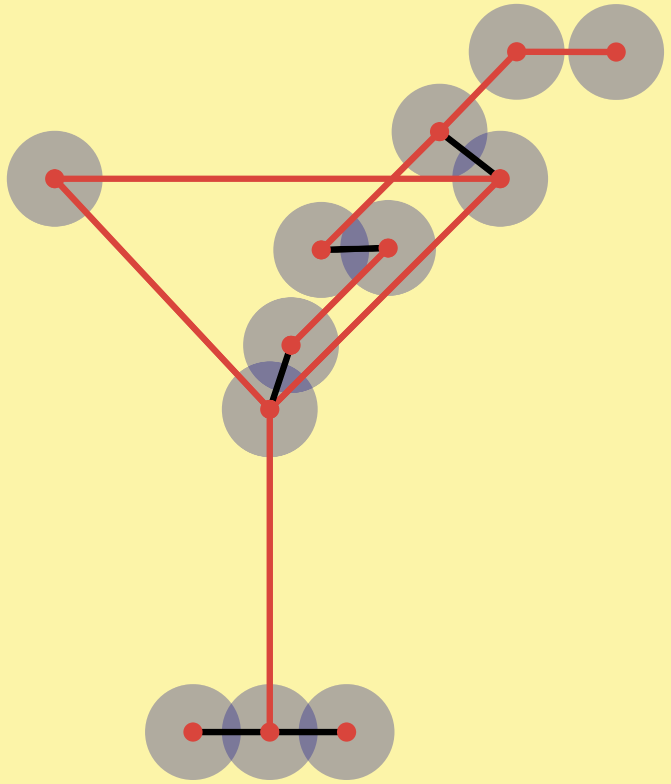
ICML | 2019
Long Beach

Connectivity-Optimized Representation Learning via Persistent Homology

Christoph D. Hofer, Roland Kwitt — University of Salzburg

Mandar Dixit — Microsoft

Marc Niethammer — UNC Chapel Hill



Unsupervised representation learning

Q: What makes a **good** representation?

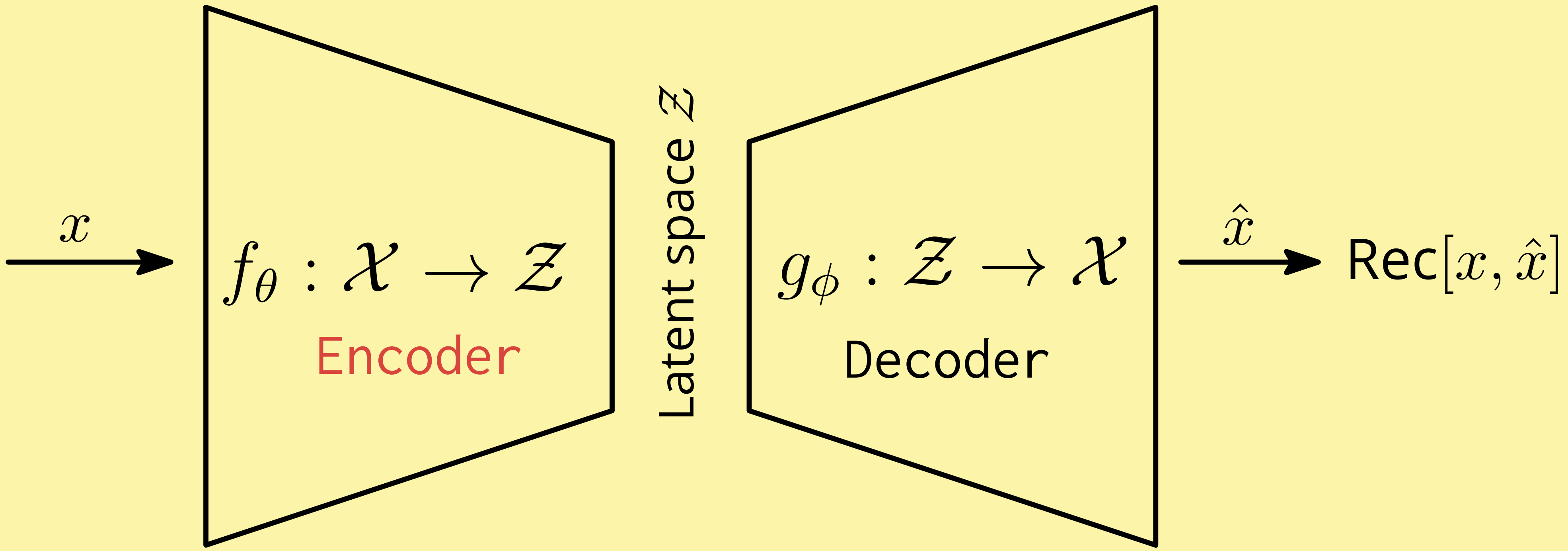
- Ability to reconstruct ($\rightarrow$ prevalance of autoencoders)
- Robust to pertubations of the input
- Useful for downstream tasks (e.g., clustering, or classification)
- etc.

Unsupervised representation learning

Q: What makes a **good** representation?

- Ability to reconstruct ($\rightarrow$ prevalance of autoencoders)
- Robust to pertubations of the input
- Useful for downstream tasks (e.g., clustering, or classification)
- etc.

Common idea: Control (/or enforce) properties of (/on) the latent representations in $\mathcal{Z}$.



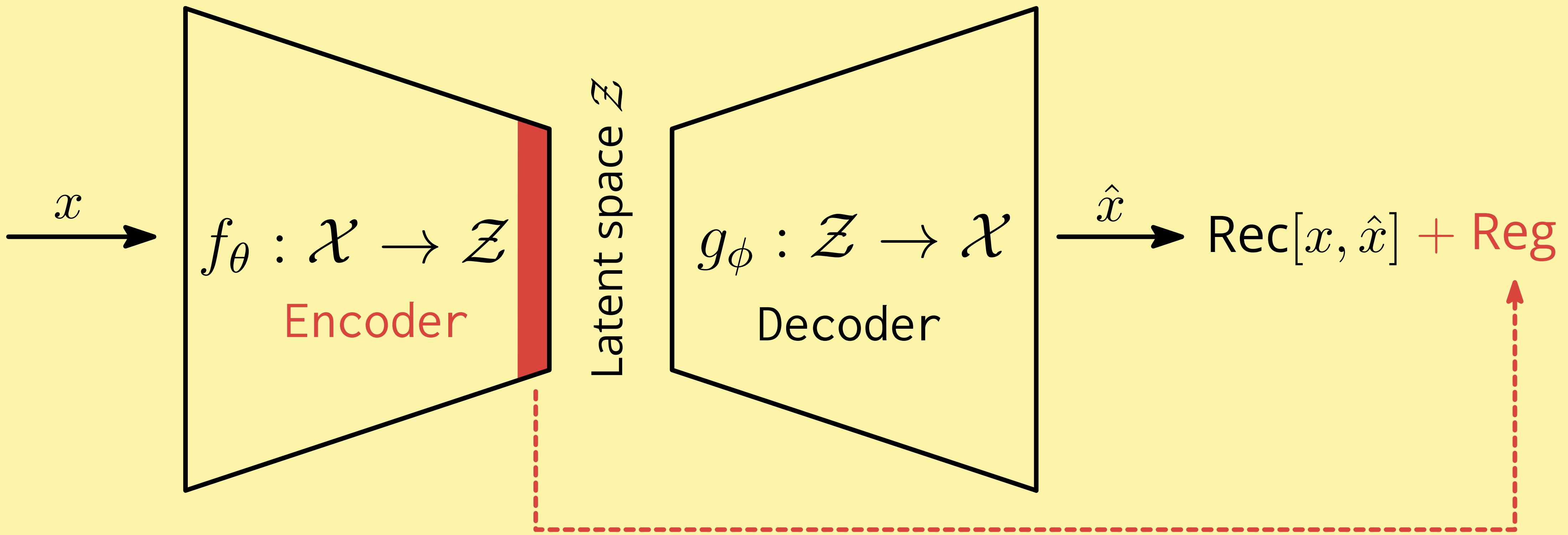
Unsupervised representation learning

Q: What makes a **good** representation?

- Ability to reconstruct ($\rightarrow$ prevalance of autoencoders)
- Robust to pertubations of the input
- Useful for downstream tasks (e.g., clustering, or classification)
- etc.

Contractive AE's [Rifai et al., ICML '11]

Common idea: Control (/or enforce) properties of (/on) the latent representations in $\mathcal{Z}$.



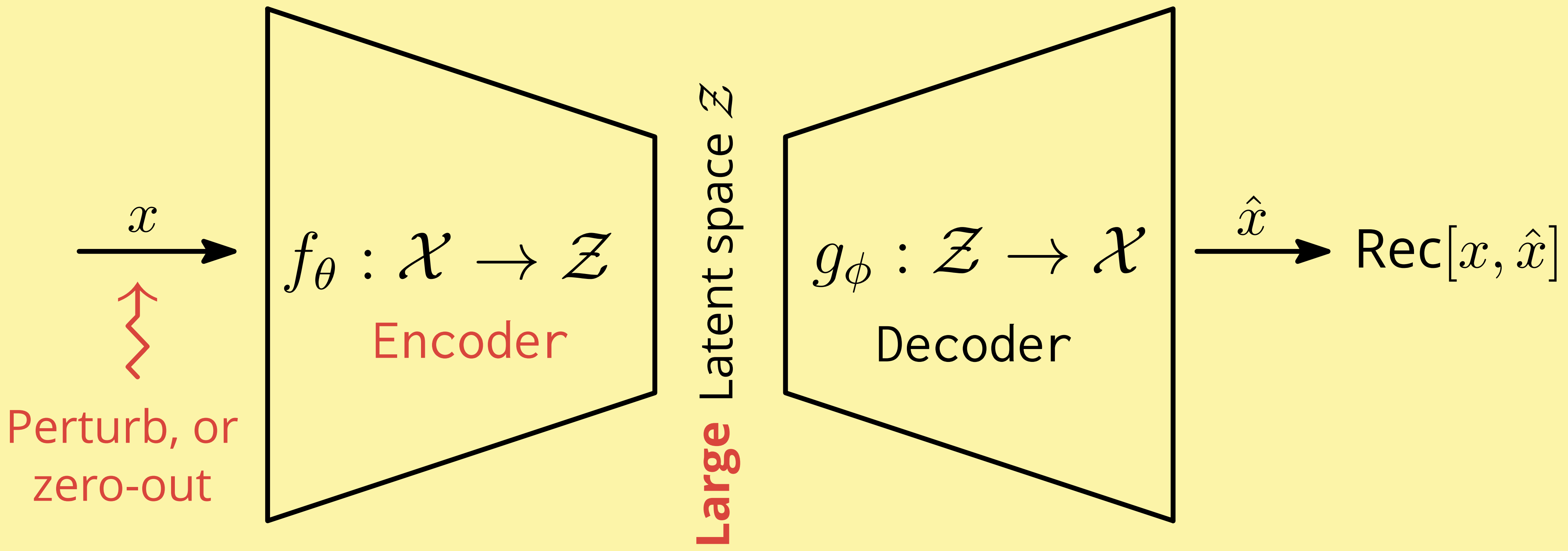
Unsupervised representation learning

Q: What makes a **good** representation?

- Ability to reconstruct ($\rightarrow$ prevalance of autoencoders)
- Robust to pertubations of the input
- Useful for downstream tasks (e.g., clustering, or classification)
- etc.

Denoising AE's [Vincent et al., JMLR '10]

Common idea: Control (/or enforce) properties of (/on) the latent representations in $\mathcal{Z}$.



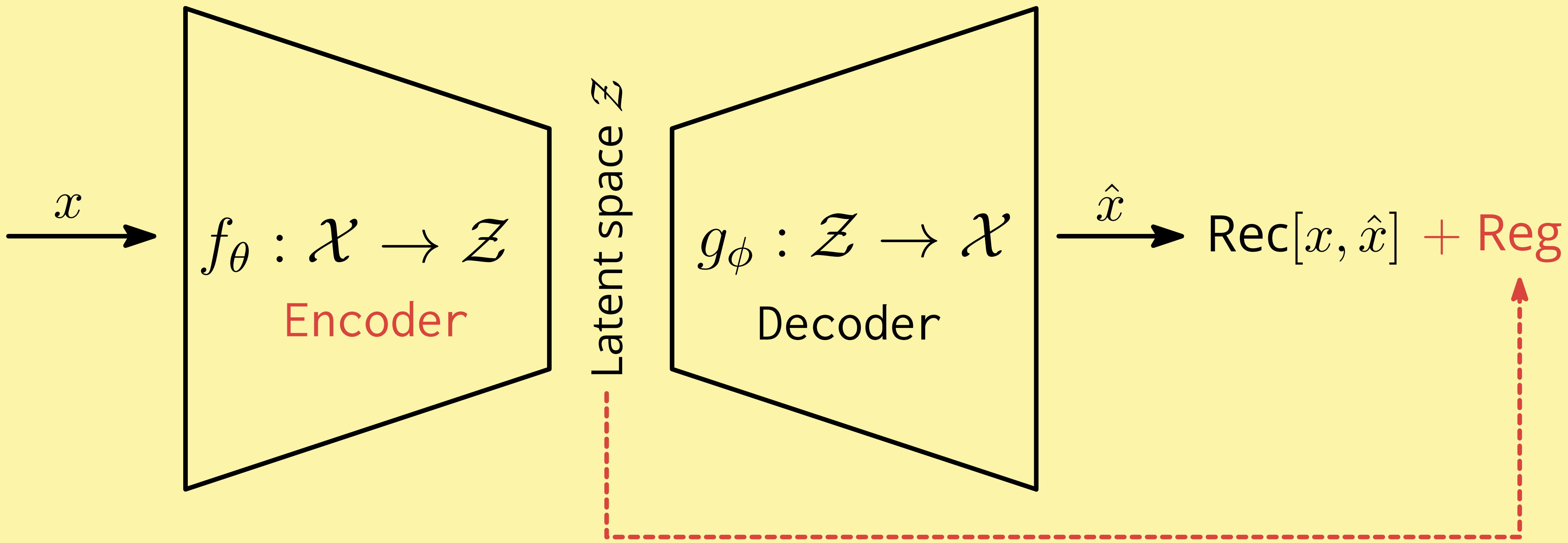
Unsupervised representation learning

Q: What makes a **good** representation?

- Ability to reconstruct ($\rightarrow$ prevalance of autoencoders)
- Robust to pertubations of the input
- Useful for downstream tasks (e.g., clustering, or classification)
- etc.

Sparse AE's [Makhzani & Frey, ICLR '14]

Common idea: Control (/or enforce) properties of (/on) the latent representations in $\mathcal{Z}$.



Unsupervised representation learning

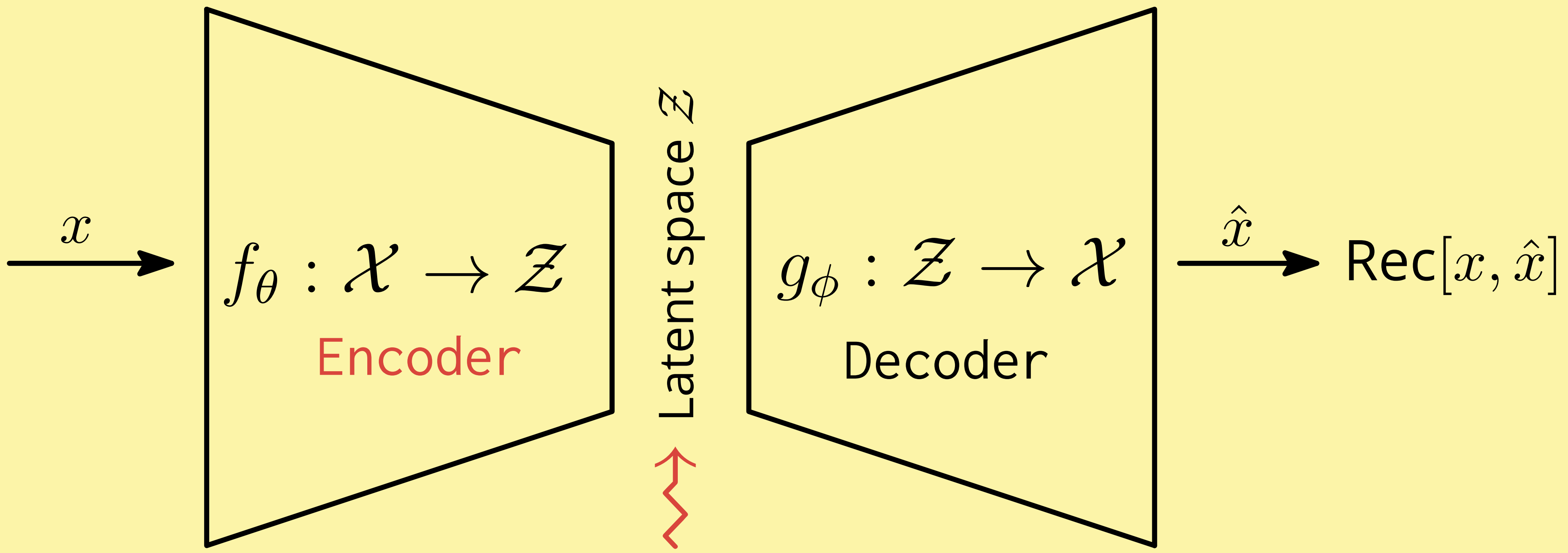
Q: What makes a **good** representation?

- Ability to reconstruct ($\to$ prevalance of autoencoders)
- Robust to pertubations of the input
- Useful for downstream tasks (e.g., clustering, or classification)
- etc.

Adversarial AE's [Makhzani et al., ICLR '16]

(by far not exhaustive)

Common idea: Control (/or enforce) properties of (/on) the latent representations in $\mathcal{Z}$.



Enforce distributional properties through **adversarial** training

Motivating (toy) example

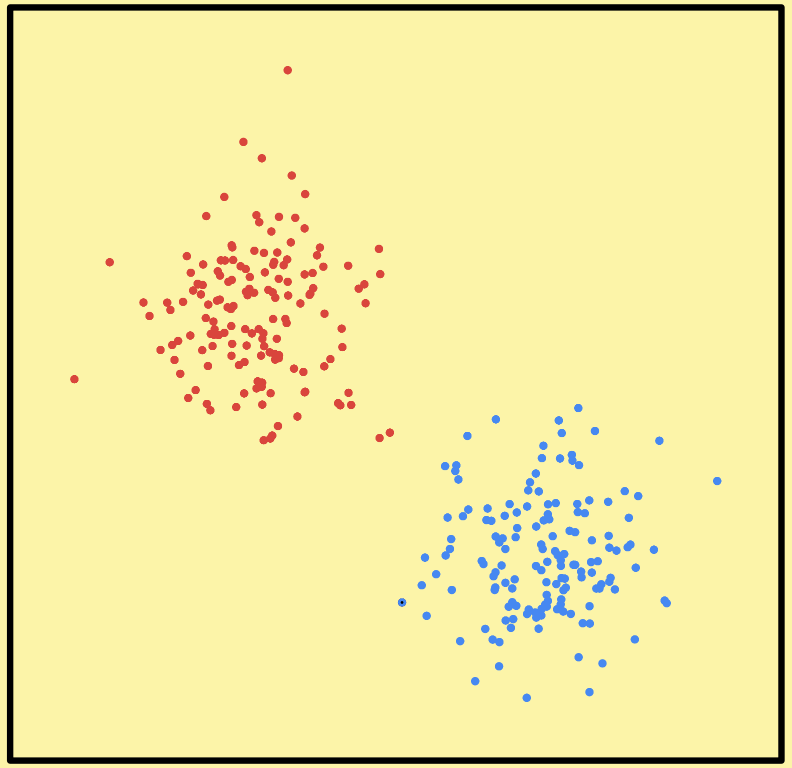
We aim to control properties of the latent space, but from a **topological point of view**!

Motivating (toy) example

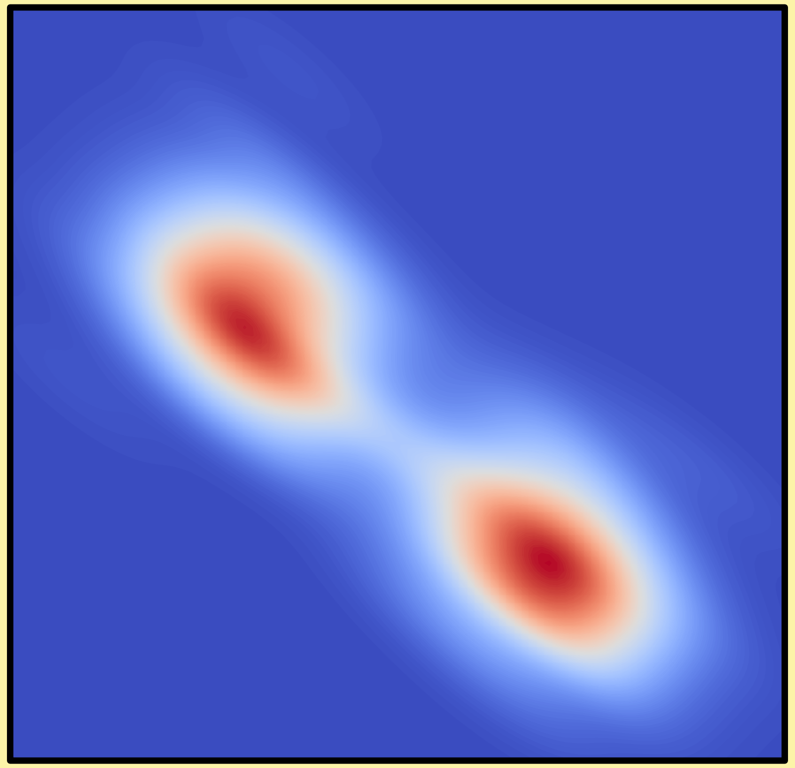
We aim to control properties of the latent space, but from a **topological point of view**!

Assume, we want to do **Kernel Density Estimation (KDE)** in the latent space $\mathcal{Z}$.

Data (z_i)



Gaussian KDE



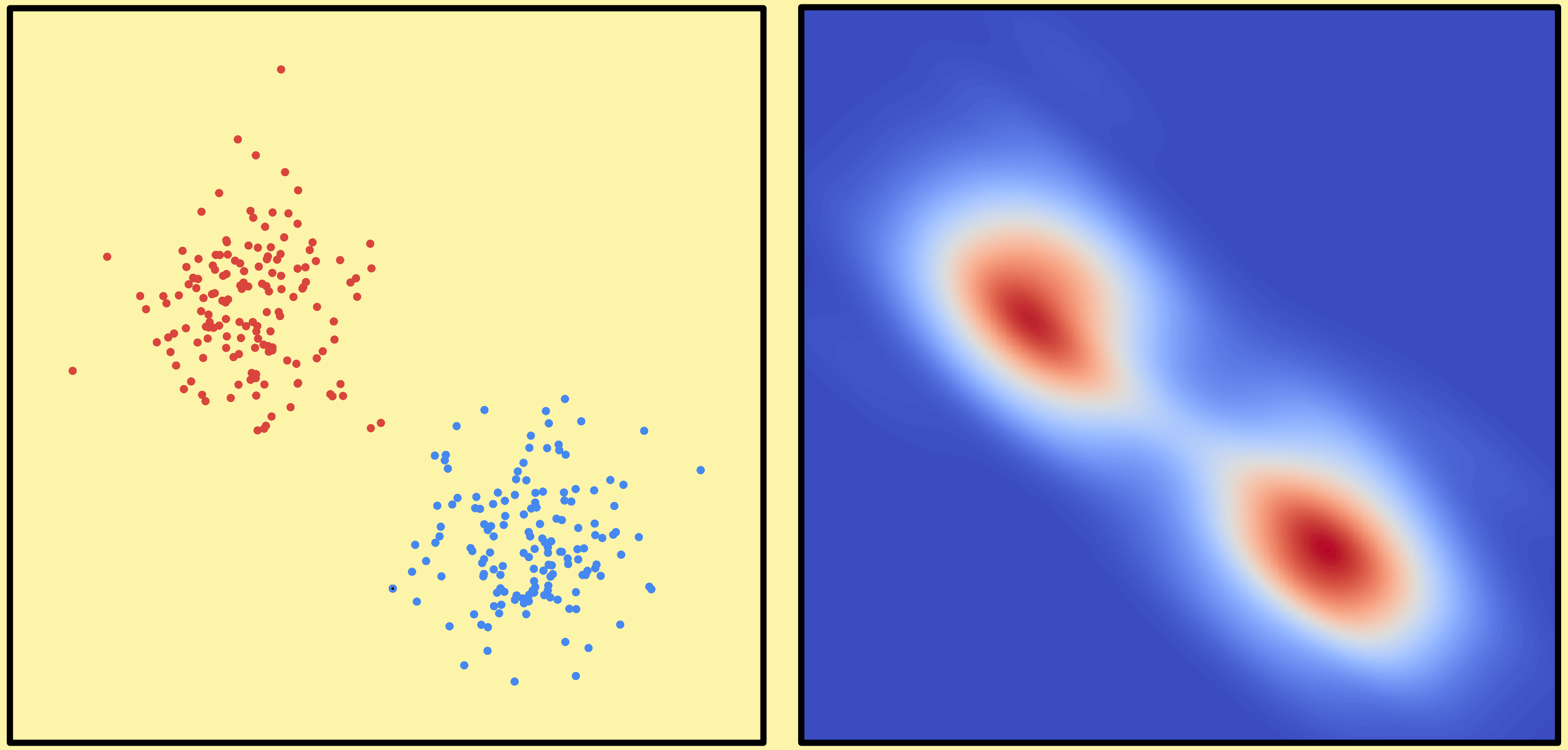
Bandwidth selection: Scott's rule [Scott, 1992]

Motivating (toy) example

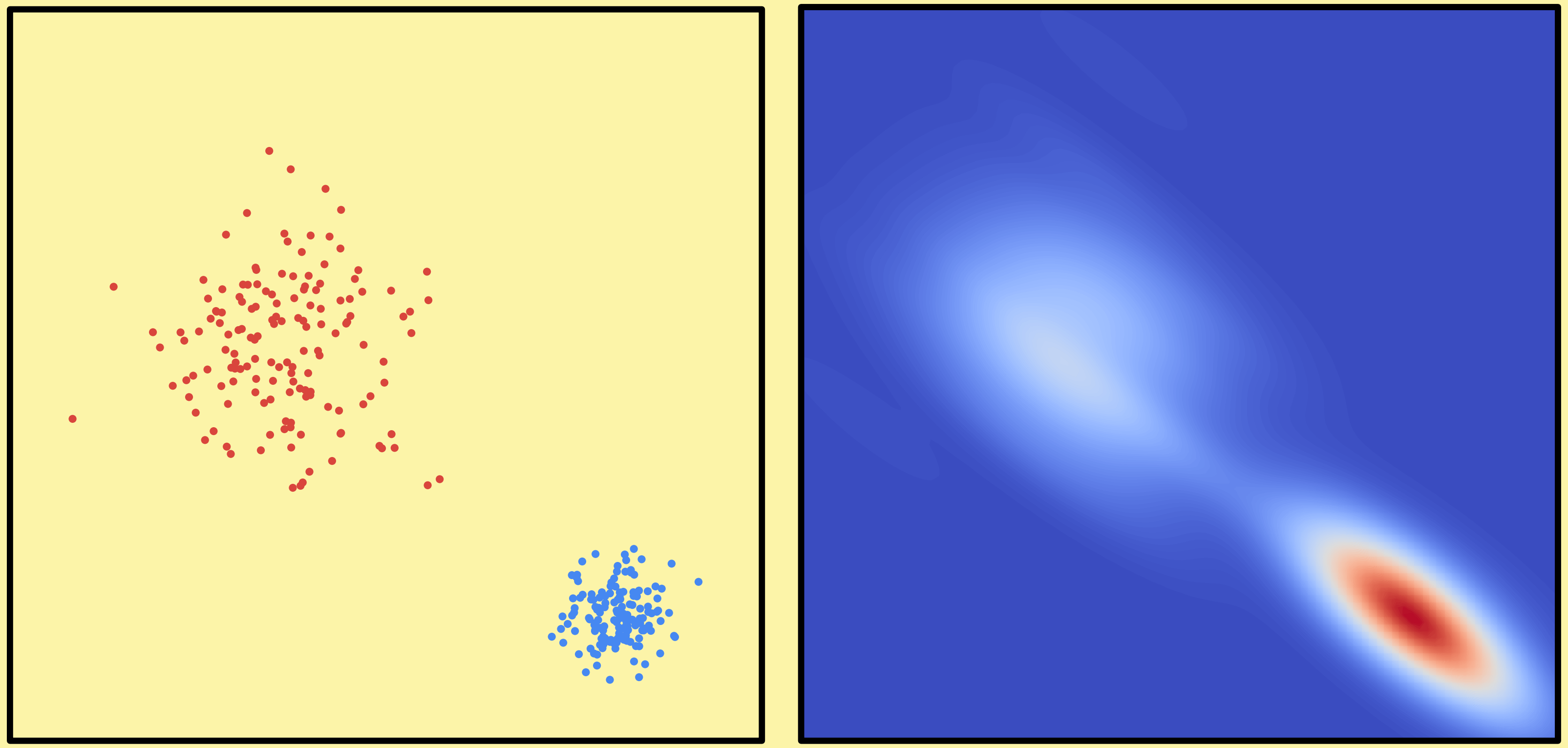
We aim to control properties of the latent space, but from a **topological point of view**!

Assume, we want to do **Kernel Density Estimation (KDE)** in the latent space $\mathcal{Z}$.

Data (z_i) Gaussian KDE



Data (z_i) Gaussian KDE

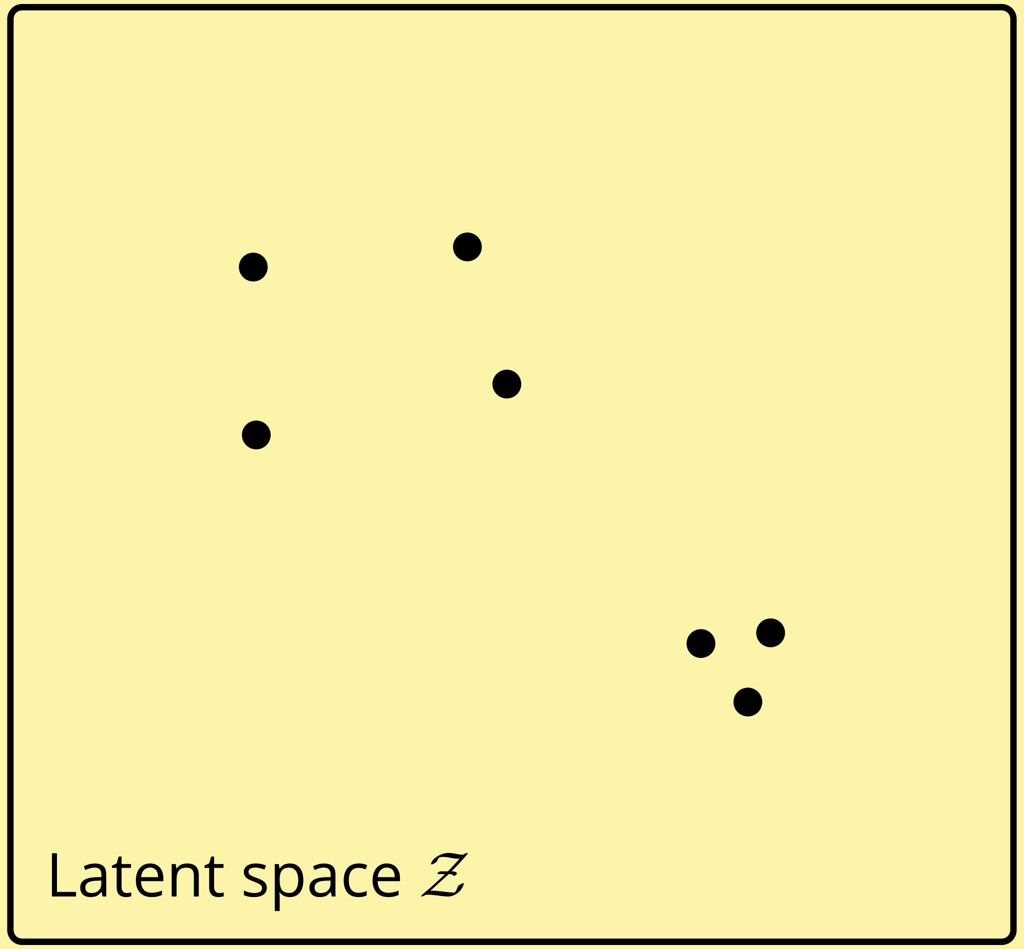


Bandwidth selection: Scott's rule [Scott, 1992]

Bandwidth selection can be challenging, as the scaling greatly differs!

Controlling connectivity

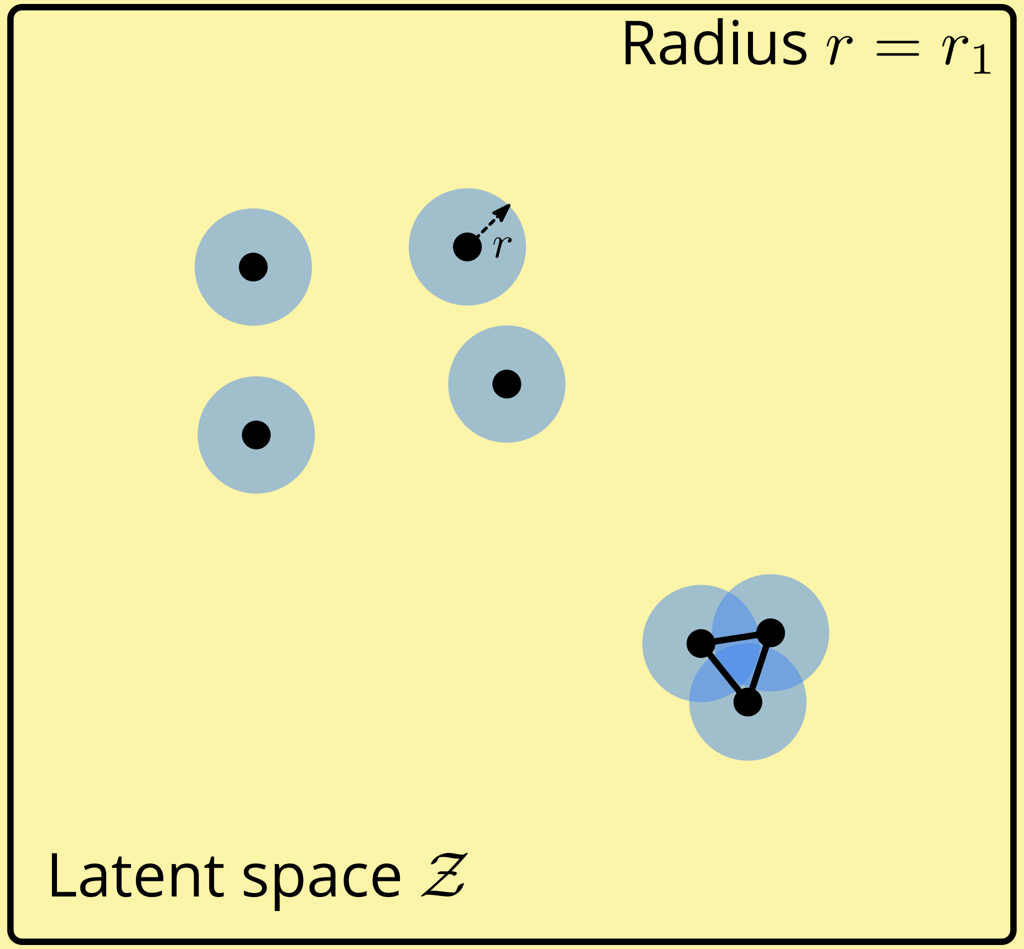
Q: How do we **capture** topological properties and what do we want to control?



Controlling connectivity

Q: How do we **capture** topological properties and what do we want to control?

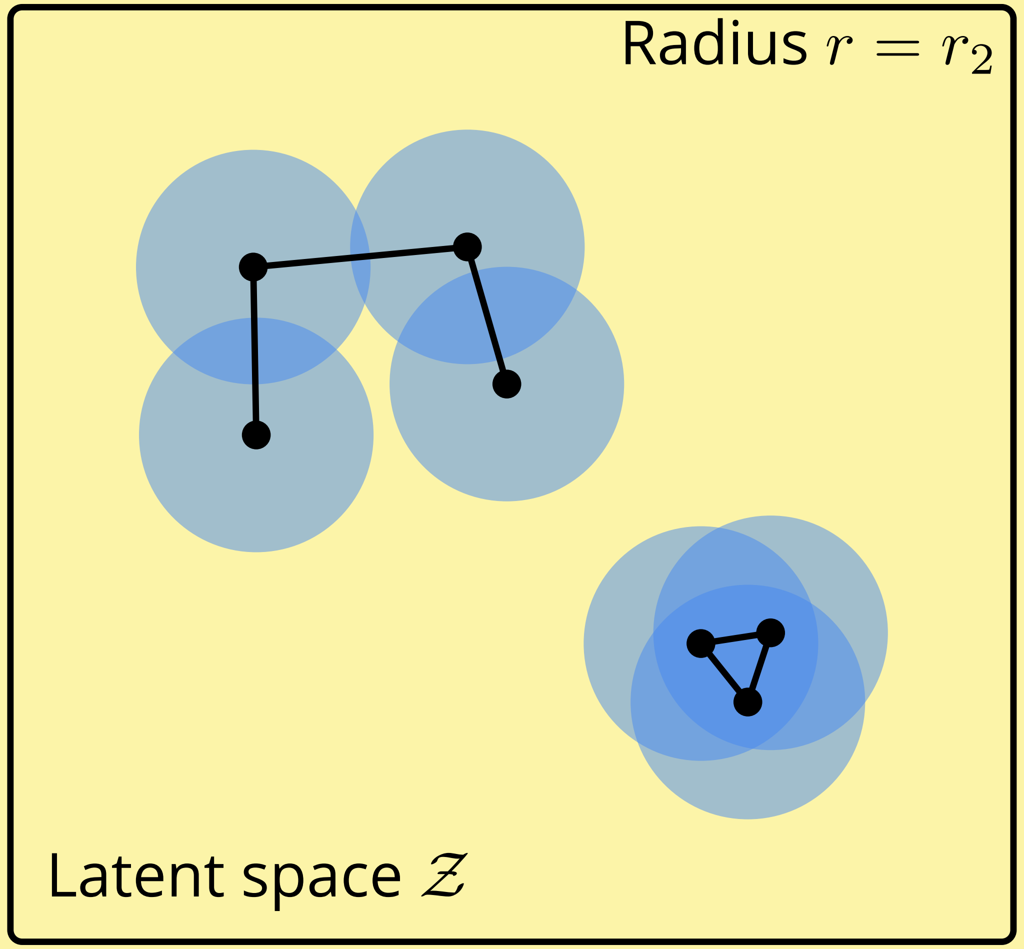
Vietoris Rips Persistent Homology (PH)



Controlling connectivity

Q: How do we **capture** topological properties and what do we want to control?

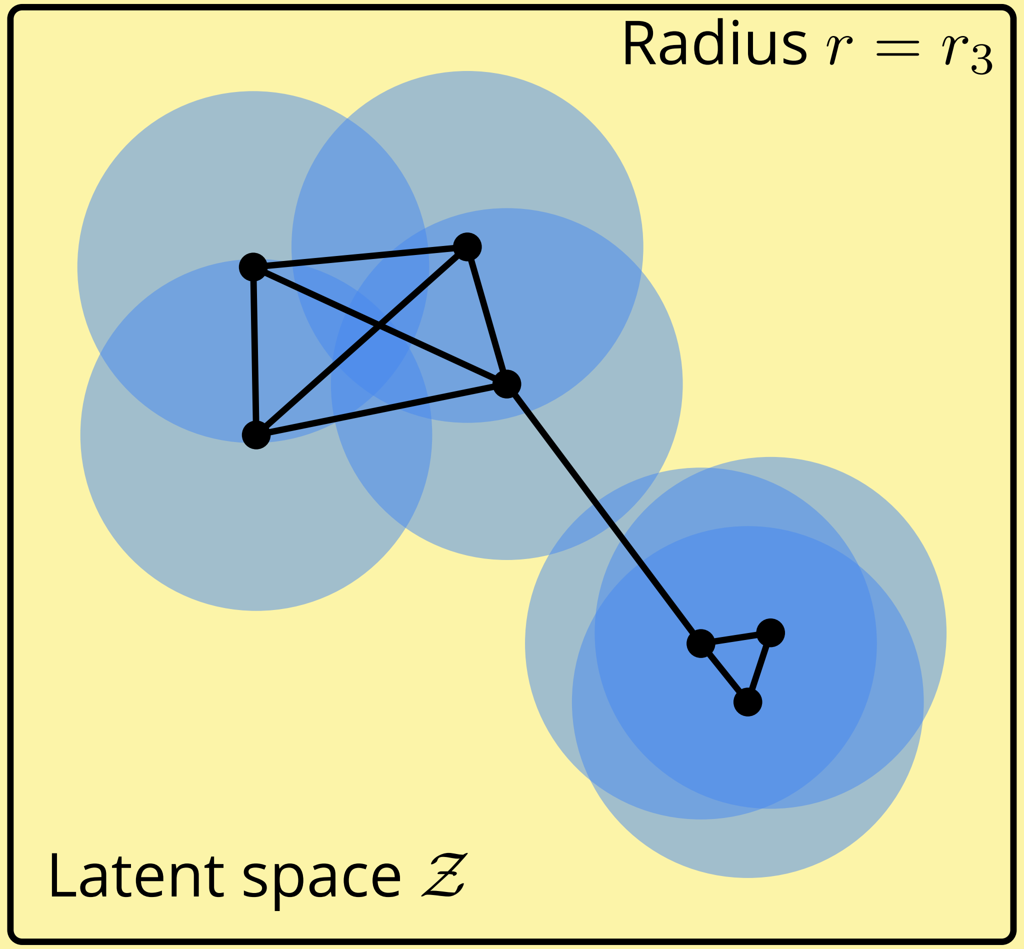
Vietoris Rips Persistent Homology (PH)



Controlling connectivity

Q: How do we **capture** topological properties and what do we want to control?

Vietoris Rips Persistent Homology (PH)

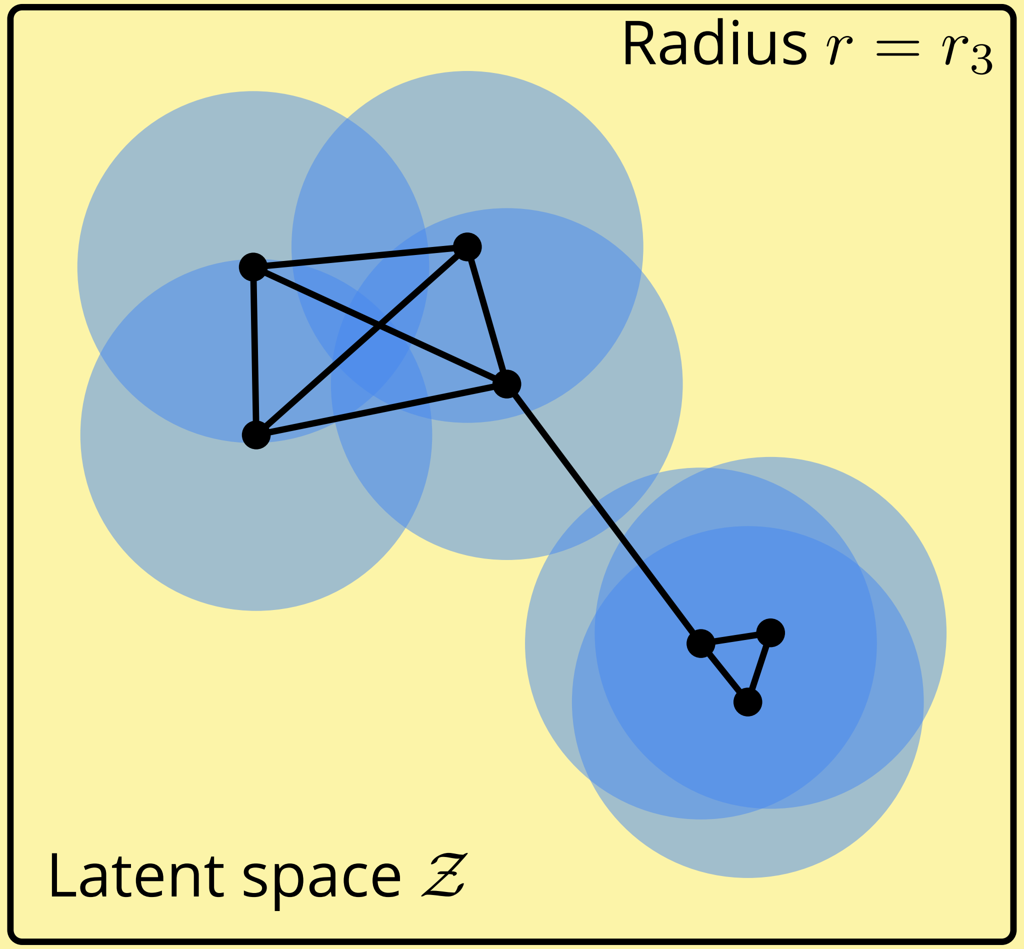


- PH tracks topological changes as the ball radius r increases
- **Connectivity information** is caputred by 0-dim. persistent homology

Controlling connectivity

Q: How do we **capture** topological properties and what do we want to control?

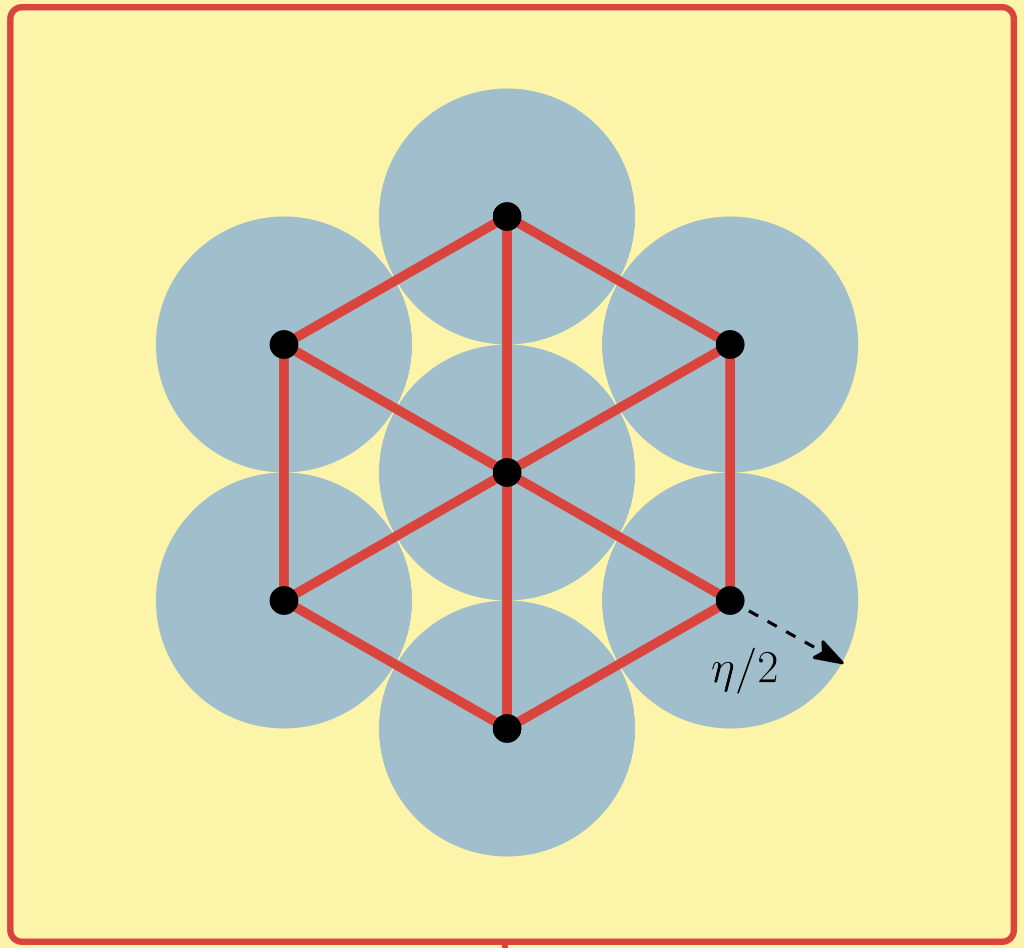
Vietoris Rips Persistent Homology (PH)



What if

$z \mapsto f_\theta(z)$

Homogeneous arrangement!

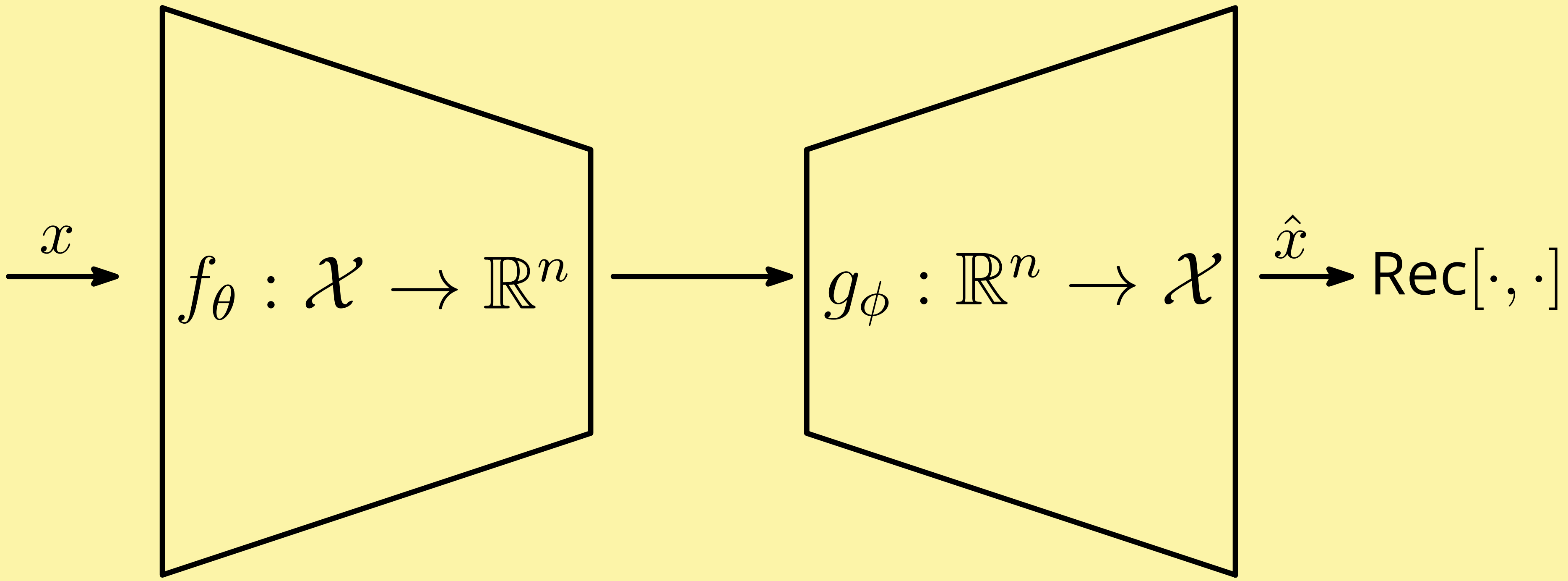


beneficial for KDE

- PH tracks topological changes as the ball radius r increases
- **Connectivity information** is caputred by 0-dim. persistent homology

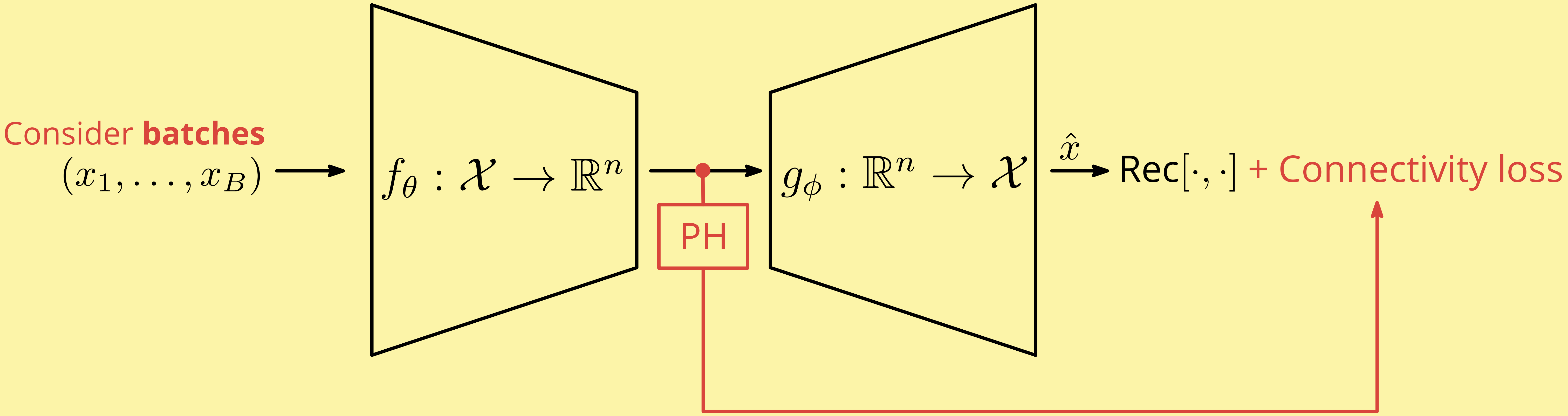
Connectivity loss

Q: How can we **control** topological properties (connectivity properties in particular)?



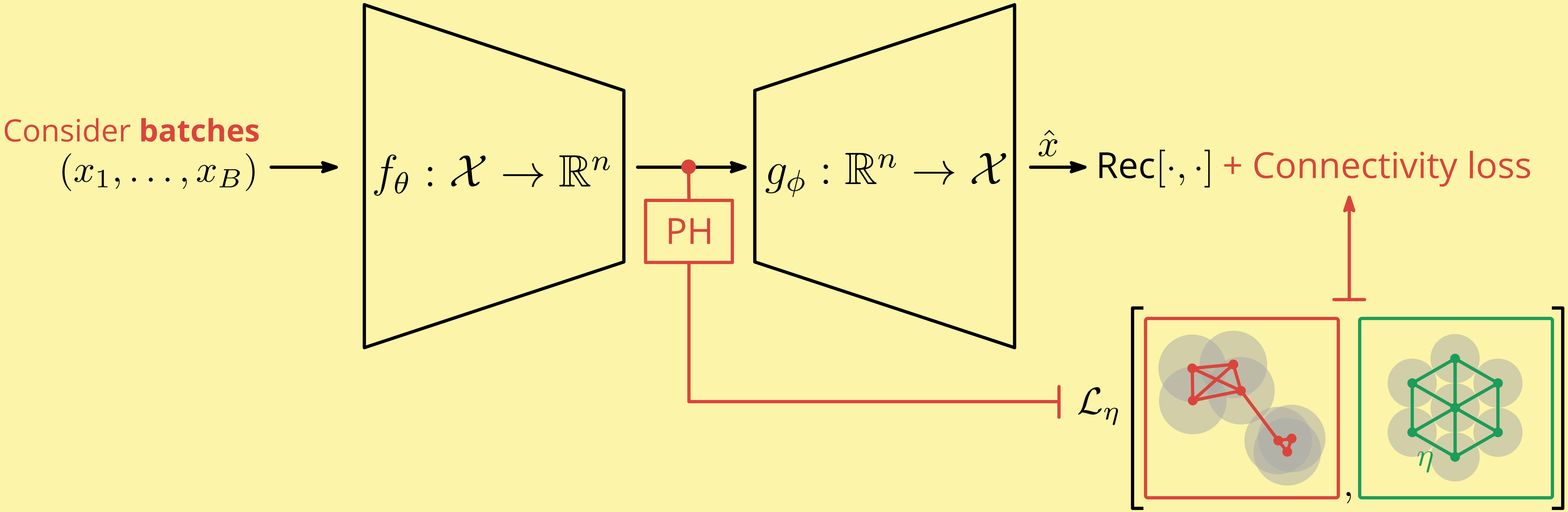
Connectivity loss

Q: How can we **control** topological properties (connectivity properties in particular)?



Connectivity loss

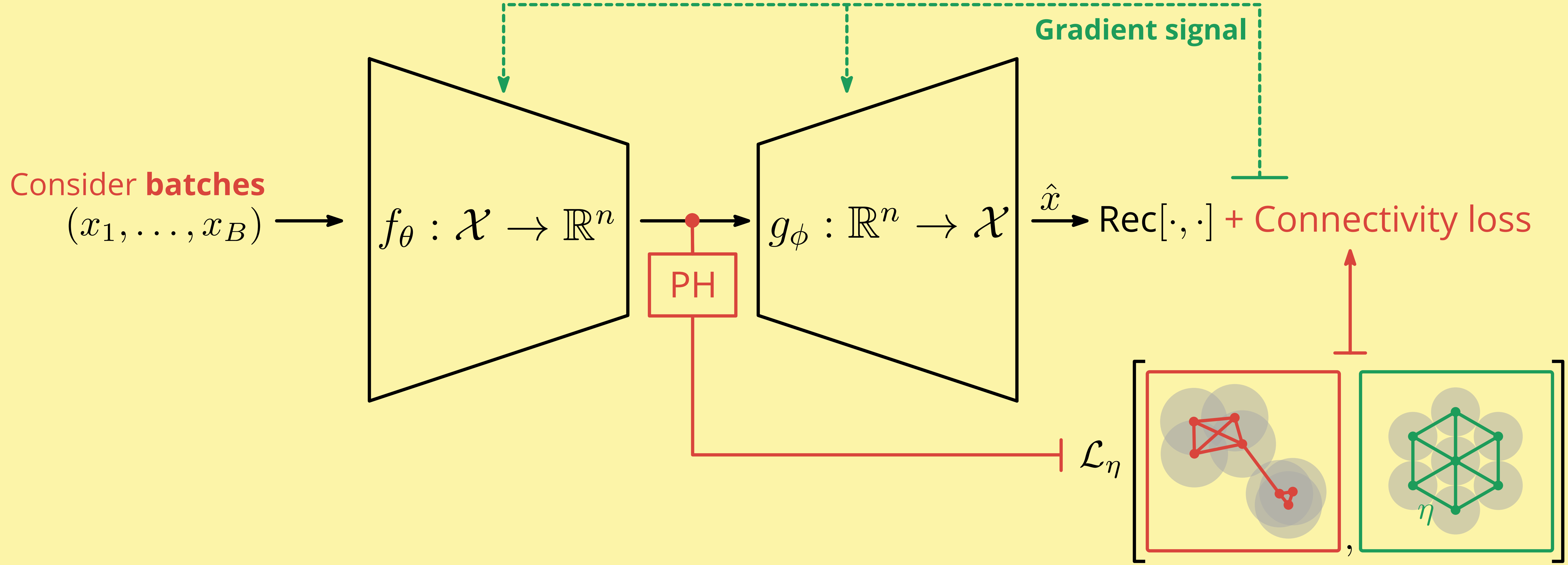
Q: How can we **control** topological properties (connectivity properties in particular)?



penalize deviation from **homogeneous arrangement** (with scale η)

Connectivity loss

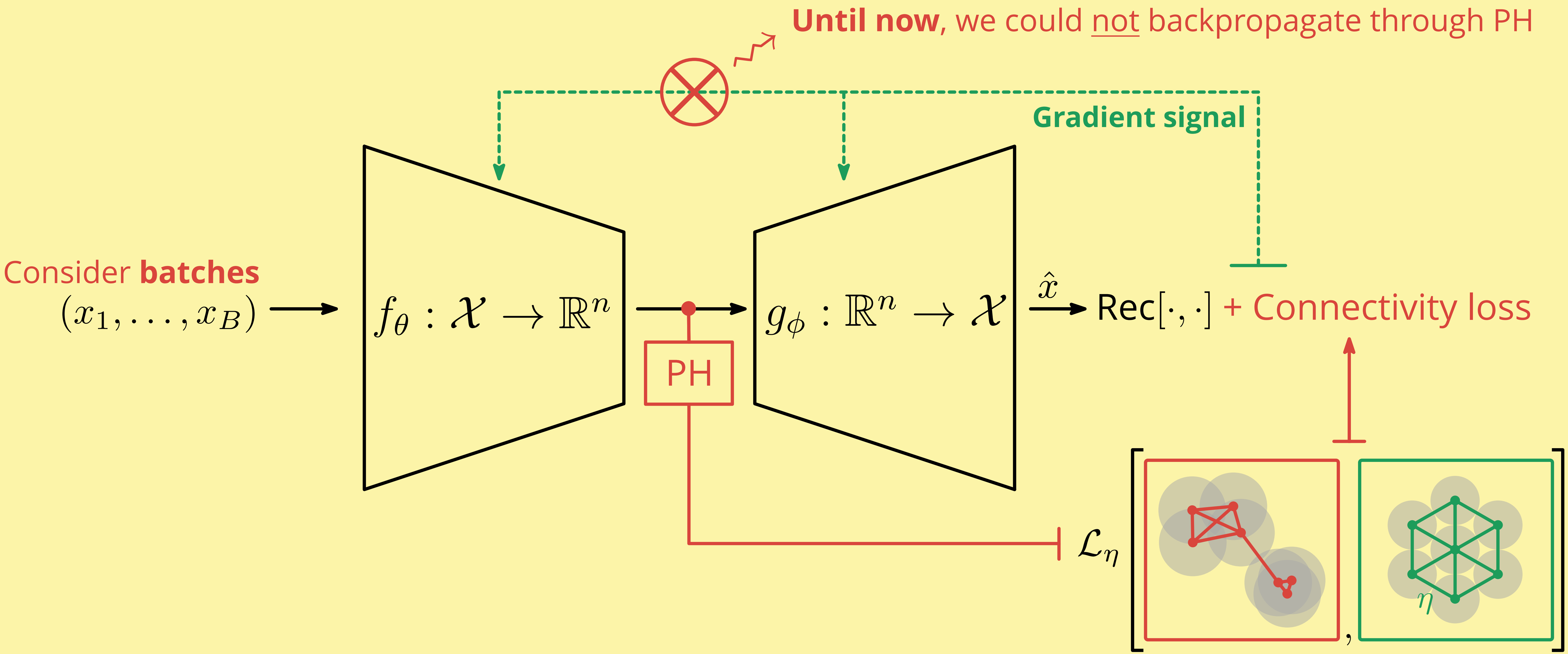
Q: How can we **control** topological properties (connectivity properties in particular)?



penalize deviation from **homogeneous arrangement** (with scale η)

Connectivity loss

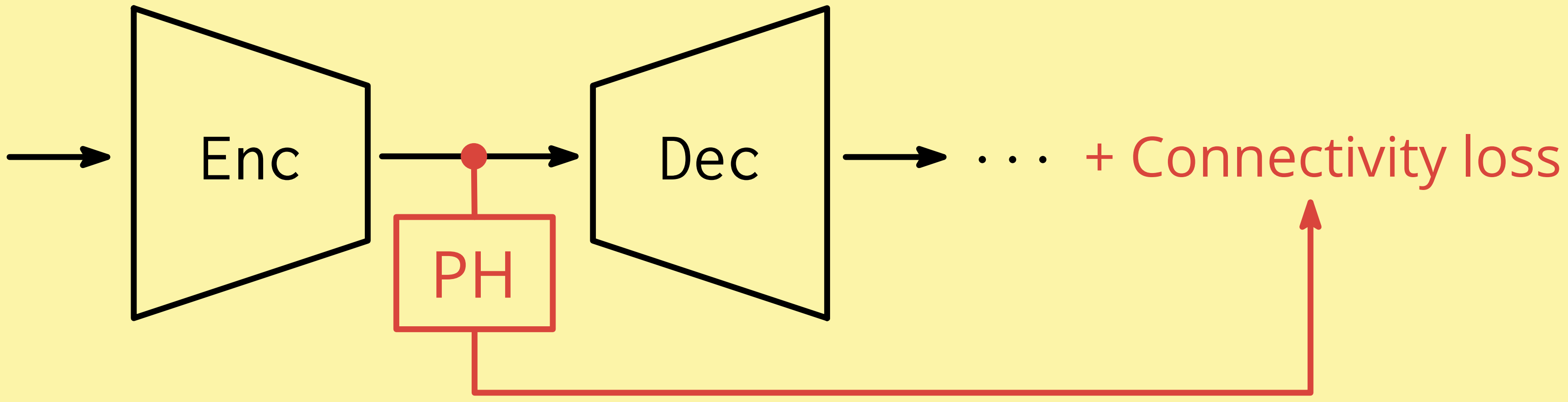
Q: How can we **control** topological properties (connectivity properties in particular)?



penalize deviation from **homogeneous arrangement** (with scale η)

Connectivity loss

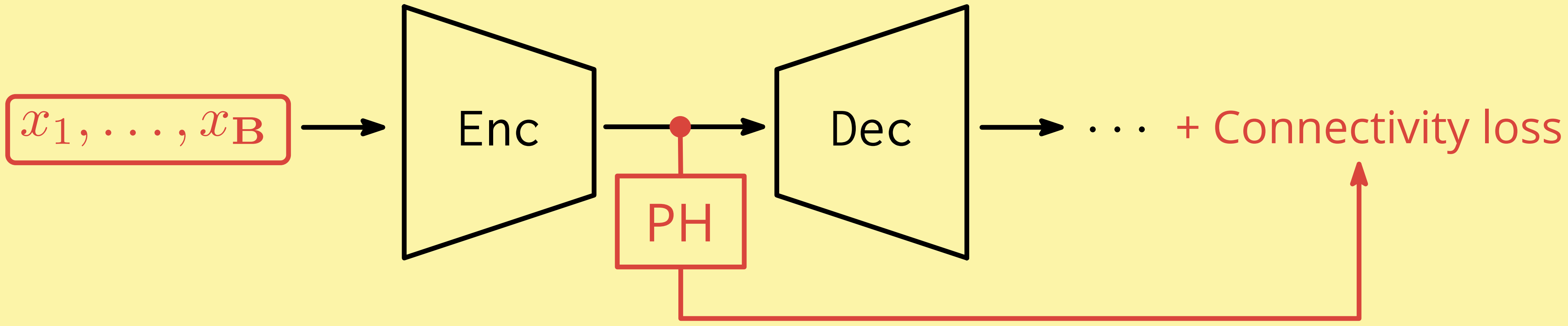
From a **theoretical perspective**, we show . . .



(1) . . . that under mild conditions, the **connectivity loss is differentiable**

Connectivity loss

From a **theoretical perspective**, we show . . .

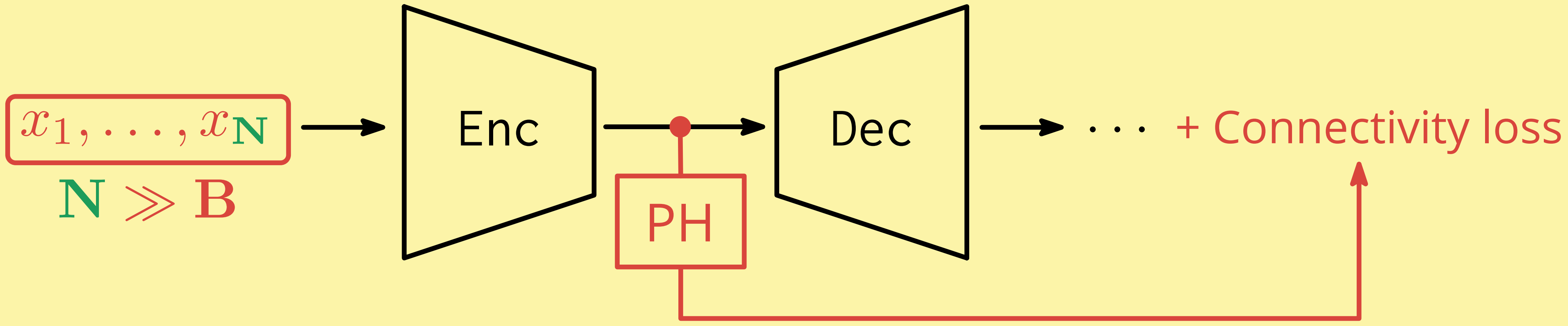


(1) . . . that under mild conditions, the **connectivity loss is differentiable**

(2) . . . metric-entropy based guidelines for choosing the training batch size $\mathbf{B}$

Connectivity loss

From a **theoretical perspective**, we show . . .



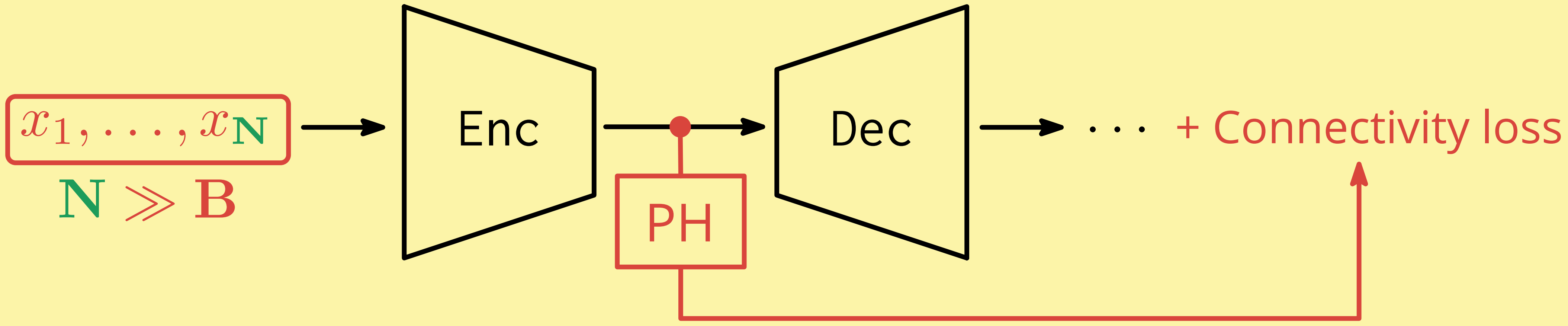
(1) . . . that under mild conditions, the **connectivity loss is differentiable**

(2) . . . metric-entropy based guidelines for choosing the training batch size $\mathbf{B}$

(3) . . . "densification" effects occur for samples, $\mathbf{N}$, larger than the training batch size $\mathbf{B}$

Connectivity loss

From a **theoretical perspective**, we show . . .



(1) . . . that under mild conditions, the **connectivity loss is differentiable**

(2) . . . metric-entropy based guidelines for choosing the training batch size $\mathbf{B}$

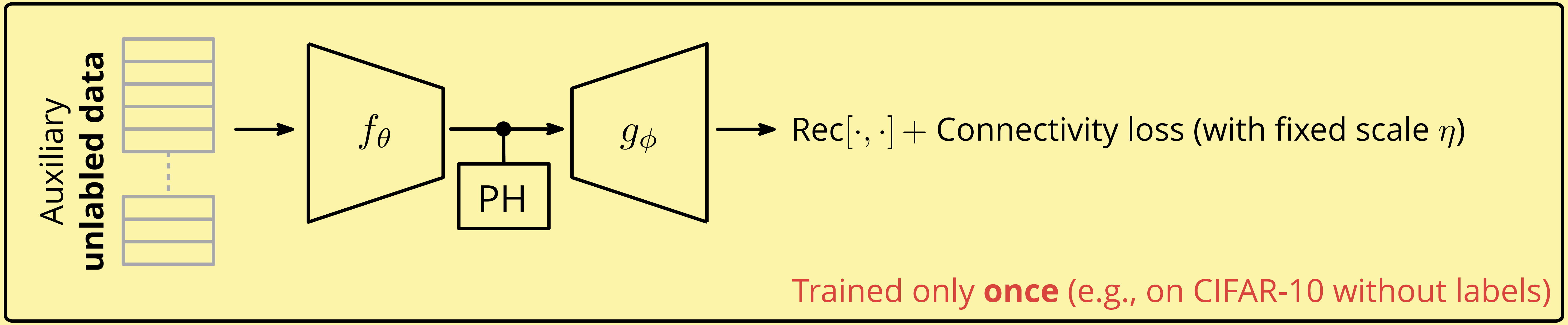
(3) . . . "densification" effects occur for samples, $\mathbf{N}$, larger than the training batch size $\mathbf{B}$

Intuitively, during training ...

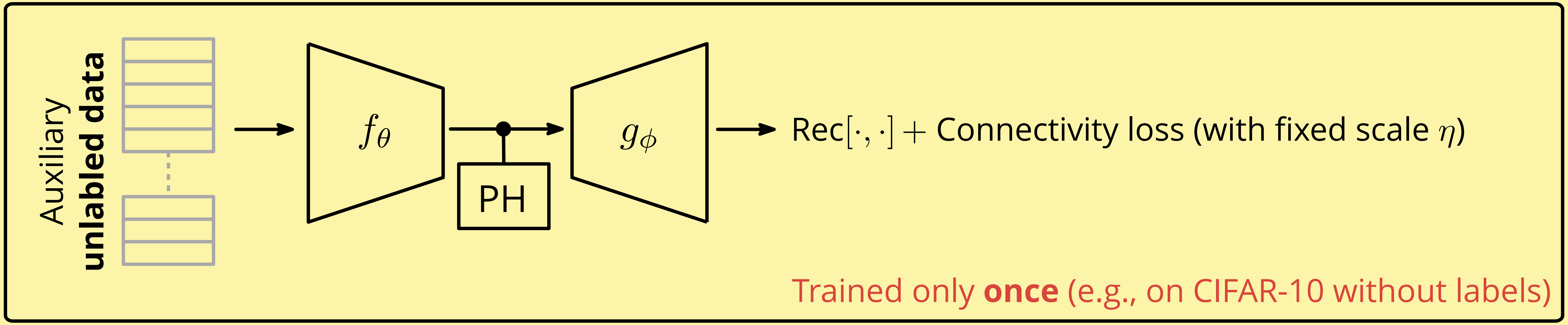
... the reconstruction loss controls **what** is worth capturing

... the connectivity loss controls **how** to topologically organize the latent space

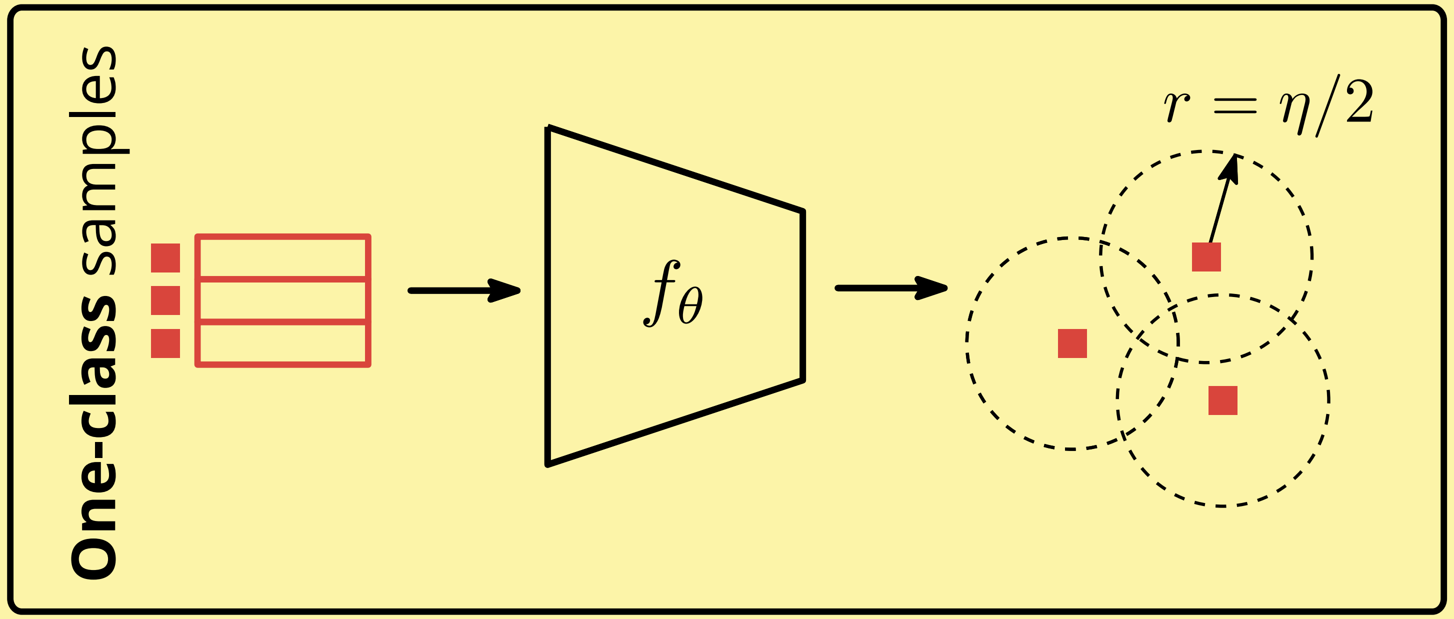
Experiments – Task: One-class learning



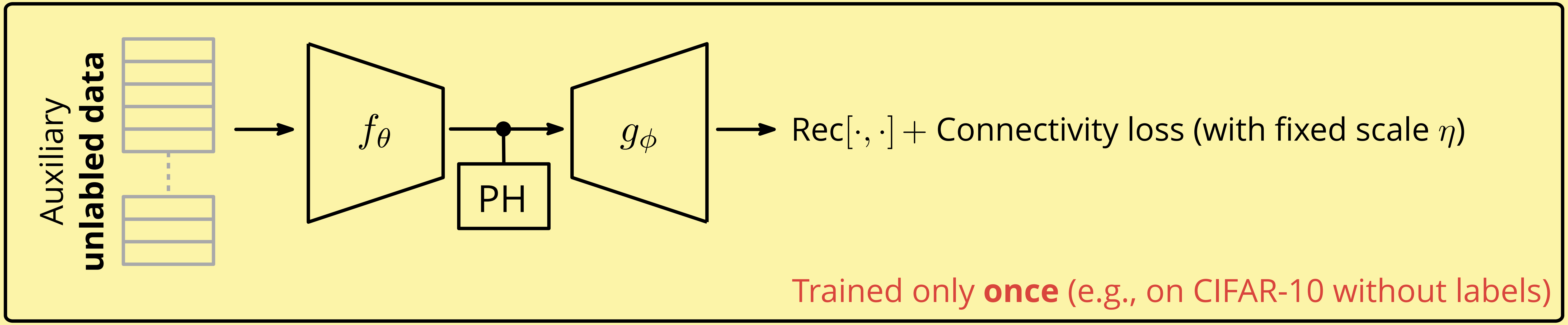
Experiments – Task: One-class learning



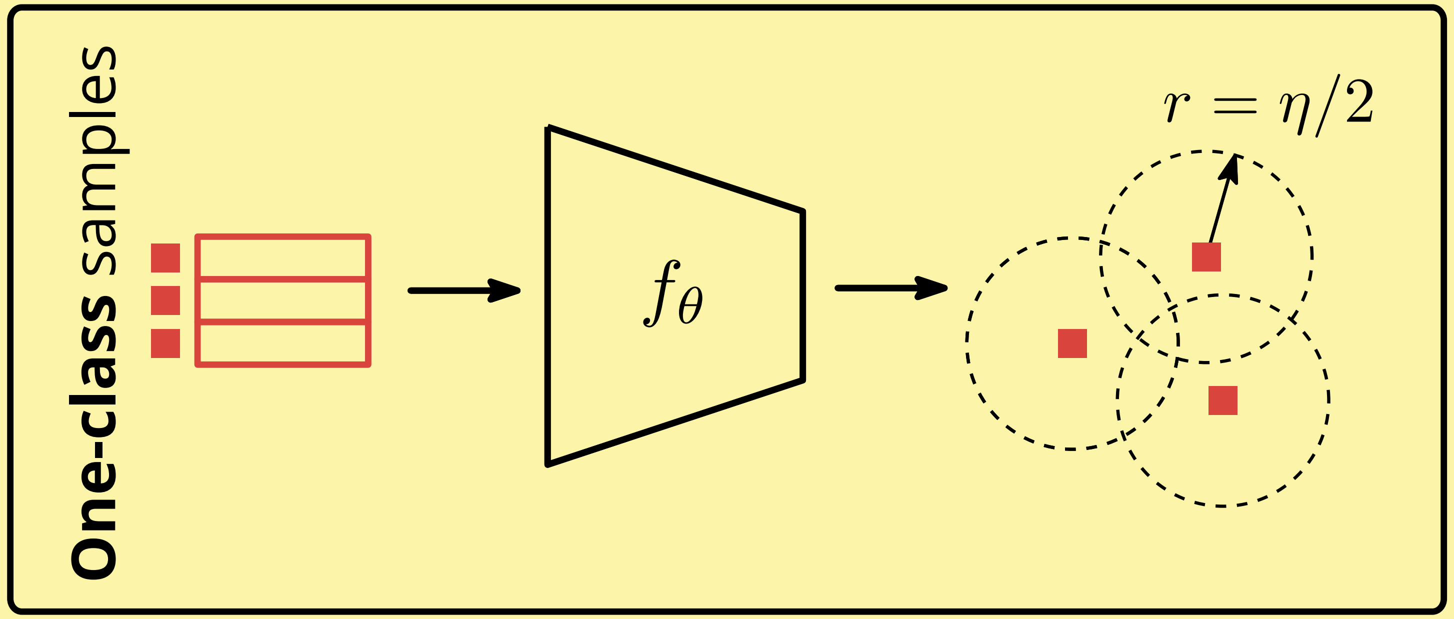
KDE-inspired **one-class** "learning"



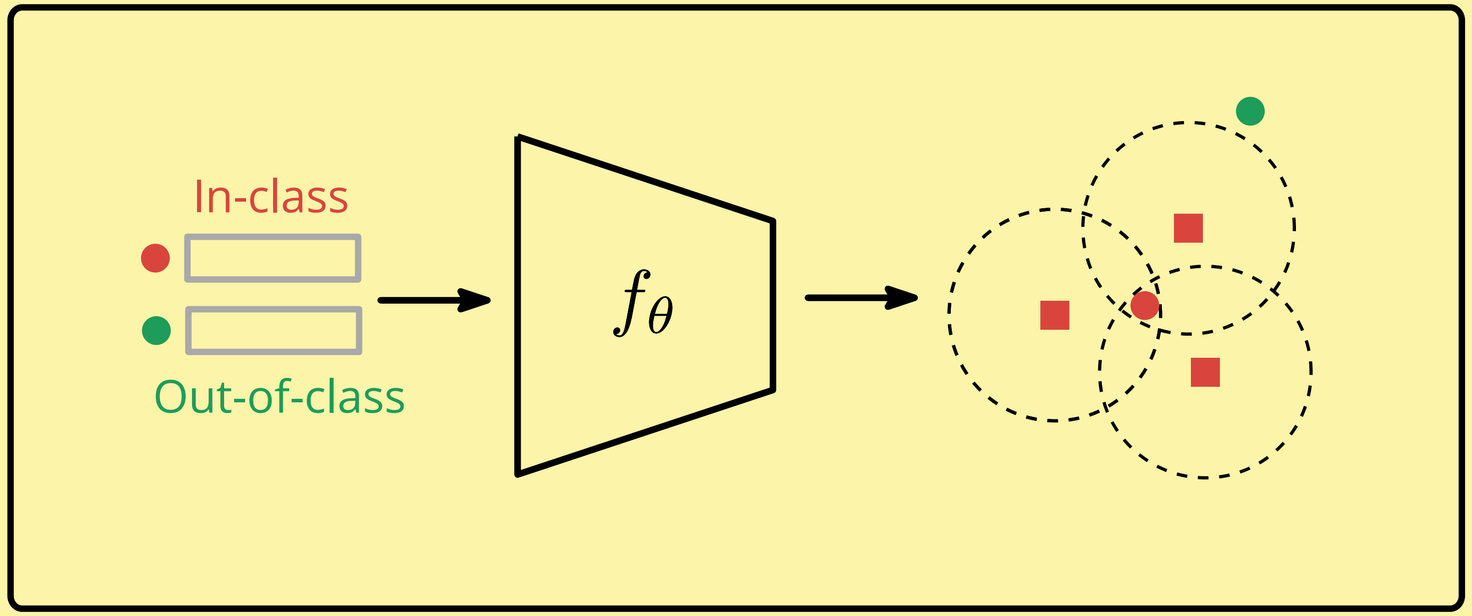
Experiments – Task: One-class learning



KDE-inspired **one-class** "learning"



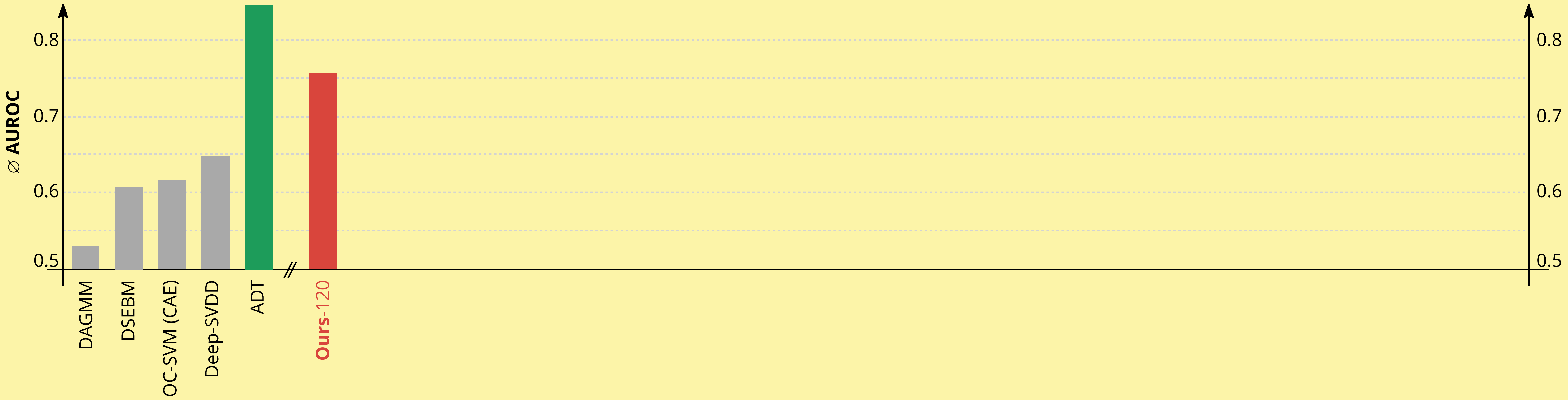
Computation of a **one-class** score



Count #samples falling into balls of radius η,
anchored at the one-class instances ■

Results – Task: One-class learning

CIFAR-10 (AE trained on CIFAR-100)



ADT [Goland & El-Yaniv, NIPS '18]

DAGMM [Zong et al., ICLR '18]

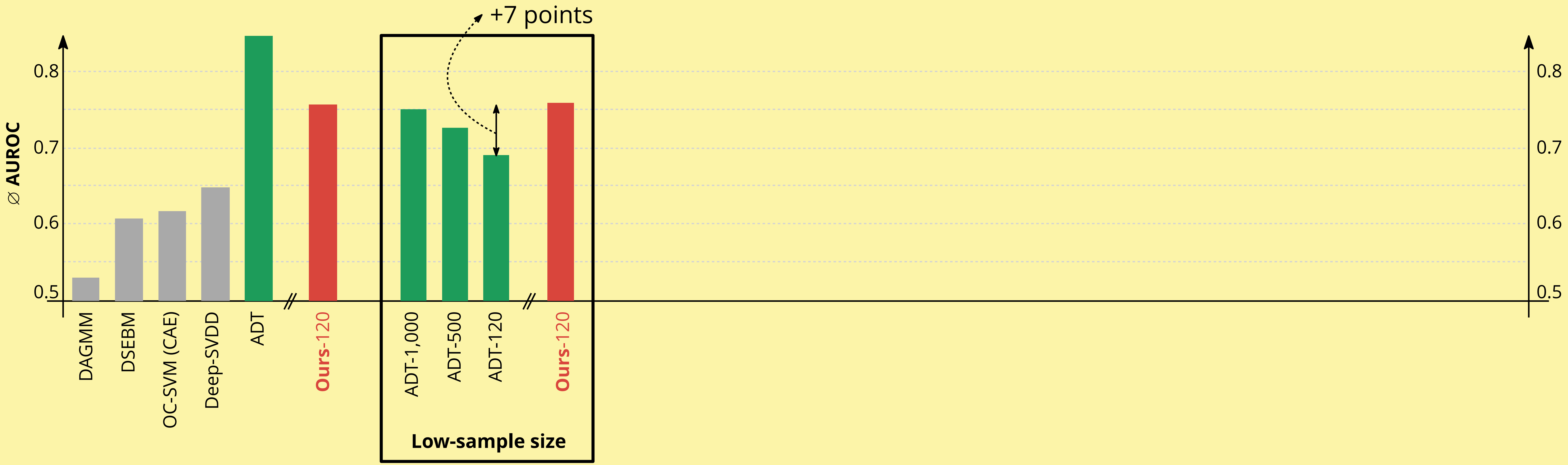
DSEBM [Zhai et al., ICML '16]

Deep-SVDD [Ruff et al., ICML '18]

Training batch size: $B = 100$

Results – Task: One-class learning

CIFAR-10 (AE trained on CIFAR-100)



ADT [Goland & El-Yaniv, NIPS '18]

DAGMM [Zong et al., ICLR '18]

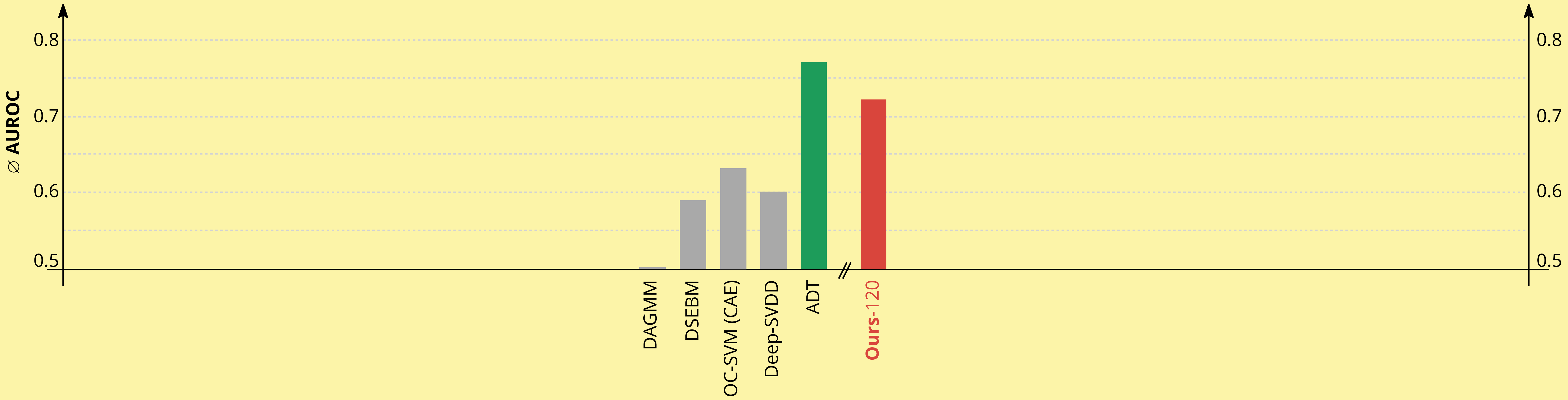
DSEBM [Zhai et al., ICML '16]

Deep-SVDD [Ruff et al., ICML '18]

Training batch size: $B = 100$

Results – Task: One-class learning

CIFAR-20 (AE trained on CIFAR-10)



ADT [Goland & El-Yaniv, NIPS ’18]

DAGMM [Zong et al., ICLR ’18]

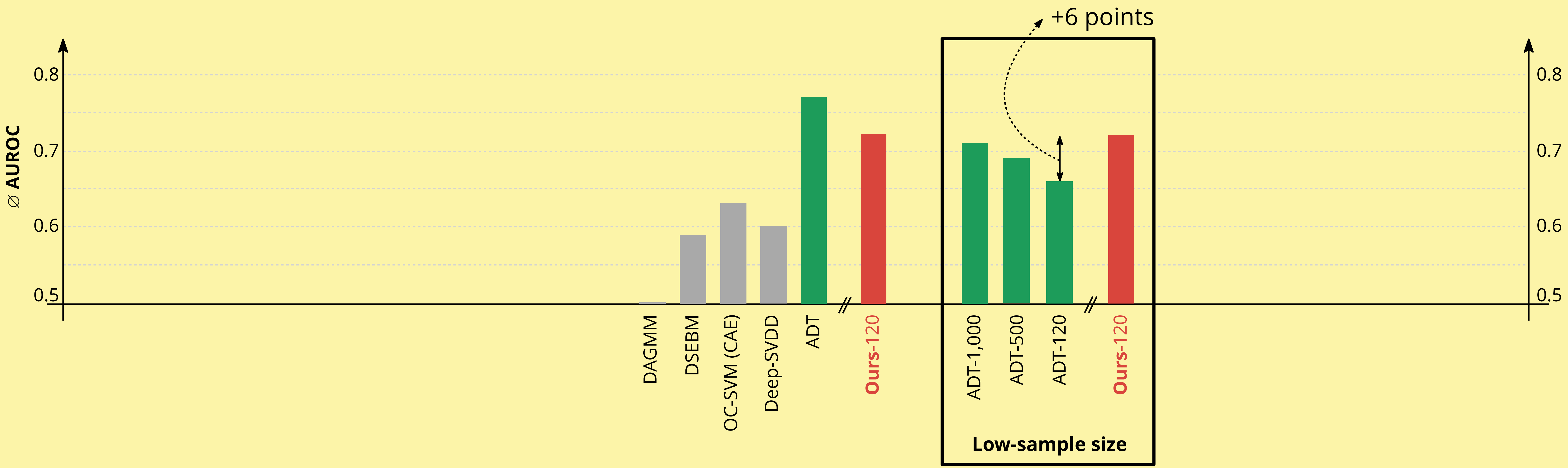
DSEBM [Zhai et al., ICML ’16]

Deep-SVDD [Ruff et al., ICML ’18]

Training batch size: $B = 100$

Results – Task: One-class learning

CIFAR-20 (AE trained on CIFAR-10)



ADT [Goland & El-Yaniv, NIPS '18]

DAGMM [Zong et al., ICLR '18]

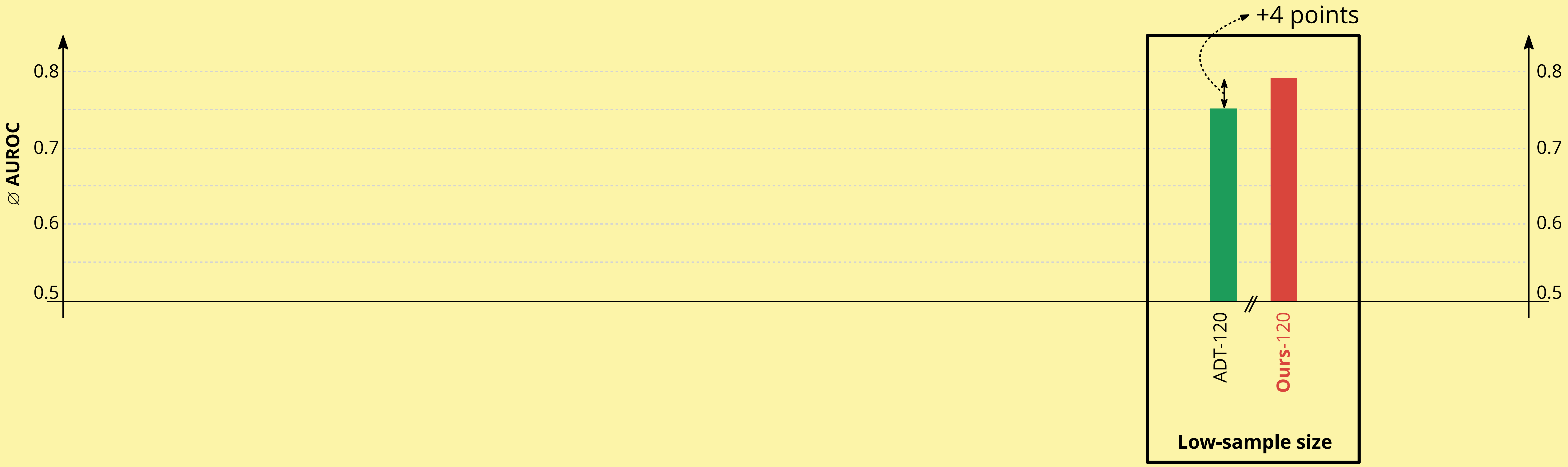
DSEBM [Zhai et al., ICML '16]

Deep-SVDD [Ruff et al., ICML '18]

Training batch size: $B = 100$

Results – Task: One-class learning

CIFAR-100 (AE trained on CIFAR-10)



ADT [Goland & El-Yaniv, NIPS '18]

DAGMM [Zong et al., ICLR '18]

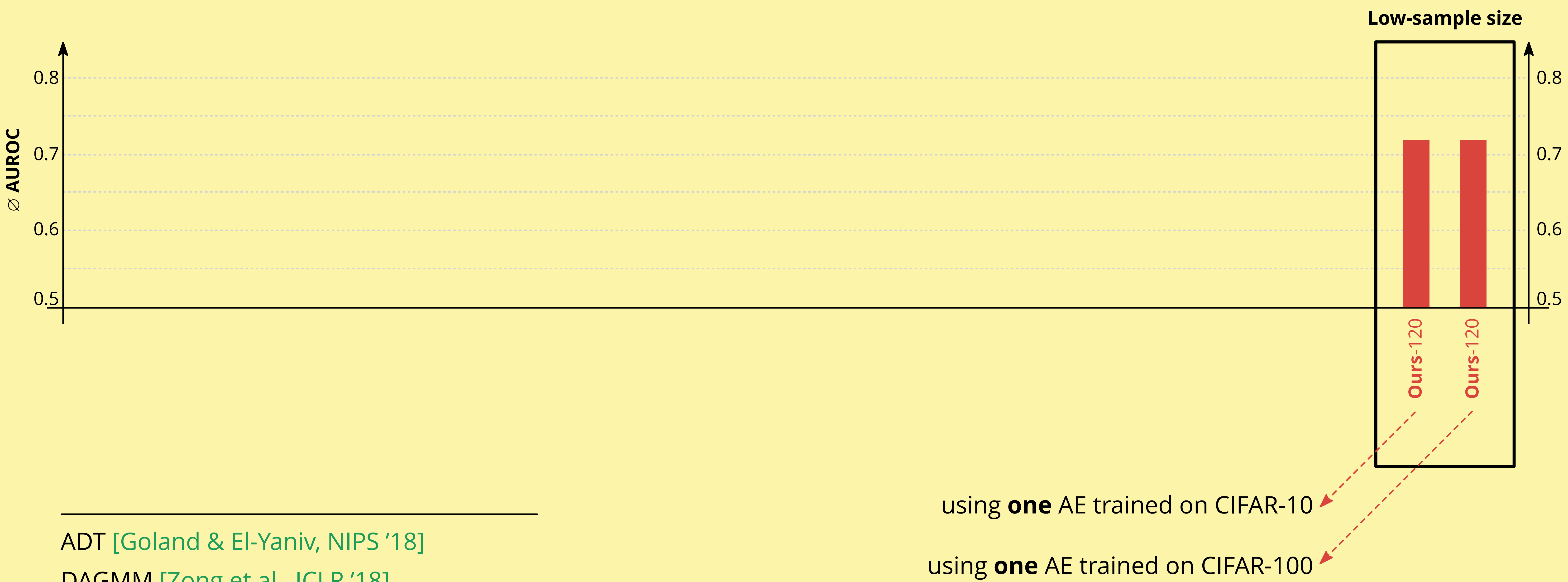
DSEBM [Zhai et al., ICML '16]

Deep-SVDD [Ruff et al., ICML '18]

Training batch size: $B = 100$

Results – Task: One-class learning

ImageNet (i.e., evaluation of **1,000** one-class models)



ADT [Goland & El-Yaniv, NIPS '18]

DAGMM [Zong et al., ICLR '18]

DSEBM [Zhai et al., ICML '16]

Deep-SVDD [Ruff et al., ICML '18]

Training batch size: $B = 100$

Come see our poster

#83

at 6.30pm (Pacific Ballroom)

```
import torch
import chofer_torchex.pershom as pershom

batch = torch.randn(10,5, requires_grad=True)
batch = batch.to('cuda')

non_ess, ess = pershom.vr_persistence_l1(batch,0,0)

example_loss = non_ess[:,1].sum()
example_loss.backward()
```

https://github.com/c-hofer/COREL_icml2019

PyTorch code available!