Transfer Learning and representations

Learning representations

How universal are representations in foundation models

Transferring representations

Antoine Cornuéjols

AgroParisTech – INRAé MIA Paris

EKINOCS research group

AgroParisTech

INRA SCIENCE & IMPACT

Outline

1. Transfer learning: definition
2. Transfering representations
3. IRM: Invariant Risk Minimization
4. Multi-task learning
5. Conclusions

Types of Out-Of-Distribution Learning

Goal

– Improve a **target** prediction function in the **target** domain using knowledge from the **source** domain

1. The **source** domain and **target** domain can be the **same**, but with different probability distributions: "**Domain Adaptation**"
   – **Co-variate shift** : same decision function $P_{Y|X}$, but ≠ distributions P_X
2. **Concept drift** : ≠ decision functions $P_{Y|X}$
3. Or they can be from **different domains**: **Transfer Learning**

Non stationary environment

- **Co-variate shift** $\mathbf{p}_{\mathcal{X}}$
  - Virtual drift
  - Non i.i.d.

E.g. moving robot

- **Change of concept** $\mathbf{p}_{\mathcal{Y}|\mathcal{X}}$
  - *Concept drift*
  - Non i.i.d. + non stationary

E.g. consummers' expectations

Concept shifts: illustrations

- **Spam** filtering
  - **Not** the same user: $P_{Y|X}$ may differ
    - E.g. for me conference announcements are important, but could be an annoyance to someone else

- **Changes in the tastes** or expectations of the consumers

- **Changes** in medicine
  - E.g. the severity of the COVID variants differs

The focus of the course

- Out-Of Distribution learning (OOD)
  - **Change of domain** between learning and testing: **Transfer** Learning

Learning →

Photo

Testing →

Cartoon

The focus of the course

- Out-Of Distribution learning (OOD)
  - Change of **domain** between learning and testing: Transfer Learning

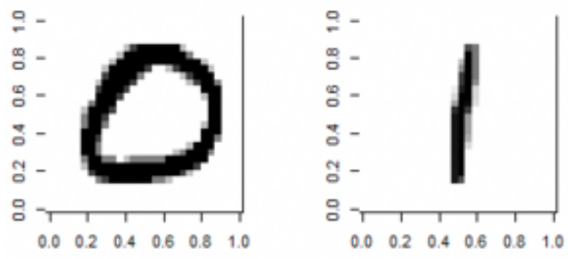
Learning ⟶


Testing ⟶


The focus of the course

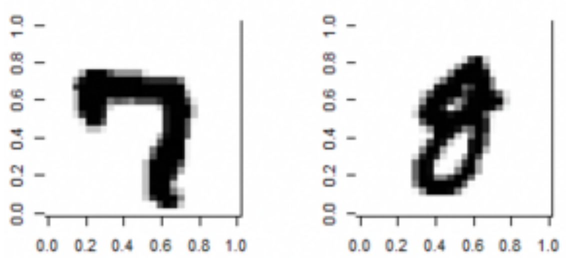
- Out-Of Distribution learning (OOD)
  - Change of **domain** between learning and testing: **Transfer** Learning

Learning



(a) Is it a zero or a one ?

Testing

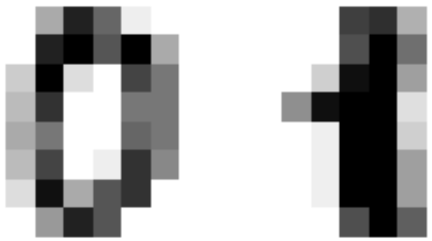


(b) Is it an eight or a seven ?

The focus of the course

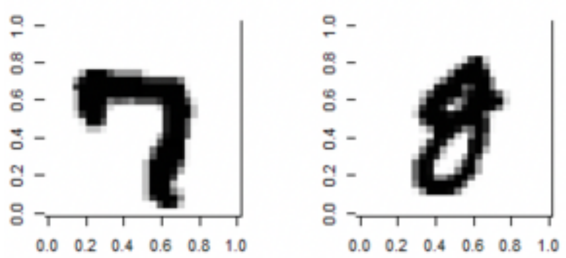
- Out-Of Distribution learning (OOD)
  - Change of **domain** between learning and testing: **Transfer** Learning

Learning →



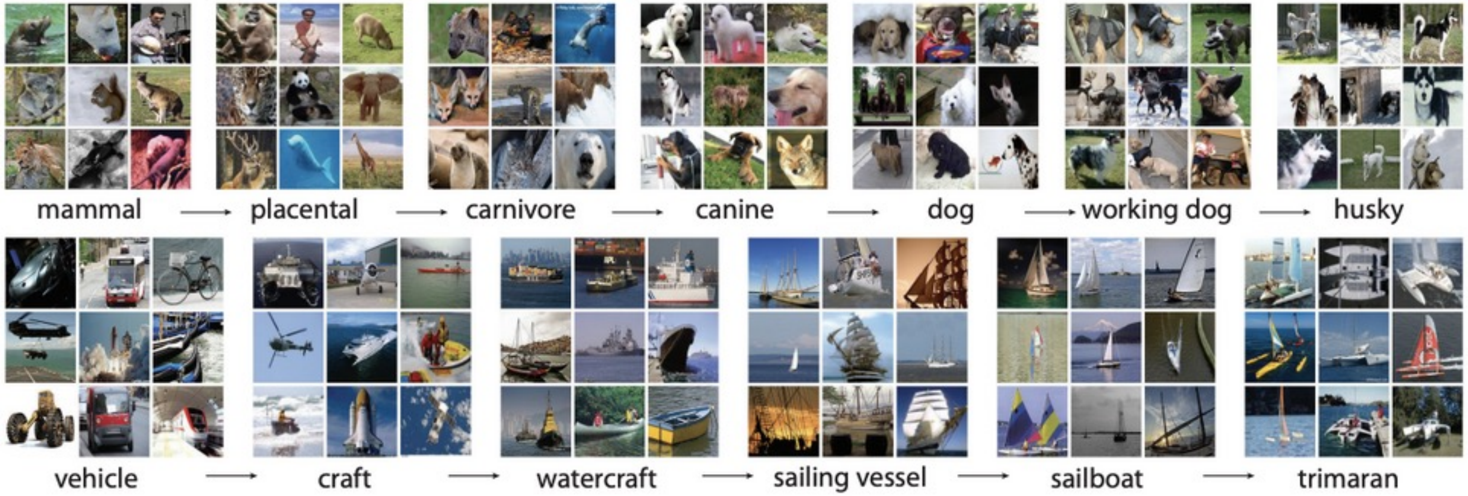
(a) Is it a zero or a one ?

Testing →



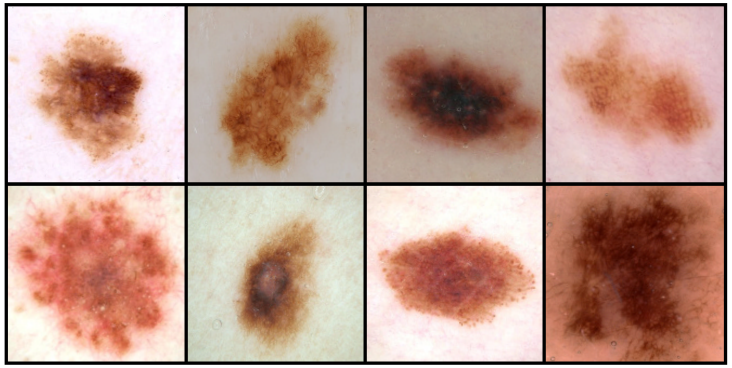
(b) Is it an eight or a seven ?

Transfer learning for **image** recognition



Endangered bird species

...



Tumors: malignant vs. benign

Transfer learning for **text styles**

A

Prove there is an infinity of prime numbers

The proof that there are infinitely many prime numbers is one of the most famous results in number theory. This proof was first attributed to the ancient Greek mathematician Euclid.

Here's a sketch of the proof:

1. **Assume the Opposite**: Let's assume, for the sake of contradiction, that there are only finitely many prime numbers. We'll denote them as $p_1, p_2, p_3, \ldots, p_n$.
2. **Form a New Number**: Consider the number $N = p_1 \cdot p_2 \cdot p_3 \cdot \ldots \cdot p_n + 1$. This is a new number formed by multiplying all the known primes and adding 1.
3. **Not Divisible by Known Primes**: Since N is not divisible by any of the known primes (by construction), there are two possibilities:
   - Either N itself is prime, which would mean there's at least one more prime beyond the ones we initially assumed.
   - Or N is composite and has a prime factor that wasn't among the original list of primes.
4. **Contradiction**: Either way, our initial assumption that we listed all primes leads to a contradiction. Thus, there must be infinitely many primes.

...

A

Prove there is an infinity of prime numbers $\boxed{\text{in alexandrines}}$

Certainly! I'll present a poetic proof in the form of alexandrines:

In numbers vast, the primes extend their reign,
Their infinite domain, they shall maintain.
Assume finitude, let's play this little game,
Suppose there's just a finite prime's name.

Now, form a number, grand and bold in might,
Multiply them all, add one to their sight.
This wondrous number, novel and pristine,
Defies division, as yet unseen.

For with the known primes, it does not align,
A contradiction, in this truth, we find.
This newborn number, either prime or wide,
Unveils new primes, our old beliefs deride.

Thus, in the realm of numbers pure and free,
Primes dance forever, an endless symphony.
No final count, no last prime shall we see,
Infinity reigns in number theory's decree.

So let it be known, in verses and in prose,
Primes march unbounded, where their story goes.
In alexandrines tall, their glory brightly glows,
An infinite tale, the mathemagician knows.

Transfer learning for **image styles**

1 Upload photo

The first picture defines the scene you would like to have painted.

Source

2 Choose style

Choose among predefined styles or upload your own style image.

Constraint of transfer

3 Submit

Our servers paint the image for you. You get an email when it's done.

Target

...

Transfer learning for **image styles**


Content

+


Style

=


Pastiche

Source

Constraint of transfer

Target

Transfer Learning: change of domains, change of tasks

- Change of **domain**
  - E.g. Recognition of the same objects but in a **different environment**

- Change of **task**
  - E.g. learning to **play chess** after having learned to **play checkers**

- Definition [Pan, TL-IJCAI'13 tutorial]
  - Ability of a system to **recognize** and **apply** knowledge and skills learned in **previous domains/tasks** to **novel domains/tasks**

- Example
  - We have **labeled images** from a **web corpus**
  - Novel task: **is there a person** in unlabeled images from a **video corpus**?


?

Person no Person Is there a Person?

Transfer Learning for sentiment analysis

	Electronics	Video games
✓	(1) Compact; easy to operate; very good picture quality; looks sharp!	(2) A very good game! It is action packed and full of excitement. I am very much hooked on this game.
✓	(3) I purchased this unit from Circuit City and I was very excited about the quality of the picture. It is really nice and sharp.	(4) Very realistic shooting action and good plots. We played this and were hooked.
✗	(5) It is also quite blurry in very dark settings. I will never_buy HP again.	(6) It is so boring. I am extremely unhappy and will probably never_buy UbiSoft again.

- Source specific: *compact, sharp, blurry*.
- Target specific: *hooked, realistic, boring*.
- Domain independent: *good, excited, nice*, *never_buy, unhappy*.

[Pan, TL-IJCAI'13 tutorial]

Notations

1. **Source** domain *S*
   - Source **training data** S_S
   - Source data **distribution** D_S
   - Source **hypothesis** h_S

2. **Target** domain *T*
   - Target **training data** S_T $(|S_T| << |S_S|)$
   - Target **data distribution** D_T
   - Target **hypothesis** h_T

What can we **transfer** from one task to another?

- In the following: a **strong assumption**

There is **something in common** between the source and the target

We will remove this assumption later on

What can we transfer

- What could be in **common**?

1. Look for a universal **representation**

2. Underlying supposedly **common regularities**

3. Learning a translation to a **common decision function**

4. **Others**

Outline

1. Transfer learning: definition
2. Transfering representations
3. IRM: Invariant Risk Minimization
4. Multi-task learning
5. Conclusions

Universal representations ?

Learning representations

with deep neural networks

Representation learning: approaches

Learning method	Learning principle	Summary
Autoencoding	Compression	Remove **redundant** information
Contrastive	Transforming space	Examples of the **same class** are **close** and **far** apart **if not** of the same class
Clustering	Compression	**Quantize** continuous data into discrete categories
Future prediction	Prediction	Predict the **future**
Imputation	Prediction	Predict **missing data**
Pretext (proxy) tasks	Prediction	Predict **abstract properties** of the data

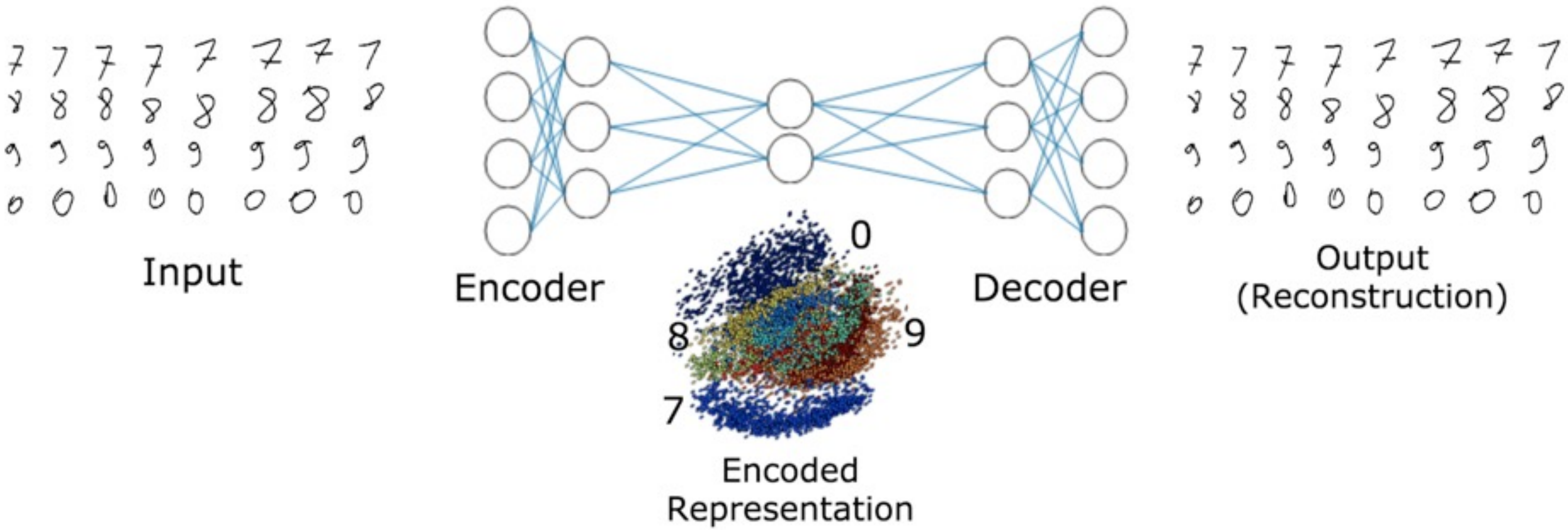
…

Inspired from [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.443]

Auto-encoding

Auto-encoding

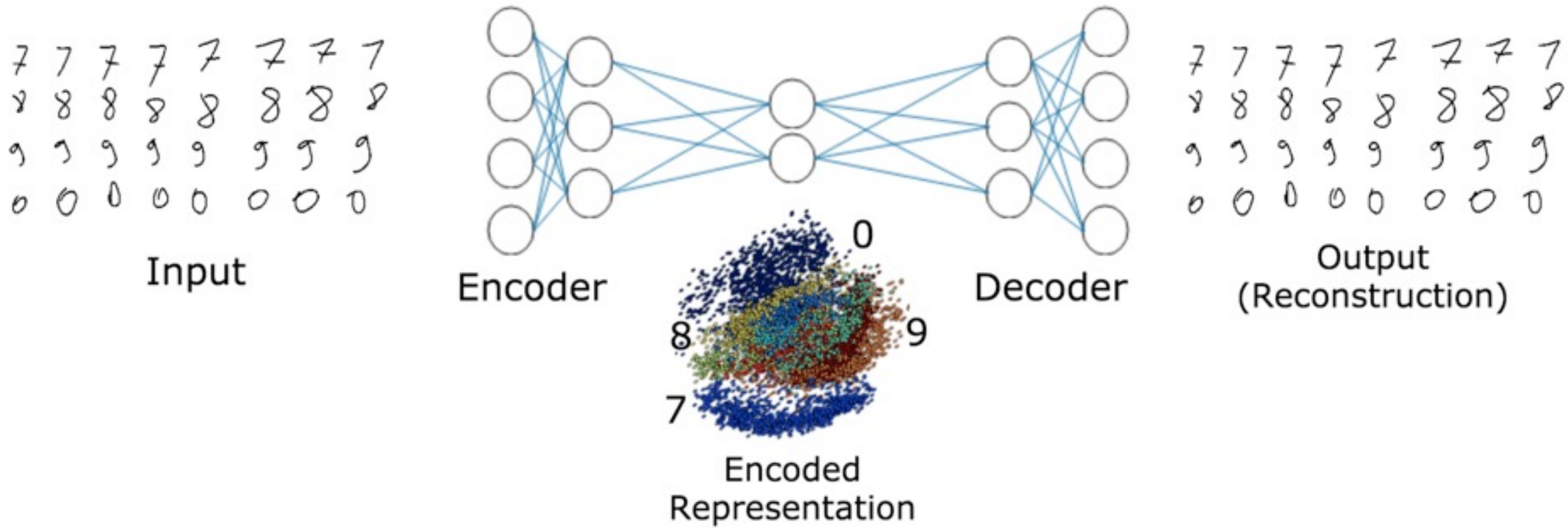
- An old idea: auto-association   (dates back to 1986)



Internal representation: the *embedding*

Auto-encoding

- An old idea: auto-association (dates back to 1986)



Supervised learning...

... without labels!

Autoencoding

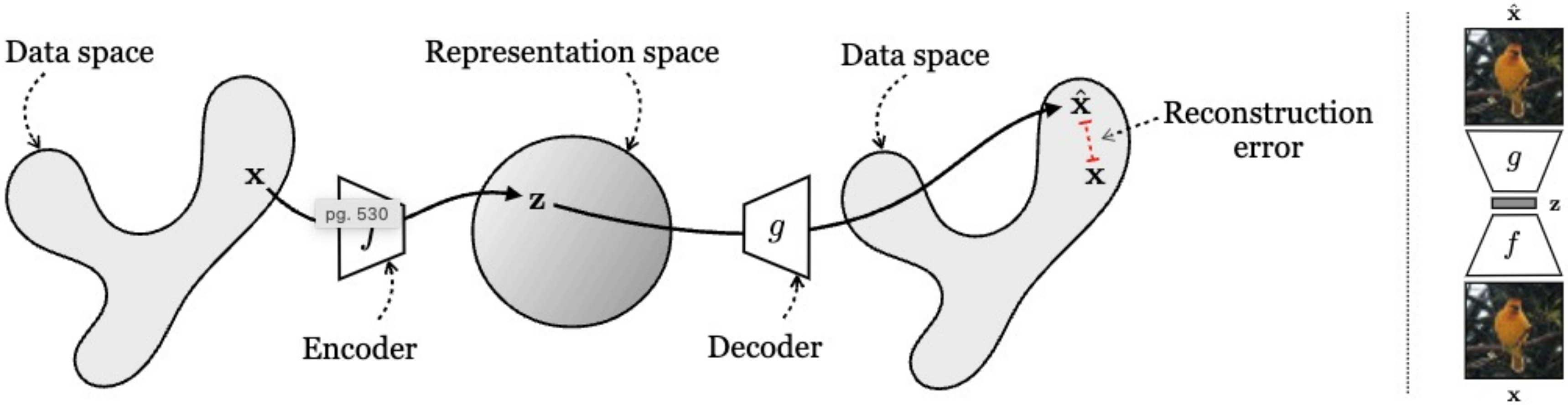


Figure 30.2: (left) An autoencoder maps from points in data space, to points in representation space, and back. (right) An example of running an autoencoder on an input image of a bird.

$$f^\star, g^\star \;=\; \underset{f,g}{\mathrm{ArgMin}}\, \mathbb{E}_X ||g(f(x)) - x||_2^2$$

If *f* and *g* are linear, this is PCA

From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.444]

Autoencoding

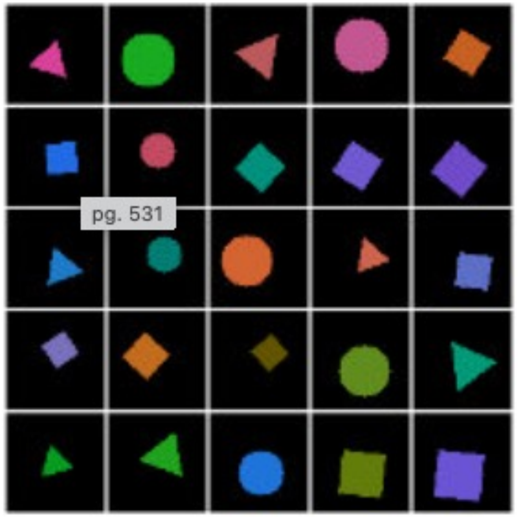
- Autoencoders learn a compress representation of the data, but *do they learn a useful representation*?

Autoencoding

- Autoencoders learn a compress representation of the data, but *do they learn a useful representation*?

- Experiment
  - **Data set** with 64,000 images
    - 8 colors
    - 3 shapes: circles, triangles, squares
    - Randomized *size*, *position* and *rotation*

Relevant



  - **Autoencoder** architecture and training
    - 6 convolutional layers (for the **encoder**, and 6 for the **decoder**)
    - Relu nonlinearities
    - **128**-dimensional bottleneck
    - Trained for 20,000 epochs of stochastic gradient with batch size = 128

From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.446]

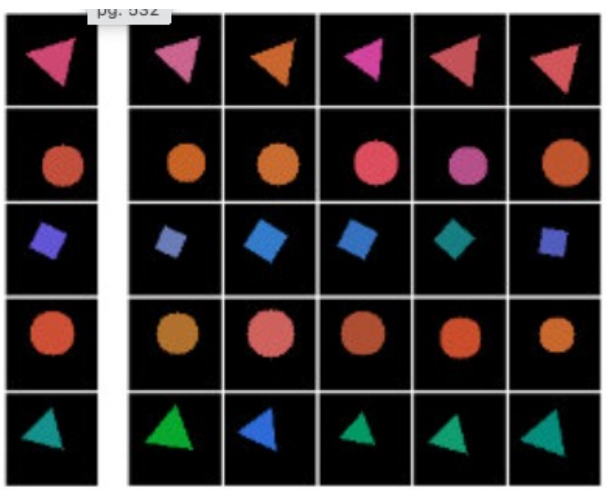
Do autoencoders learn **useful representations**?

For a given **query** (a pattern)

1. Visualize the images in the data set whose embeddings are **nearest neighbors** to the **query's embedding** (layer 6)

2. Measure the accuracy of a **one-nearest-neighbor** in embedding space

Query



Nearest neighbors are **similar** in terms of colors, shapes, positions and rotations in the embedding of the **6th layer**.

From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.446]

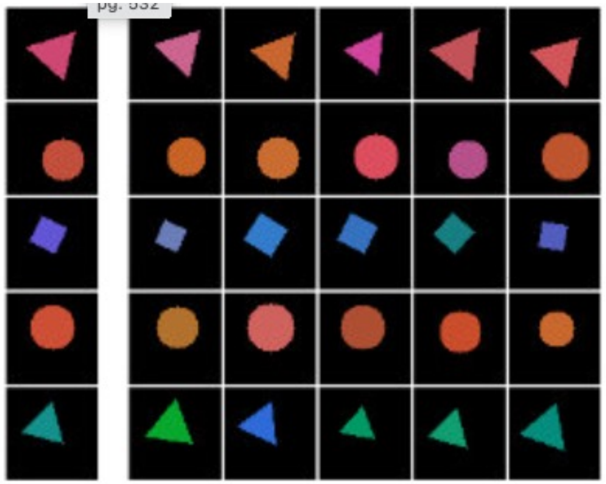
Do autoencoders learn **useful representations**?

For a given **query** (a pattern)

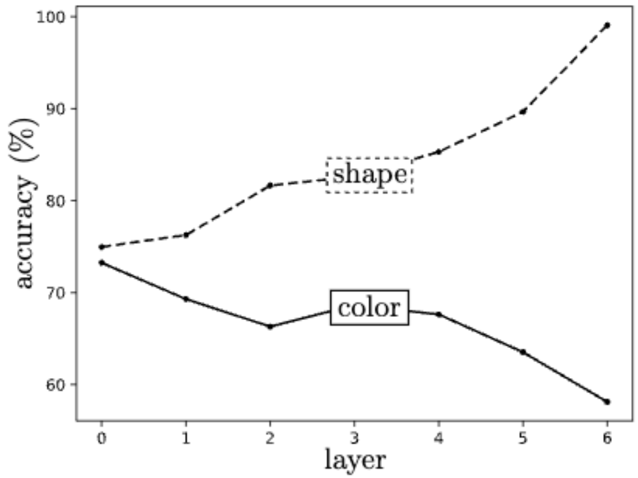
1. Visualize the images in the data set whose embeddings are **nearest neighbors** to the **query's embedding** (layer 6)

2. Measure the accuracy of a **one-nearest-neighbor** in embedding space

Query



Nearest neighbors are **similar** in terms of colors, shapes, positions and rotations in the embedding of the 6th **layer**.



The **color classifier** performs best in the 1st layer embedding (raw data) whereas the **shape classifier** performs best on the 6th layer (more abstract).

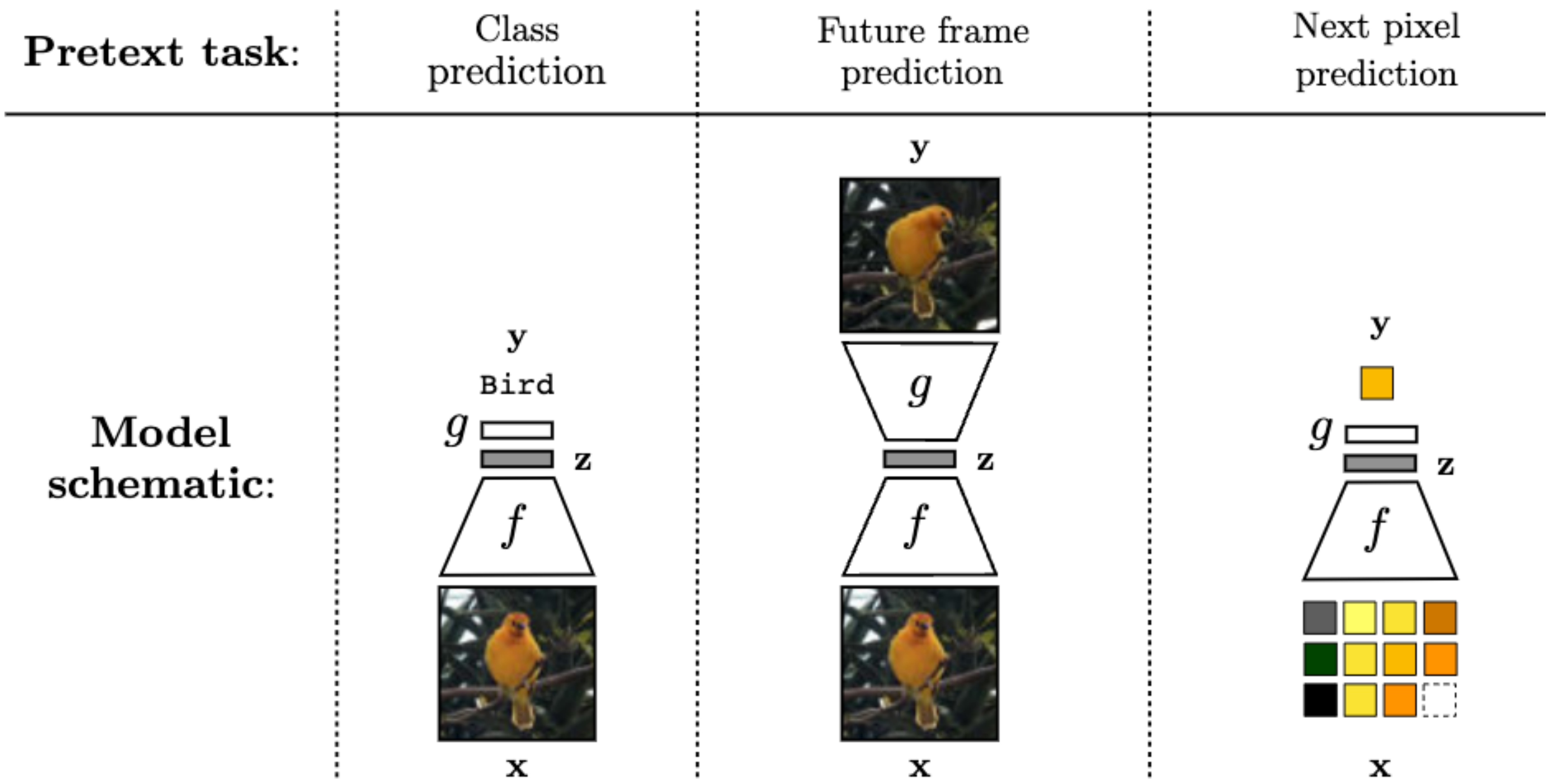
No representation is UNIVERSAL

From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.446]

Predictive encoding

Predictive encoding

- We make prediction to learn a **useful representation**
  - Not to predict per se -- > a pretext (or proxy) learning task

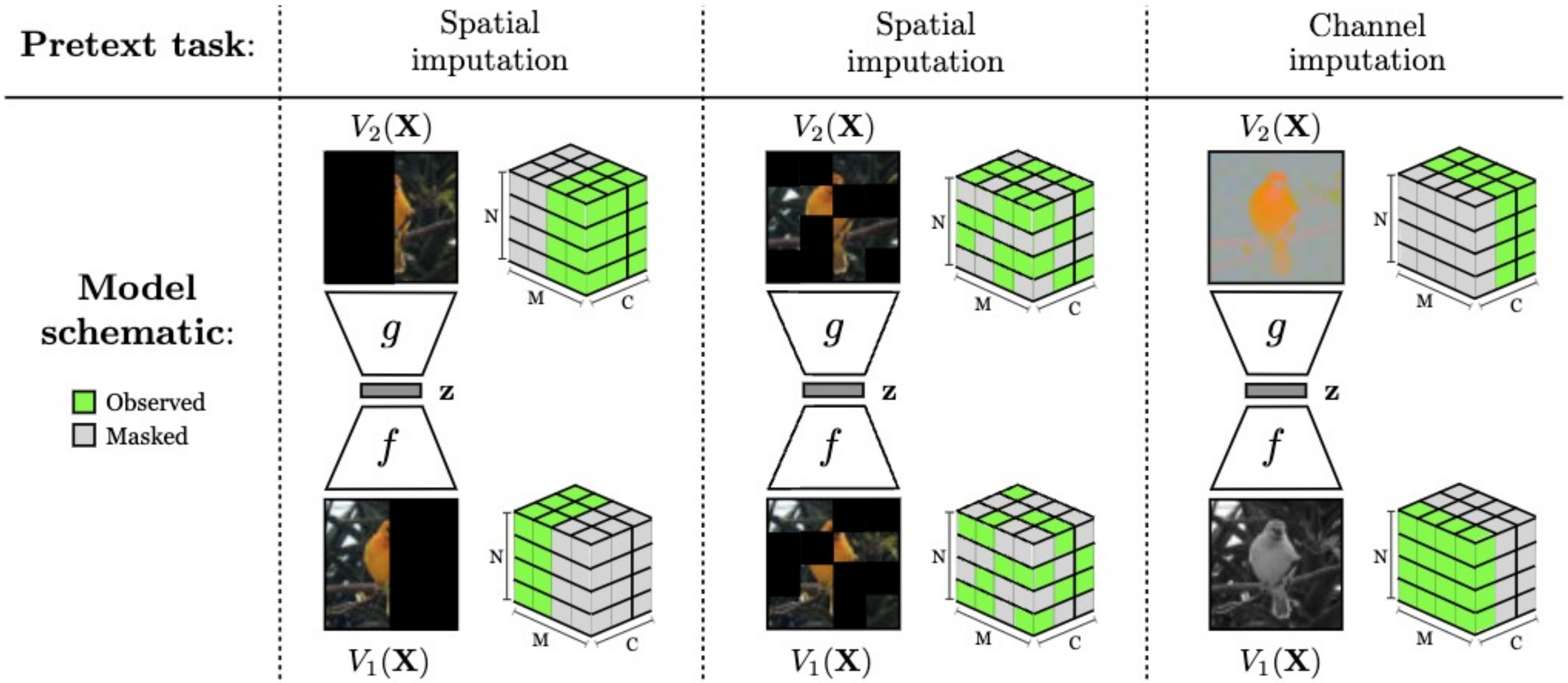


From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.447]

Self-supervised learning

Learning by **imputation**

- **Imputation** is a special case of **self-supervised learning**, where the prediction targets are missing elements of the input data

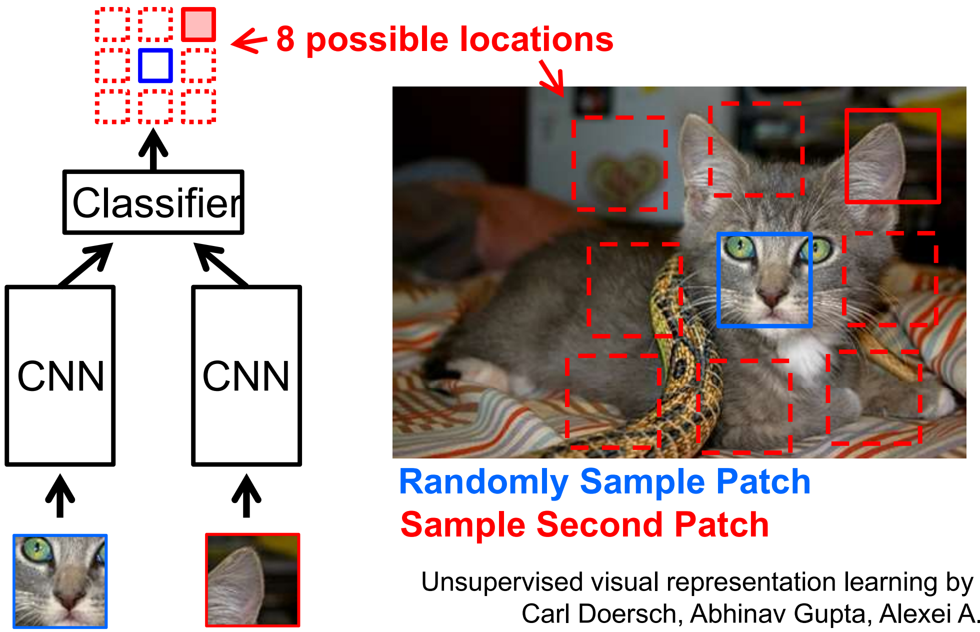


From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.450]

Self-supervised learning

- A type of supervised learning but where the labels are provided by the learner itself, therefore possible from unsupervised data!
  - Example: train a network so that it predicts the relative positions of two subimages

  It is thus hoped that the system will learn spatial **relationships** that will be useful for other tasks related to images



Unsupervised visual representation learning by context prediction,
Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015

Self-supervised learning

In **sequences**

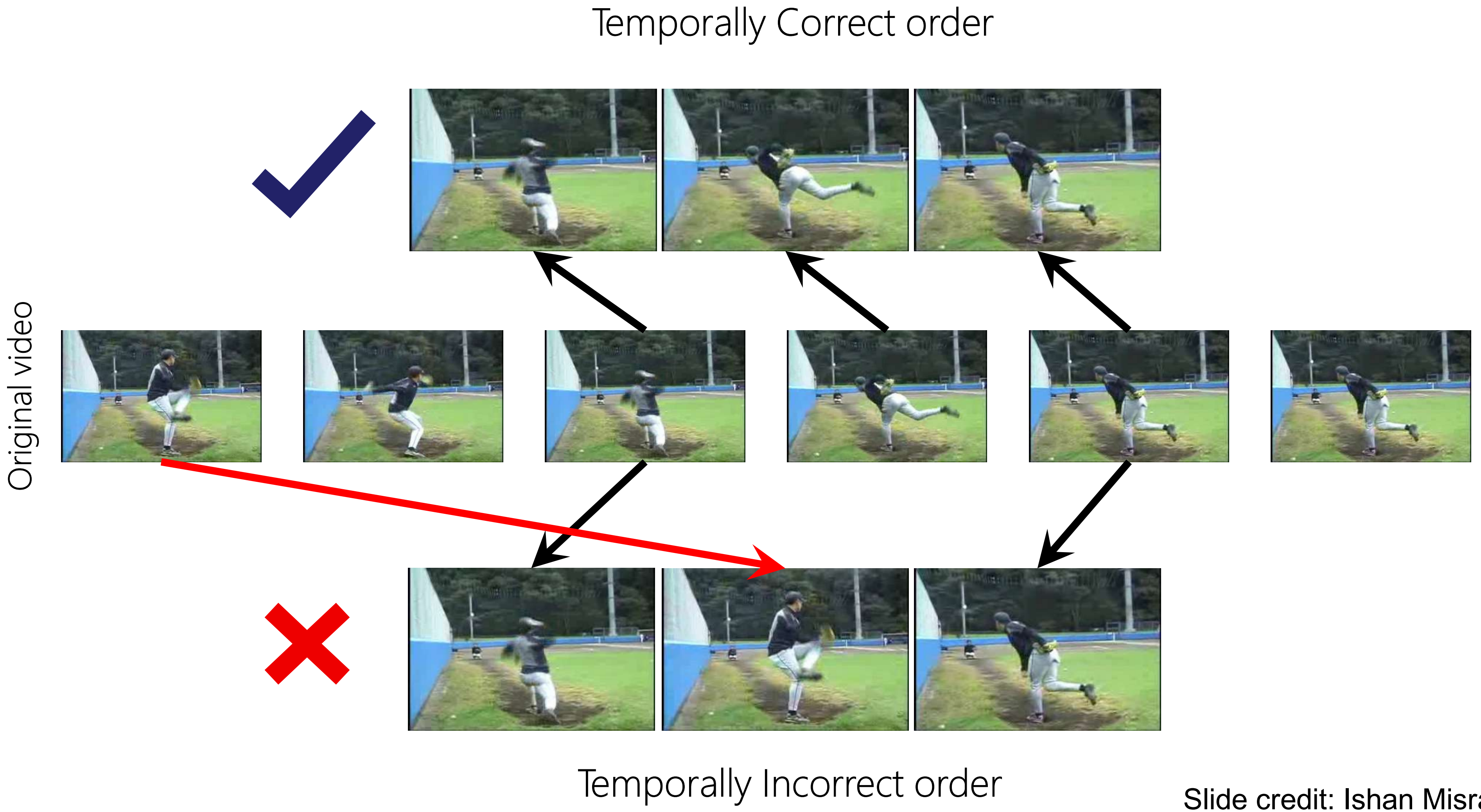
- Is this a **valid** sequence?

Original video

Sun and Giles, 2001; Sun et al., 2001; Cleermans 1993; Reber 1989
Arrow of Time - Pickup et al., 2014

Slide credit: Ishan Misra

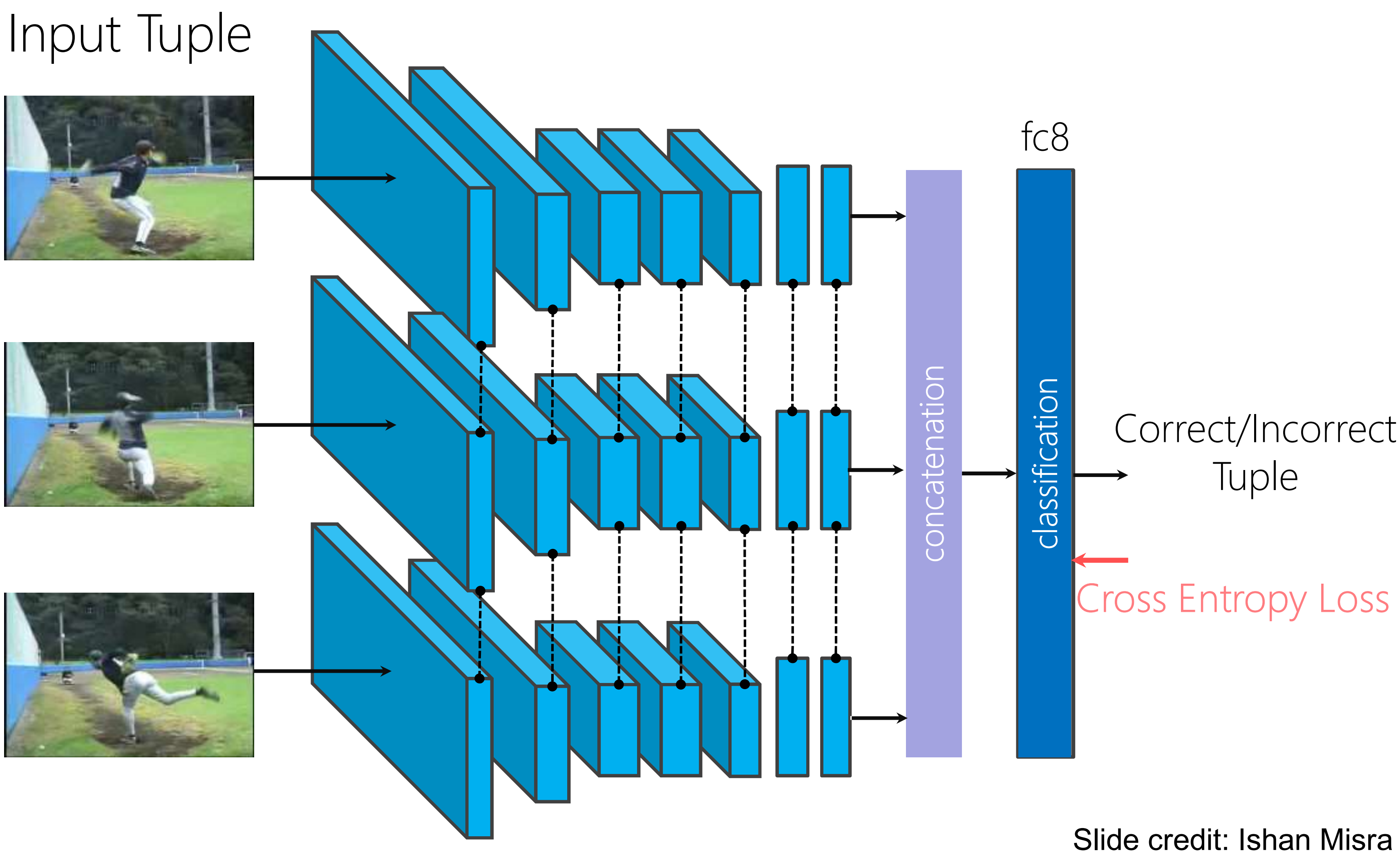
Self-supervised learning



Slide credit: Ishan Misra

Self-supervised learning

...

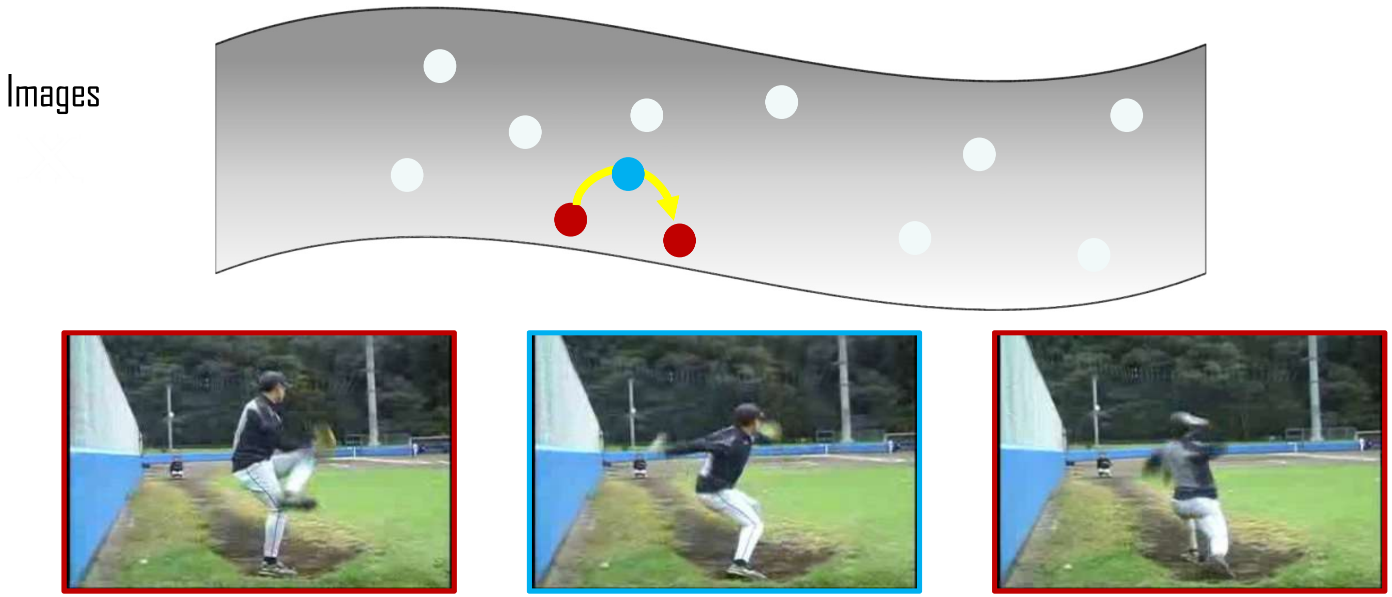


Slide credit: Ishan Misra

Self-supervised learning

...

Learning an embedding that satisfies the order of sequences



Given a start and an end, can this point lie in between?

Shuffle and Learn – I. Misra, L. Zitnick, M. Hebert – ECCV 2016

Slide credit: Ishan Misra

ITHACA (March 2022)

- Given an incomplete tablet, ITHACA generates predictions for the missing words in order to recover the whole text.
- The historians choose the final answer using their expertise
- Accuracy = **62**% vs. 25% for the experts alone!!
- Provides also a probability for the geographic source and the date between 800 BC et 800 AC.

Data augmentation

Data augmentation

- Principle: **augment the data set** by adding random transformations of each data point without changing their class
- Transformations should correspond to **invariances** in the data

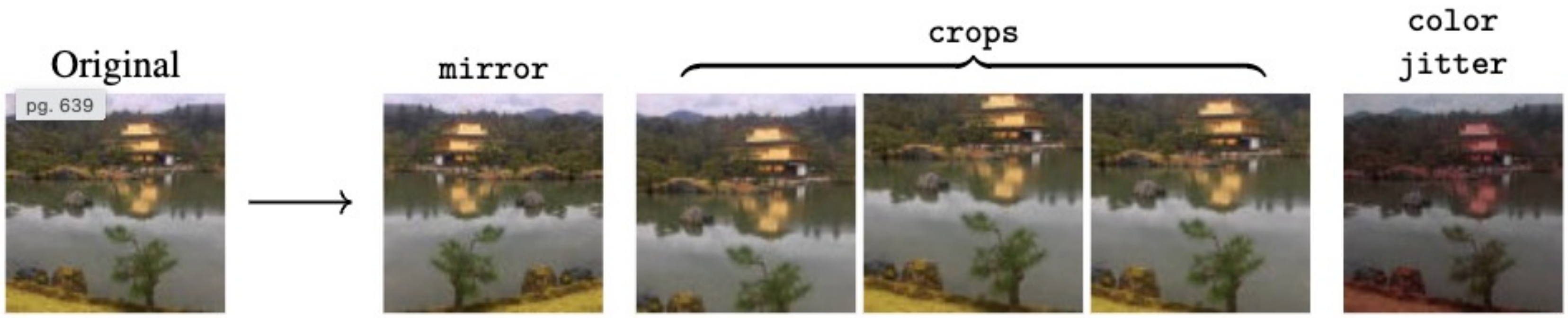


Figure 36.1: A few common types of data augmentation.

From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.539]

Data augmentation

...

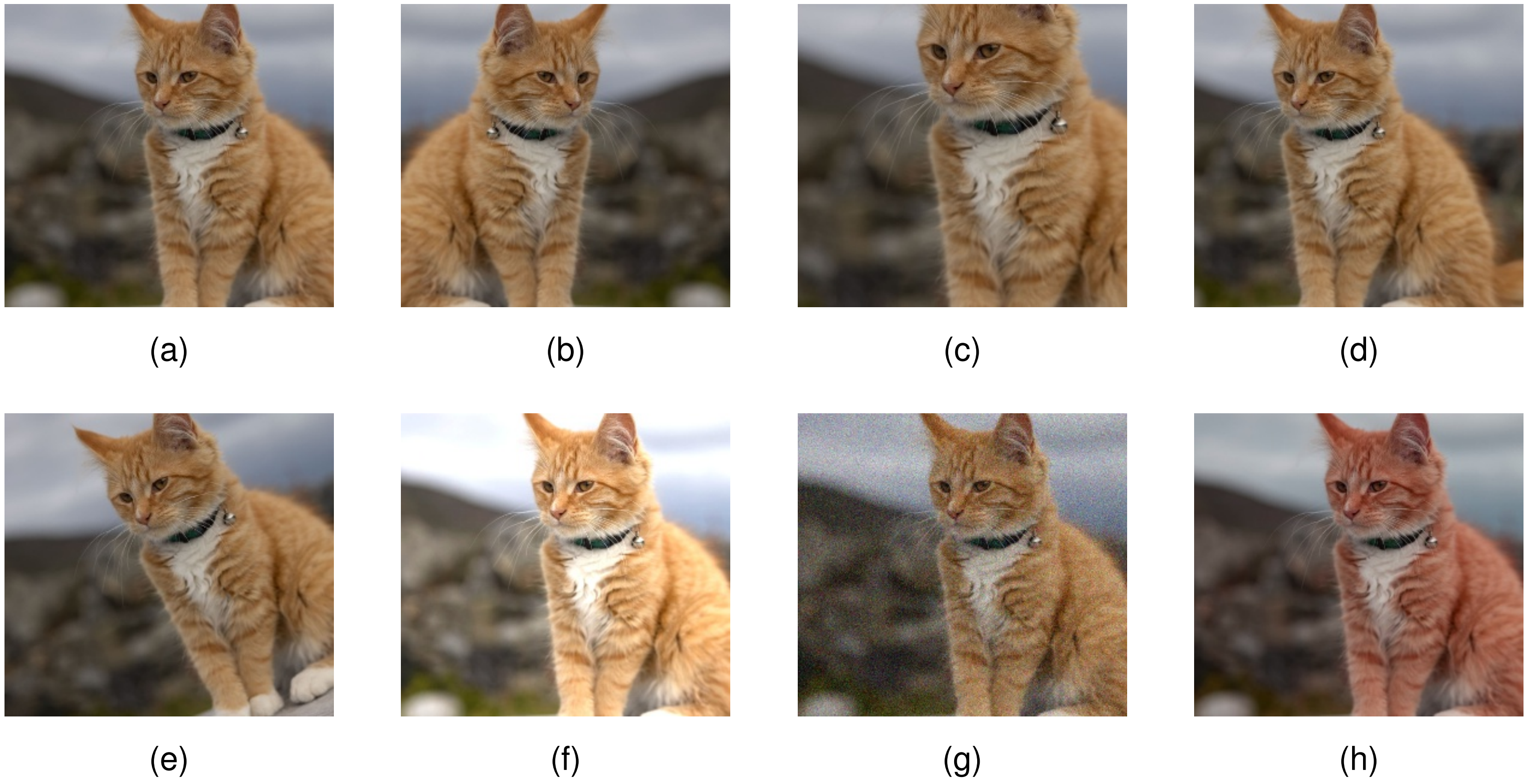


Figure 9.1 Illustration of data set augmentation, showing (a) the original image, (b) horizontal inversion, (c) scaling, (d) translation, (e) rotation, (f) brightness and contrast change, (g) additive noise, and (h) colour shift.

From [Christopher Bishop. **Deep learning. Foundations and concepts**. Springer, 2024, p.258]

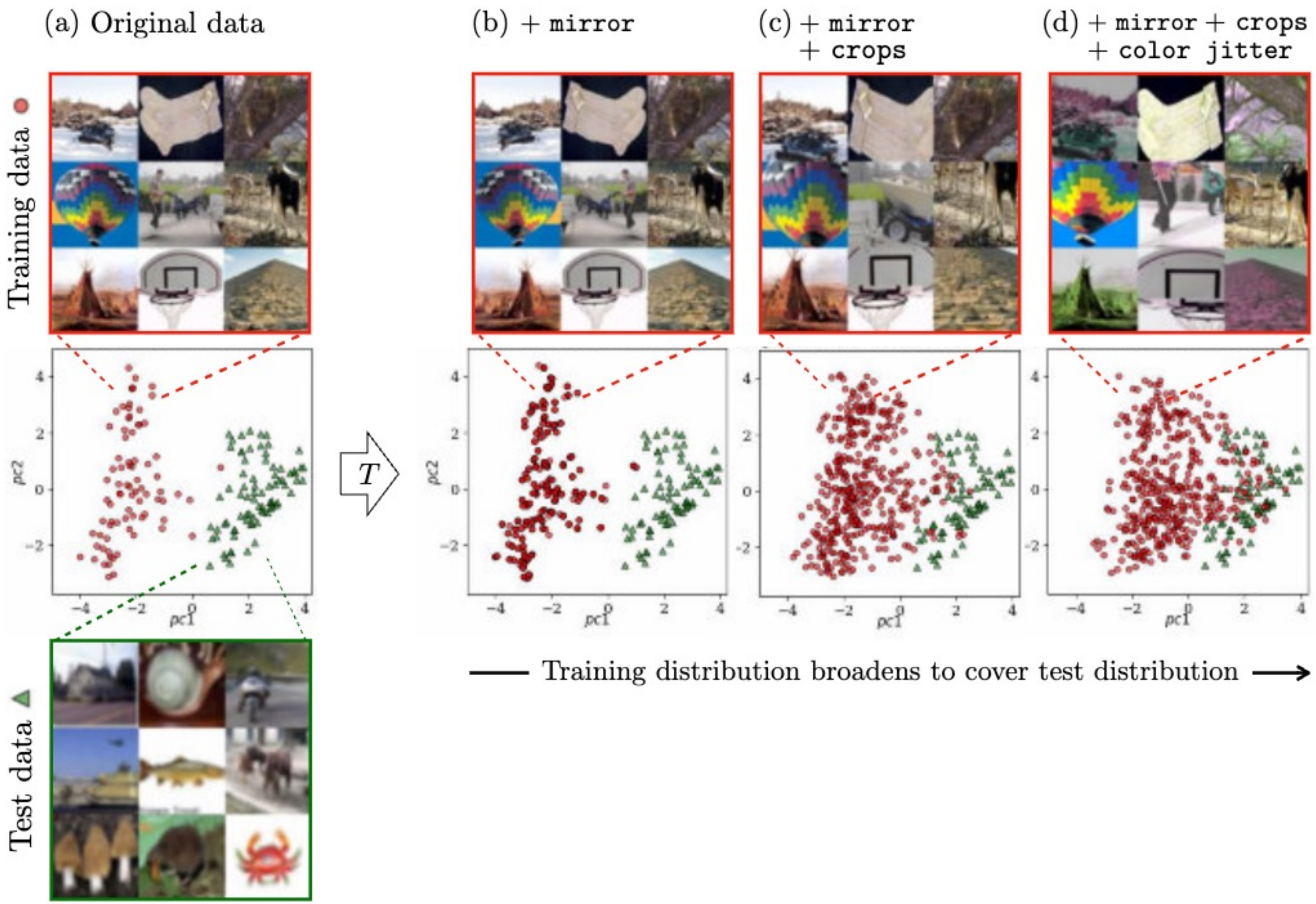


Figure 36.2: Data augmentation broadens the training distribution so that it might better cover the test cases. (a) Training data are from Caltech256 [175] and test data are from CIFAR100 [278]. The scatter plots show these data in a 2D feature space [the first two principle components (pcs) of CLIP [394]]. (b) Training data after `mirror` augmentations (random horizontal flips). (c) The same plus `crops` (crop then rescale to the original size). (d) The same plus `color jitter` (random shifts in color and contrast).

...

From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.541]

Remarks on data augmentation

- With data augmentation, we aim at **enlarging** as much as possible the **training distribution** so that it covers the test distribution as well
- **But** it is usually impossible to train on all possible queries we might encounter, we often need to rely on **transfer learning** to adapt to new kinds of queries and new tasks

See course on transfer learning to come

Contrastive learning

Contrastive learning

- We want to **learn invariant representations** to certain transformations
  - This is what **Convolutional NNs** do: **translational** invariance
    - The hypothesis space is tailored for that
  - We can **force** invariances
    - Through the objective function
  - **Penalize** deviations from the invariance we want
    - Suppose **T** is a transformation we wish our representation to be invariant to
    - f is the encoder
    - **Loss** function $||f(T(x)) - f(x)||_2^2$

Contrastive learning

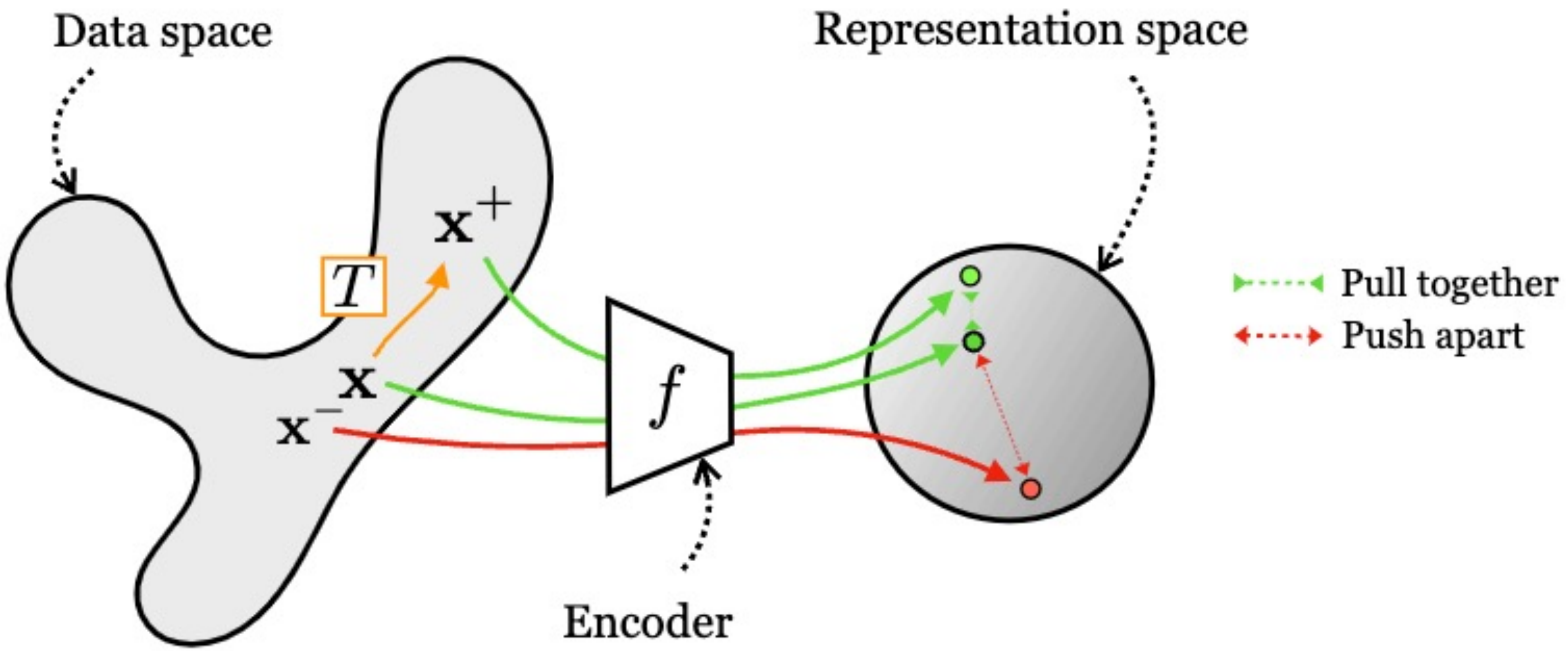
– **Penalize** deviations from the invariance we want

- Suppose **T** is a transformation we wish our representation to be invariant to
- *f* is the encoder
- Loss function $||f(T(x)) - f(x)||_2^2$ *Alignment loss*

- But we want to **avoid** that *f* projects all points to the same vector. (The *collapsing* phenomenon)
- **Second loss function** that pushes apart embeddings of data points for which we do not want an invariant representation
- **Positive** pairs $\{x, x^+\}$ with $x^+ = T(x)$
- **Negative** pairs $\{x, x^-\}$

Contrastive learning

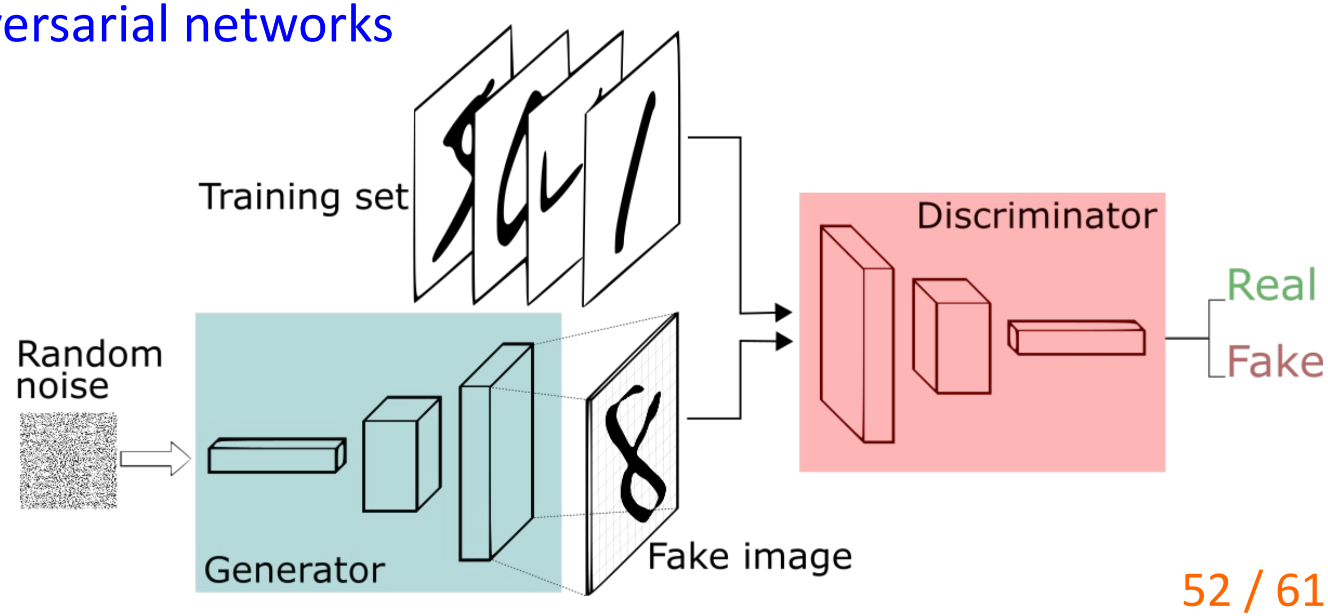
- Illustration



From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.456]

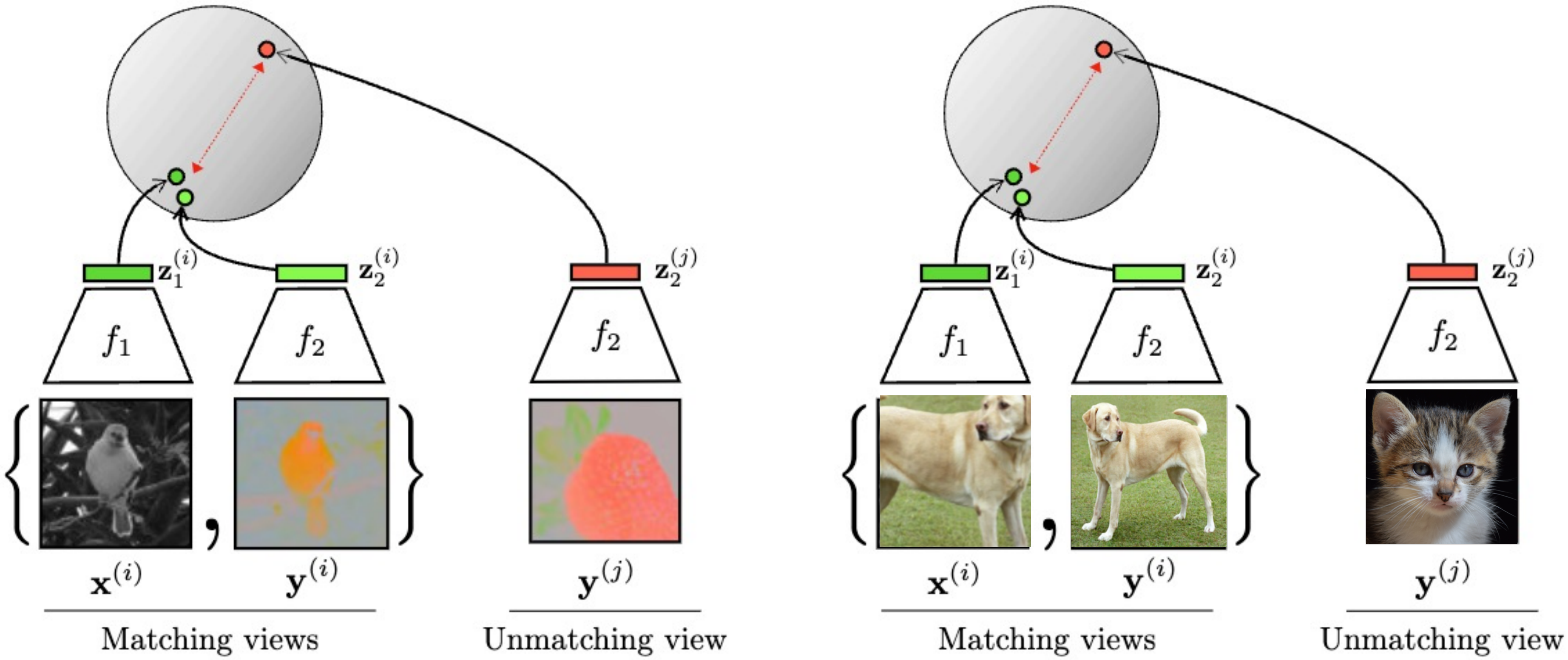
Constrastive learning

- Typically, the examples x^+ are obtained through **data augmentation** techniques from x
  - **Rotation**
  - **Mirror** image
  - Change of **luminosity**
  - Change of **scale**
  - Generative adversarial networks (**GAN**)



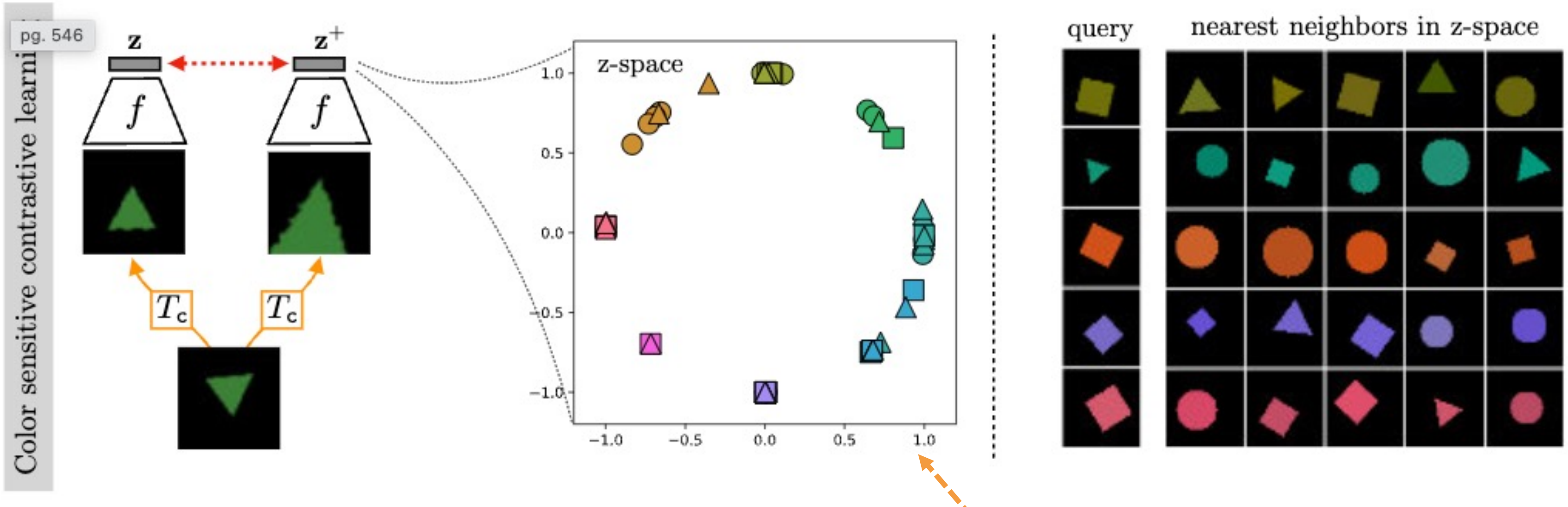
Contrastive learning

- Illustrations



From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.456]

Contrastive learning

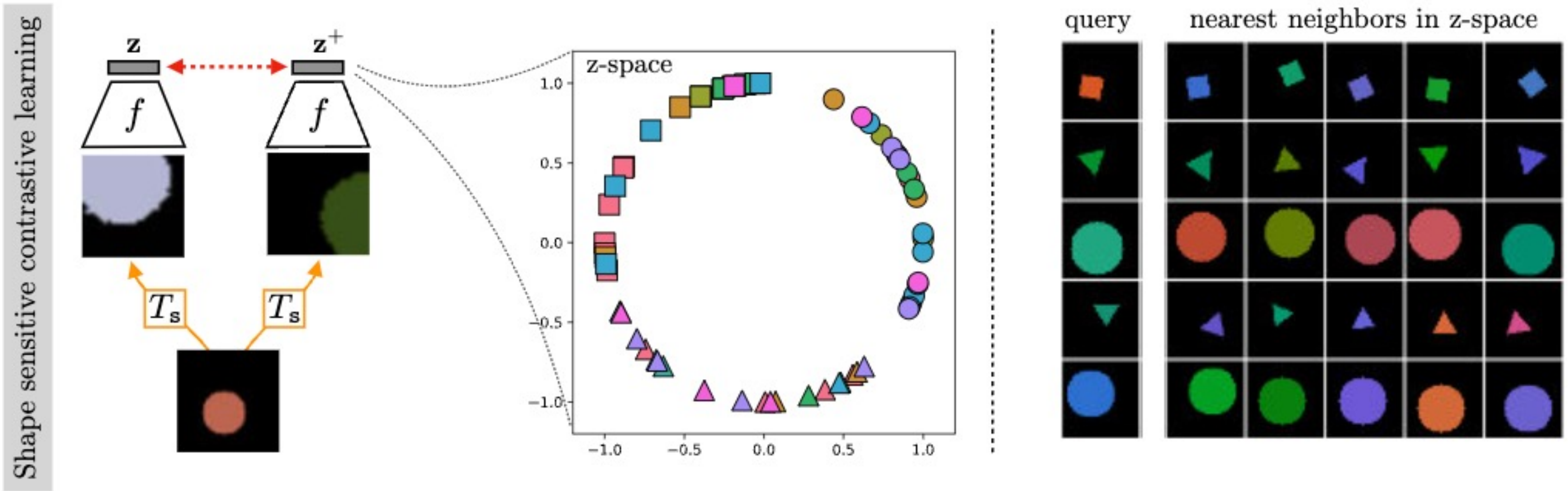


- Same experiments with autoencoding as for measuring the usefulness of representations learned by autoencoders, but with **embedding of dimension 2** instead of 128
- Transformations T_c sensitive to **color** and **invariant to shapes**
- The **embedding** becomes **invariant to shapes** and spreads the colors rather uniformly

From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.459]

Contrastive learning

- Same experiments with autoencoding as for measuring the usefulness of representations learned by autoencoders, but with embedding of dimension 2 instead of 128
- Transformations T_S sensitive to **shapes** and **invariant to colors**
- The **embedding** becomes **invariant to colors** and spreads the shapes rather uniformly



From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.459]

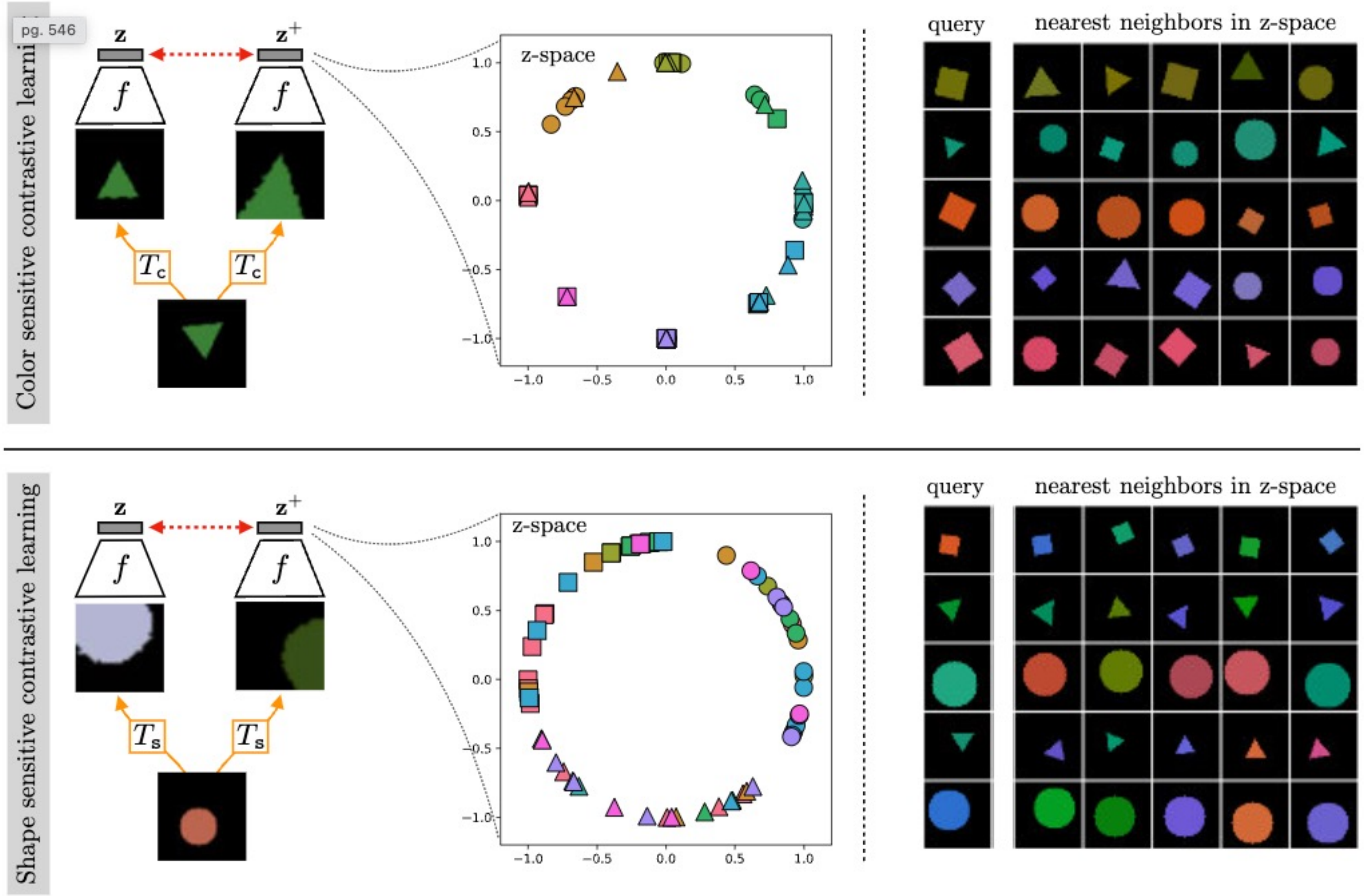


Figure 30.13: Contrastive learning on colored shapes using two different transformations for creating positive pairs. The choice of transformation controls which features the embedding becomes sensitive to and which it becomes invariant to.

…

From [Antonio Torralba et al. **Foundations of Computer Vision**. MIT Press, 2024, p.459]

Constrastive learning

- The triplet loss

$$\mathcal{L}(x, x^+, x^-) = \mathrm{Max}\big(D(f(x), f(x^+)) - D(f(x), f(x^-)) + m, 0\big)$$

The distance of the **negative pair** must be

at least the distance of the **positive one** + m

Conclusions

- In all cases, the hope is to learn a **representation** that captures **regularities**
  - that are **useful** for the task at hand,
  - and possibly for a **variety of tasks**.

Kind of an **invariance** wrt. the tasks

Kind of an **invariance** wrt. the tasks

Are **foundation models** the ultimate solution?