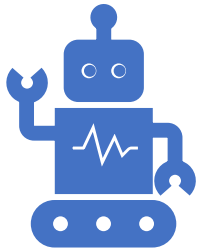


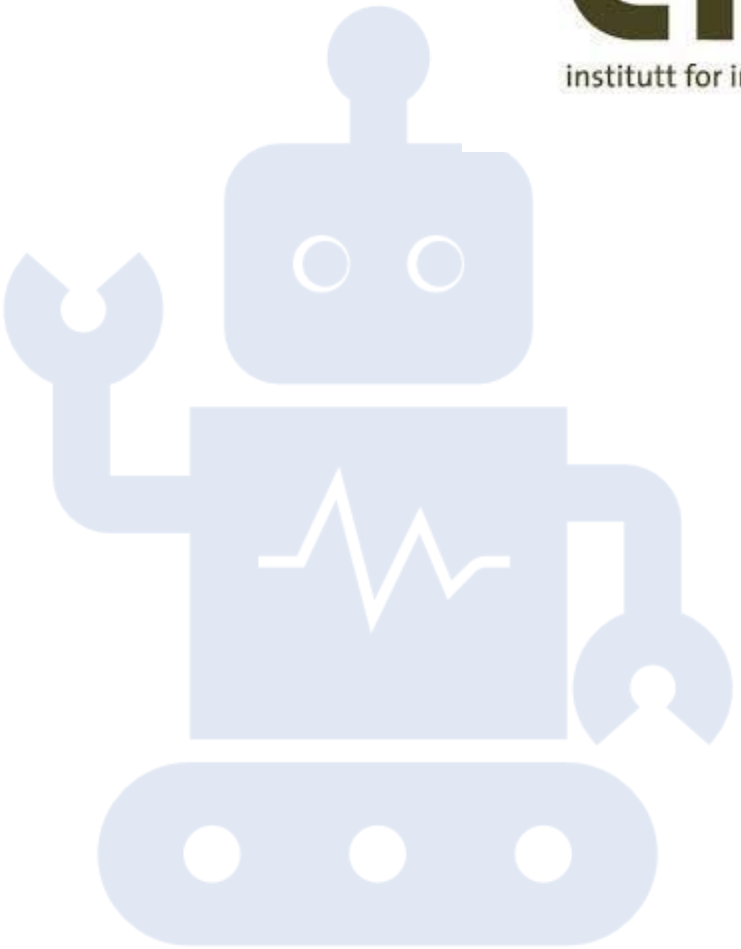
UiO : University of Oslo

institutt for informatikk



IN3050/IN4050 - Introduction to Artificial Intelligence and Machine Learning

Unsupervised Learning



Plan for today

- Guest Lecture by Benedikte Wallace - Modelling dance with machine learning

- Short summary of Unsupervised Learning
- Self-Supervised Learning

Unsupervised Learning

1: Introduction

3: Learning how data are grouped

- The most typical method of unsupervised learning

4: Learning how to compress data

5: Learning how to represent data in low dimension

2: Learning how to generate similar data

- Debatable whether this counts as “unsupervised learning”

Supervised vs. unsupervised learning

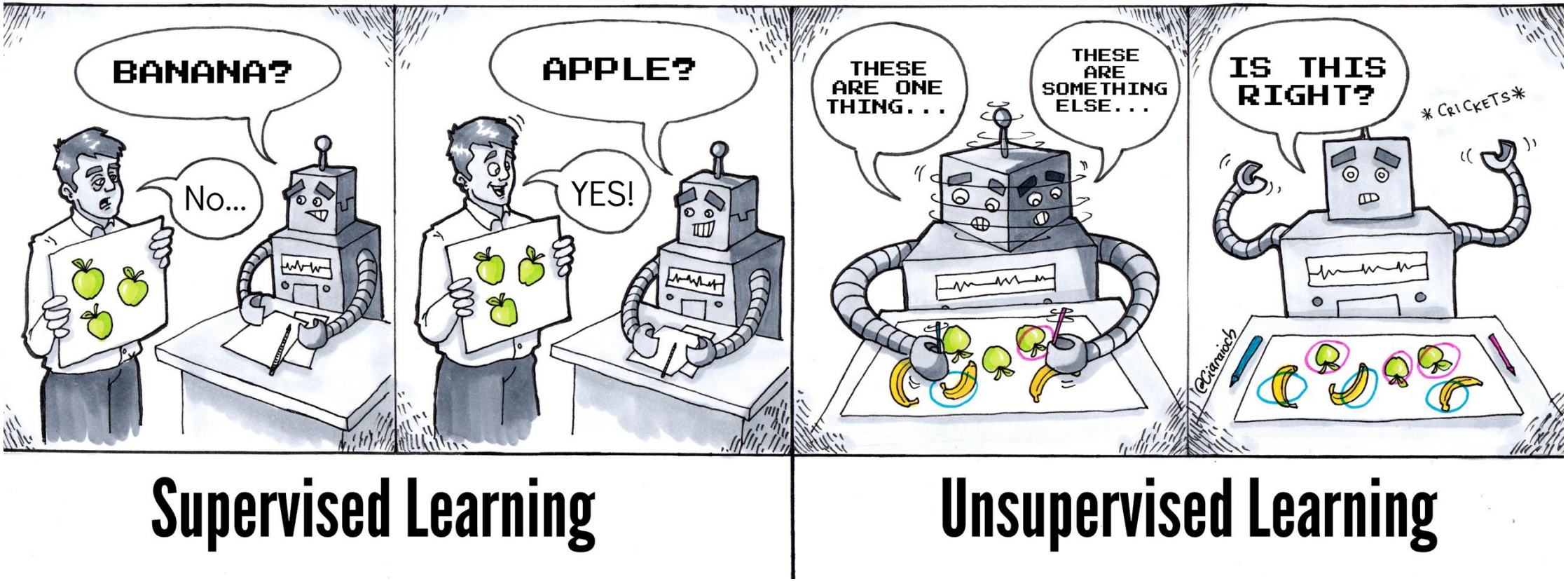
Supervised learning:

- Training set consisting of
  - a feature vector $\boldsymbol{x} = (x_1, \ldots, x_m)$ for each sample
  - a response/target y for each sample ("label")
- Goal: learn the mapping f: $\boldsymbol{x} \mapsto y$.

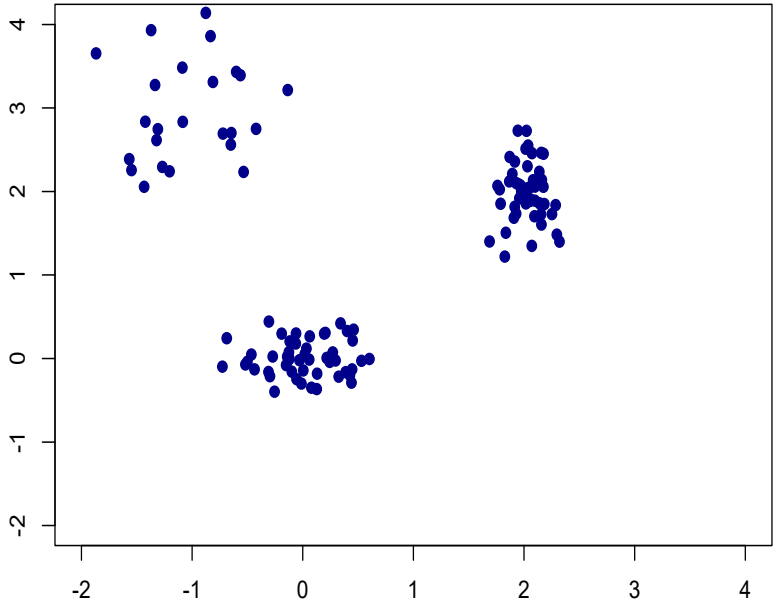
Unsupervised learning:

- Training set consisting of
  - a feature vector $\boldsymbol{x} = (x_1, \ldots, x_m)$ for each sample
- Goal: learn *something* about the distribution of the $\boldsymbol{x}$'s.

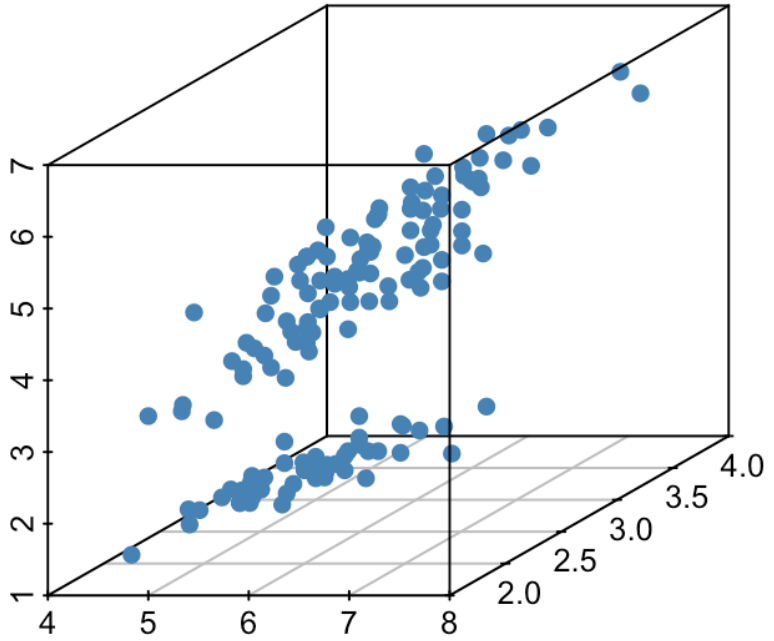
Supervised vs. unsupervised learning



Unsupervised learning = discovering structural features



Two features $\boldsymbol{x} = (x_1, x_2)$



Three features $\boldsymbol{x} = (x_1, x_2, x_3)$

Key unsupervised learning tasks:

- **Discovering groups of similar objects**
- **Data compression**
- **Dimensionality reduction**
- **Learning to generate fake data**

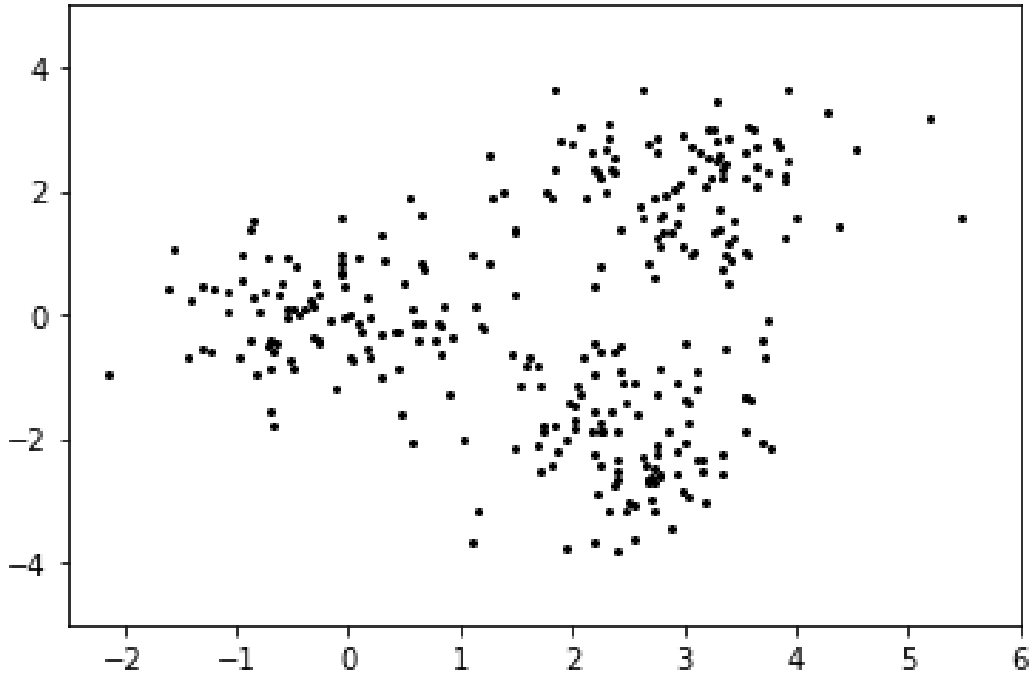
Discovering groups of similar objects

Clustering:

Learning how the data are grouped.

Key clustering methods:

- Hierarchical clustering
- **k-means clustering**
- Competitive learning



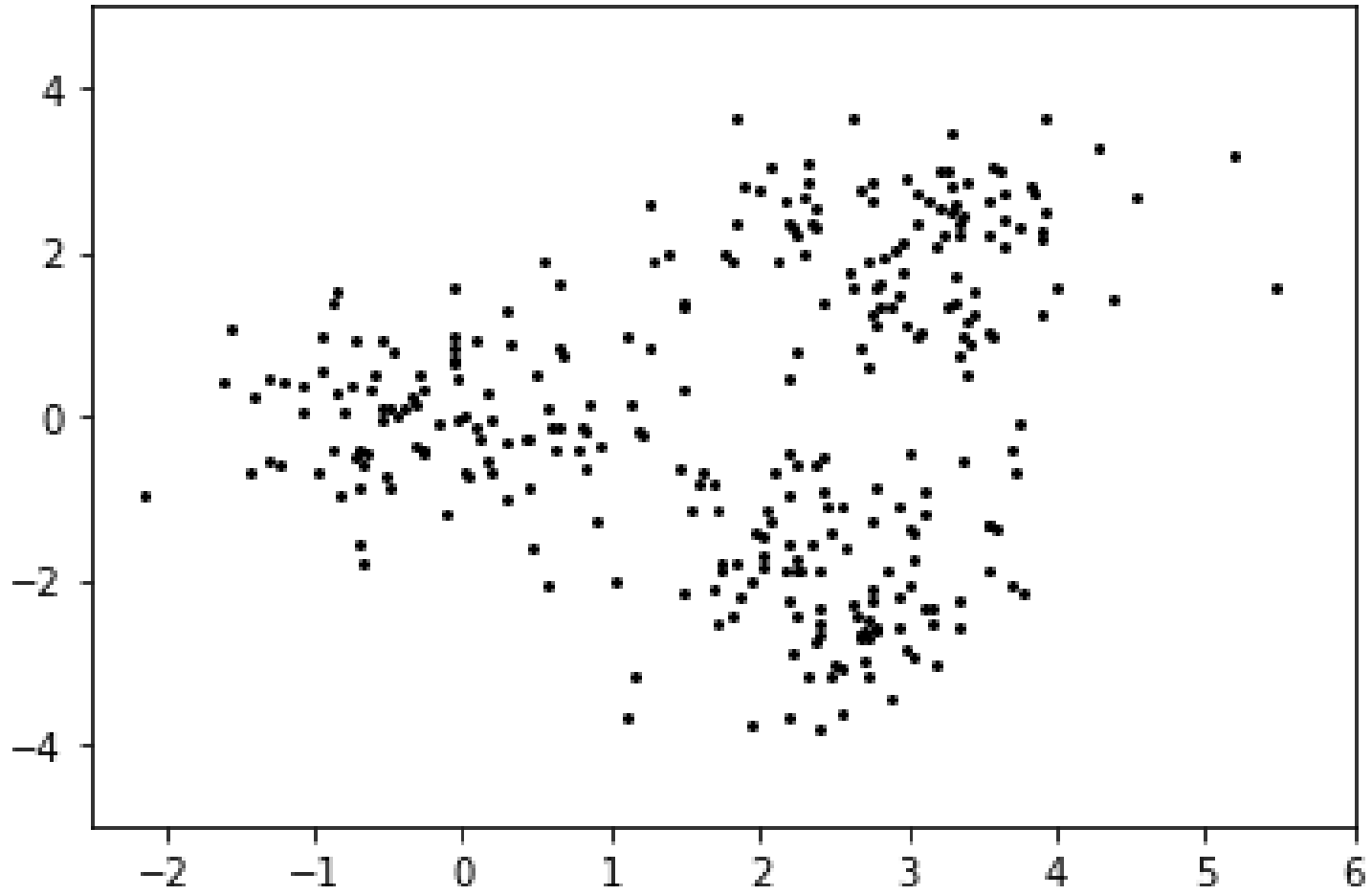
k-means clustering

Suppose we wish to cluster samples $x_1, \ldots, x_n \in \mathbf{R}^p$

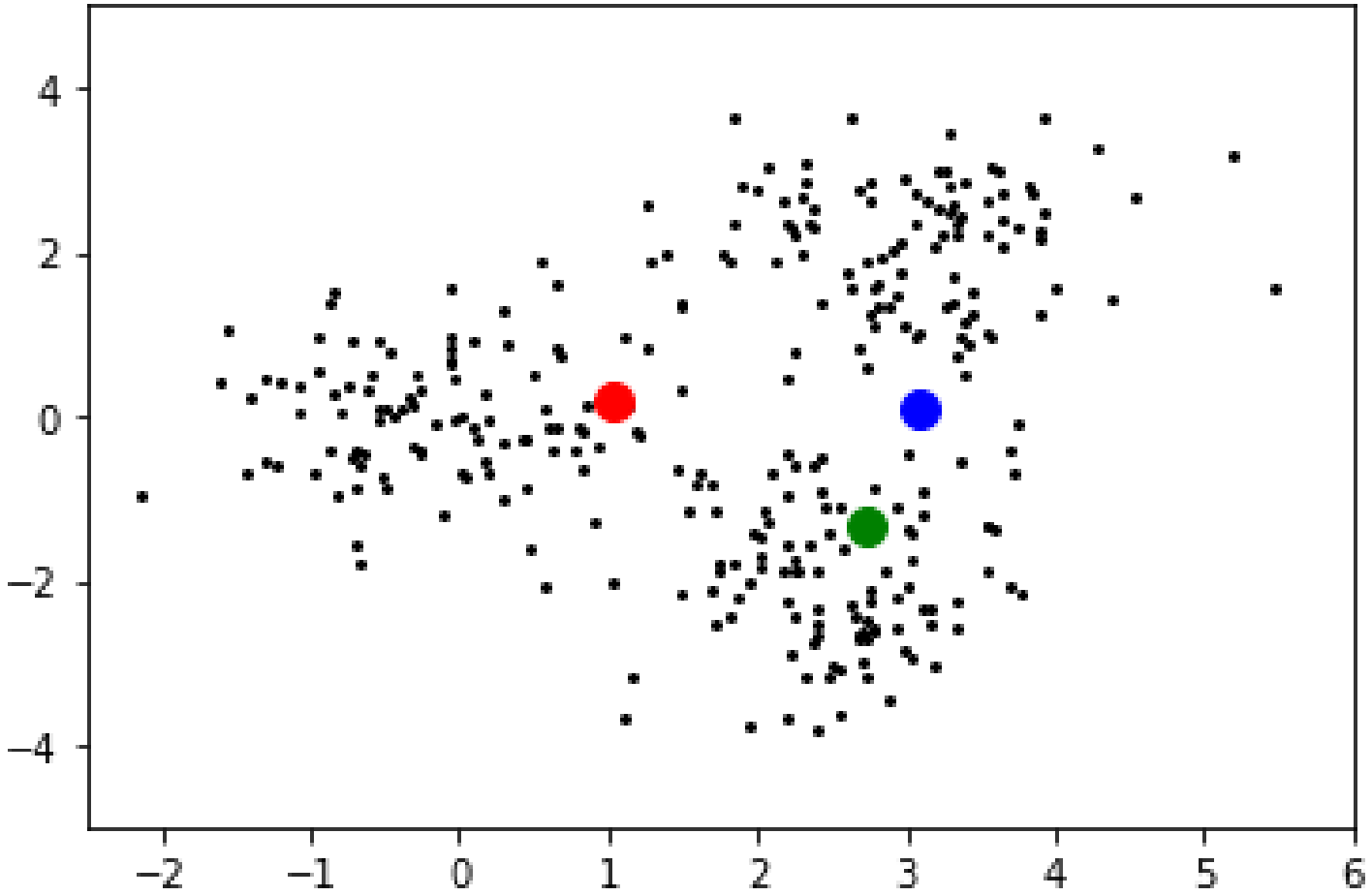
Algorithm:

1) Randomly select k cluster centers $c_1, \ldots, c_k \in \mathbf{R}^p$
2) Assign each sample to the nearest cluster center
3) Set cluster centers to mean of samples in cluster
4) Repeat steps 2-3 until convergence

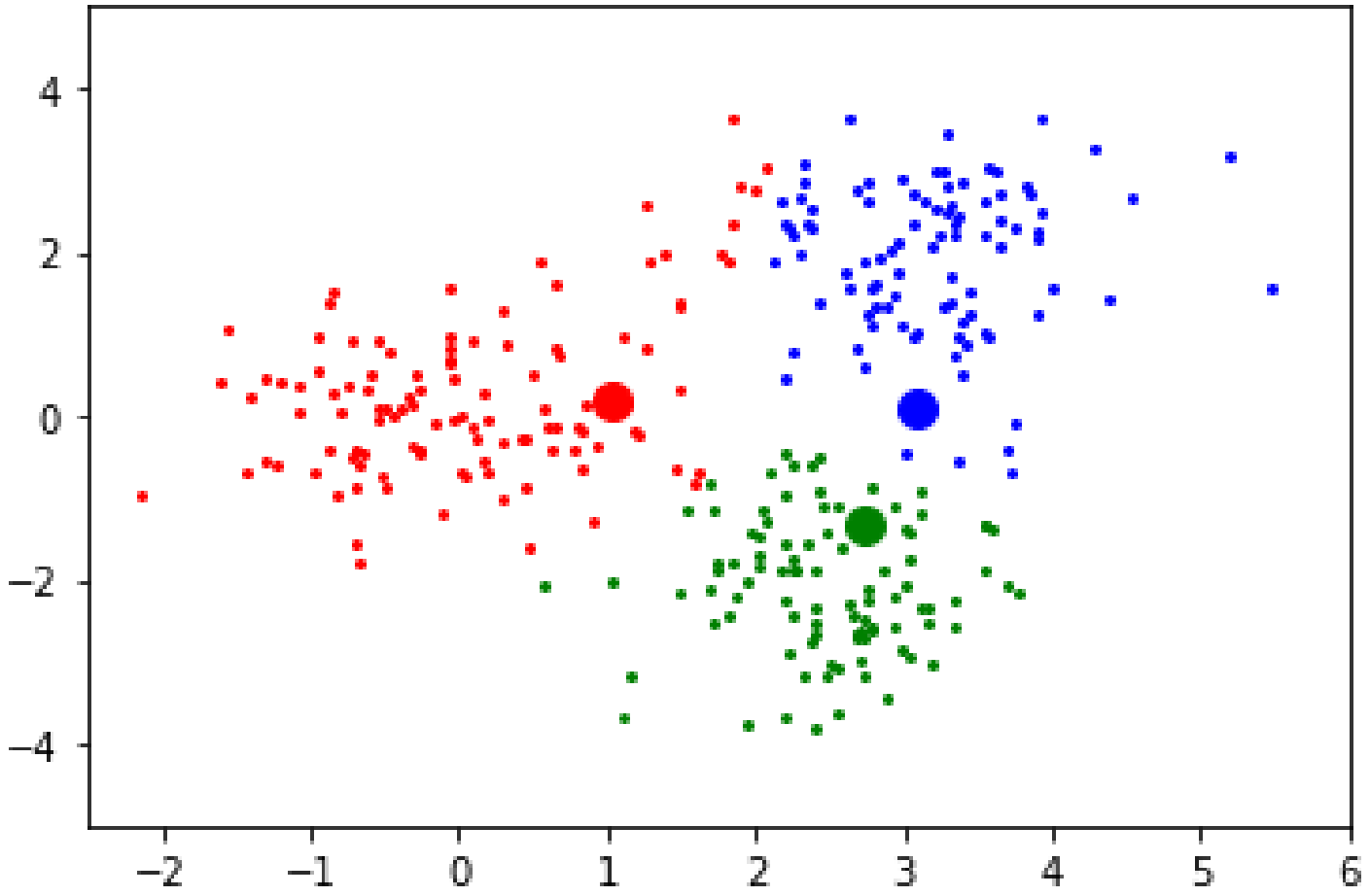
Step 0: the data



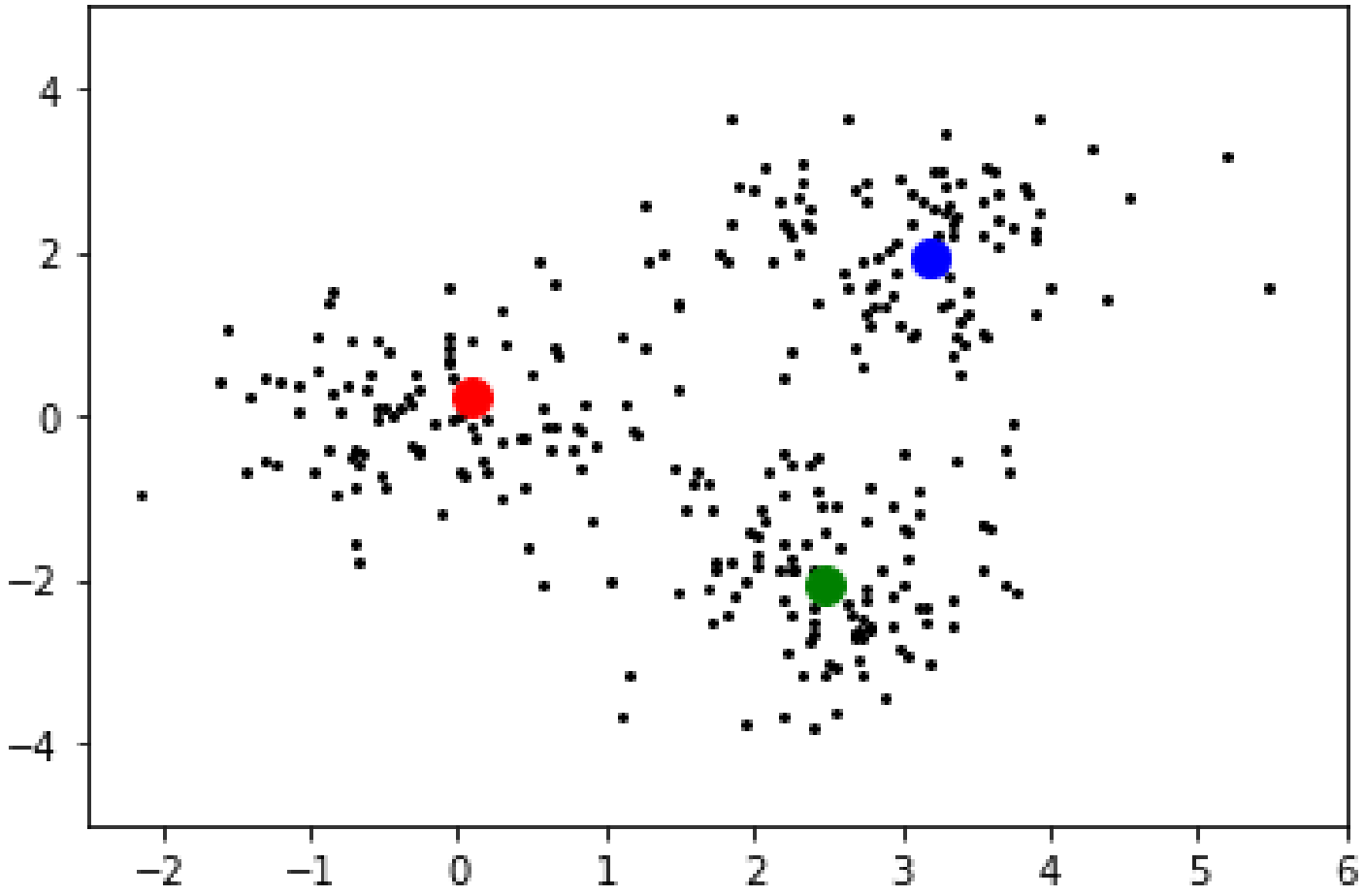
Step 1: randomly select 3 cluster centers



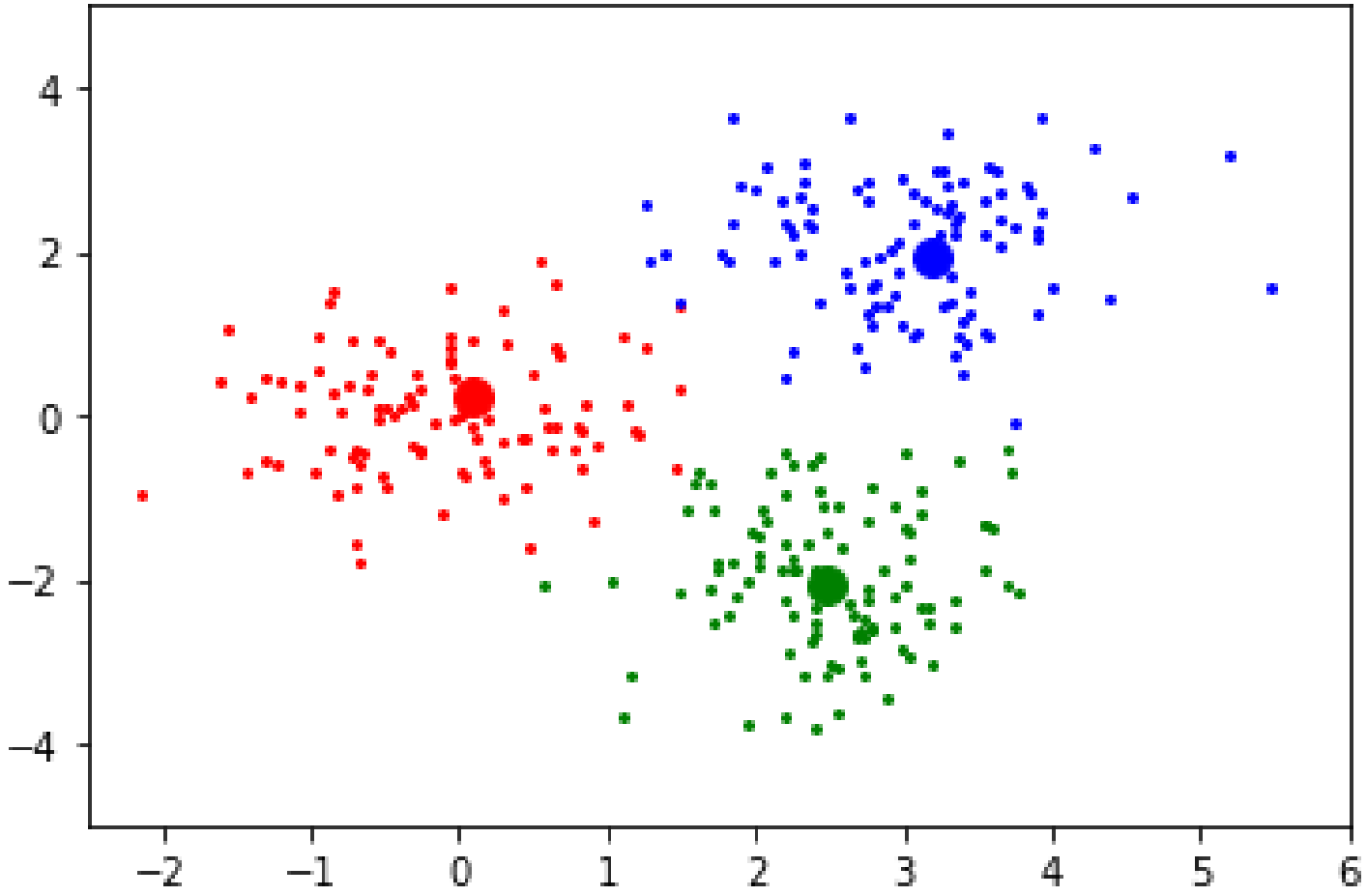
Step 2: assign each sample to the nearest cluster center



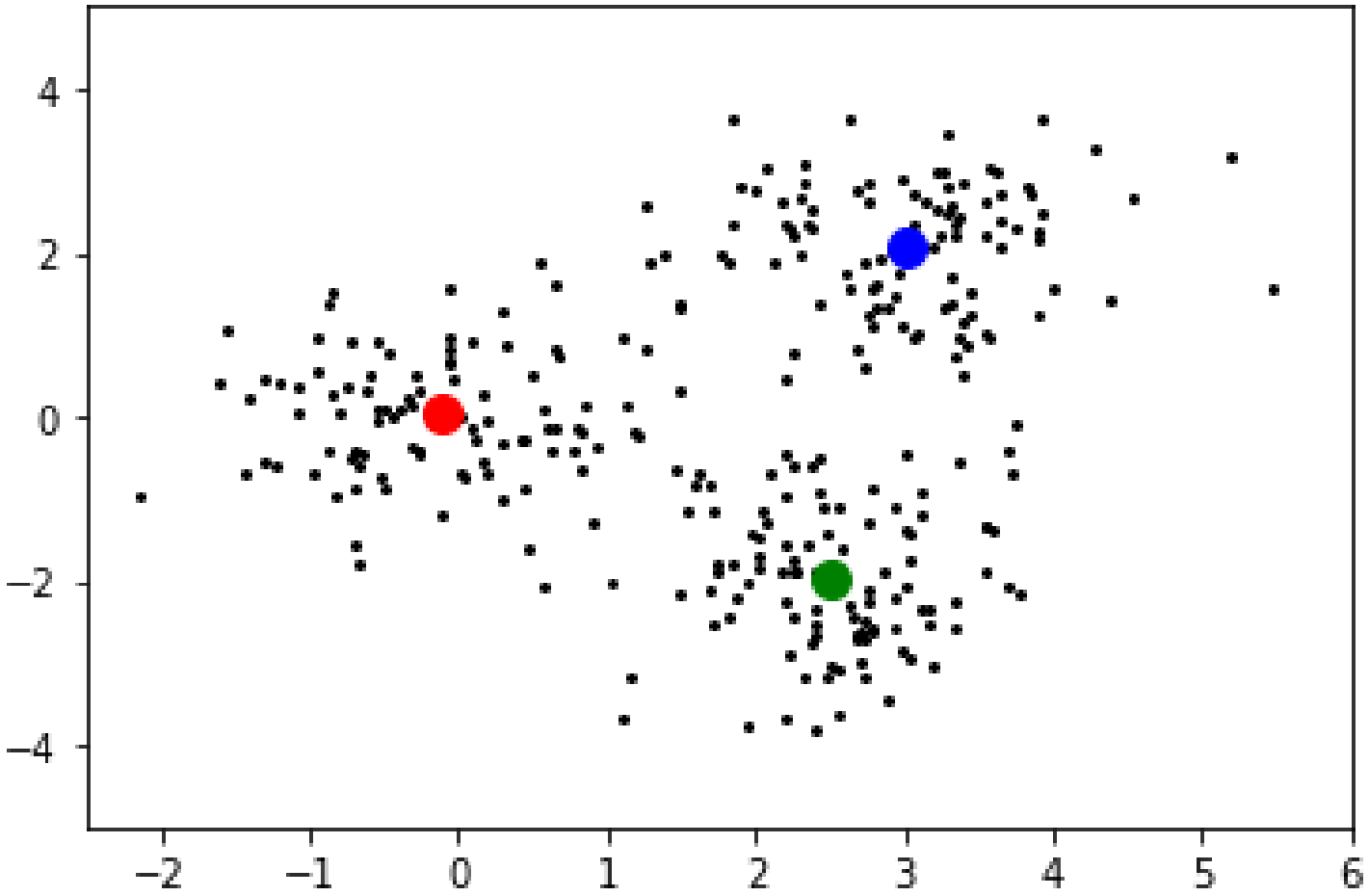
Step 3: update cluster centers



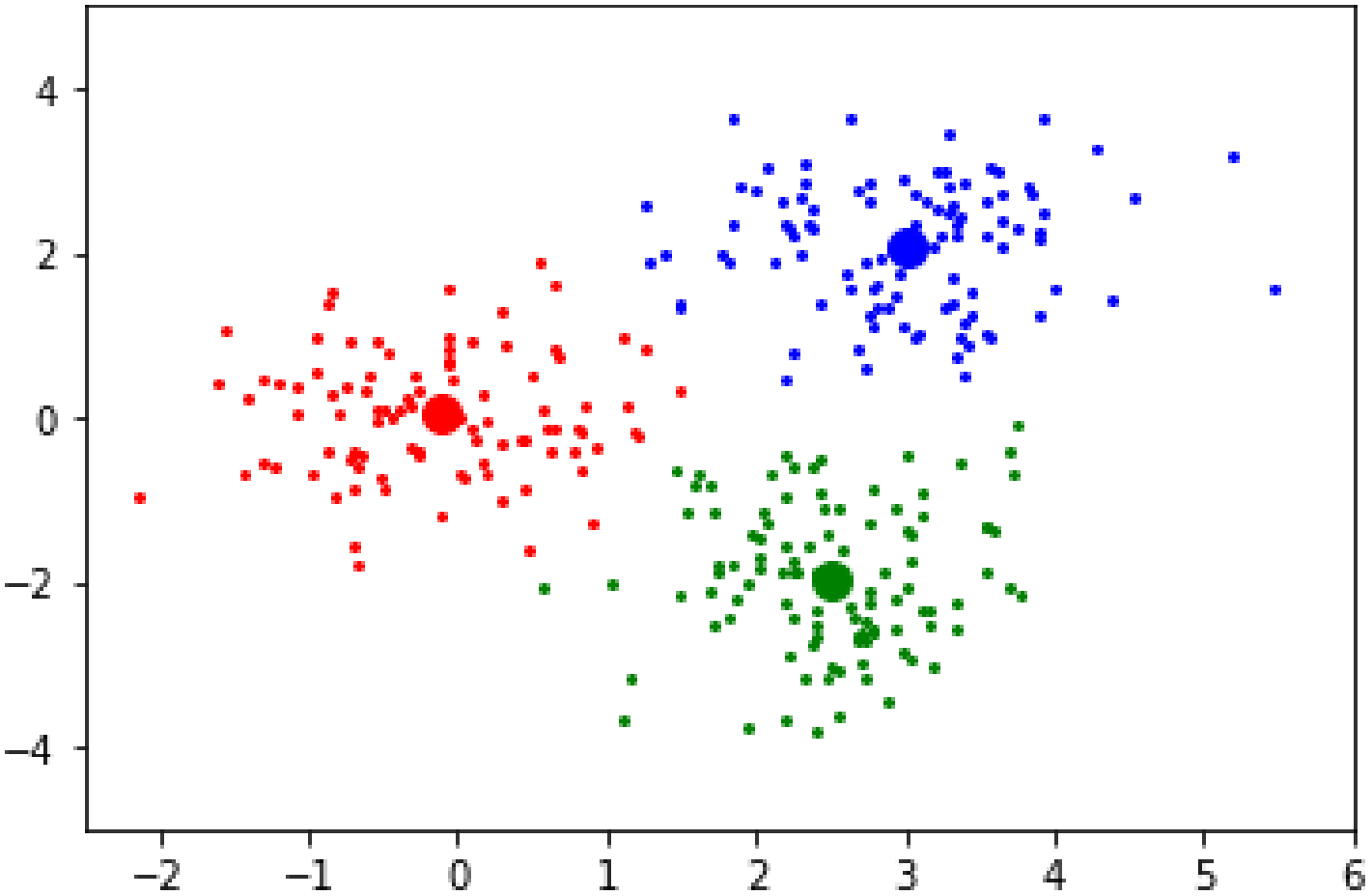
Step 4: assign each sample to the nearest cluster center



Step 5: update cluster centers



Step 6: assign each sample to the nearest cluster center



Interactive demo

http://alekseynp.com/viz/k-means.html

Dimensionality reduction methods

Goal: find Low-Dimensional Representation (LDR) of the data

Principal components analysis (PCA):

- Maximize variance

Multidimensional scaling (MDS)

- Preserve pairwise distances

T-distributed stochastic neighbour embedding (t-SNE)

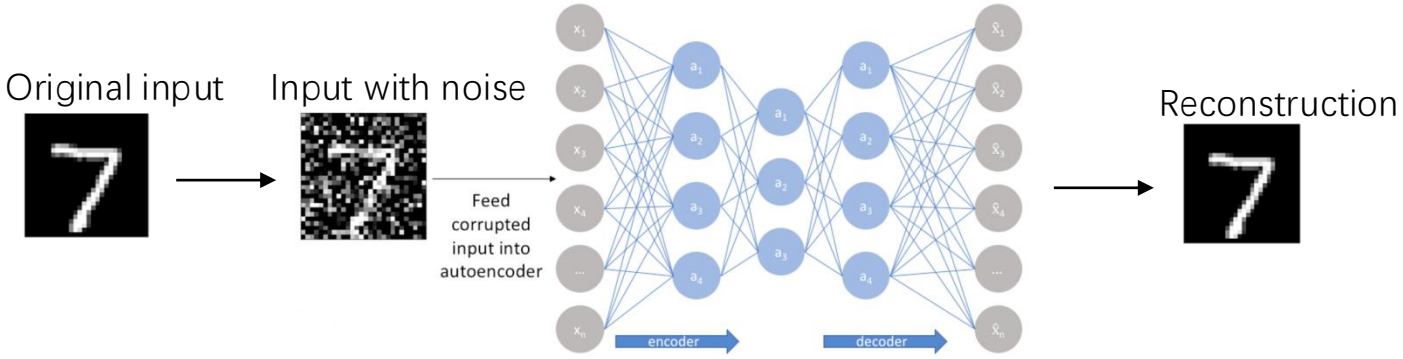
- Preserve neighbor relations

Uniform Manifold Approximation and Projection (UMAP)

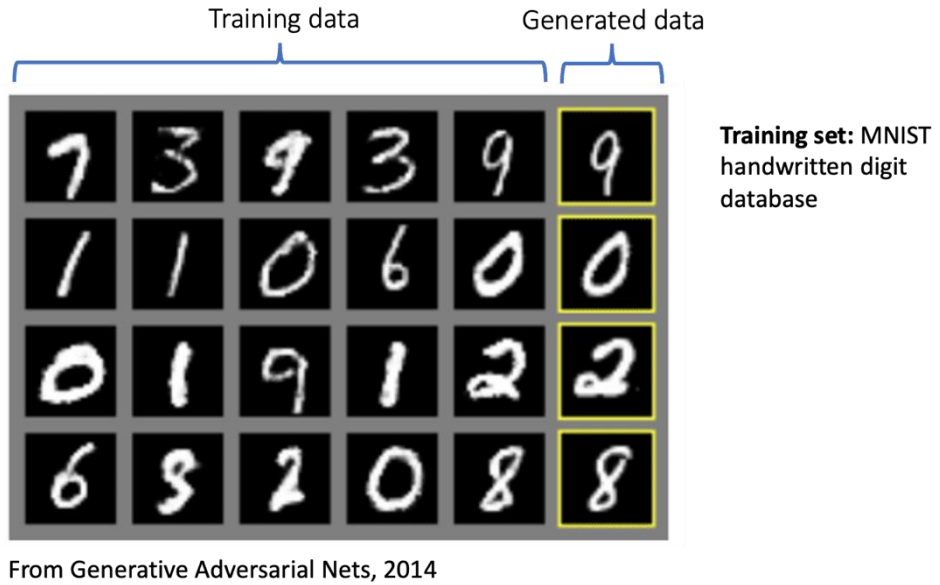
- Preserve topological structure

Generation tasks

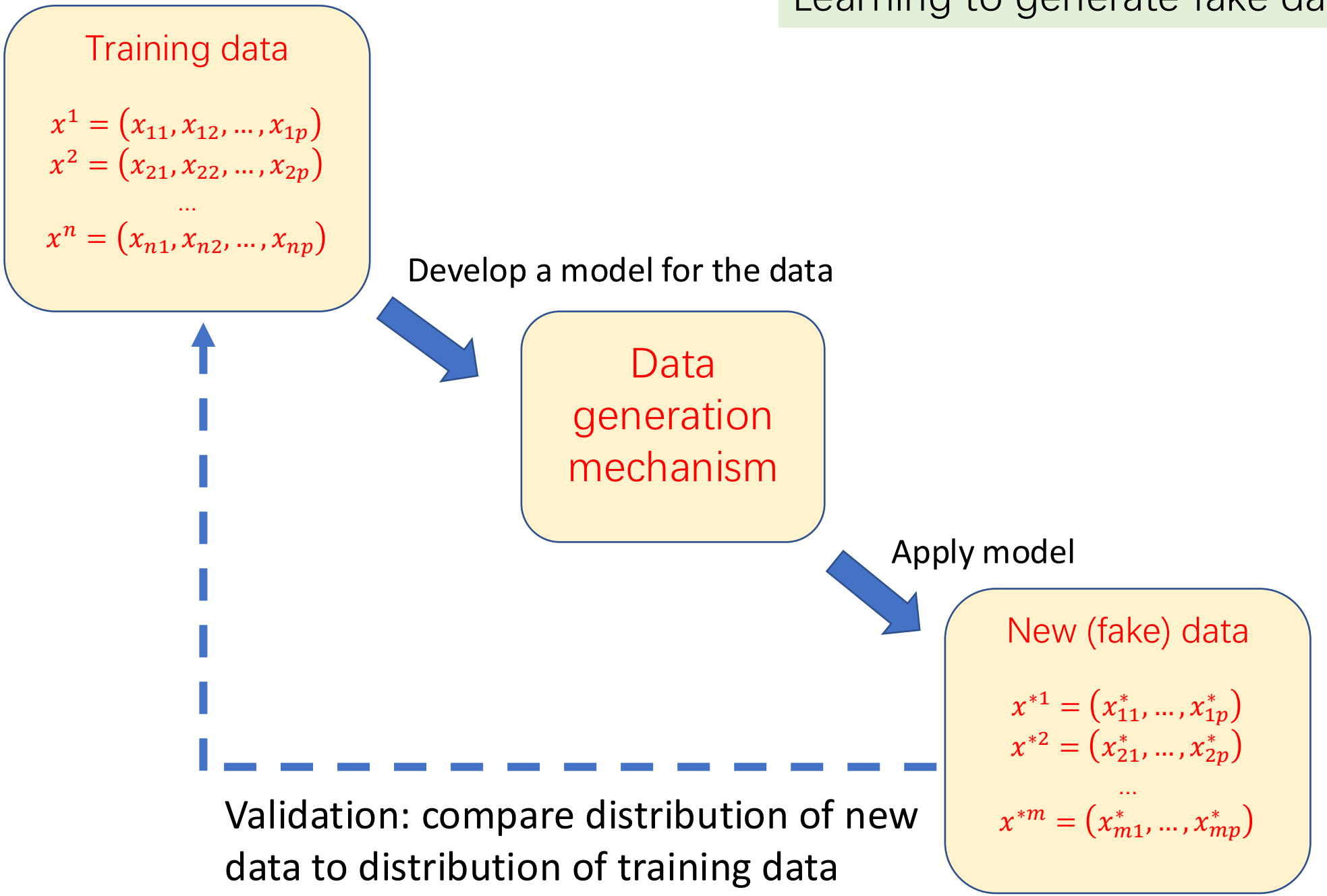
- Autoencoders



- Generative adversarial networks



Learning to generate fake data



Generation tasks

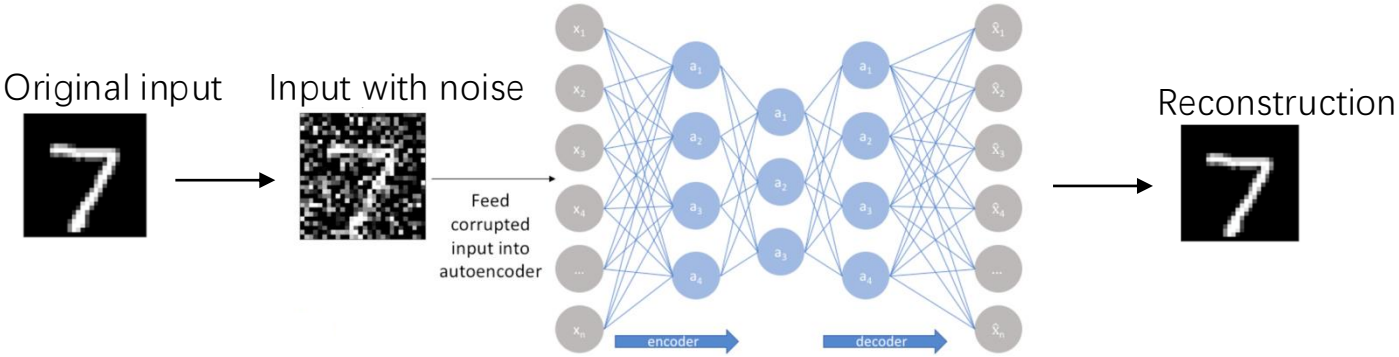
- Why is it useful to generate data (that looks like the input data)?
- Is generation really unsupervised learning?

Why is it useful to generate data?

- Some tasks are inherently generative (in particular for text):
  - Machine translation
  - Summarization
- LLMs are good at a wide variety of text generation tasks
- Generative models for other modalities have become popular too:
  - Image generation
  - Video generation
  - Music generation
  - …

Why is it useful to generate data **that looks like the input data**?

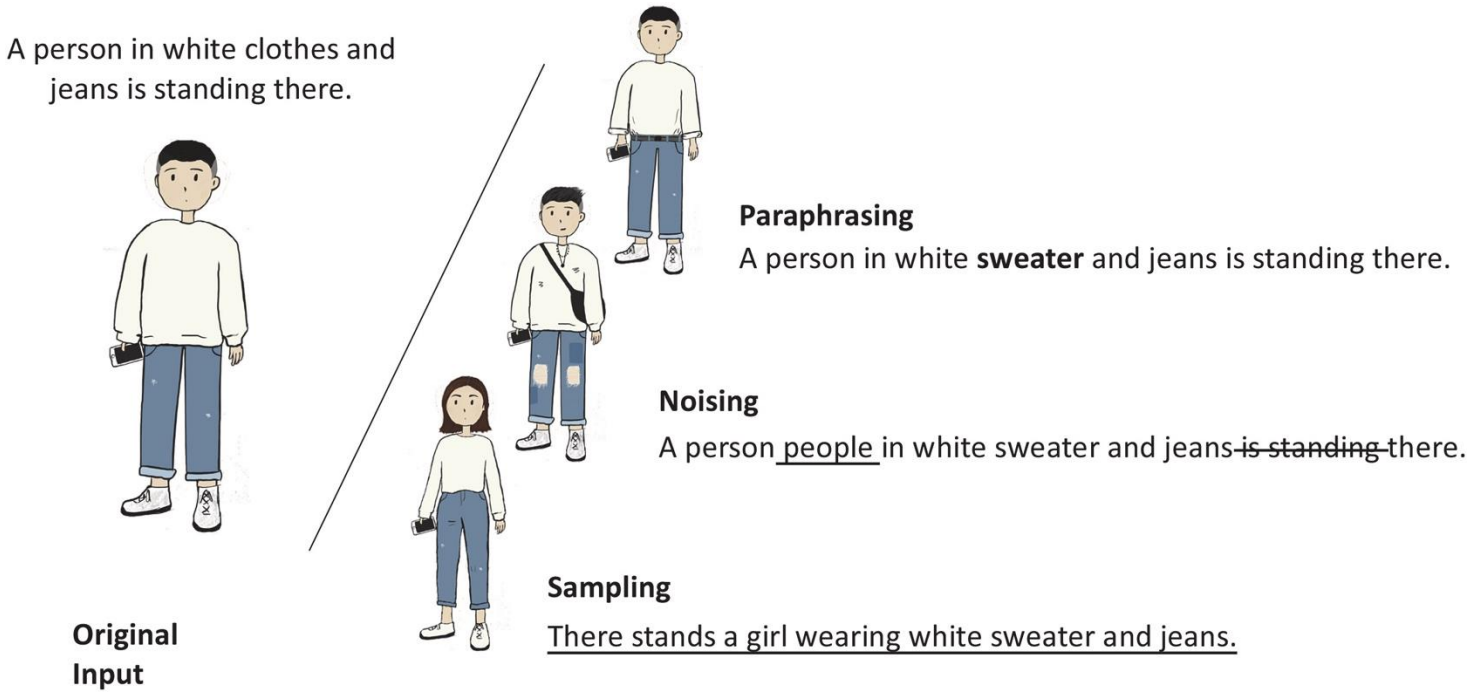
- Denoising is a useful task in itself: **denoising autoencoder**



- Other applications of denoising?

Why is it useful to generate data that looks like the input data?

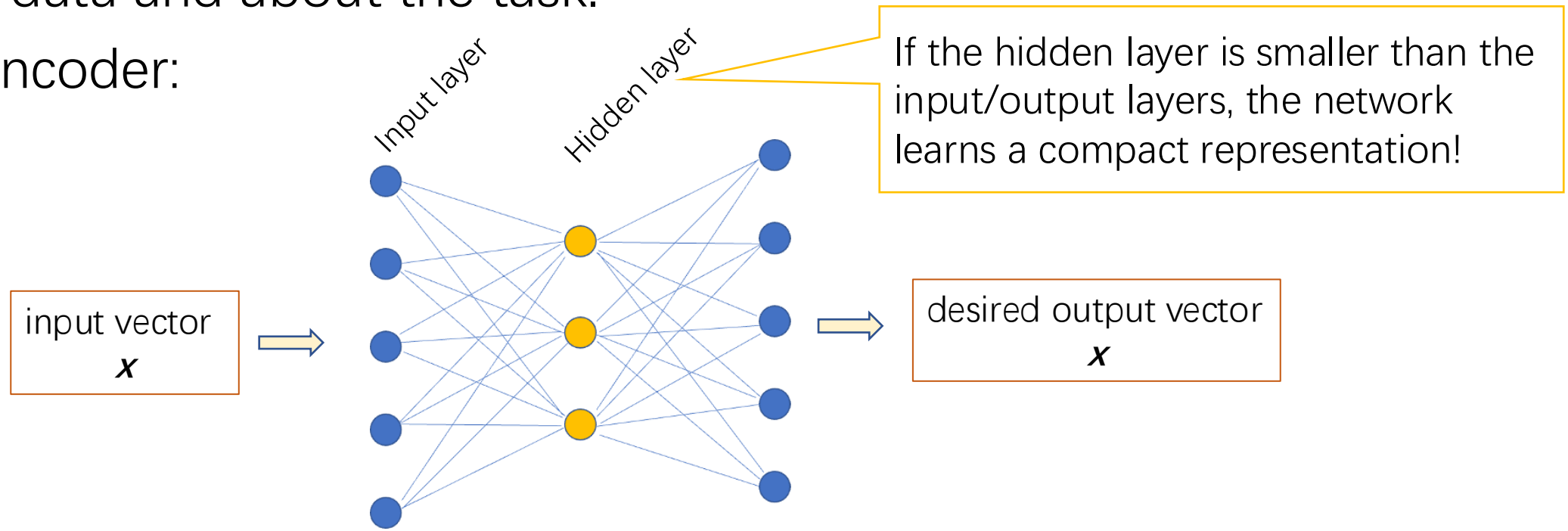
- **Data augmentation**
  - If the amount of labeled training data is insufficient, one can generate additional artificial training examples



https://www.sciencedirect.com/science/article/pii/S2666651022000080

Why is it useful to generate data that looks like the input data?

- For generation, we need to train a model on the input data. We can inspect the model itself to find out things about the structure of the data and about the task.
- Autoencoder:

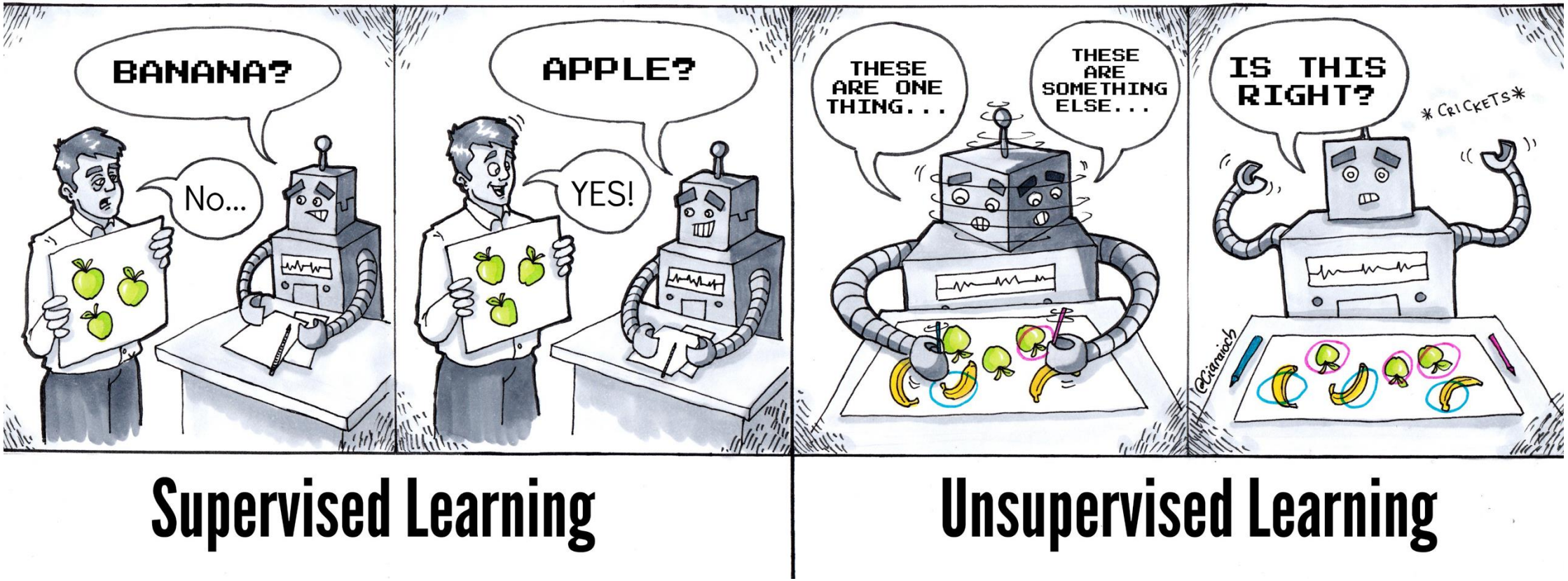


Why is it useful to generate data that looks like the input data?

- Autoencoders can be used to compress data
- However: Dimensionality reduction is not compression!

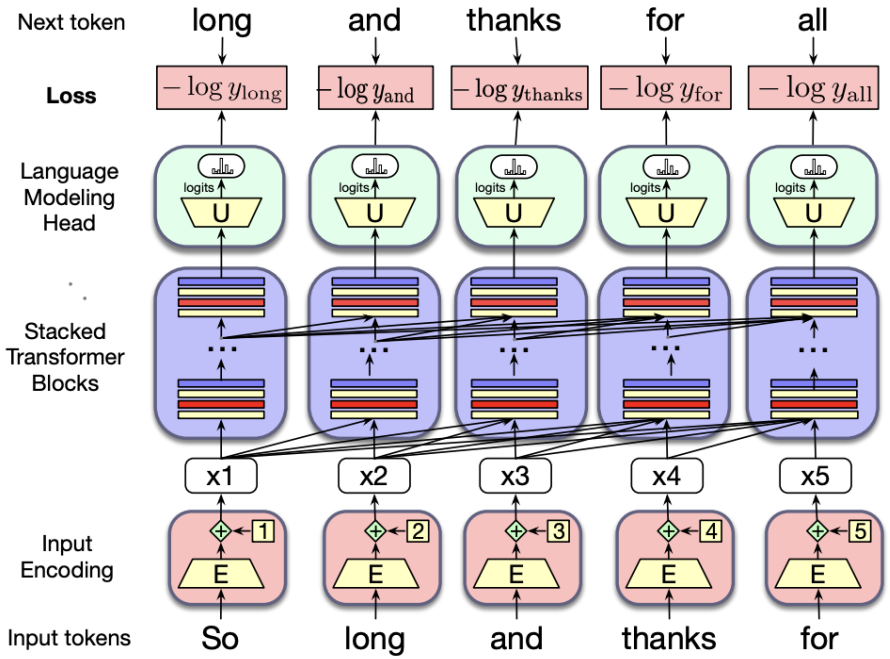
- Compression:
  - Encoded version of data should be **compact** and **possible to decode** to original format
- Dimensionality reduction:
  - Encoded version of data should be **useful for human interpretation** or as input to other analysis steps

Is generation really unsupervised learning?



Is generation really unsupervised learning?

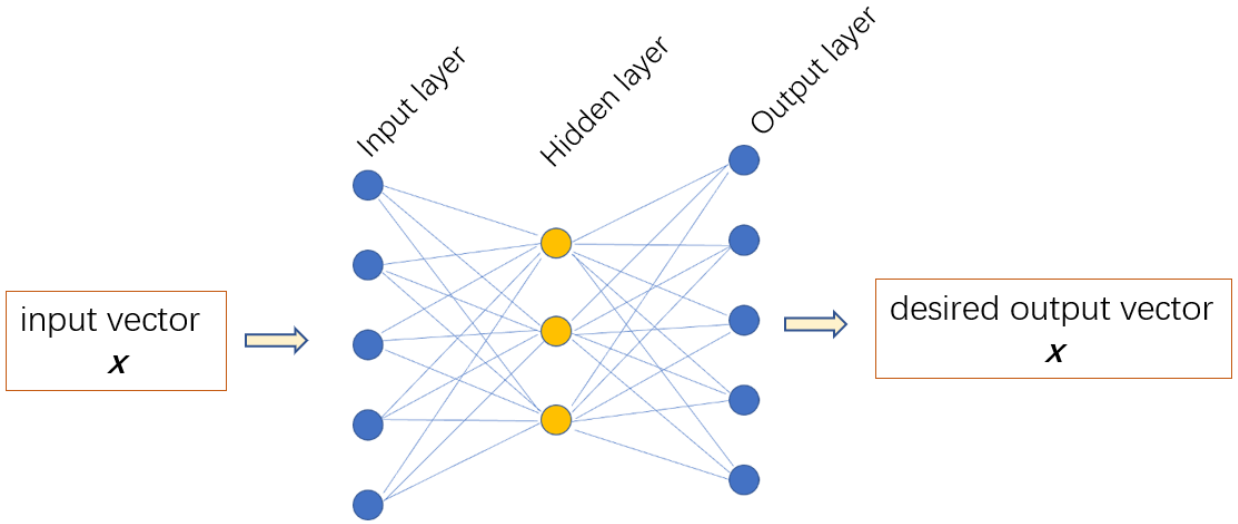
- Language modeling = next word prediction
- The ”labels” are the next words
- If we train on full sentences, the labels are already there
- No need for any manual annotation
- Is this supervised or unsupervised learning?



Are autoencoders UL or SL?

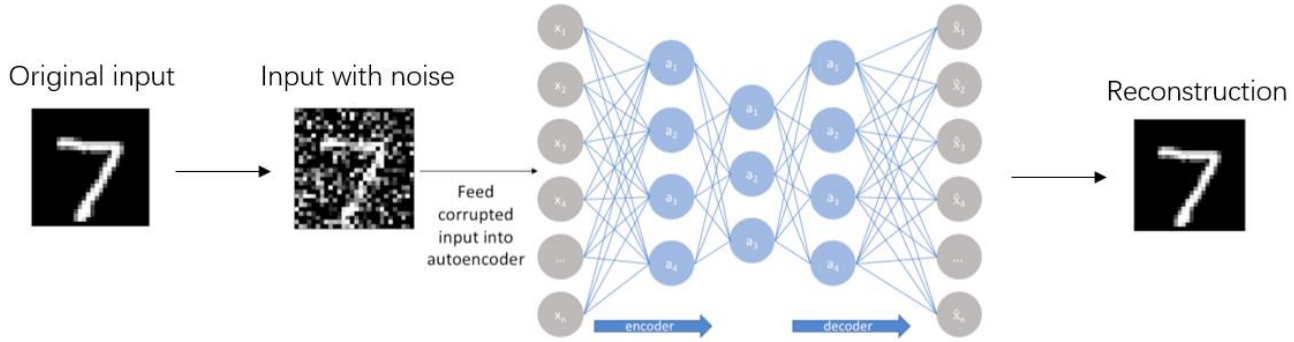
- It has sets of inputs (x) and targets (x)
  - Like SL, but inputs and targets happen to be the same
- It doesn't need any labels – the labels come from the data themselves
  - Like UL

An autoencoder is a neural network that learns to produce an output similar to the input:



Are denoising autoencoders UL or SL?

- It has sets of inputs (x) and targets (y)
  - Like SL – the inputs and targets are not the same here
- We don't need labeled data, the labels are derived directly from the data
  - Like UL



Do generative models really count as unsupervised learning?

- The reality is more complex than we have been pretending
- In practice, ML is **not always supervised or unsupervised**
- We often **combine elements of SL and UL**
- Some types of learning are somewhere between SL and UL
  - In particular, for generation tasks

Do generative models really count as unsupervised learning?

- Different people use these terms slightly differently
- Perhaps because of this, many have started to refer to this kind of learning as **self-supervised learning**.

I now call it “self-supervised learning”, because “unsupervised” is both a loaded and confusing term. … Self-supervised learning uses **way more supervisory signals than supervised learning, and enormously more than reinforcement learning**. That’s why calling it “unsupervised” is totally misleading.
by Yann LeCun (2019. 04. 30)
AI602: Recent Advances in Deep Learning: Lecture 07



What is self-supervised learning?

Self-supervised learning is a machine learning technique that uses unsupervised learning for tasks that conventionally require supervised learning. Rather than relying on labeled datasets for supervisory signals, self-supervised models generate implicit labels from unstructured data.

Source: https://www.ibm.com/topics/self-supervised-learning

Self-supervised learning - Examples



Text: Given a long text sequence as input – predict the most likely next word



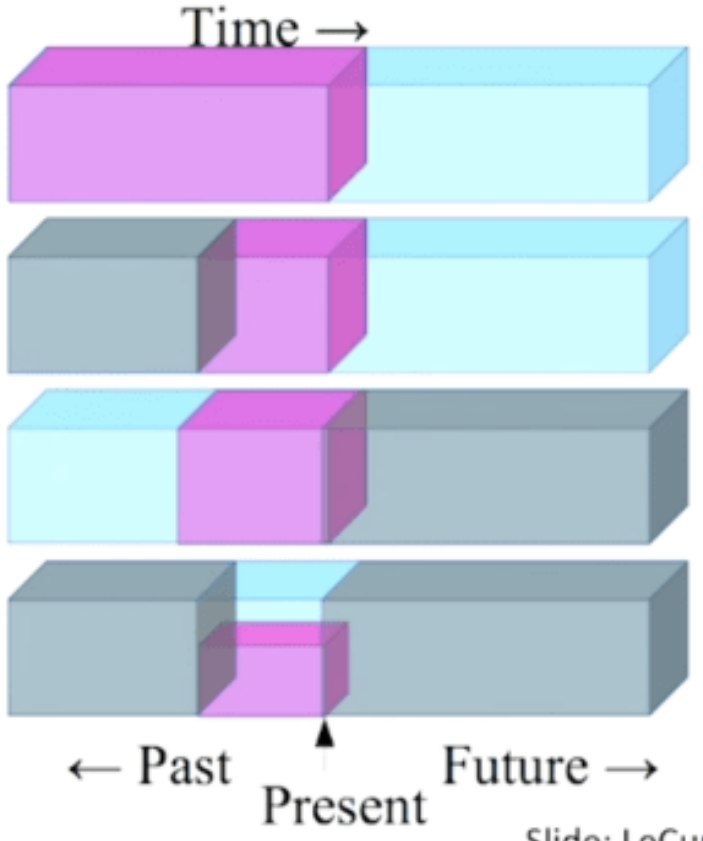
Images: Given a part of an image, predict what is in another part



Video: Given 5 seconds of video, predict the next frame

Self-supervised learning
Example: Learning from videos

- Predict any part of the input from any other part.
- Predict the future from the past.
- Predict the future from the recent past.
- Predict the past from the present.
- Predict the top from the bottom.
- Predict the occluded from the visible
- Pretend there is a part of the input you don't know and predict that.



Source: https://www.youtube.com/watch?v=7I0Qt7GALVk&ab_channel=EPFL

Self-supervised learning is behind much of the recent progress in Generative AI

- Generating images
  - https://thispersondoesnotexist.com/
- Generating video
  - 2023: https://youtu.be/Itbc12qXr30
  - 2024: https://youtu.be/2dfNHDGH1Co
- Generating text
- Next step: Generating an understanding of the environment for robots