases} p & \text{if } x = 1, \\ q & \text{if } x = 0. \end{cases}$$

MLE: $\bar{w}^*_{MLE} = \arg\max \; p(D|\bar{w})$

$$p(D|\bar{w}) = \prod_1^n p(y^{(i)} | \bar{x}^{(i)}, \bar{w})$$

[samples assumed to be i.i.d.]

$$= \prod_1^n \underbrace{\left[\frac{1}{1+e^{-z}}\right]}_{\pi_i}^{y^{(i)}} \underbrace{\left[1 - \frac{1}{1+e^{-z}}\right]}_{1-\pi_i}^{(1-y^{(i)})}, \quad \text{with } z = \bar{w}^T \bar{x}^{(i)}$$

NLL: $C(\bar{w}) = -\log p(D|\bar{w})$

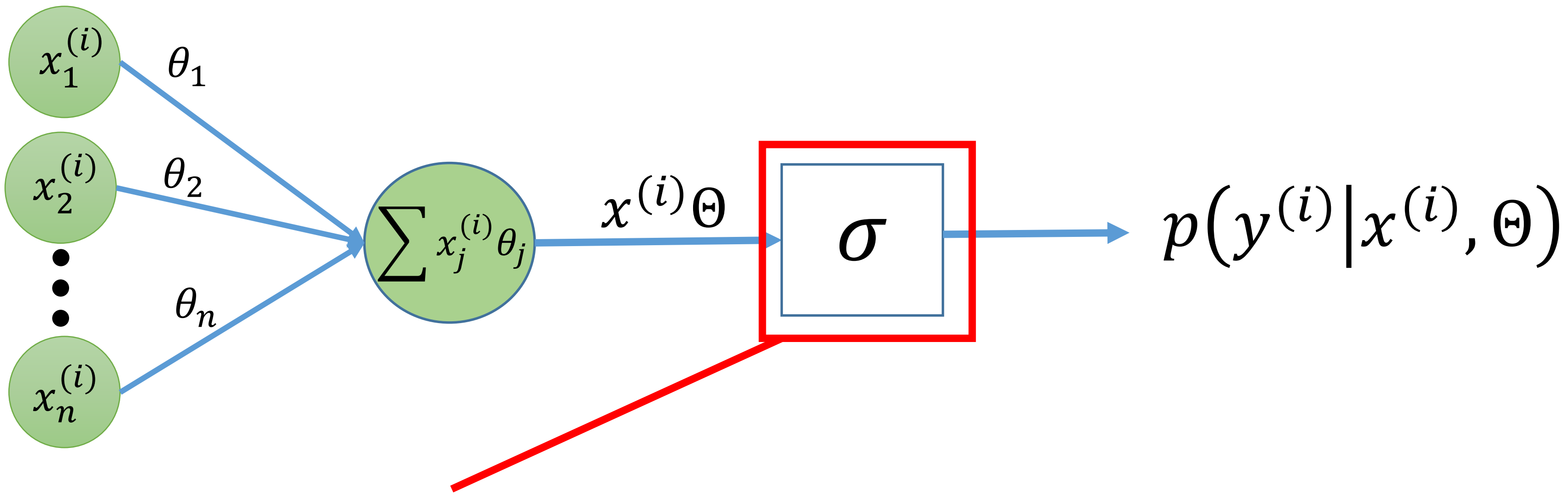
negative log likelihood

$$= -\frac{1}{N} \sum y^{(i)} \log(\pi_i) + (1-y^{(i)}) \log(1-\pi^{(i)})$$

← Log-loss function (0/1 loss)

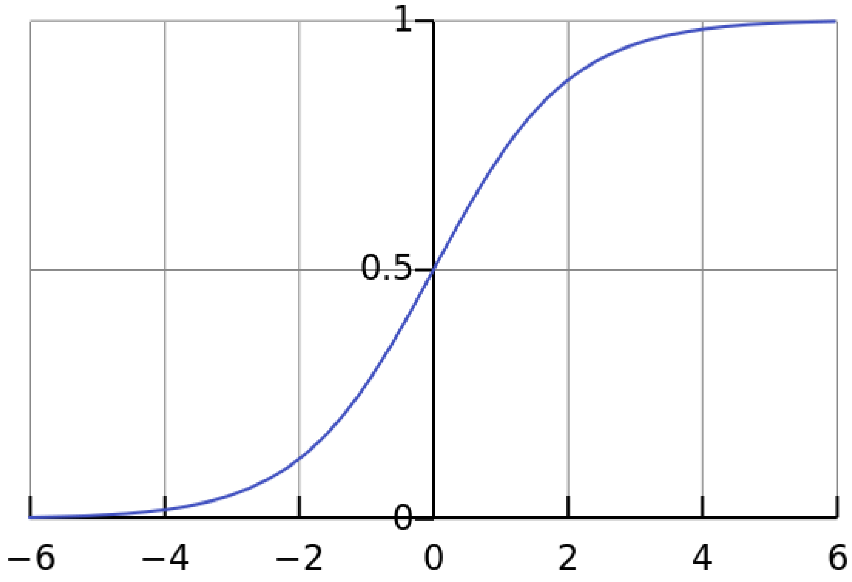
"cross-entropy"

Binary classification and sigmoid function

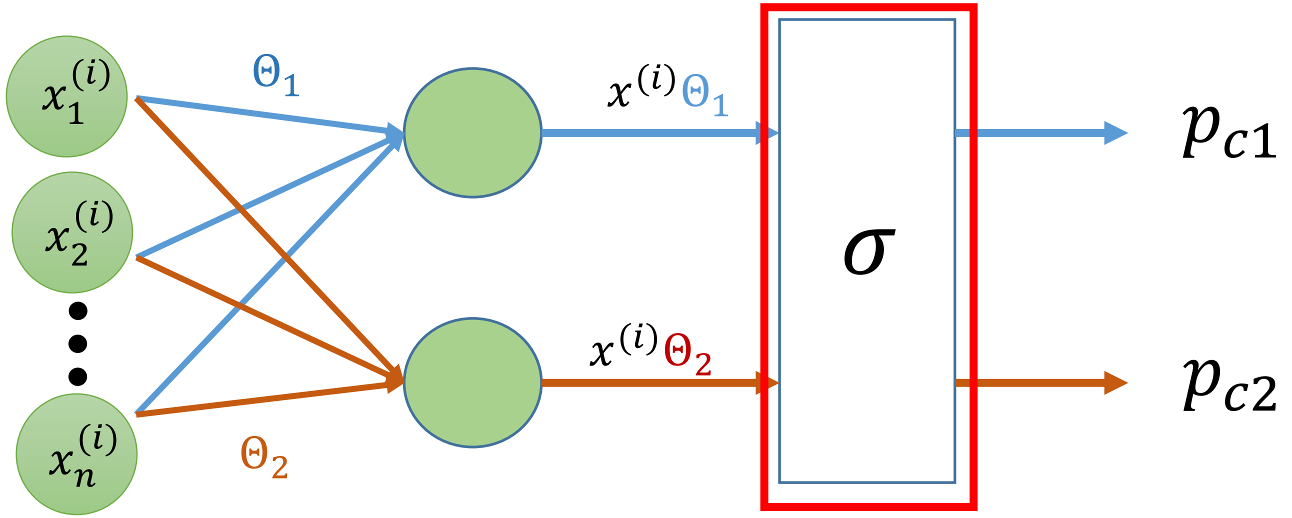


Requirements for the function:
(1)Output must be positive
(2)Output must between [0, 1]

$$\sigma(x) = sigm(x) = \frac{1}{1+e^{-x}} = \frac{\mathrm{e}^x}{\mathrm{e}^x+1}$$



Extending to Multiclass: Softmax



Softmax Function vs Sigmoid Function

As mentioned above, the softmax function and the sigmoid function are similar. The softmax operates on a vector while the sigmoid takes a scalar.

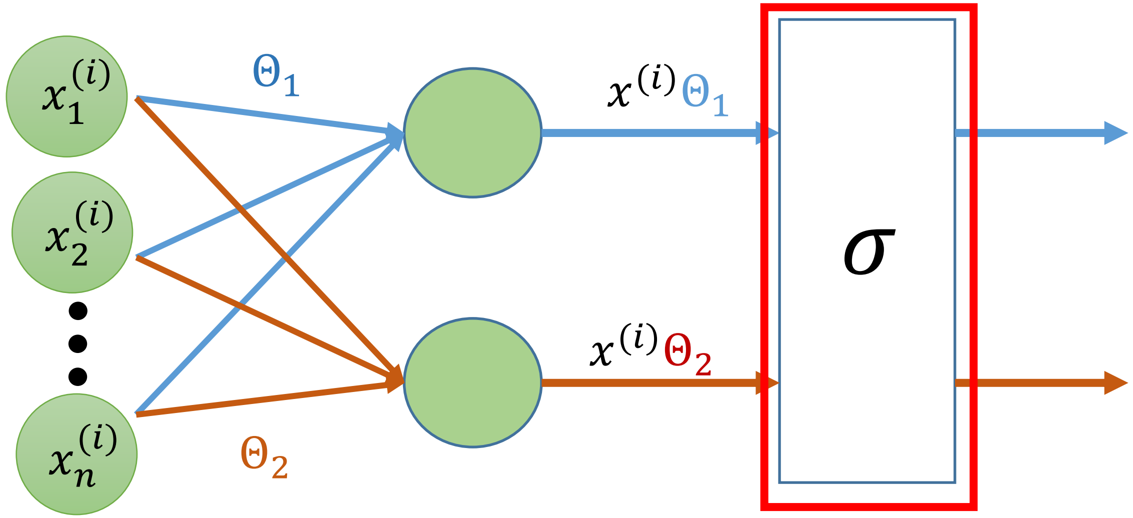
$$\text{Softmax } \sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

$$\text{Sigmoid } S(x) = \frac{1}{1+e^{-x}}$$

Requirements for this function:
(1) Output must be positive
(2) Output must between [0, 1]
(3) Summing up all outputs is 1

$$softmax(\eta_i) = \frac{\exp(\eta_i)}{\sum_j \exp(\eta_j)} = \frac{\exp(\eta_i)}{\exp(\eta_1) + \exp(\eta_2) + \cdots + \exp(\eta_c)}$$

Extending to Multiclass: Softmax



$$p_{c1} = \frac{\exp(x^{(i)}\Theta_1)}{\exp(x^{(i)}\Theta_1)+\exp(x^{(i)}\Theta_2)}$$

$$p_{c2} = \frac{\exp(x^{(i)}\Theta_2)}{\exp(x^{(i)}\Theta_1)+\exp(x^{(i)}\Theta_2)}$$

Requirements for this function:
(1) Output must be positive
(2) Output must between [0, 1]
(3) Summing up all outputs is 1

$$\sum_m softmax(\eta_i) = \frac{\exp(\eta_i)}{\sum_j \exp(\eta_j)} = \frac{\exp(\eta_1)+\exp(\eta_2)+\cdots+\exp(\eta_c)}{\exp(\eta_1)+\exp(\eta_2)+\cdots+\exp(\eta_c)} = 1$$

Maximum-likelihood Estimation (MLE)

!

- Step 1: Write down probability distribution
- Step 2: Decompose into per sample probability
- Step 3: Minimize negative log likelihood

Left as exercise

Optimization procedure – SGD in a nutshell

Iterative gradient descent to find parameters Θ:

Initialize weights with small random values

Initialize biases with 0 or small positive values

Compute gradients

Update parameters with SGD

We'll study this next week

SGD in a nutshell:

Compute the negative gradient at θ^0

$\Rightarrow -\nabla C(\theta^0)$

Times the learning rate η

$\Rightarrow -\eta \nabla C(\theta^0)$

Course Logistics

Course Logistics

Machine Perception (263-3710-00L)
2L + 1U + 1A
5 ECTS Credits

Audience:

- Computer Science Master | RSC

Lecture

- Thursday 10:15 – 11:55 CAB G 11 »

Exercise:

- Thursday 13:15 – 15:00 CAB G 11 » or
- Friday 13:15 – 15:00 CAB G 11 »
- 4 pen & paper assignments (not graded)
- 4 interactive tutorials
- 2 lecture-style classes with practical tips
- 1 multi-week project **(40% of grade)**

Slides will be available the day of the lecture:

https://ait.ethz.ch/teaching/courses/2020-SS-Machine-Perception/

user: mp-2020
pw: Perception20

Write this down now!

Course Logistics

Machine Perception (263-3710-00L)
2L + 1U + 1A
5 ECTS Credits

Audience:

- Computer Science Master | RSC

The lecture will be recorded

https://www.video.ethz.ch/lectures/d-infk/2020/spring/263-3710-00L

user: hil-20s
pw: 3Ewk6Fv

Lecture

- Thursday 10:15 – 11:55 CAB G 11 »

Exercise:

- Thursday 13:15 – 15:00 CAB G 11 » or
- Friday 13:15 – 15:00 CAB G 11 »
- 4 pen & paper assignments (not graded)
- 4 interactive tutorials
- 2 lecture-style classes with practical tips
- 1 multi-week project **(40% of grade)**

Course Logistics – Cont'd

Machine Perception (263-3710-00L)

2L + 1U + 1A

5 ECTS Credits

Communication policy:

All questions (regarding content, organization etc.) on Piazza

Sign up for the Piazza forum here

We will **not** be able to answer individual emails

Course Logistics – Cont'd

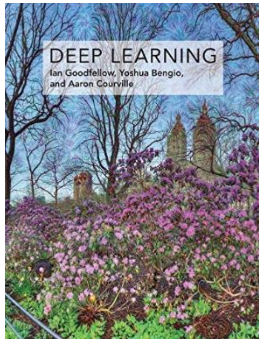
Machine Perception (263-3710-00L)

2L + 1U + 1A

5 ECTS Credits

Additional recommended reading:

- **Deep Learning**, MIT Press, *Ian Goodfellow and Yoshua Bengio and Aaron Courville*
- Available for free online: http://www.deeplearningbook.org/



Additional (catch-up) reading:

- **Pattern Recognition and Machine Learning**, *Christopher Bishop*
- **Mathematics for machine learning**, *Mark Deisenroth et al*.
- Available for free online: https://mml-book.github.io/

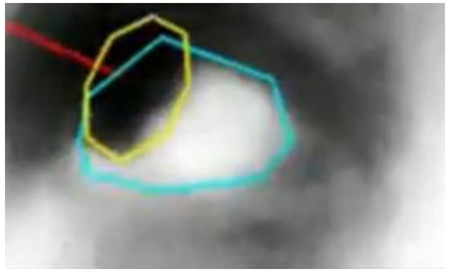
Course Logistics – Projects

Dynamic Gesture Recognition

Classification of hand gestures from multi-modal data including RGB, depth, segmentation mask and skeletal information for videos.

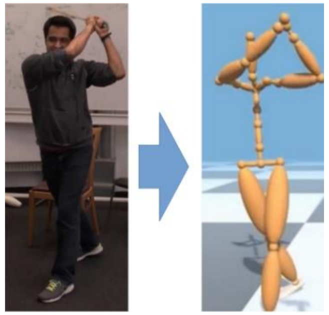
Eye Gaze Estimation

In which 3D direction a person's eye is looking towards, as seen from the perspective of a webcam.



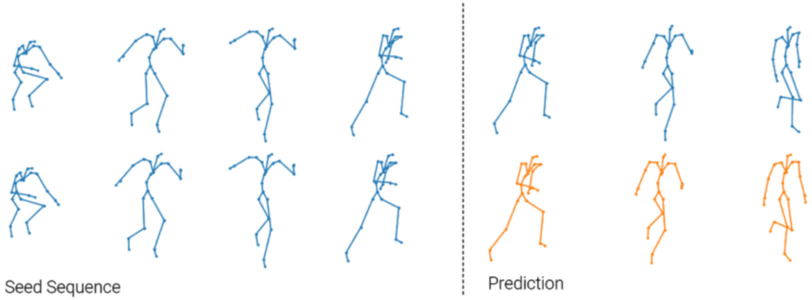
Body Pose Estimation

Estimation of 3D location of body joints from a 2D human image.



Human Motion Prediction

Given a sequence of human motion data, predict how the motion continues for several frames in the future.



Course Logistics - Compute

You will need some GPUs!

We use the Leonhard cluster:

- 1 GPU at a time per group
- Will be made available around week 3
- In the beginning, use TensorFlow CPU-version for simple tests
- Leonhard has limited resources, faire usage will be monitored

Deep Learning Basics

Week 3&4

- MLP
- Fullly connected networks
- Data, tasks, loss functions
- Backprop
- Activation functions, etc.

CNNs, RNNs & Co

Week 5-7

- CNNs
- RNNs
- LSTM / GRU / Backprop through time
- Fully convolutional networks
- instance segmentation and other advanced vision tasks

Generative Modelling

Week 8-12

- Latent variable models
- Implicit models
- Autoregressive models
- Recent research in DL-based machine perception

Deep RL

Week 12-14

- RL & policy learning

https://ait.ethz.ch/teaching/courses/2020-SS-Machine-Perception/#schedule

Wk.	Date	Content	Material	Exercise Session
1	20.02.	**-- No Class --**		
2	27.02.	**-- No Class --**		
3	05.03.	**Deep Learning Introduction** Class content & admin, Feedforward Networks, Representation Learning		**Tutorial** Implement your own MLP
4	12.03.	**Training Neural Networks** Backpropagation		**Tutorial** Linear Regression **Pen & Paper** Backpropagation
5	19.03.	**Convolutional Neural Networks**		**Tutorial** CNNs in TensorFlow **Pen & Paper** CNN
6	26.03.	**Recurrent Neural Networks** LSTM, GRU, Backpropagation through time		**Tutorial** RNNs in TensorFlow **Pen & Paper** RNN
7	02.04.	**Fully Convolutional Neural Networks** Advanced Vision Topics		**Class** Tips for Training Part 1
8	09.04.	**Generative Models I: Latent variable models** Variational Autoencoders, etc.		**Class** Tips for Training Part 2 **Pen & Paper** VAE
9	16.04.	**-- No Class (Easter) --**		
10	23.04.	**Generative Models II: Implicit Models** Generative Adversarial Networks & Co		**Pen & Paper** GAN
11	30.04.	**Generative Models III: Autoregressive Models** PixelCNN, PixelRNN, WaveNet, Stochastic RNNs		**Pen & Paper** Autoregressive VAE
12	7.05.	**Applications** Hands & Full body, Eyes		
13	14.05.	**Reinforcement Learning I**		**Pen & Paper** RL
14	21.05.	**-- No Class (Ascension Day) --**		
15	28.05.	**Reinforcement Learning II**		

60% Final 40% project

https://ait.ethz.ch/teaching/courses/2020-SS-Machine-Perception/

Next week

What functions can neural networks represent?

- Universal approximation theorem

And how do we train them?

- Backprop algorithm