Self-supervised learning of ‘deep’ representations

Wulfram Gerstner

EPFL, Lausanne, Switzerland

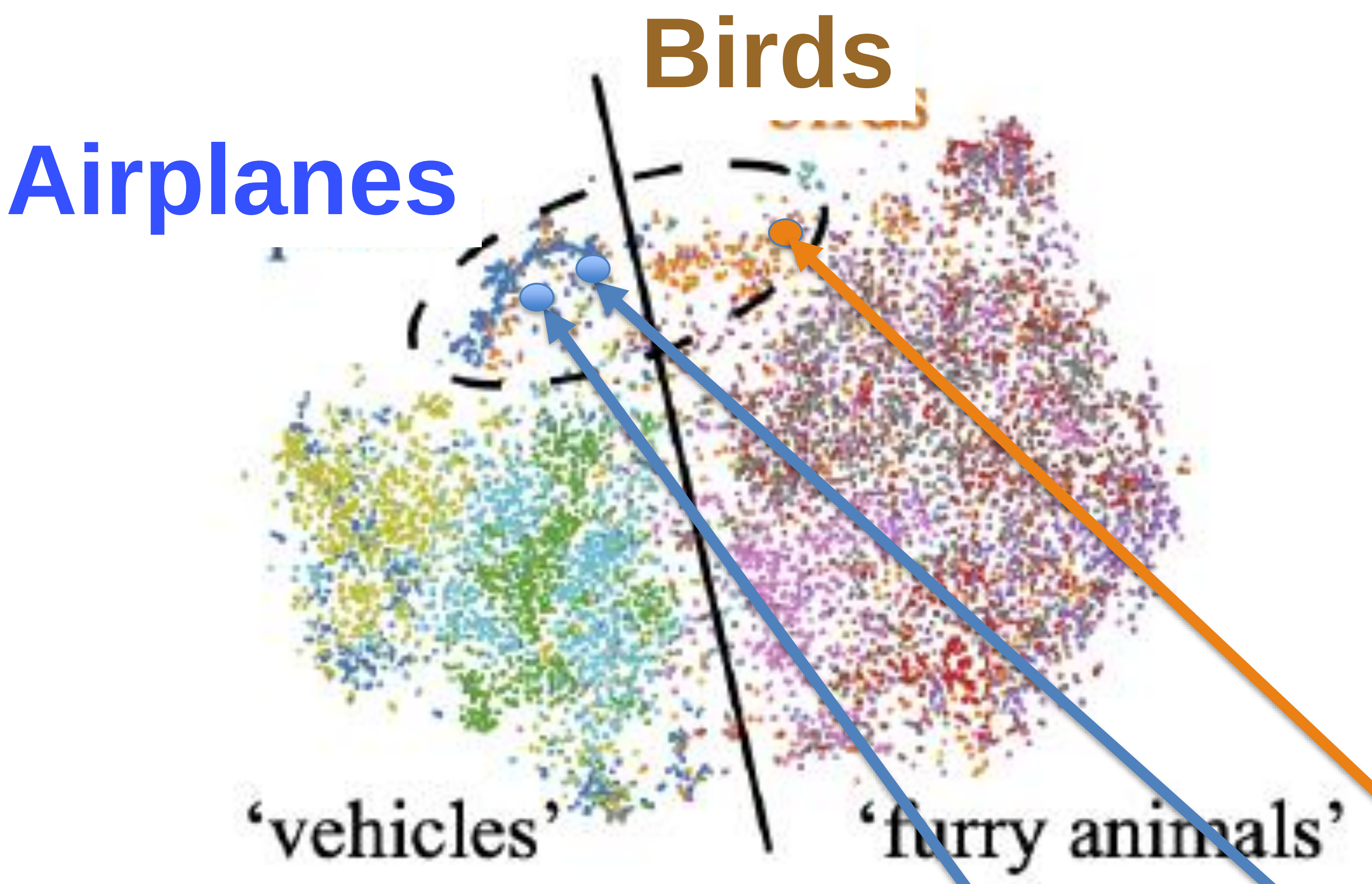
1) Introduction (review)
2) **Self-supervised Learning: use ‘sameness’**
3) Representation Learning with CPC
4) Representation Learning with SimCLR
5) Noncontrastive self-supervised learning with VICReg

Previous slide.

The following slides are partially a review and partially an extension of the topics we have previously seen in representation learning

Review: Good representation

→ needs deep network

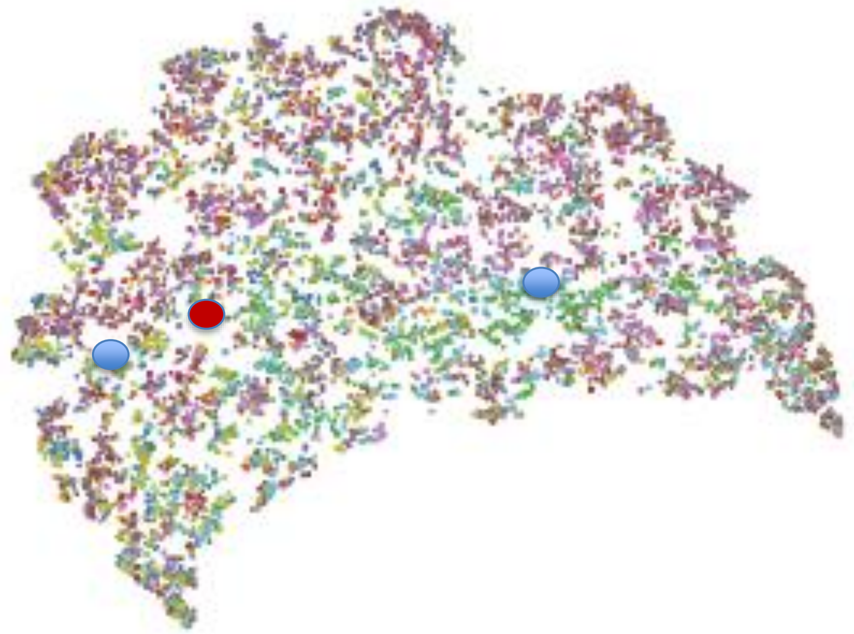


→ Objects of same class are neighbors

→ Raise arm if airplane →

→ could be learned with 3-factor rule

bad representation



Previous slide.
What would be a good state representation?

Suppose we have many unlabeled images, some of these with airplanes others with birds. At the end of the visual processing stream (say in IT), we want a representation such that two images of airplanes are represented similarly, but different from two images of birds.

This idea requires a deep network since the pixel images of an airplane from below and a bird from below may be more similar, that the image of a black airplane from below and a white airplane from above.

All single-layer methods such as PCA, ICA, or clustering would therefore not work!

The similarity in pixel space is not always a good predictor for similarity in the space of ‘meaning’ that is developed in deep areas such as IT.

However, if we have a good representation in a deep area, then it will be easy to learn a reward-based task such as raise your arm if you see an airplane.

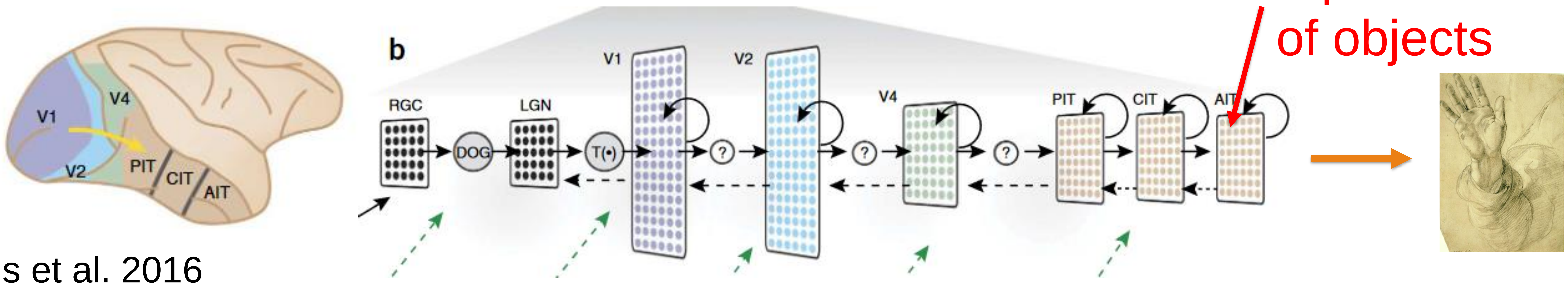
Learn a ‘good representation’!

Loss functions??
Learning rules???
Network architecture???
What kind of feedback???

Big question:

1) In ML: Loss function?

2) In Biology/Hardware:
local learning rules
(with several global signals)



representation of objects

image: Yamins et al. 2016

Previous slide.

In the following we discuss the loss functions of different contrastive and noncontrastive learning rules.

Self-supervised learning: same object other object

Representation of

same object in different views:

“your bicycle” from different views

(and not the “class of bicycles)

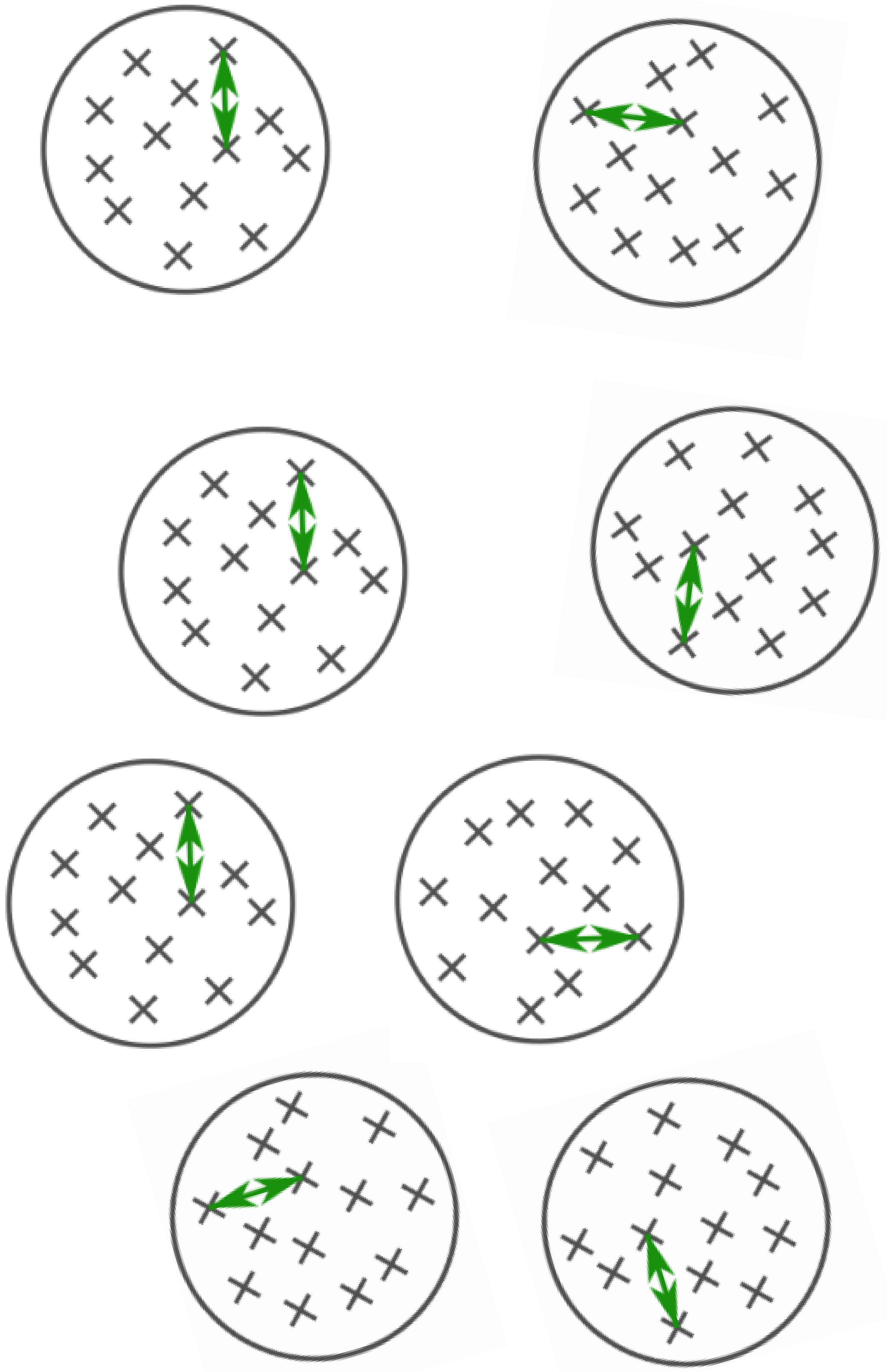
same image in different styles

“augmented image”

same song with different instruments

same content of text

but rewrite with different synonyms



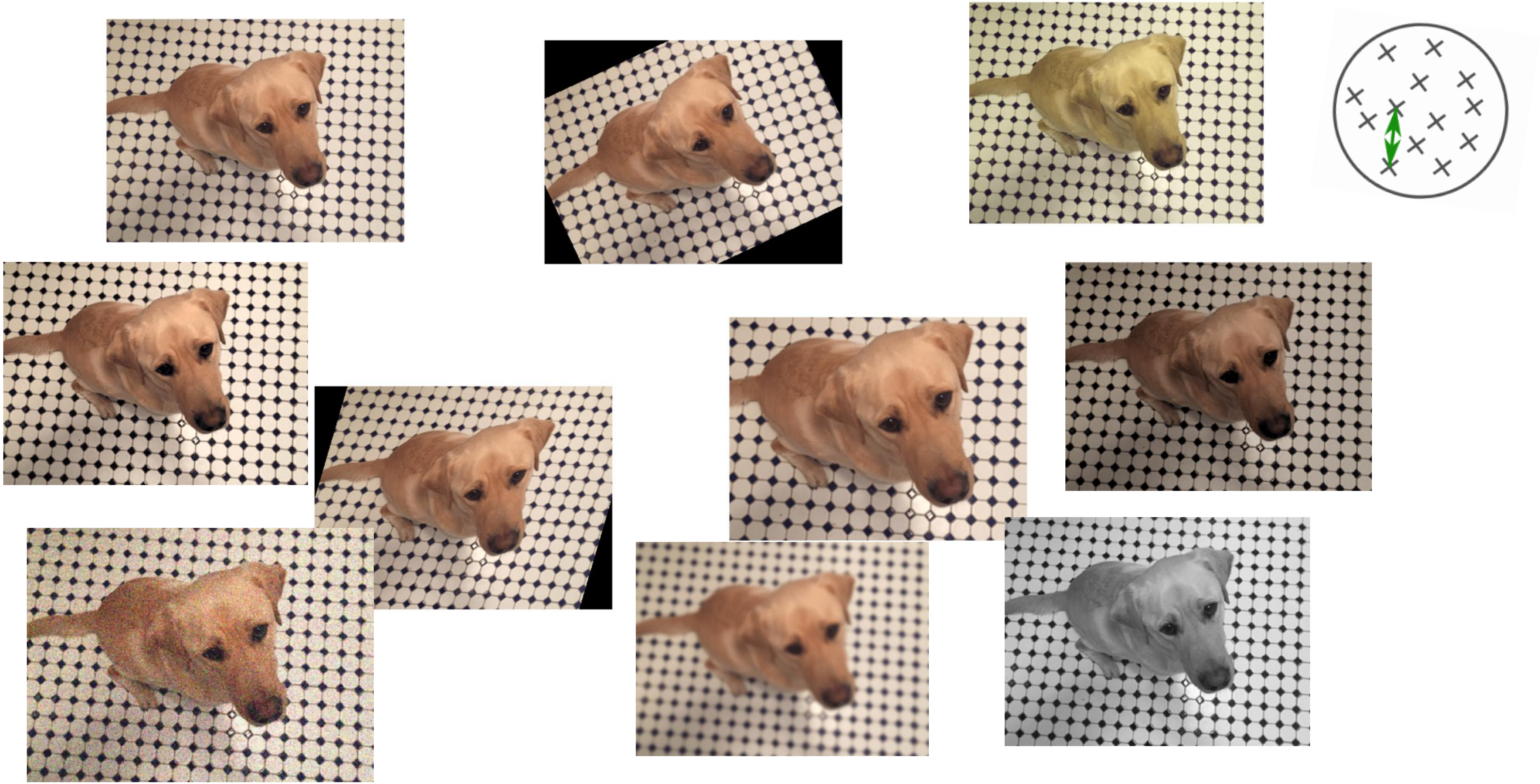
Previous slide.

Self-supervised learning aims to make different representations of the same object close to each other.

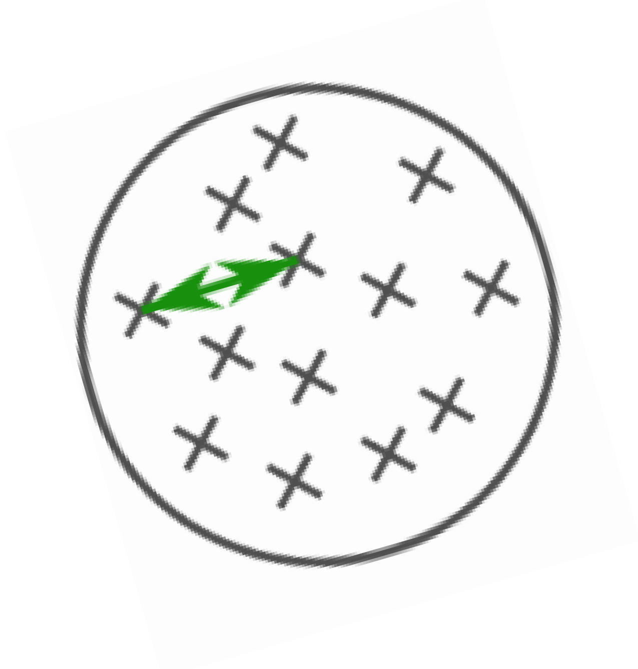
The term object is used here in a loose sense:
The object can be a song, the meaning of a sentence, or your specific bicycle.

Self-supervision and image augementation

https://ch.mathworks.com/help/deeplearning/ug/image-augmentation-using-image-processing-toolbox.html



Self-supervision and style augmentation



Self-supervision and text: same content

"*they have a huge family house*«

→ "Their family house is huge"

→ (huge - big - great - impressive)
→ (house - mansion - villa) o
→ (have - own - posess)
→ (family - parents)

→ "Their parents own a mansion"

Previous slide.

Self-supervised learning aims to make different representations of the same object close to each other.

The term object is used here in a loose sense:
The object can be a song, the meaning of a sentence, or your specific bicycle.

Self-supervised learning of ‘deep’ representations

Wulfram Gerstner

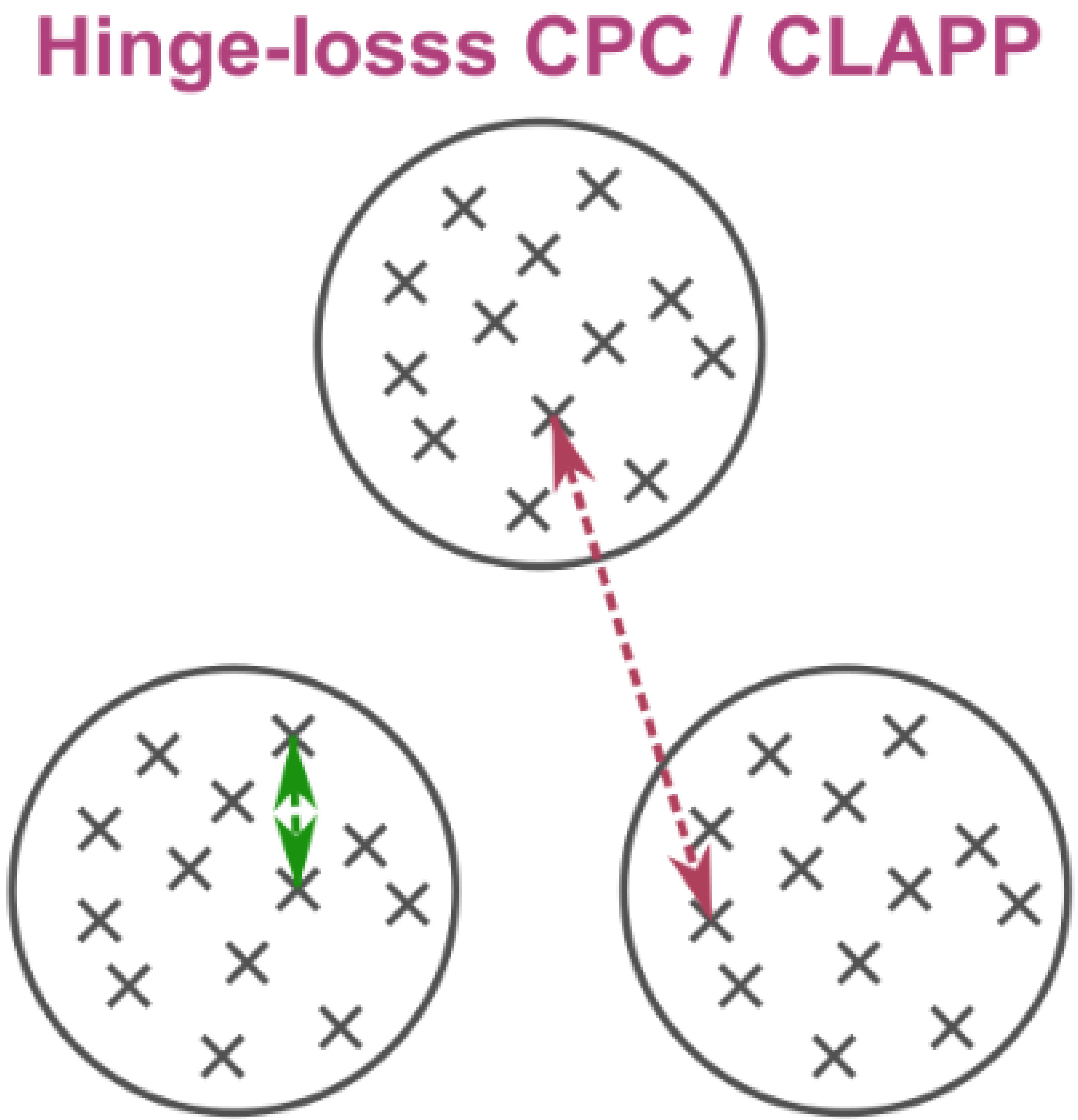
EPFL, Lausanne, Switzerland

1) Introduction (review)
2) Self-supervised Learning: use ‘sameness’
3) **Representation Learning with CPC**
4) Representation Learning with SimCLR
5) Noncontrastive self-supervised learning with VICReg

Previous slide.

Contrastive self-supervised learning aims to make different representations of the same object similar to each other AND to make representations of other objects different.

Contrastive self-supervised learning: CPC



CPC:
Contrastive Predictive Coding

y^t = sameness/contrastive signal

$$y^t = \begin{cases} 1 \; if \; same \; sample \\ -1 \; if \; other \; sample \end{cases}$$

CPC:
Van den Oord, 2019

Previous slide.

Contrastive self-supervised learning aims to make different representations of the same object similar to each other AND to make representations of other objects different.

The first example is Contrastive Predictive Learning (CPC) that we discuss here in a form where it has a Hinge Loss – this is the version that is closely related to CLAPP.

CLAPP Loss = CPC Hinge Loss

Illing, et al. NeurIPS 2021

$$L_{CPC}^{t,L} = \max(0,1 - y^t \cdot u_t^{t+\delta t,l})$$

$$L_{CLAPP}^{t,l} = \max(0,1 - y^t \cdot u_t^{t+\delta t,l})$$

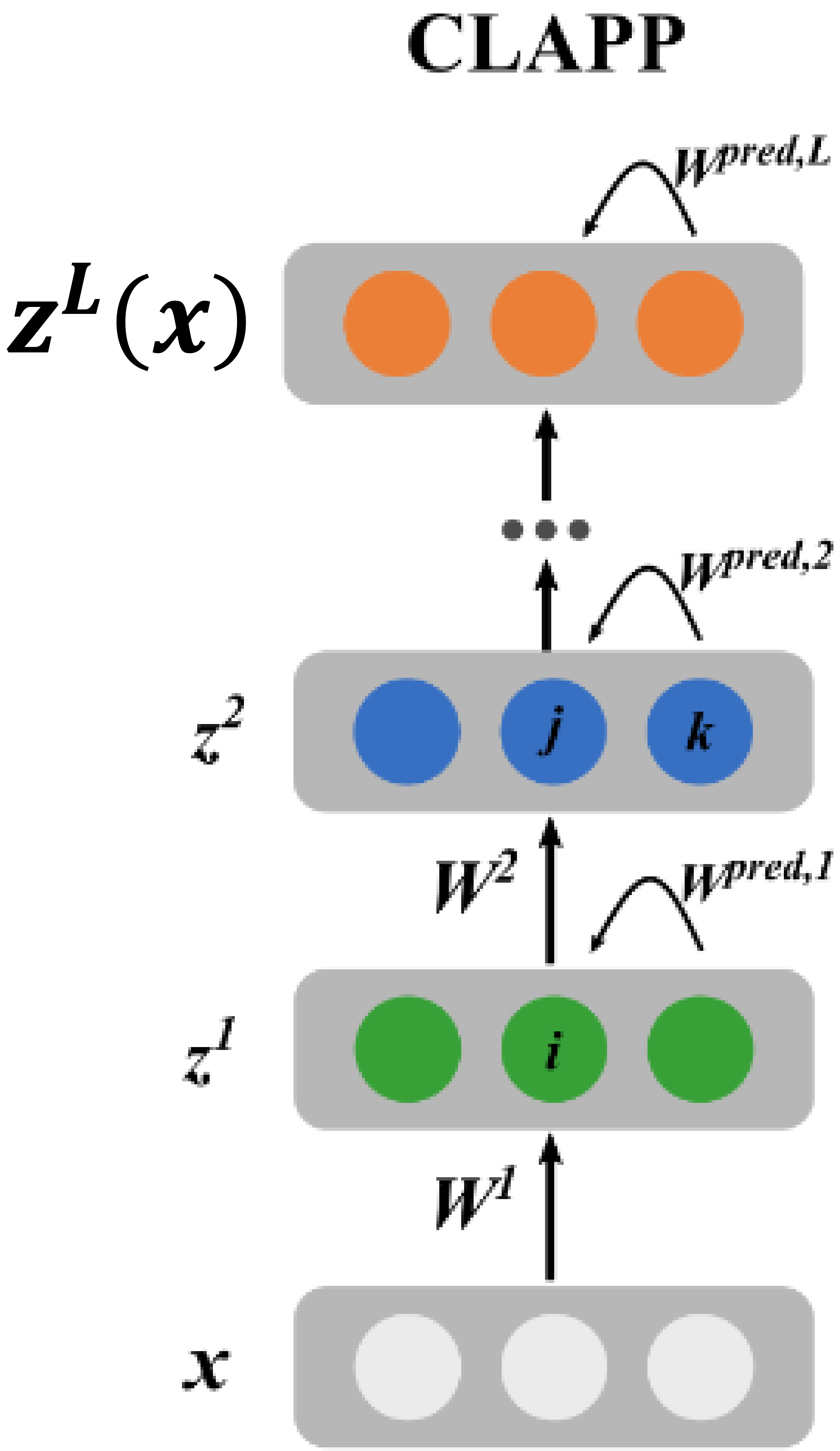
$u_t^{t+\delta t,l}$ = similarity:

$$u_t^{t+\delta t,l} = \mathbf{z}^{t+\delta t,l} \underbrace{\mathbf{W}^{pred,l} \mathbf{c}^{t,l}}$$

feedforward vs lateral prediction

y^t = sameness signal/contrastive signal

$$y^t = \begin{matrix} 1 \ \textit{if same sample} \\ -1 \ \textit{if next sample} \end{matrix}$$



Previous slide. CPC is an example of a contrastive self-supervised algorithm.
The small index l is the layer index. The last layer is l=L.
In standard contrastive learning the loss is applied at the last layer and backprop is used to update weights in all layers.
Let us look at the loss in the last layer. It is a hinge-loss: either zero or linear in u.
The variable u is a measure of the similarity between the activity state vector **z** in layer l and the lateral prediction from OTHER neurons **c = z** in the same layer.

If variable y tells whether the prediction comes from the SAME object (y=1) or a different object (y=-1).

The boldface **z** refers to all neuron in a layer. For an interpretation it is easier to look at individual neurons such as neuron i in layer l. Its activity depends ONLY on the feedforward pathway

$$z_i^{t+\delta t,l} = g(\sum_j w_{ij}^l \, z_j^{t,l-1})$$

Self-supervised learning of ‘deep’ representations

Wulfram Gerstner

EPFL, Lausanne, Switzerland

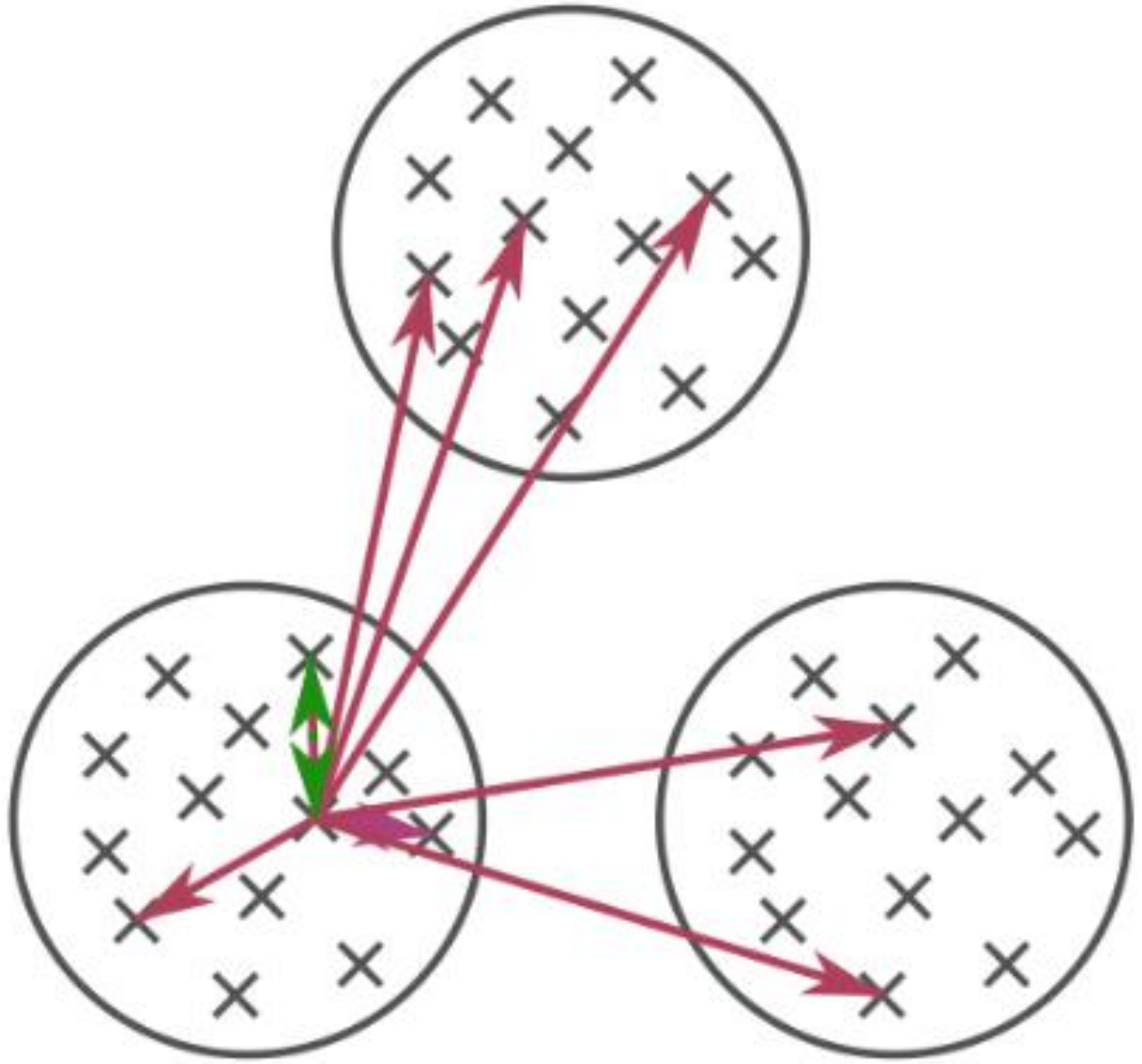
1) Introduction (review)
2) Self-supervised Learning: use ‘sameness’
3) Representation Learning with CPC
4) **Representation Learning with SimCLR**
5) Noncontrastive self-supervised learning with VICReg

Previous slide.

Contrastive Self-supervised learning with SimCLR is similar in spirit to CPC, but the pool of negative examples is constructed differently.

Contrastive self-supervised learning: SimCLR

SimCLR



-two representations $\boldsymbol{z_i}$ and $\boldsymbol{z'_i}$

-maximize objective

$$\boldsymbol{L_{i,}^{SimCLR}} = \frac{exp(\boldsymbol{z_i z'_i})}{\sum_{\boldsymbol{k}\neq i} exp(\boldsymbol{z_i z_k})}$$

- similarity measure

$$\boldsymbol{sim}(\boldsymbol{z_i}, \boldsymbol{z_j}) = \boldsymbol{z_i} \cdot \boldsymbol{z_j}$$

T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. Proc. of the 37th Int. Conf. Mach. Learn. PMLR, 119, 2020.

Previous slide.

Contrastive Self-supervised learning with SimCLR compares two representations representations $\boldsymbol{z_i}$ and $\boldsymbol{z_i'}$ of the SAME object i (numerator) with a random selection of other examples compared again to i.
All comparisons are anchored on one example $\boldsymbol{z_i}$.

The evaluation is done with a softmax

Self-supervised learning of ‘deep’ representations

Wulfram Gerstner

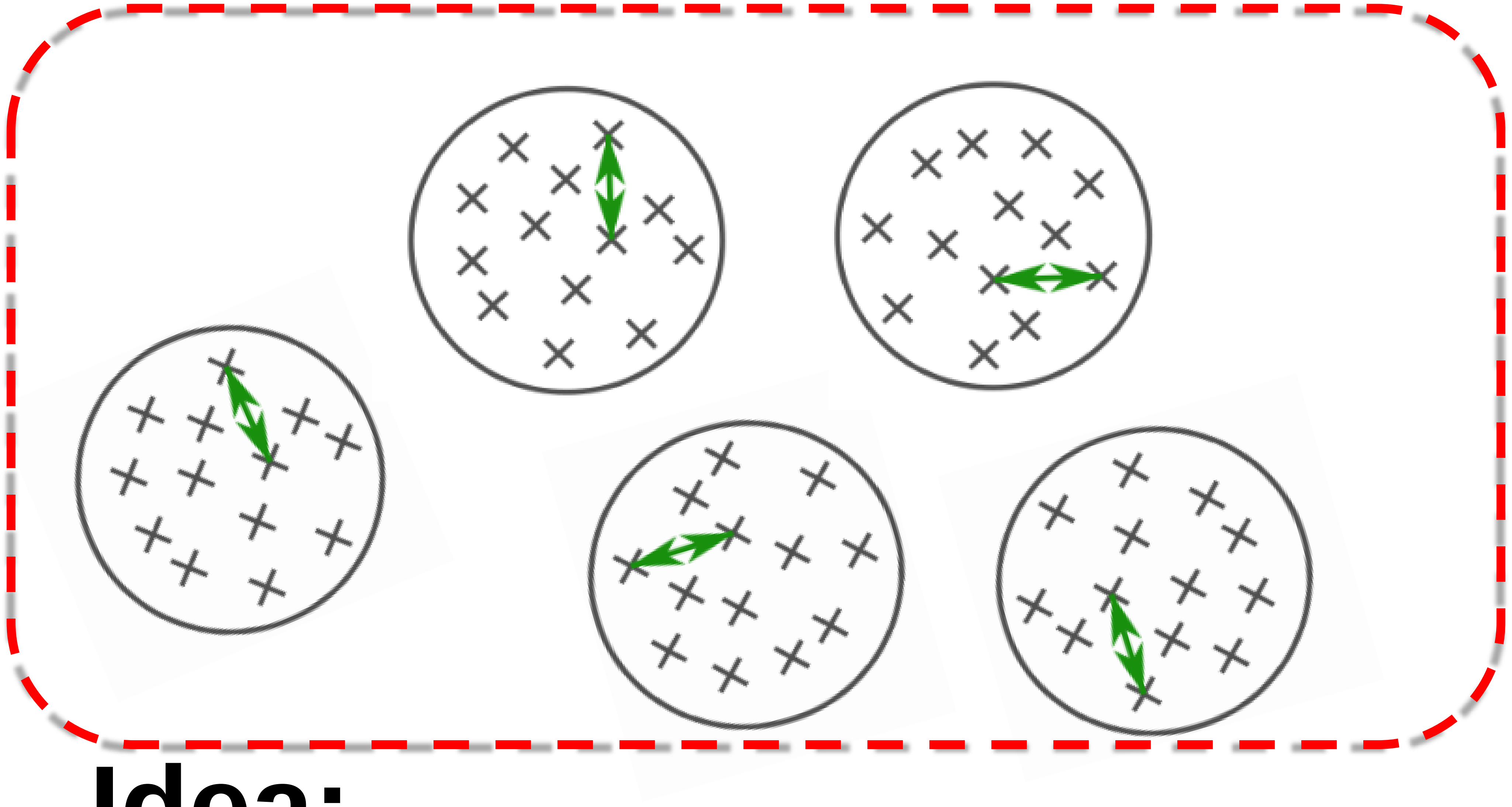
EPFL, Lausanne, Switzerland

1) Introduction (review)
2) Self-supervised Learning: use ‘sameness’
3) Representation Learning with CPC
4) Representation Learning with SimCLR
5) **Noncontrastive self-supervised learning with VICReg**

Previous slide.

Noncontrastive learning does not explicitly use ‘negative’ examples.

Noncontrastive self-supervised learning VICReg



Idea:

- make alternative representations similar
- make all neurons respond
- assign neurons different roles

Variance-**I**nvariance-**C**ovariance **Reg**ularization: VICReg

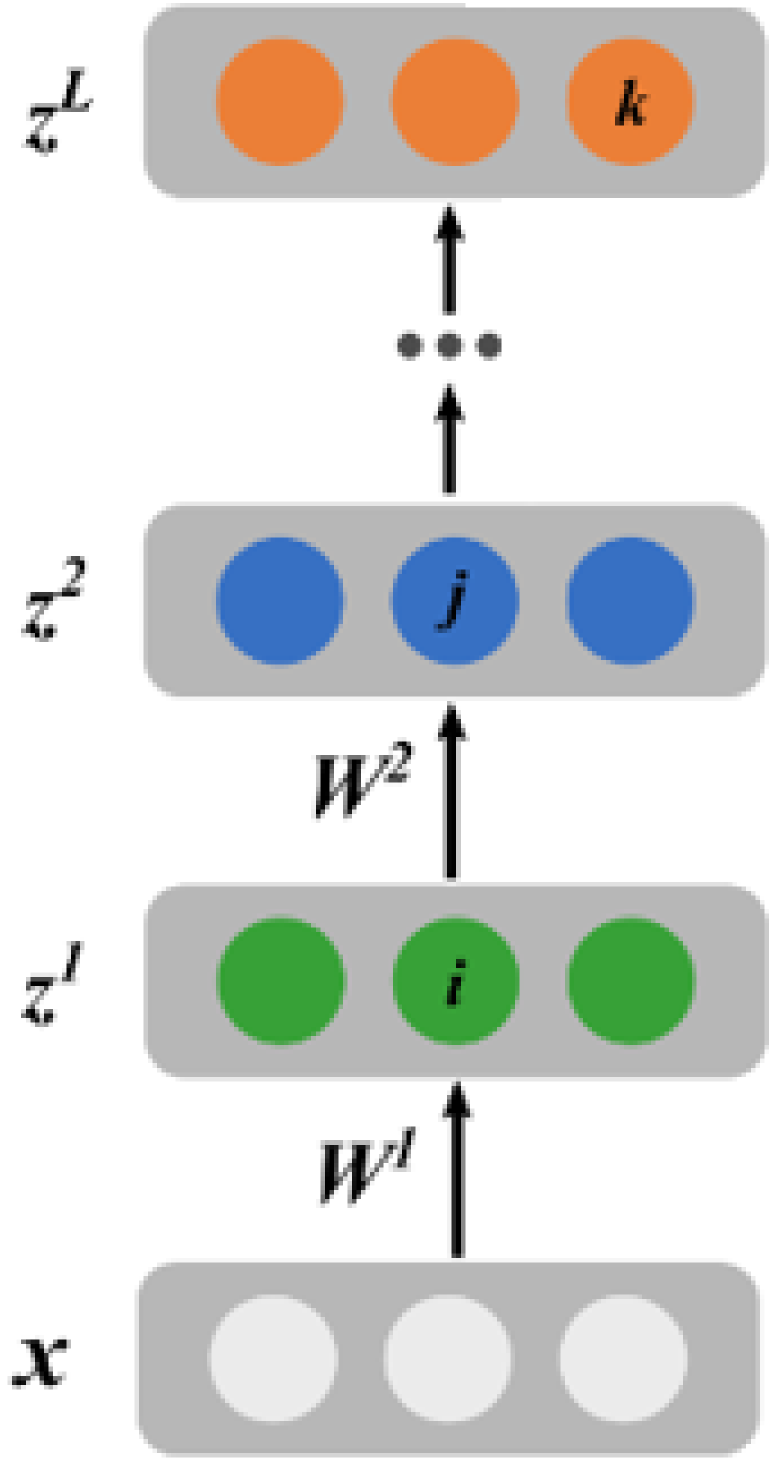
A. Bardes, J. Ponce, and Y. LeCun. Vicreg: variance-invariance-covariance regularization for self-supervised learning. ICLR, arXiv:2105.04906v3, 2022

Previous slide.

Noncontrastive learning does not explicitly use ‘negative’ examples. Instead, noncontrastive learning exploits the statistics of the representations $\boldsymbol{z}_{\boldsymbol{n}}^{\boldsymbol{L}}$ in the output layer across ALL samples n in the data set.

Noncontrastive self-supervised learning VICReg

Variance-**I**nvariance-**C**ovariance **Reg**ularization: VICReg



Representations in layer L:

$\boldsymbol{z^L(x_1), z^L(x_2)\ldots\ z^L(x_N)}$ of N objects.

$\boldsymbol{z_1^L = z^L(x_1), z_2^L = z^L(x_2), \ldots}$

Alternative representations in layer L

$\boldsymbol{z_1^{L\prime} = z^L(x_1{}'), z_2^{L\prime} =}$ ... of ‘augmented objects’.

Idea:

- Make alternative representations similar
- make all neurons respond
- assign neurons different roles

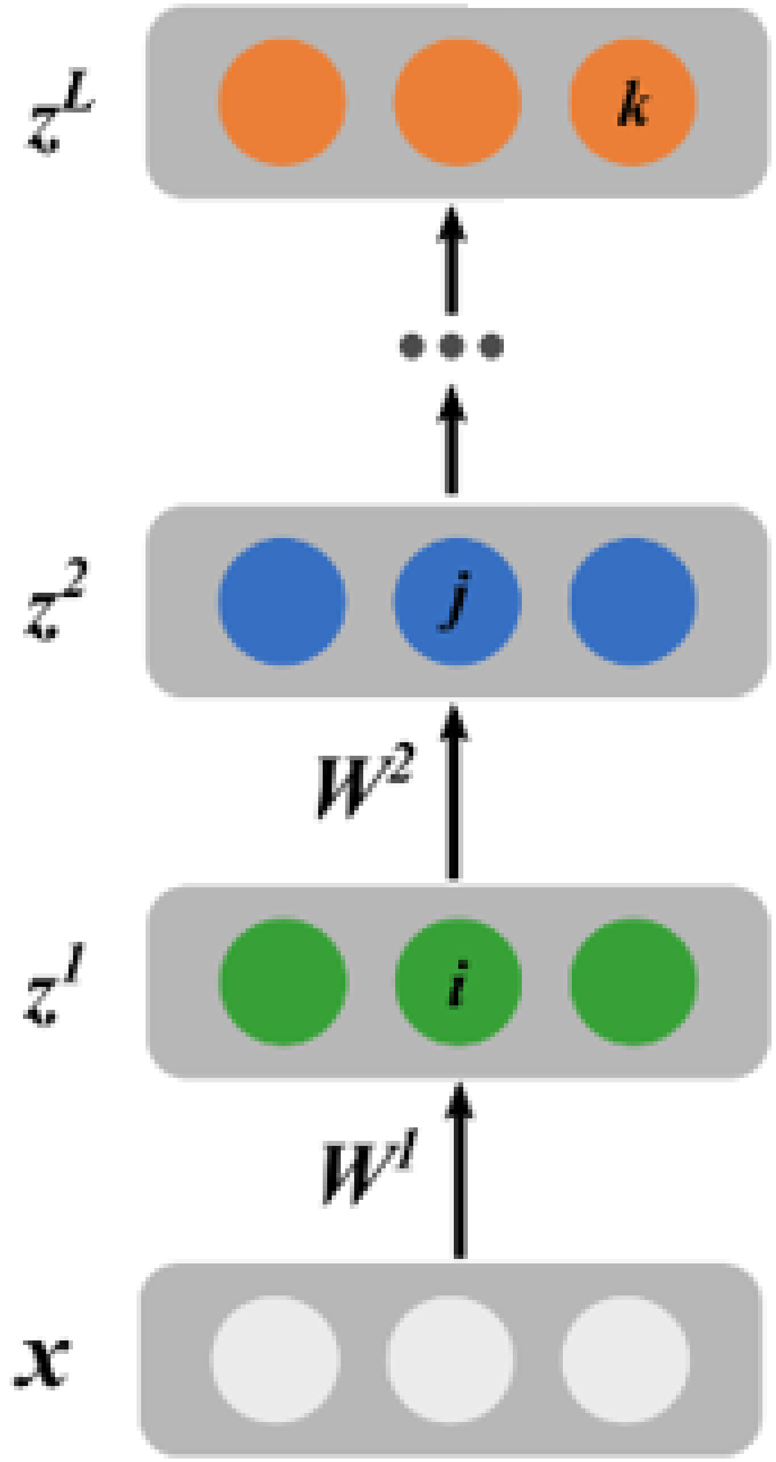
A. Bardes, J. Ponce, and Y. LeCun. Vicreg: variance-invariance-covariance regularization for self-supervised learning. ICLR, arXiv:2105.04906v3, 2022

Previous slide.

For each object we have many possible inputs. Two inputs (e.g., original and augmented) for object n are called $\boldsymbol{x_n}$ and $\boldsymbol{x_n}$’.

These inputs generate representations in the output layer, called $\boldsymbol{z_n^L}$ and $\boldsymbol{z_n^L}$’.

Self-supervision: different objects in VICReg



Variance: All neurons must participate → control $\boldsymbol{SD}(k)$

minimize $\boldsymbol{max}(\boldsymbol{0}, \boldsymbol{1} - \boldsymbol{SD}(k))$.

Invariance: Make alternative représentations similar

minimize $(\boldsymbol{z_n^L}$ - $\boldsymbol{z_n^{L}}{'})^2$.

Covariance: assign neurons different roles → covariance

minimize $\boldsymbol{cov}(j,k)$.

Regularization by variance and covariance

A. Bardes, J. Ponce, and Y. LeCun. Vicreg: variance-invariance-covariance regularization for self-supervised learning. ICLR, arXiv:2105.04906v3, 2022

SD(j) is the standard deviation of output component j of the output vector $\boldsymbol{z_n^L}$ across all representations and and all object n.
→ SD(j) should be close to one for each component j to make sure that all neurons participate

cov(j,k) is the covariance between output components j and k.
→ cov(j,k) should be minimal for all pairs j and k so as to ensure that neurons have different roles (and do not all respond together to each stimulus).

Summary: Self-supervised Learning:

Examples:

- Predict original image from augmented image
- Predict original song from augmented song

→ Align representation in representation layer

BUT:

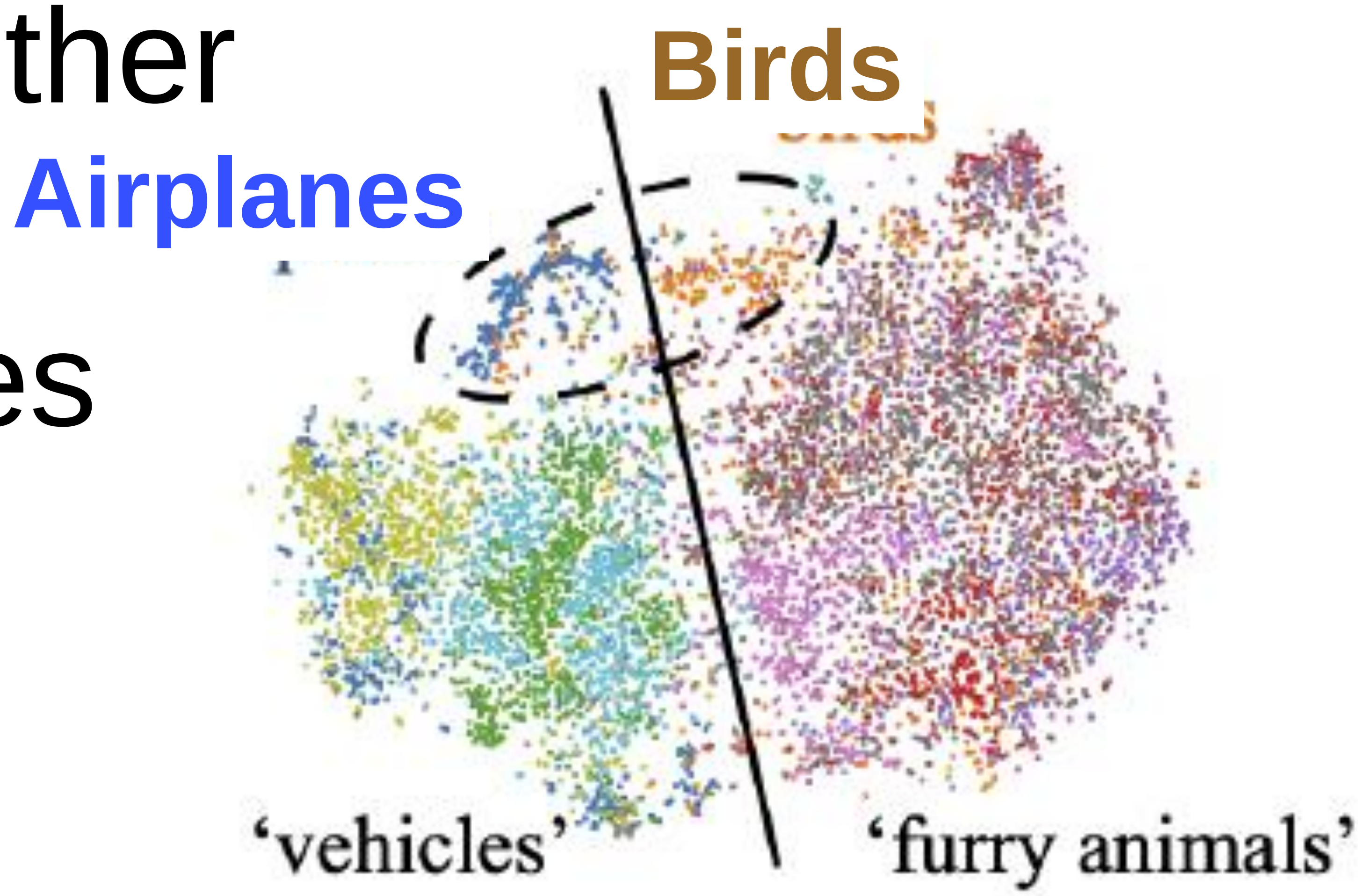
- Avoid collapse of representation

Solution 1 = contrastive learning

→Move different ‘objects’ far away from each other

Solution 2 = noncontrastive learning

→all neurons participate, but have different roles



Previous slide. Summary

Discussion: Last week

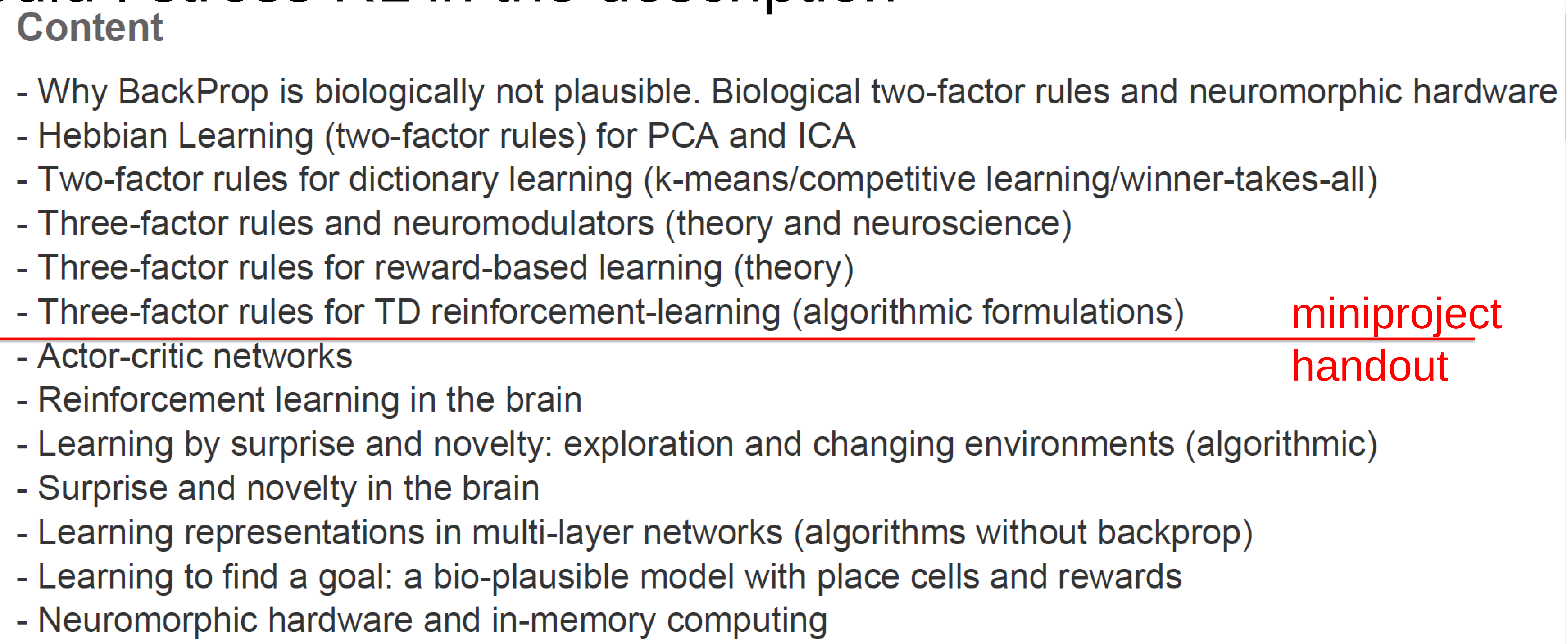
Semester plan

Should I reorder: RL earlier?

Should I stress RL in the description

Wulfram Gerstner

EPFL, Lausanne, Switzerland



Content

3 weeks

- Why BackProp is biologically not plausible. Biological two-factor rules and neuromorphic hardware
- Hebbian Learning (two-factor rules) for PCA and ICA
- Two-factor rules for dictionary learning (k-means/competitive learning/winner-takes-all)

4 weeks

- Three-factor rules and neuromodulators (theory and neuroscience)
- Three-factor rules for reward-based learning (theory)
- Three-factor rules for TD reinforcement-learning (algorithmic formulations)

miniproject handout

flexible topics

- Actor-critic networks
- Reinforcement learning in the brain
- Learning by surprise and novelty: exploration and changing environments (algorithmic)
- Surprise and novelty in the brain
- Learning representations in multi-layer networks (algorithms without backprop)
- Learning to find a goal: a bio-plausible model with place cells and rewards
- Neuromorphic hardware and in-memory computing

Assessment methods

Oral exam (70 percent) plus miniproject (30 percent).

If more than 45 students participate, the oral exam is replaced by a written exam.

Oral exam (27 min):

- 12 min Presentation of a research paper related to class.
- Followed by questions to paper and to lectures.
- Sample session during last 2 weeks (TA=role of student)

Questions?

For those who are not available on Tuesday 2pm-4pm,
we offer an alternative exercise sessions Wednesday, 4pm or 5pm

Previous slide. Repetition of slides from week 1.

The first 7 weeks are a very systematic introduction to

- Representation learning with two-factor rules
- Reinforcement learning with three-factor rule

Then we hand out the miniprojects. You can choose one of two projects:

(i) receptive field learning with two-factor rules
(ii) learning to navigate in a maze with three-factor rules

I will handle the last weeks a bit more flexibly in terms of topics.

The course finishes with an oral exam (unless the number of students is above 45).

Discussion:

Alternative titles for this class:

[ ] Learning in Neural Networks (existing title)

[ ] Learning in biological neural networks

[ ] Learning algorithms of the brain

[ ] Brain-style learning in Neural Networks

[ ] other

I wish you Good Luck and
a lot of success for your exams.