Neighborhood Reconstructing Autoencoders

Yonghyeon Lee

Seoul National University

Hyeokjun Kwon

Seoul National University

Frank C. Park

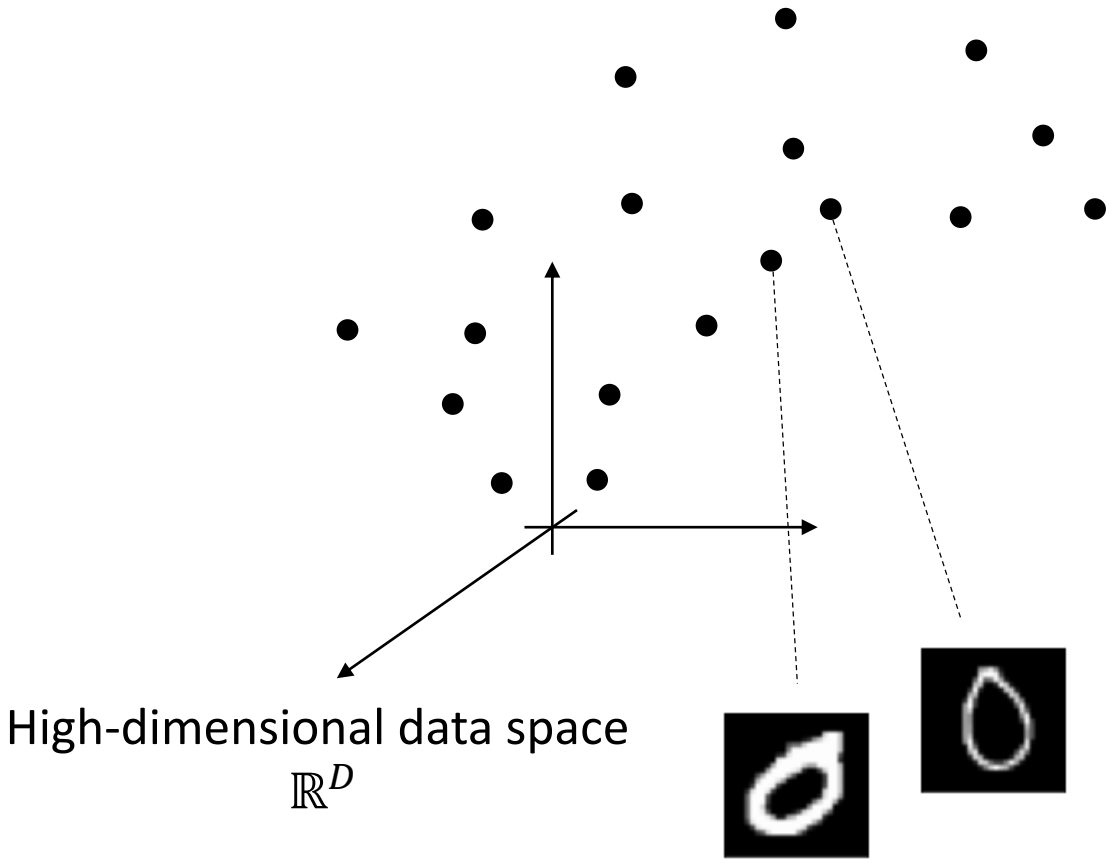
Seoul National University
Saige Research

NeurIPS 2021, 35th Conference on Neural Information Processing Systems

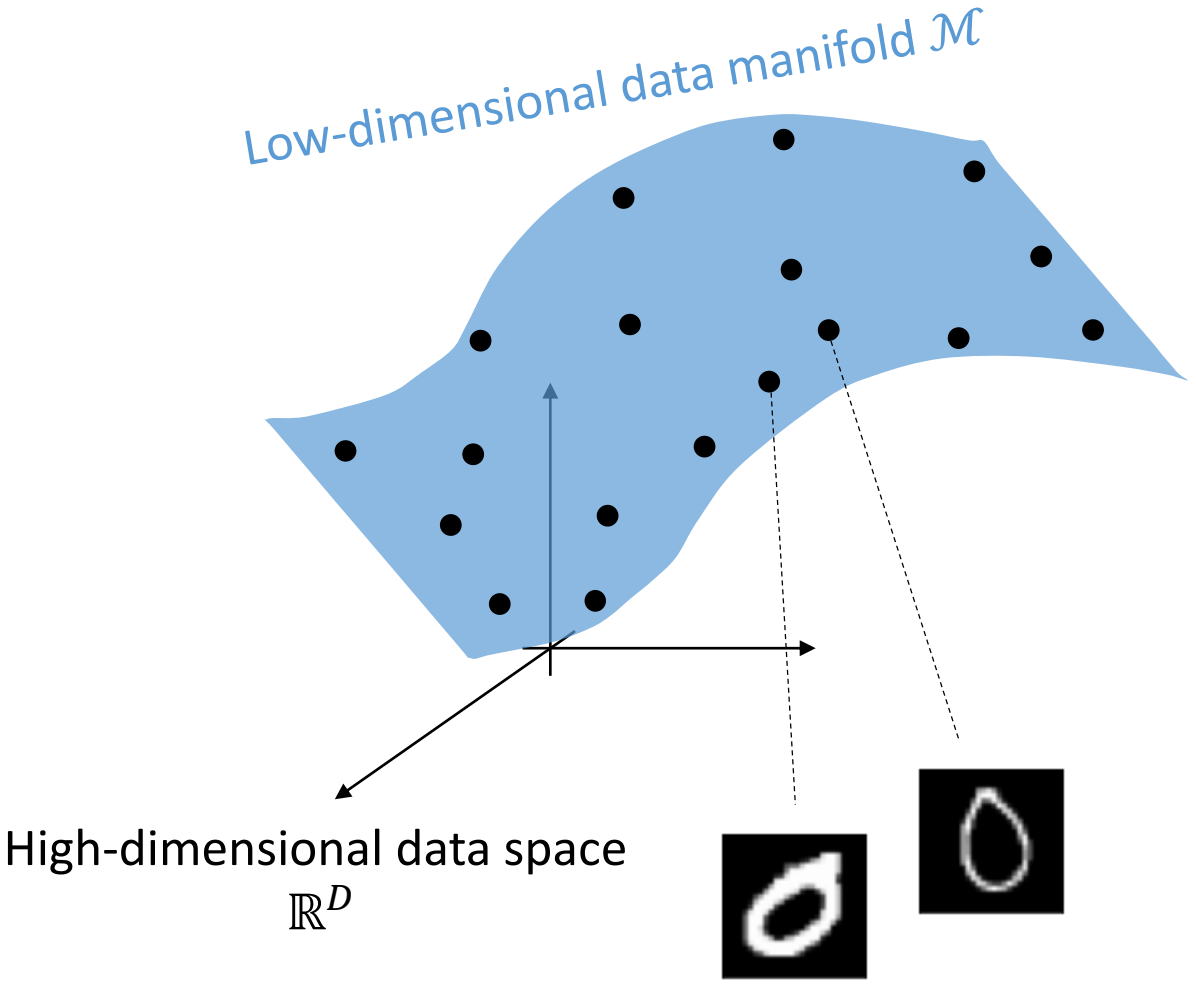
Neighborhood Reconstructing Autoencoder (NRAE)

is a **graph**-based autoencoder that explicitly accounts for the **local connectivity and geometry** of the data, and consequently learns a **more accurate data manifold and representation.**

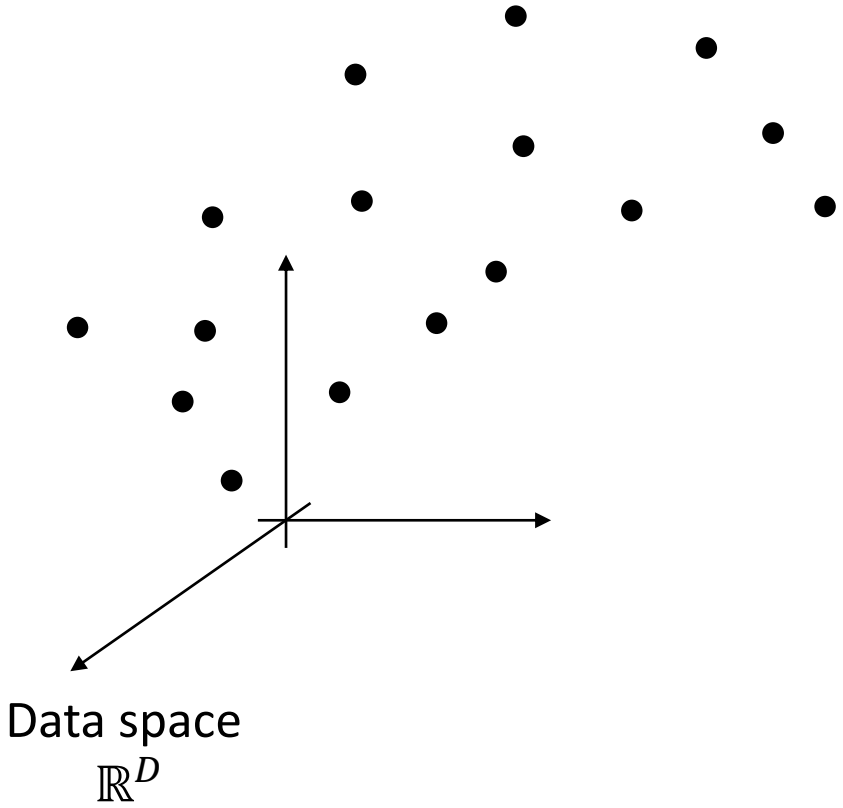
Premise: Manifold Hypothesis



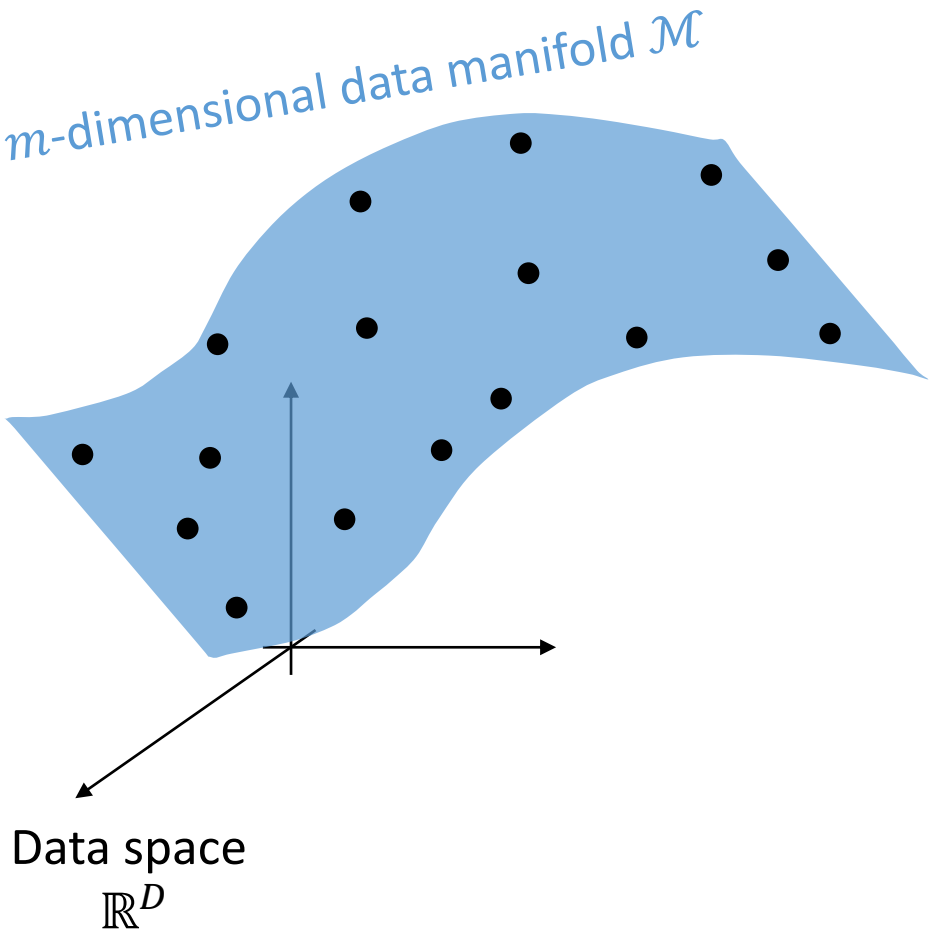
Premise: Manifold Hypothesis



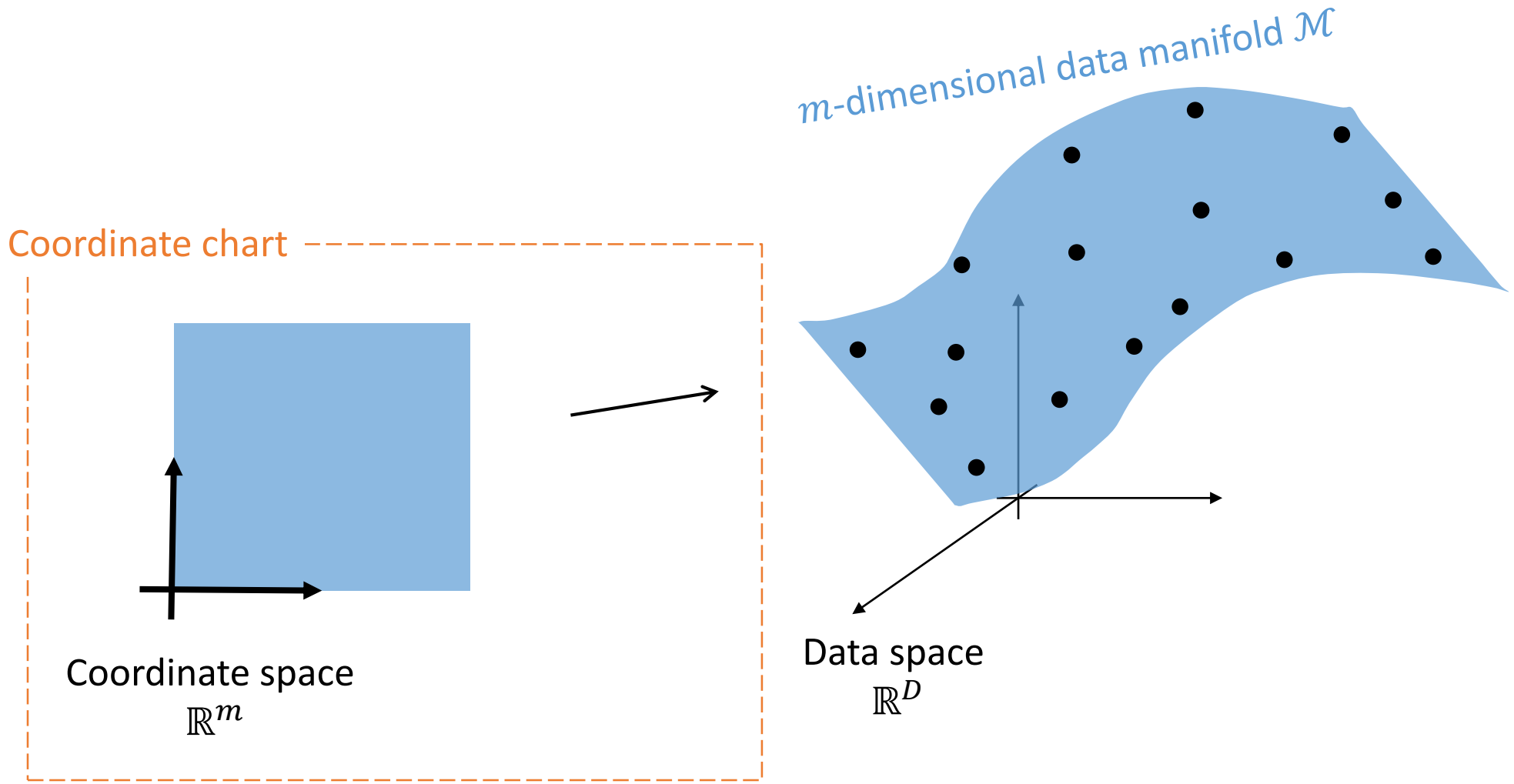
Manifold Learning



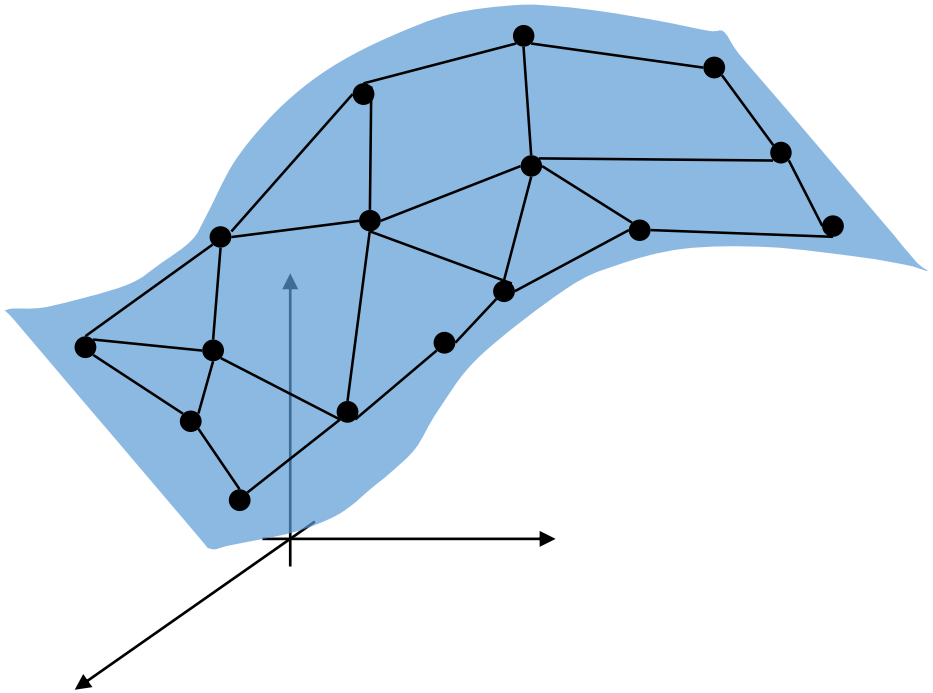
Manifold Learning



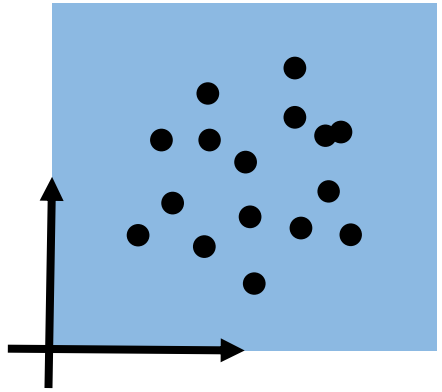
Manifold Learning



Traditional Manifold Learning

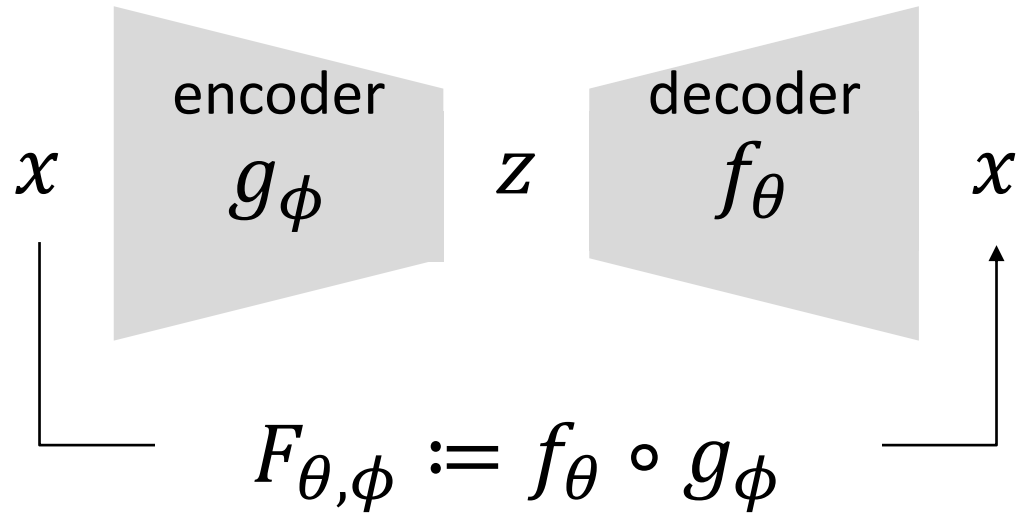


1. Represent data manifold as a graph



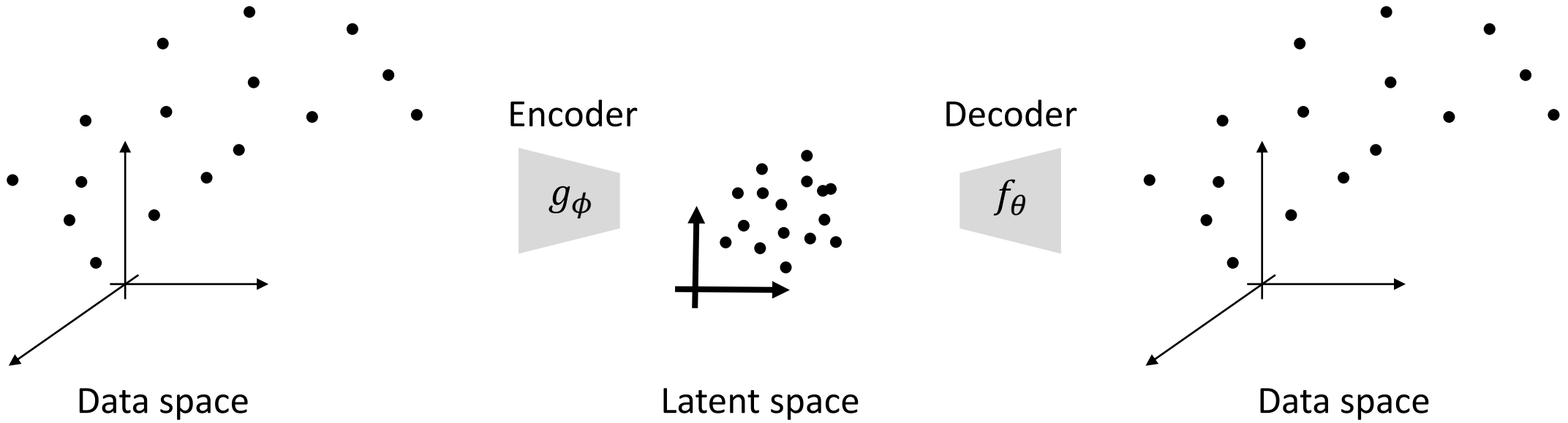
2. Compute low-dimensional embedding

Autoencoder-Based Manifold Learning

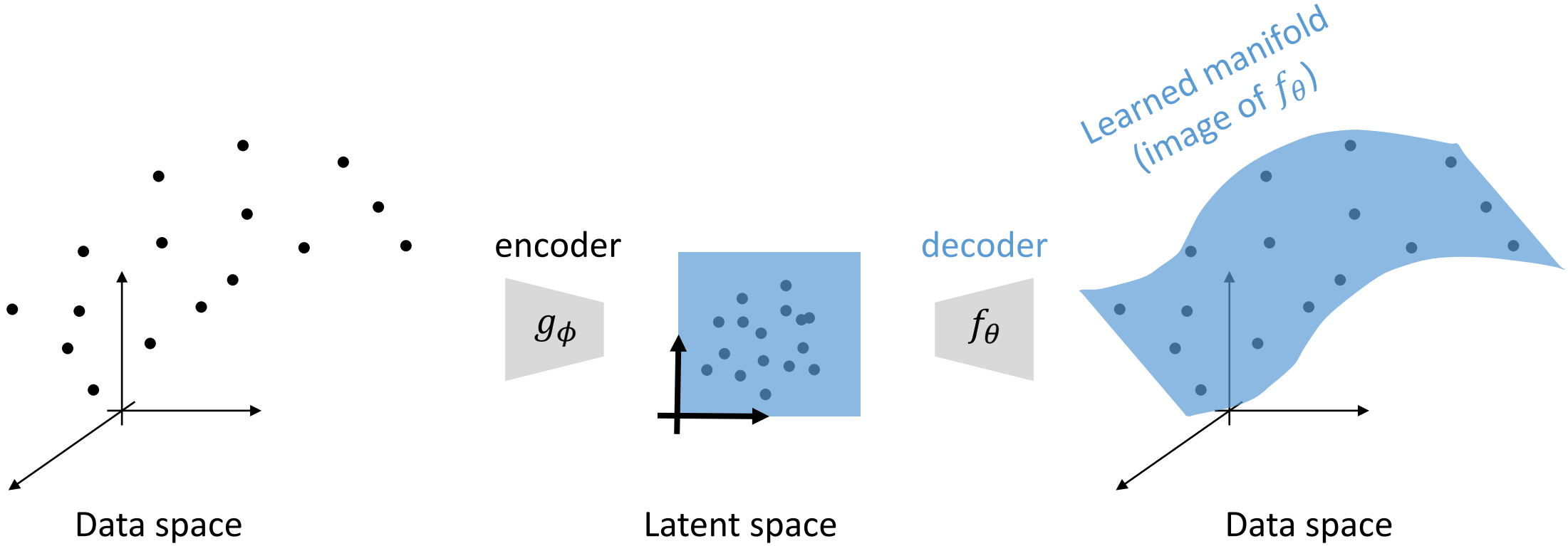


$$\min_{\theta,\phi} \sum_i ||F_{\theta,\phi}(x_i) - x_i||^2$$

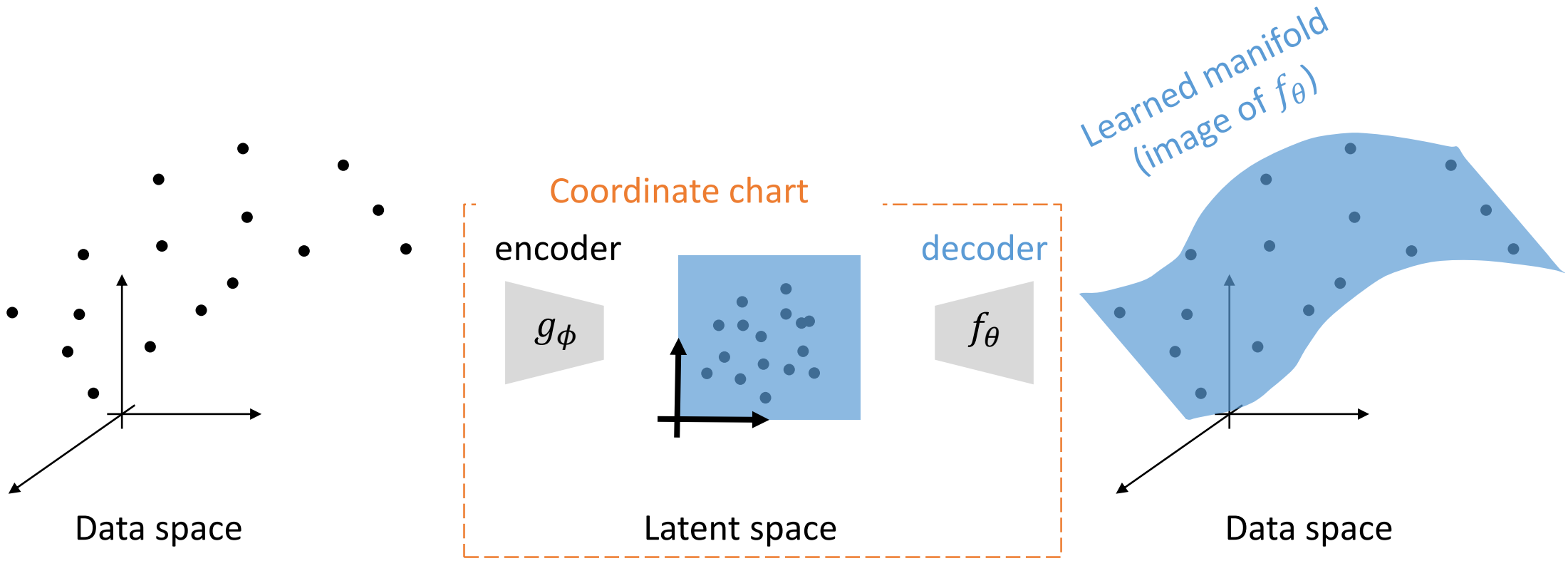
Autoencoder-Based Manifold Learning



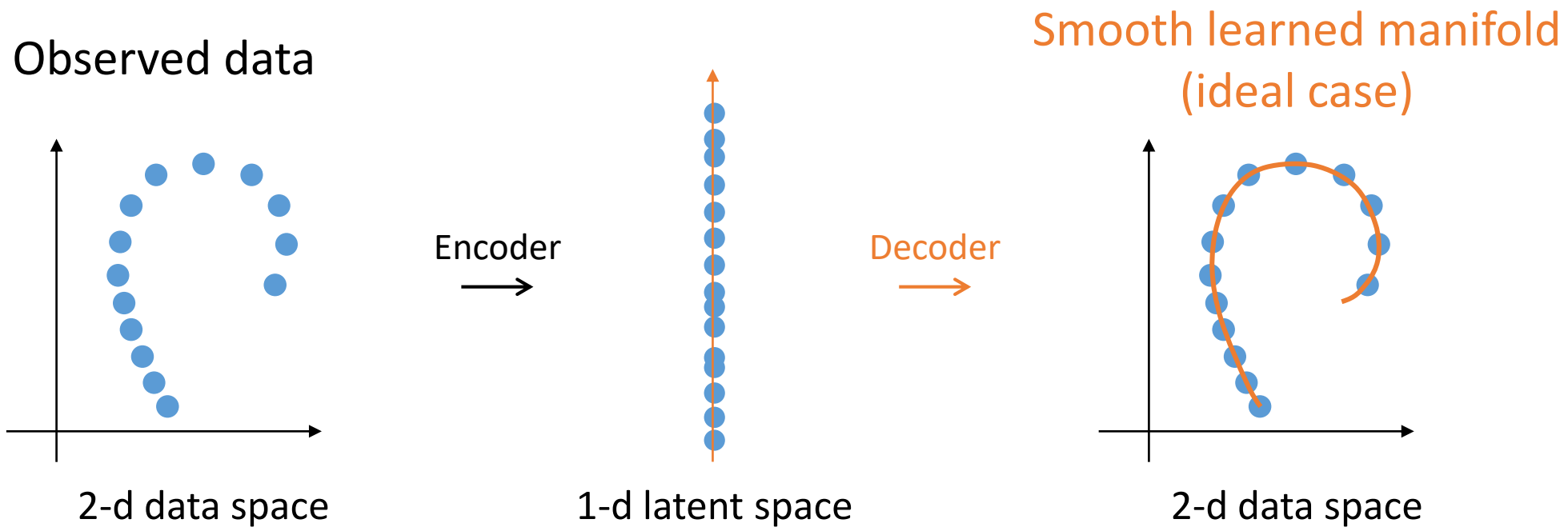
Autoencoder-Based Manifold Learning



Autoencoder-Based Manifold Learning

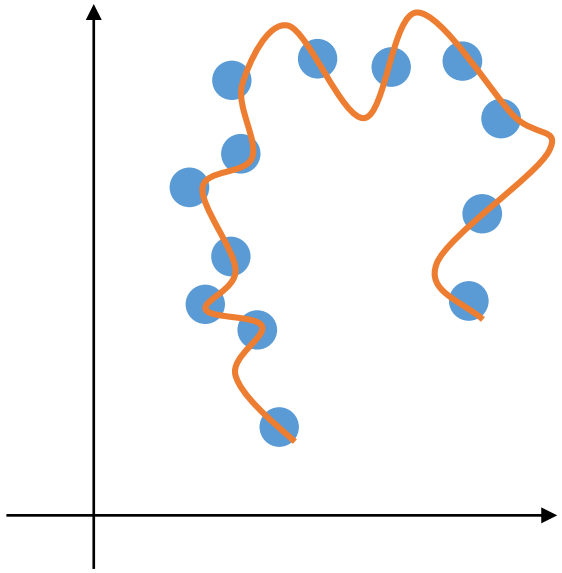


Example: One-Dimensional Manifold

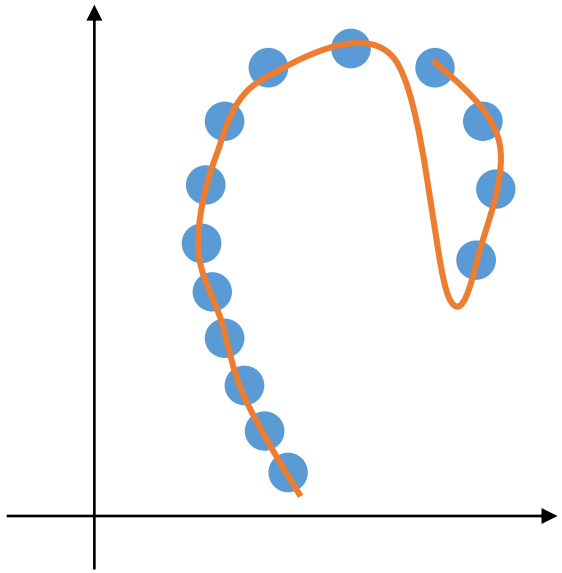


Autoencoders sometimes learn the wrong manifold

1. Overfitting

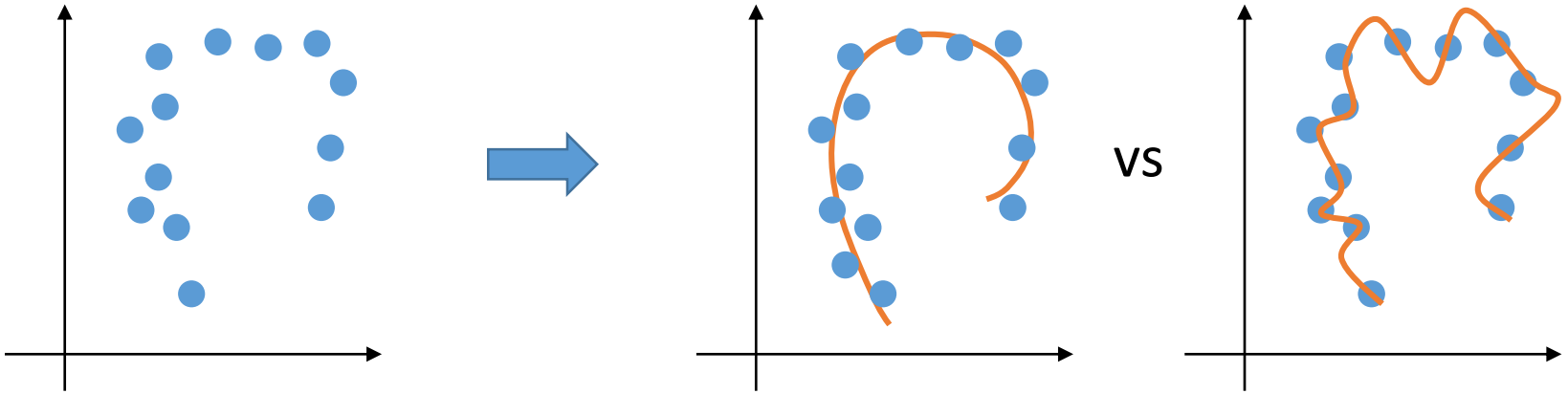


2. Wrong local connectivity

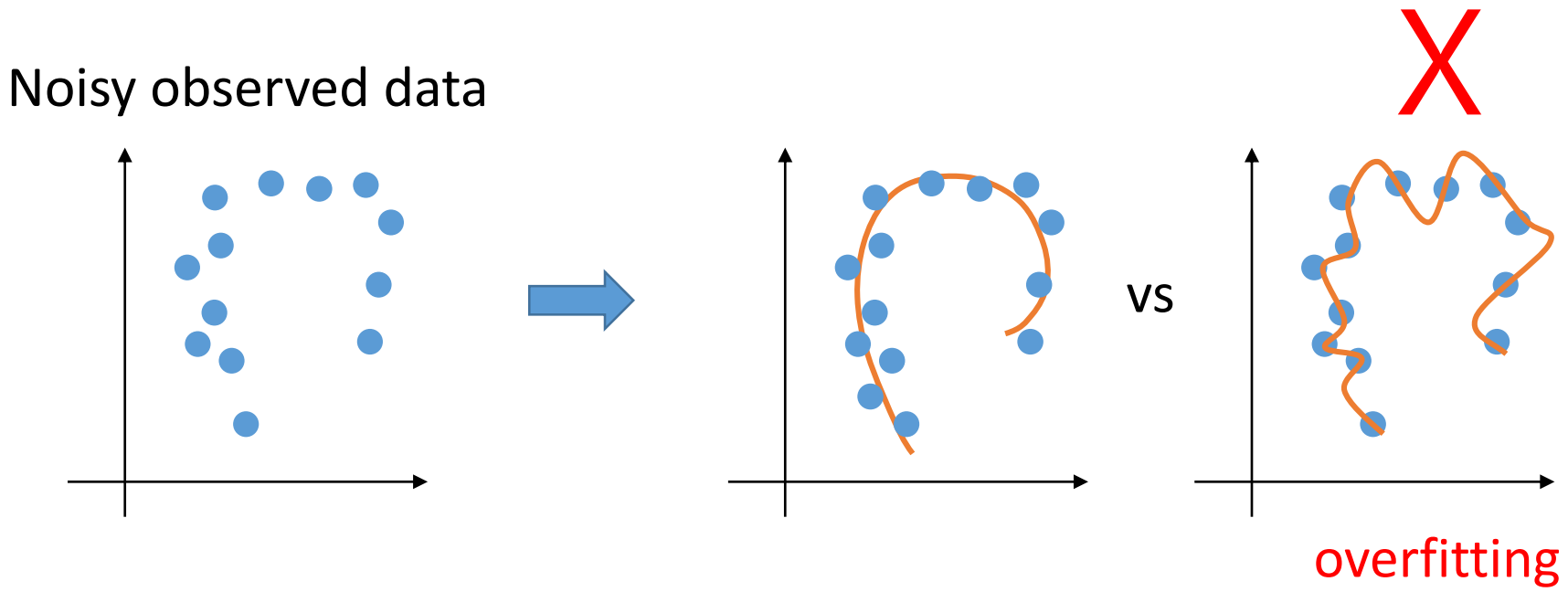


1. Learned manifold can overfit the training data

Noisy observed data

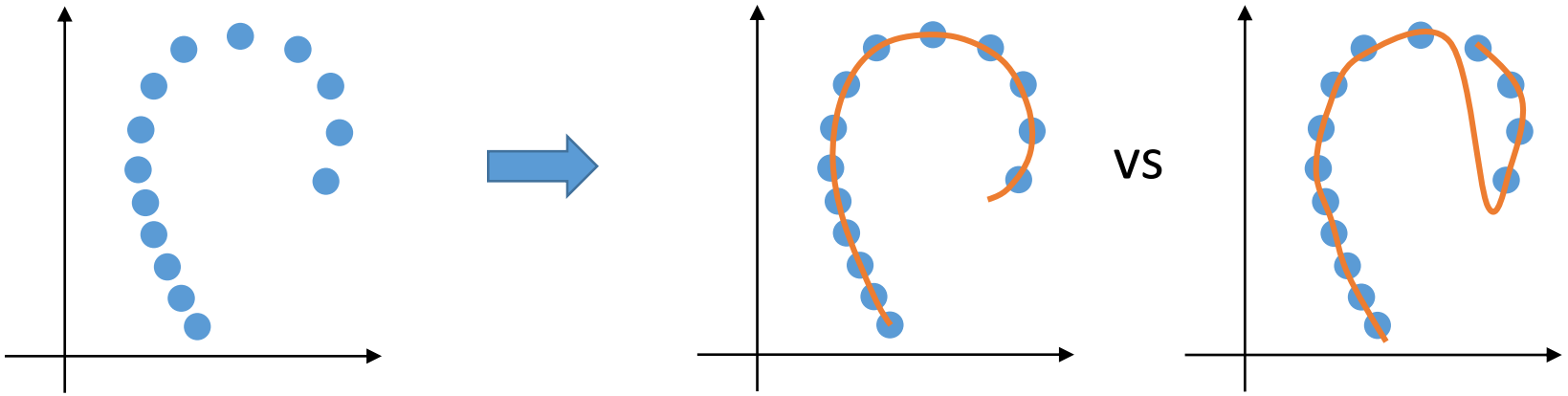


1. Learned manifold can overfit the training data

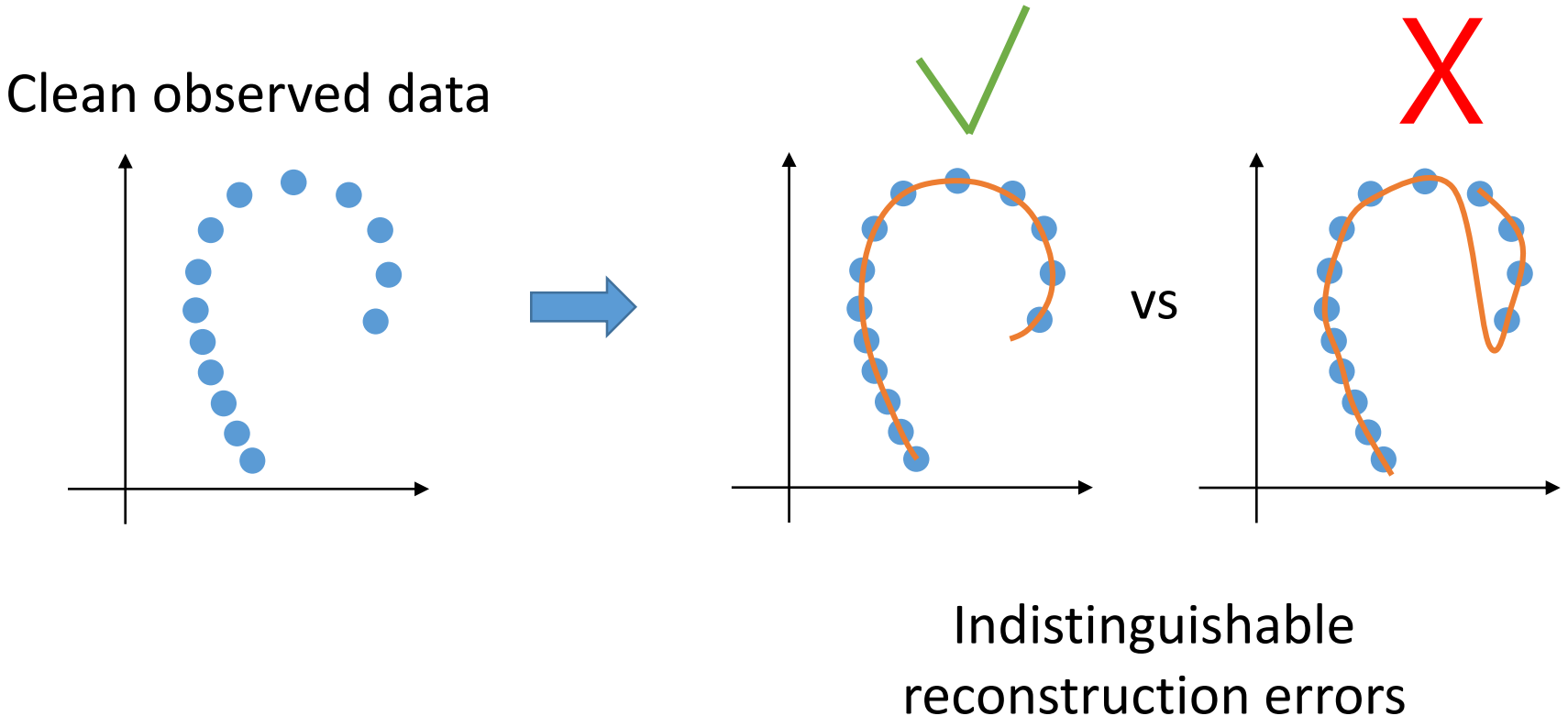


2. Learned manifold can have the wrong local connectivity

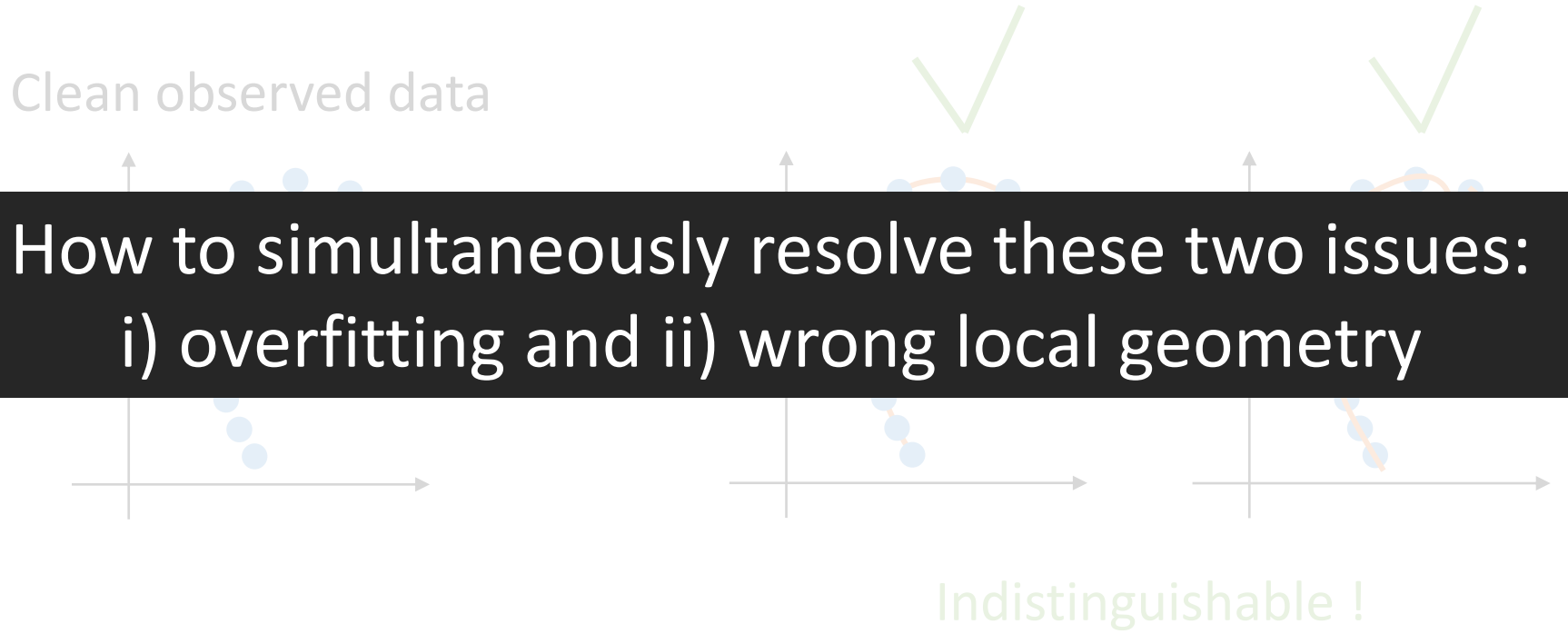
Clean observed data



2. Learned manifold can have the wrong local connectivity

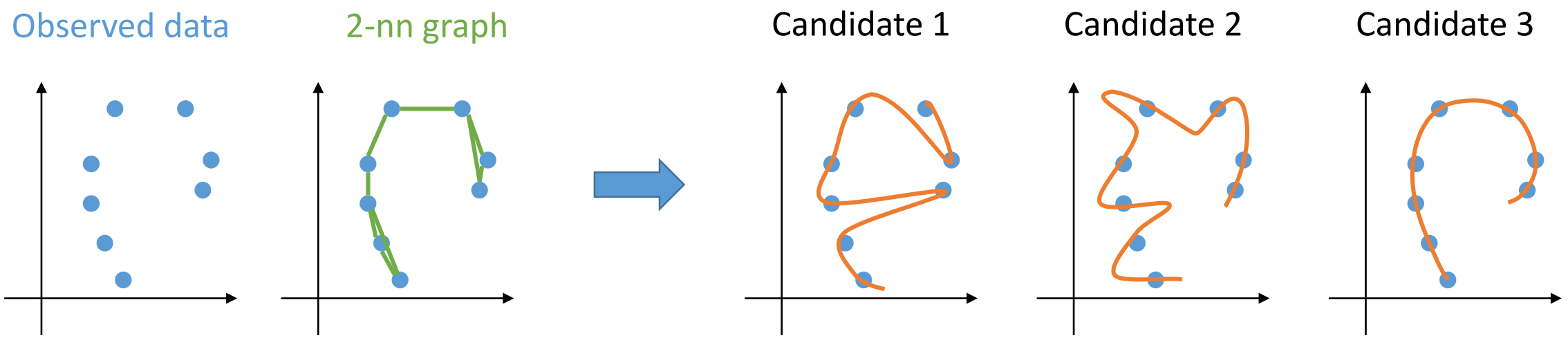


Learned manifold can have the wrong local geometry

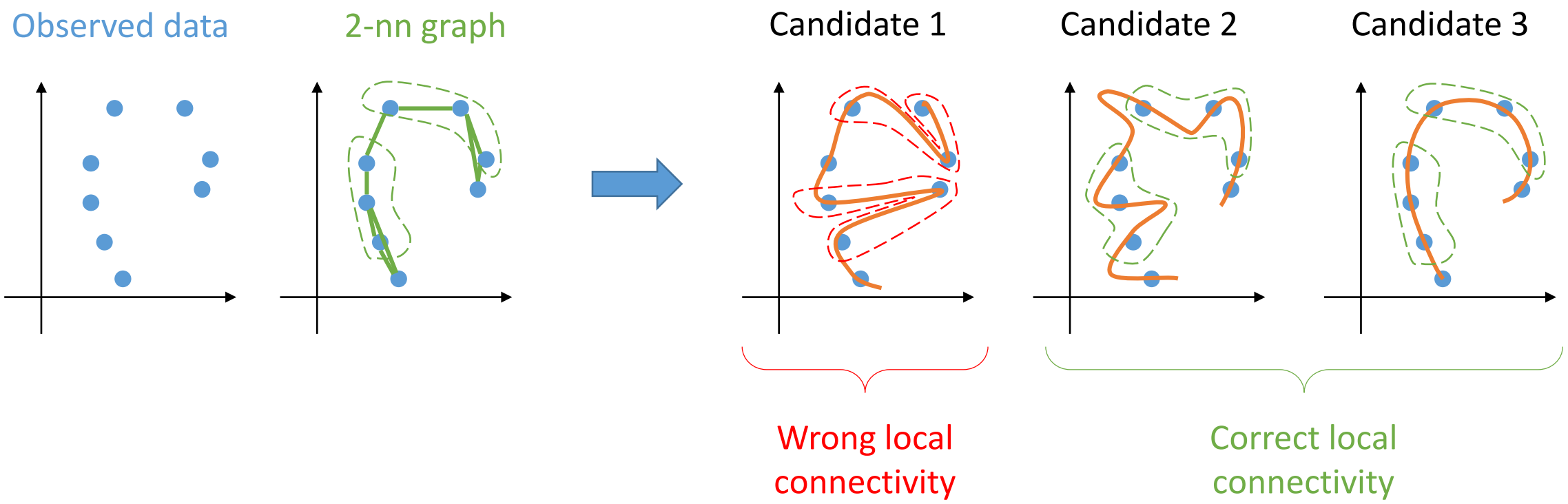


How to simultaneously resolve these two issues:
i) overfitting and ii) wrong local geometry

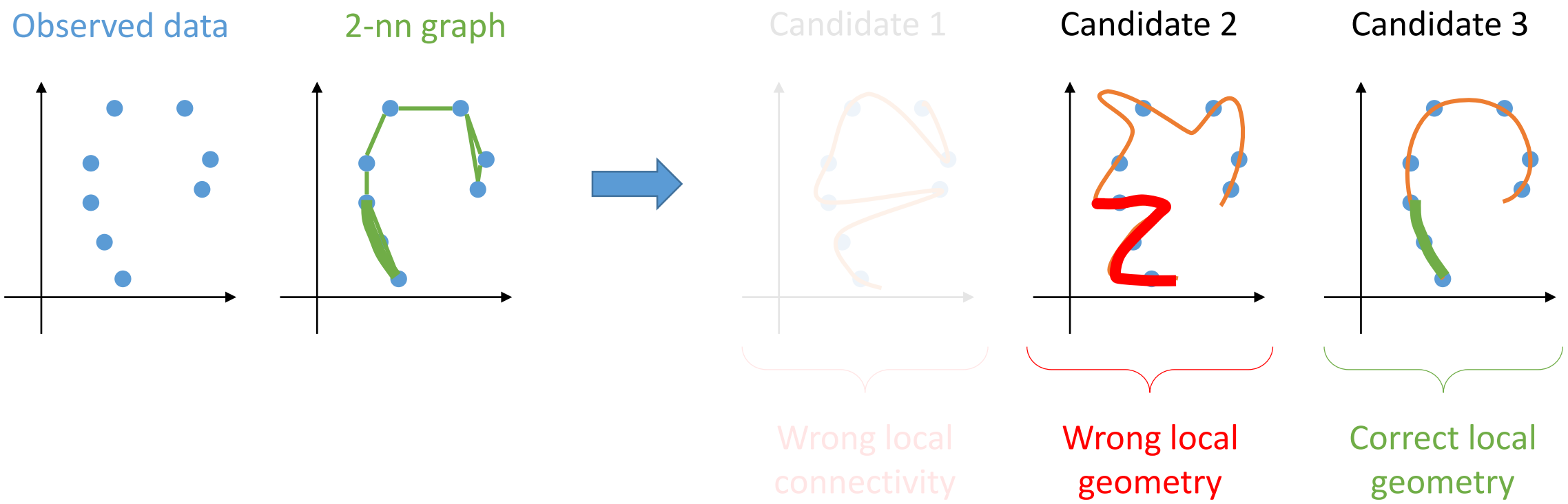
Key Insight: Use the Neighborhood Graph



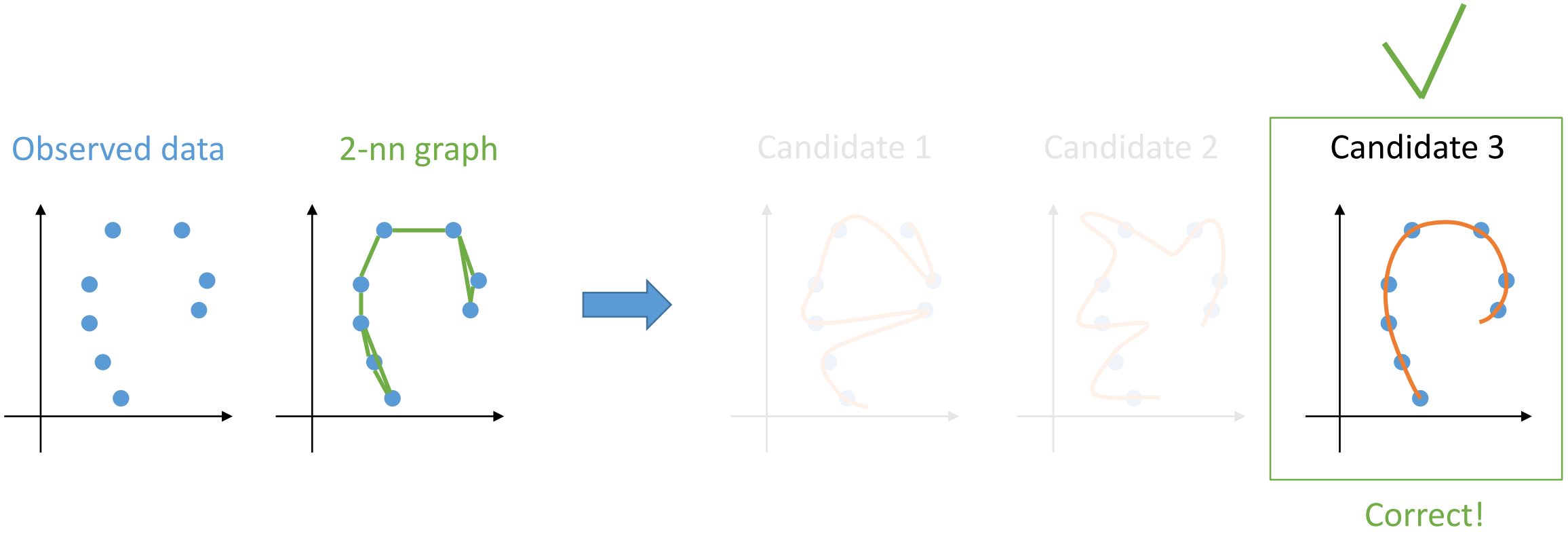
Key Insight: Use the Neighborhood Graph



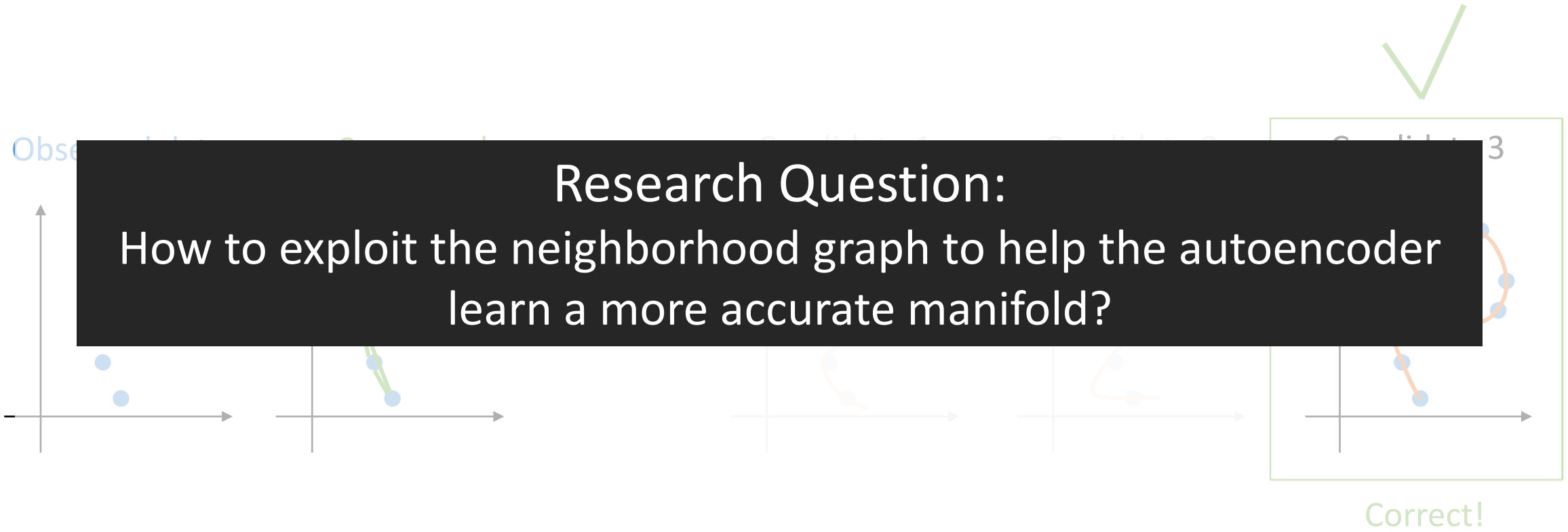
Key Insight: Use the Neighborhood Graph



Key Insight: Use the Neighborhood Graph



Key Insight: Use the Neighborhood Graph



Research Question:

How to exploit the neighborhood graph to help the autoencoder learn a more accurate manifold?

Existing Graph-Based Autoencoders

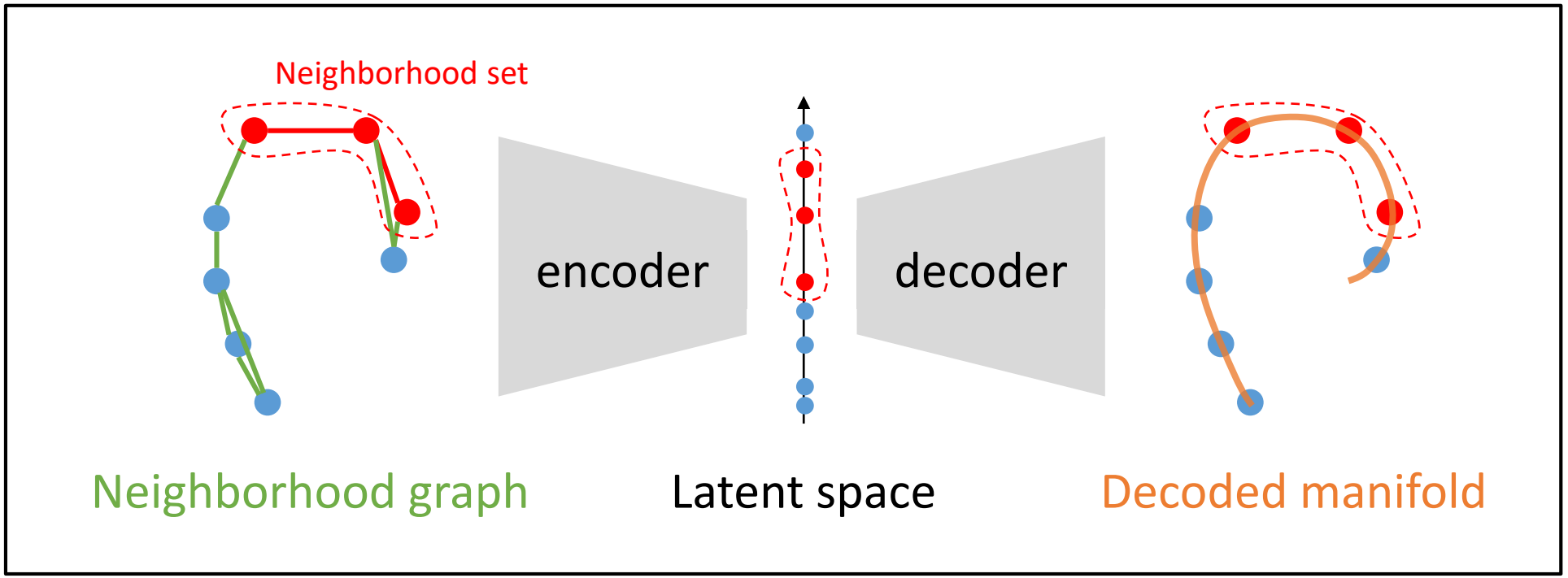
- Topological Autoencoder (M. Moor *et al*, ICML 2020)
- Witness Autoencoder (S. Schoenenberger *et al*, NeurIPS 2020 Workshop)
- Geometry Regularized Autoencoder (Andres F. Duque *et al*, ICBDB 2020)
- Structure-Preserving Autoencoder (Xingyu Chen *et al*, TNNLS 2021)

Existing Graph-Based Autoencoders

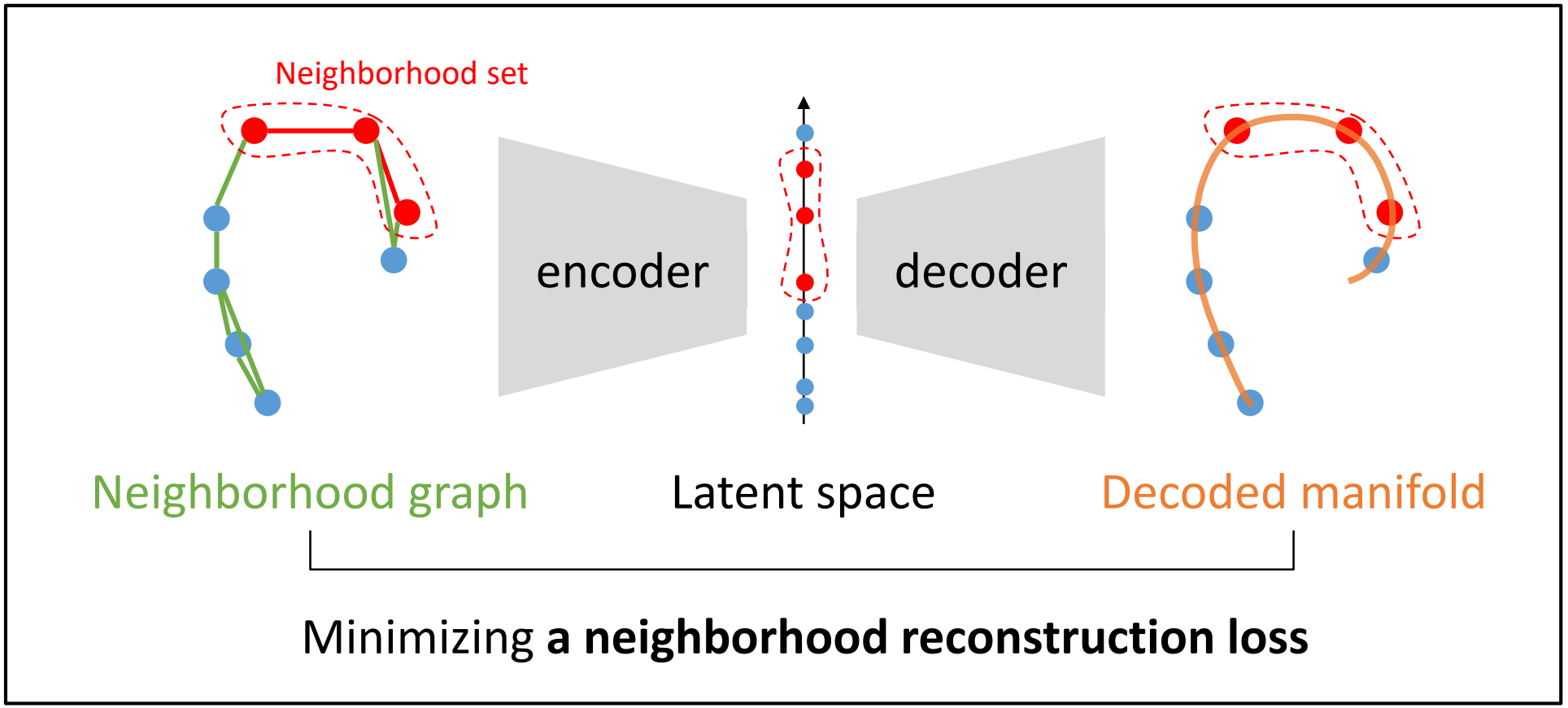
- Topological Autoencoder (M. Moor *et al*, ICML 2020)
- Witness Autoencoder (S. Schoenenberger *et al*, NeurIPS 2020 Workshop)
- Geometry Regularized Autoencoder (Andres F. Duque *et al*, ICBDB 2020)
- Structure-Preserving Autoencoder (Xingyu Chen *et al*, TNNLS 2021)

The above methods focus primarily on regularization of the latent space distribution: **the regularization terms contain the encoder function only**

Neighborhood Reconstructing Autoencoder (NRAE)



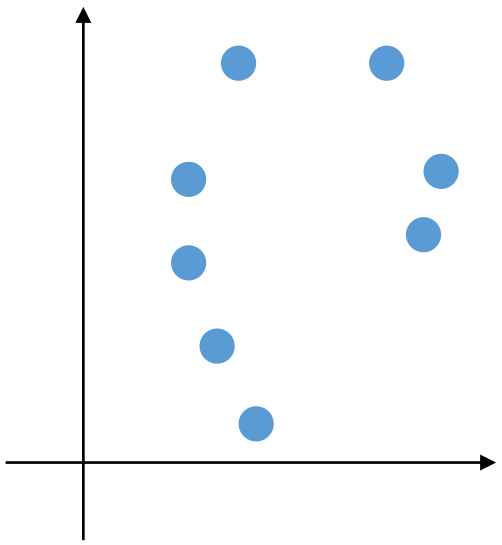
Neighborhood Reconstructing Autoencoder (NRAE)



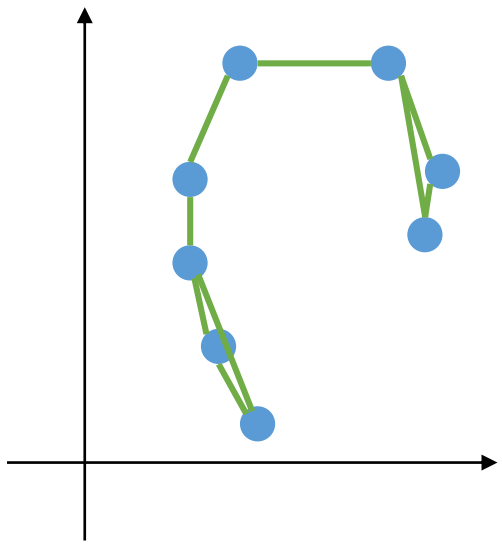
Definition) Neighborhood Reconstruction Loss

First construct a neighborhood graph

Observed data



2-nn graph

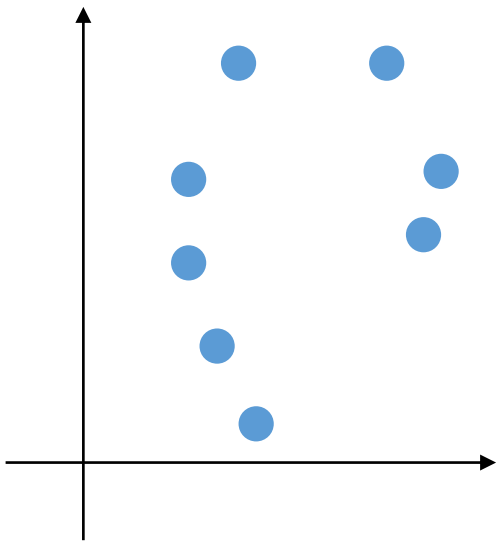


Definition) Neighborhood Reconstruction Loss

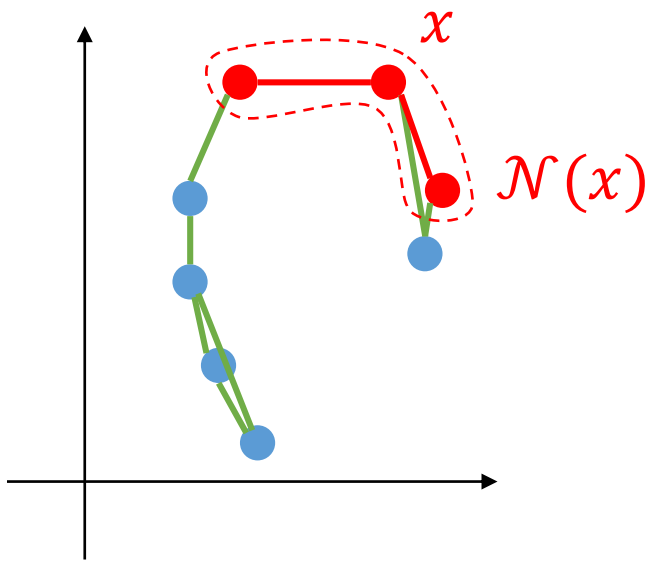
First construct a neighborhood graph

Denote the neighborhood set of x by $\mathcal{N}(x)$

Observed data



2-nn graph



Definition) Neighborhood Reconstruction Loss

- ***The neighborhood point reconstruction*** at $x' \in \mathcal{N}(x)$ is a local quadratic approximation of the composition $F_{\theta,\phi}(x') := f_\theta \circ g_\phi(x')$ with respect to x, i.e., for $x = (x^1, \dots, x^D)$,

$$\tilde{F}_{\theta,\phi}(x'; x) := F_{\theta,\phi}(x) + \sum_i \frac{\partial F_{\theta,\phi}}{\partial x^i}(x)(x' - x)^i + \frac{1}{2}\sum_{i,j} \frac{\partial^2 F_{\theta,\phi}}{\partial x^i \partial x^j}(x)(x' - x)^i (x' - x)^j .$$

- A **linear approximation** can also be used in place of a quadratic approximation.

Definition) Neighborhood Reconstruction Loss

- ***The neighborhood point reconstruction*** at $x' \in \mathcal{N}(x)$ is a local quadratic approximation of the composition $F_{\theta,\phi}(x') := f_\theta \circ g_\phi(x')$ with respect to x, i.e., for $x = (x^1, \dots, x^D)$,

$$\tilde{F}_{\theta,\phi}(x'; x) := F_{\theta,\phi}(x) + \sum_i \frac{\partial F_{\theta,\phi}}{\partial x^i}(x)(x' - x)^i + \frac{1}{2}\sum_{i,j} \frac{\partial^2 F_{\theta,\phi}}{\partial x^i \partial x^j}(x)(x' - x)^i (x' - x)^j .$$

- ***The neighborhood reconstruction loss*** for $\mathcal{N}(x)$ is the sum of the neighborhood point reconstruction errors:

$$\sum_{x' \in \mathcal{N}(x)} ||x' - \tilde{F}_{\theta,\phi}(x'; x)||^2 .$$

Definition) Neighborhood Reconstruction Loss

- ***The neighborhood point reconstruction*** at $x' \in \mathcal{N}(x)$ is a local quadratic approximation of the composition $F_{\theta,\phi}(x') := f_\theta \circ g_\phi(x')$ with respect to x, i.e., for $x = (x^1, \dots, x^D)$,

$$\tilde{F}_{\theta,\phi}(x'; x) := F_{\theta,\phi}(x) + \sum_i \frac{\partial F_{\theta,\phi}}{\partial x^i}(x)(x' - x)^i + \frac{1}{2}\sum_{i,j} \frac{\partial^2 F_{\theta,\phi}}{\partial x^i \partial x^j}(x)(x' - x)^i (x' - x)^j .$$

- ***The neighborhood reconstruction loss*** for $\mathcal{N}(x)$ is the sum of the neighborhood point reconstruction errors:

$$\sum_{x' \in \mathcal{N}(x)} ||x' - \tilde{F}_{\theta,\phi}(x'; x)||^2 .$$

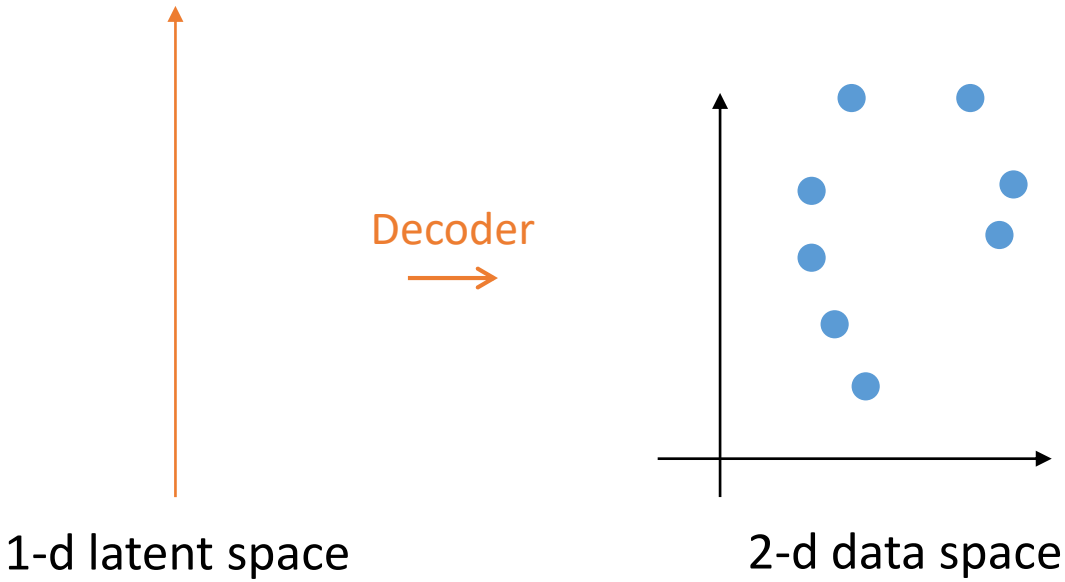
- The neighborhood reconstruction loss enforces the **local smoothness of the autoencoder,** where "**local**" is specified by the neighborhood graph.

The Decoder is More Important

- We find that approximating the decoder, i.e., enforcing the smoothness of the decoder, plays a key role in learning the correct manifold.

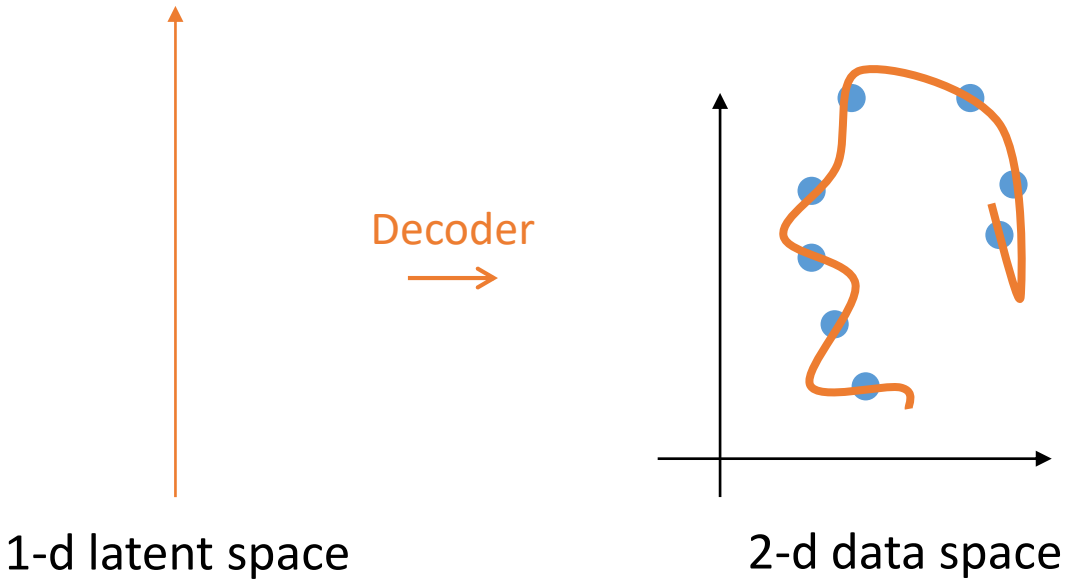
The Decoder is More Important

- We find that approximating the decoder, i.e., enforcing the smoothness of the decoder, plays a key role in learning the correct manifold.



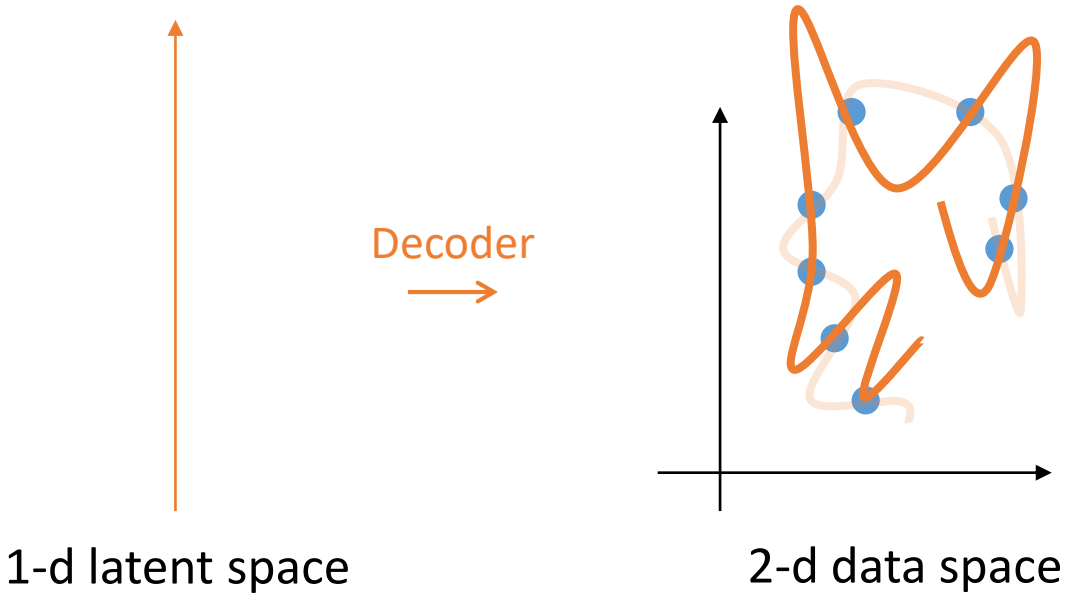
The Decoder is More Important

- We find that approximating the decoder, i.e., enforcing the smoothness of the decoder, plays a key role in learning the correct manifold.



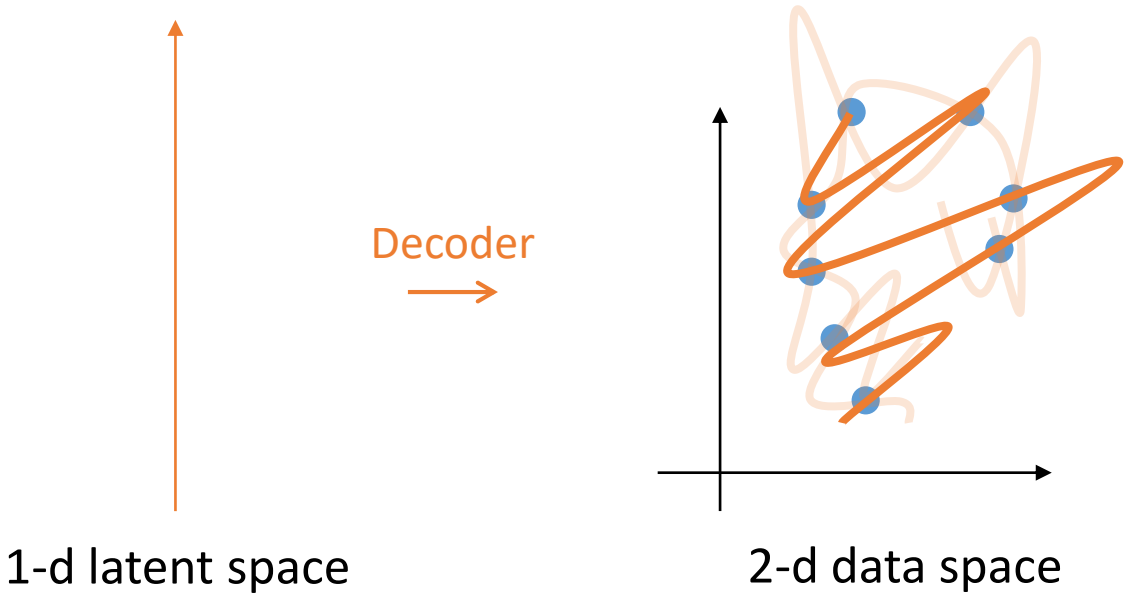
The Decoder is More Important

- We find that approximating the decoder, i.e., enforcing the smoothness of the decoder, plays a key role in learning the correct manifold.



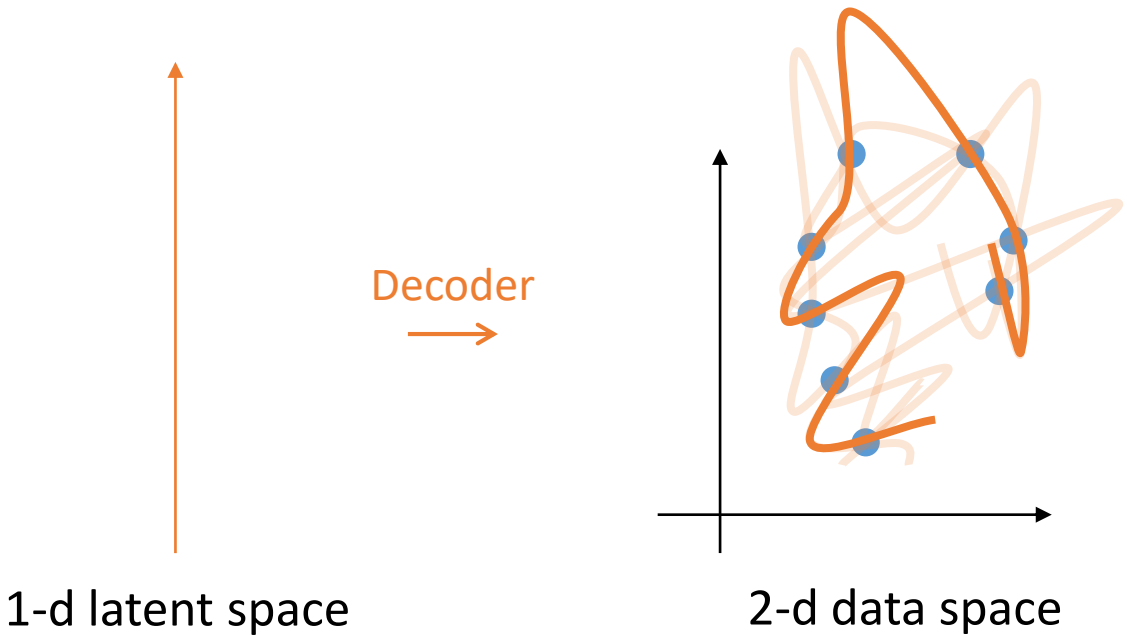
The Decoder is More Important

- We find that approximating the decoder, i.e., enforcing the smoothness of the decoder, plays a key role in learning the correct manifold.



The Decoder is More Important

- We find that approximating the decoder, i.e., enforcing the smoothness of the decoder, plays a key role in learning the correct manifold.



Definition) Neighborhood Reconstruction Loss

- We find that approximating the decoder part, i.e., enforcing the smoothness of the decoder, plays a key role in learning the correct manifold.

- We approximate the decoder part only, i.e., the neighborhood reconstruction loss for $\mathcal{N}(x)$ is

$$\sum_{x' \in \mathcal{N}(x)} ||x' - \tilde{f}_\theta\big(g_\phi(x'); g_\phi(x)\big)||^2 ,$$

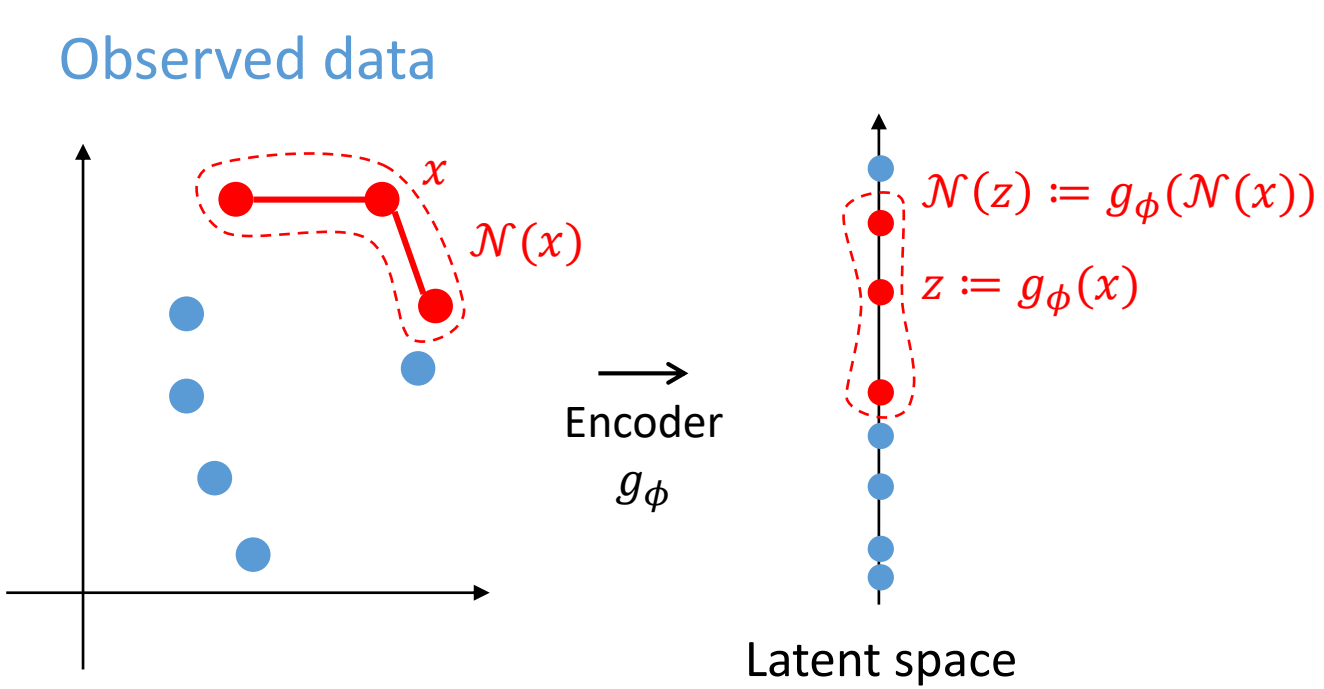
where $\tilde{f}_\theta\,(z'; z)$ is a local quadratic approximation of $f_\theta(z')$ with respect to z.

Geometric Interpretation of the Neighborhood Reconstruction Loss

$$\sum_{x' \in \mathcal{N}(x)} ||x' - \tilde{f}_\theta\big(g_\phi(x'); g_\phi(x)\big)||^2$$

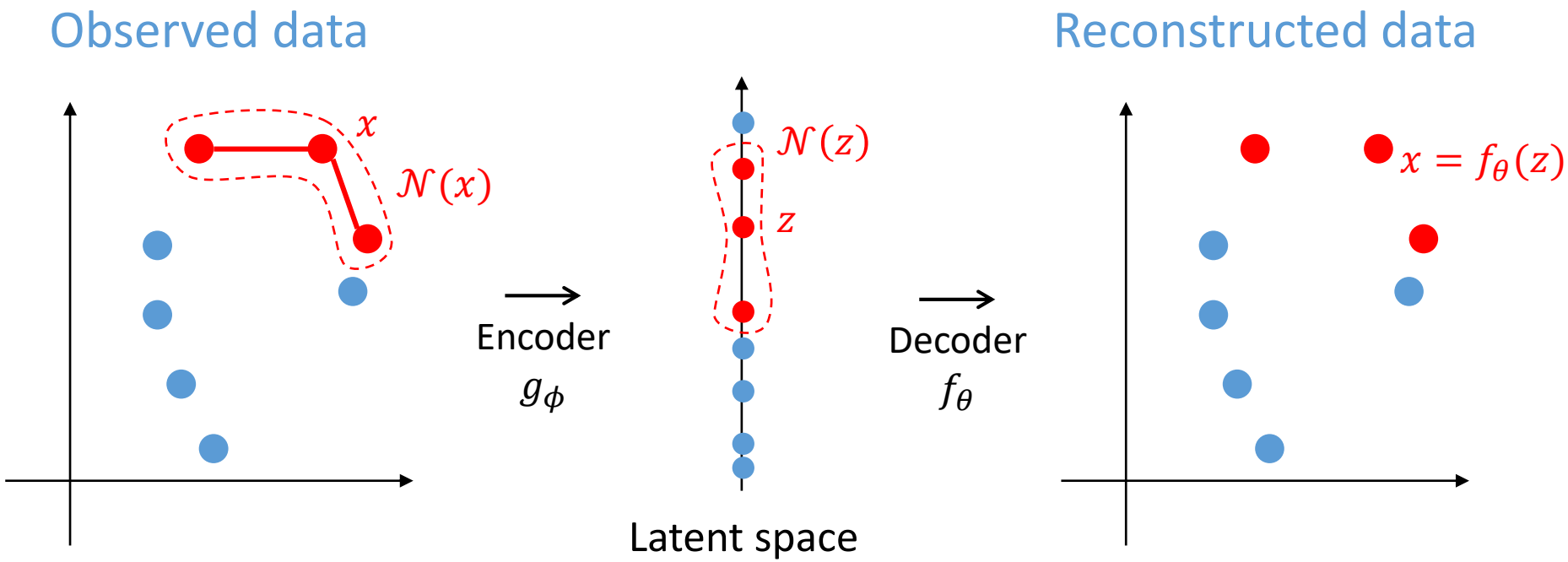
Geometric Interpretation of the Neighborhood Reconstruction Loss

$$\sum_{x' \in \mathcal{N}(x)} ||x' - \tilde{f}_\theta\big(g_\phi(x'); g_\phi(x)\big)||^2$$



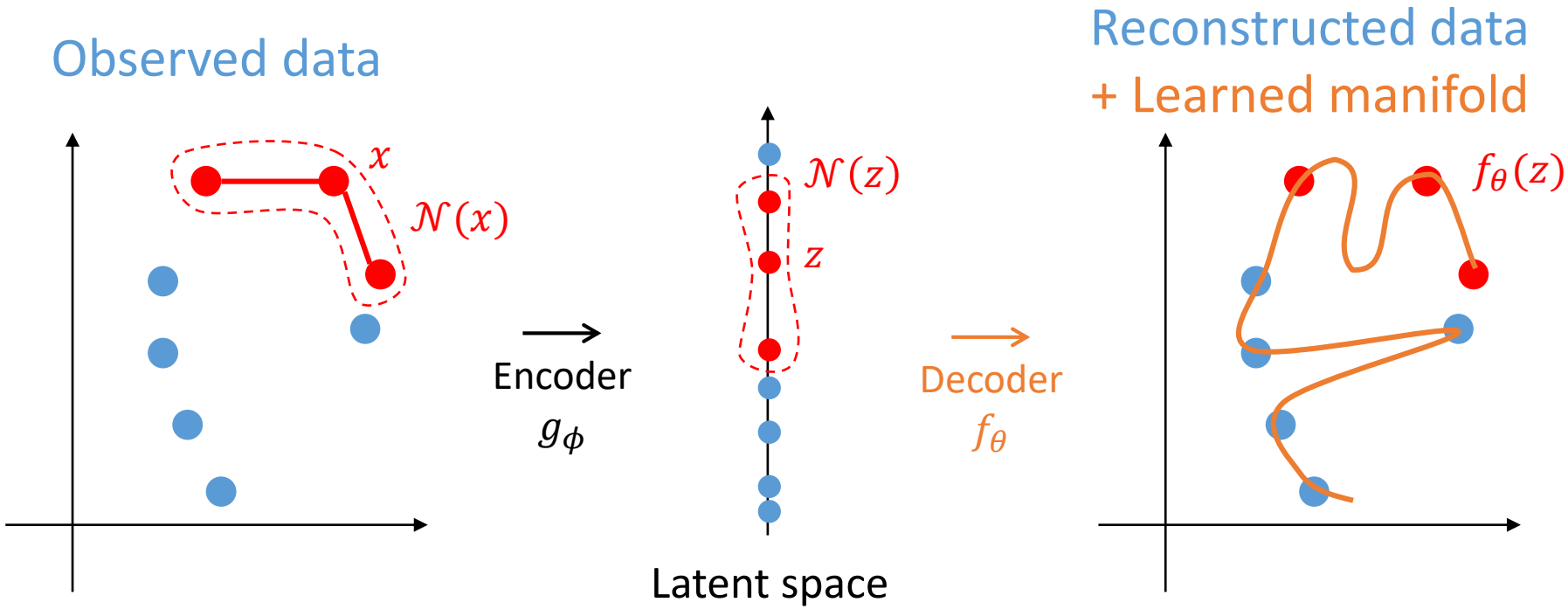
Geometric Interpretation of the Neighborhood Reconstruction Loss

$$\sum_{x' \in \mathcal{N}(x)} ||x' - \tilde{f}_\theta\big(g_\phi(x'); g_\phi(x)\big)||^2$$



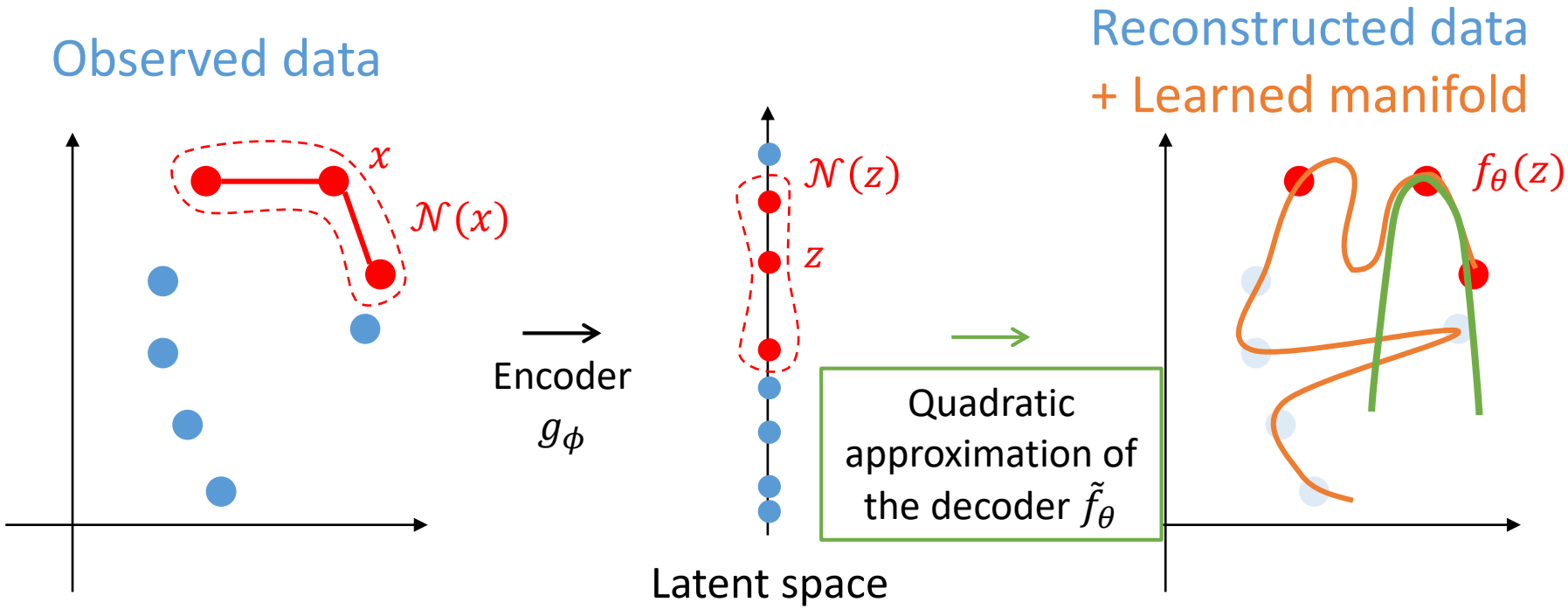
Geometric Interpretation of the Neighborhood Reconstruction Loss

$$\sum_{x' \in \mathcal{N}(x)} ||x' - \tilde{f}_\theta\big(g_\phi(x'); g_\phi(x)\big)||^2$$



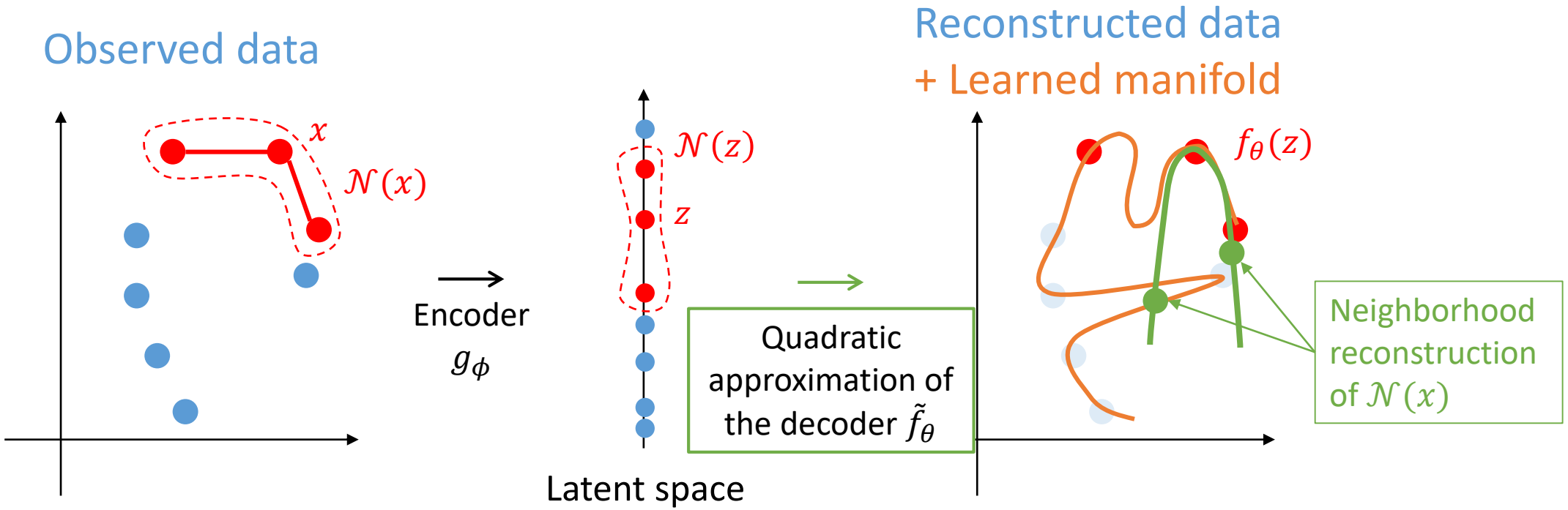
Geometric Interpretation of the Neighborhood Reconstruction Loss

$$\sum_{x' \in \mathcal{N}(x)} ||x' - \tilde{f}_\theta\big(g_\phi(x'); g_\phi(x)\big)||^2$$



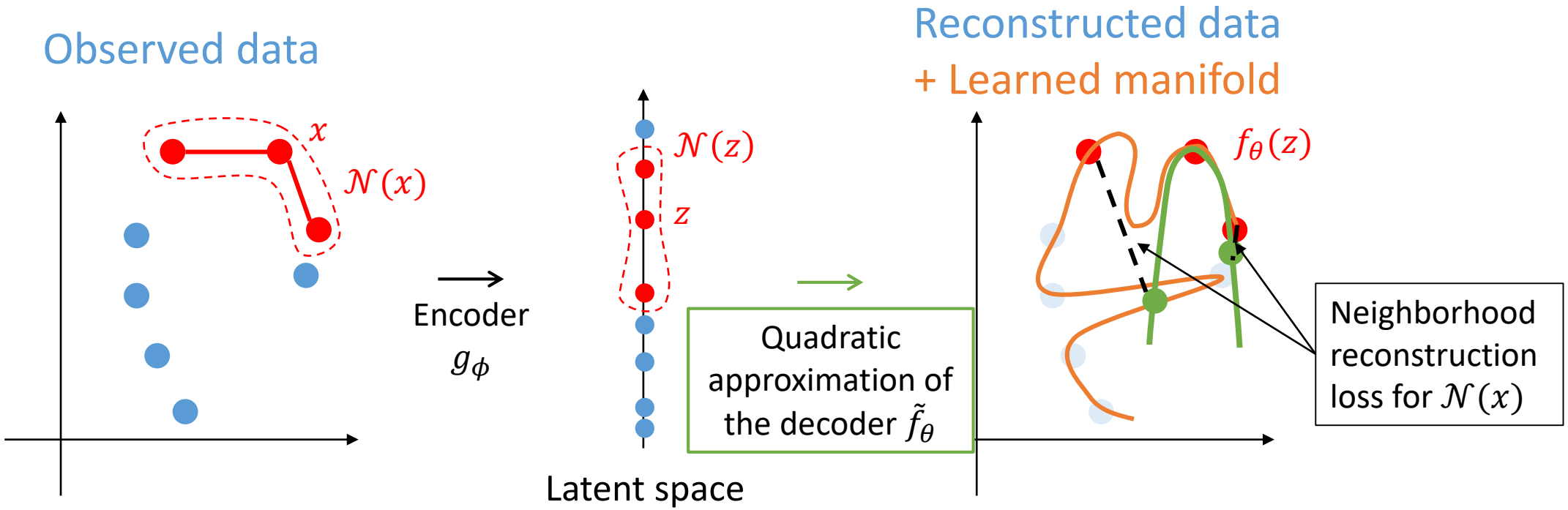
Geometric Interpretation of the Neighborhood Reconstruction Loss

$$\sum_{x' \in \mathcal{N}(x)} ||x' - \tilde{f}_\theta\big(g_\phi(x'); g_\phi(x)\big)||^2$$

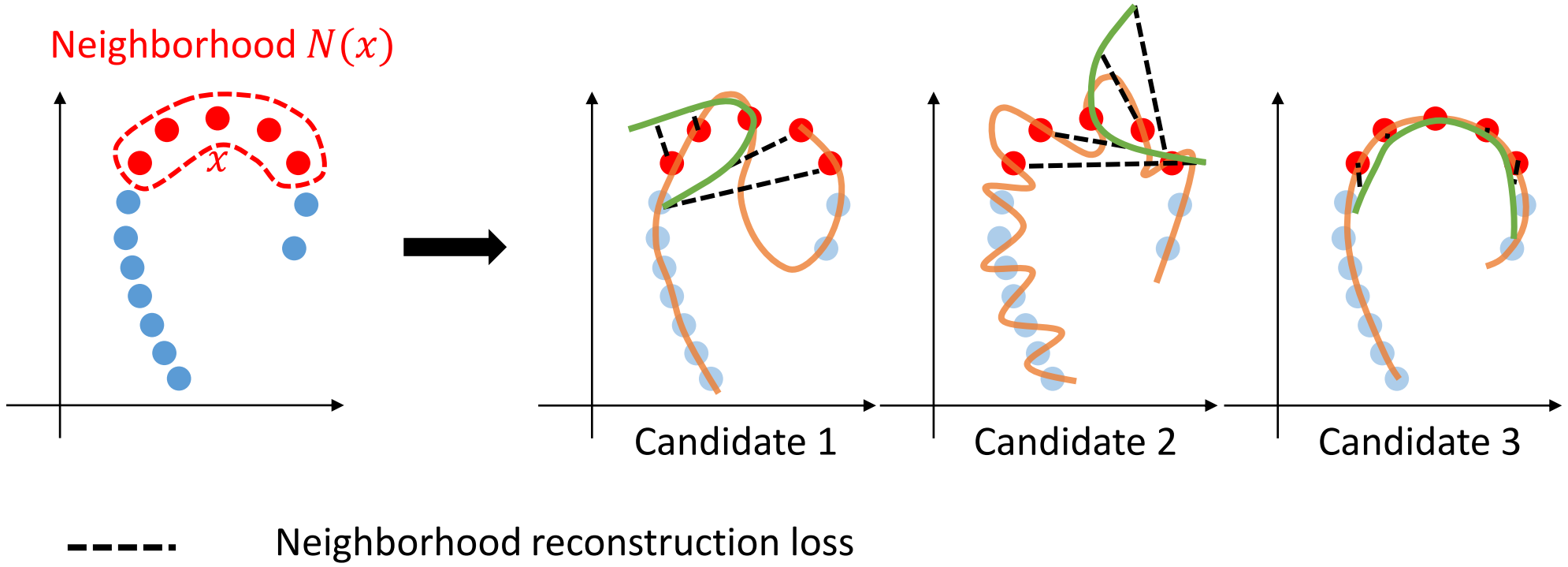


Geometric Interpretation of the Neighborhood Reconstruction Loss

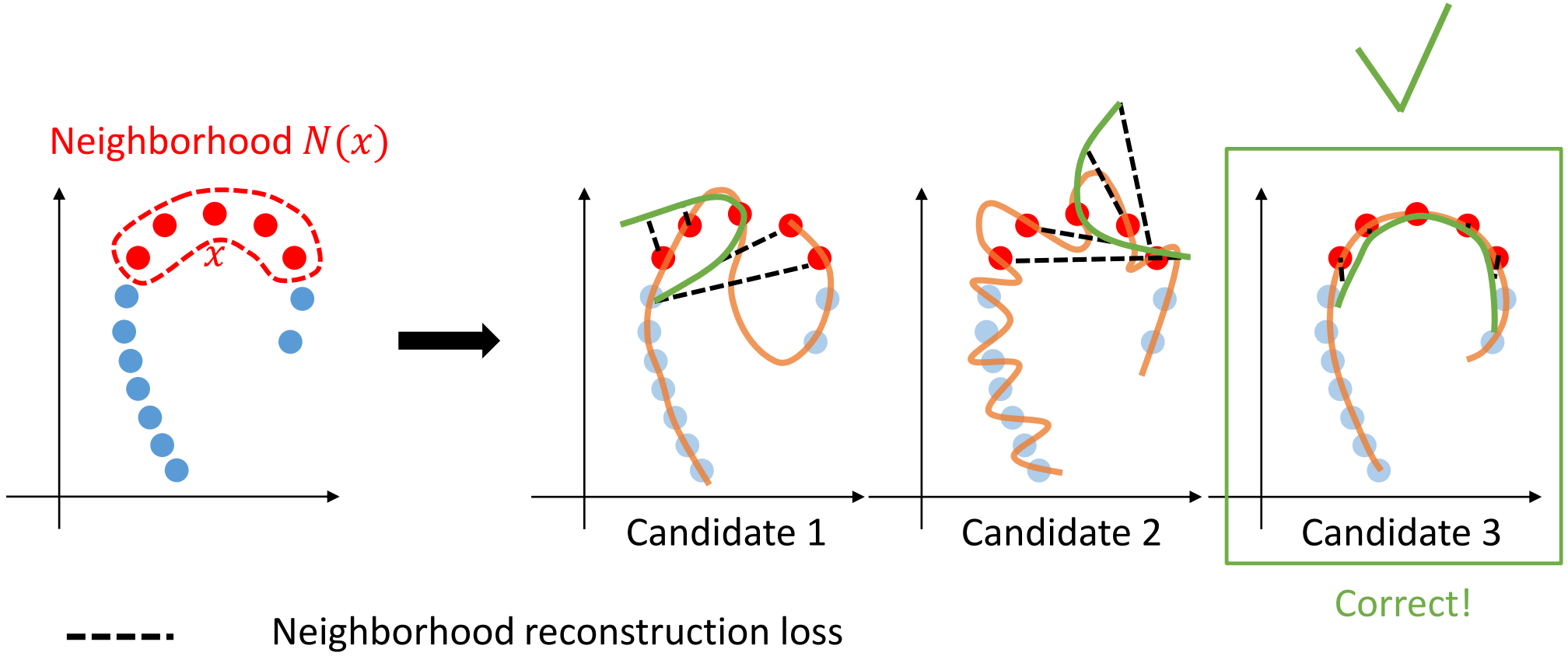
$$\sum_{x' \in \mathcal{N}(x)} ||x' - \tilde{f}_\theta\big(g_\phi(x'); g_\phi(x)\big)||^2$$



Geometric Interpretation of the Neighborhood Reconstruction Loss



Geometric Interpretation of the Neighborhood Reconstruction Loss



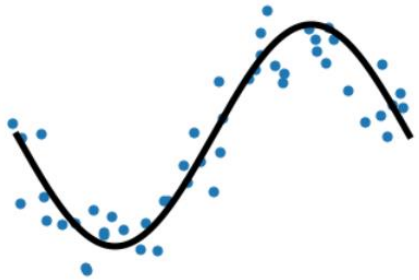
Remarkably Simple Implementation

```
def point_recon_loss(encoder, decoder, x):
  z = ecnoder(x)
  recon_x = decoder(z)
  return norm(x-recon_x)**2

def jvp(f, x, v):
  return (Dot product between Jacobian of "f" at the point "x" and a vector "v")

def neighborhood_recon_loss(encoder, decoder, x, x_n):
  """linear approximation of the decoder is used.
  New arg:
    x_n: a neighborhood point of "x" ("x_n" can be "x" itself)
  """
  z = encoder(x)
  z_n = encoder(x_n)
  recon_x = decoder(x)
  return norm(x_n - recon_x - jvp(decoder, z, v=z_n-z))**2
```

Some Experimental Results

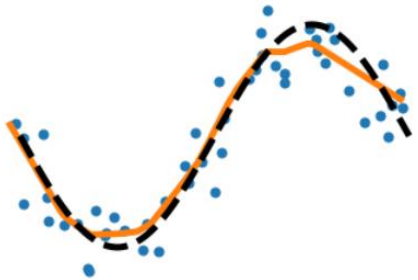
NRAE-L : NRAE with linear approximation of the decoder
NRAE-Q : NRAE with quadratic approximation of the decoder

De-noising Property

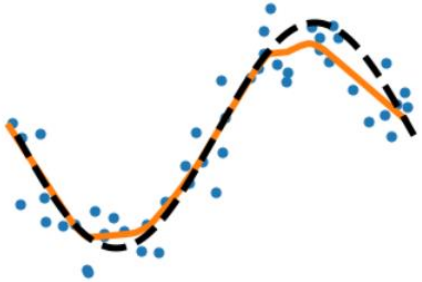


Ground-truth manifold
+ noisy samples

Vanilla AE



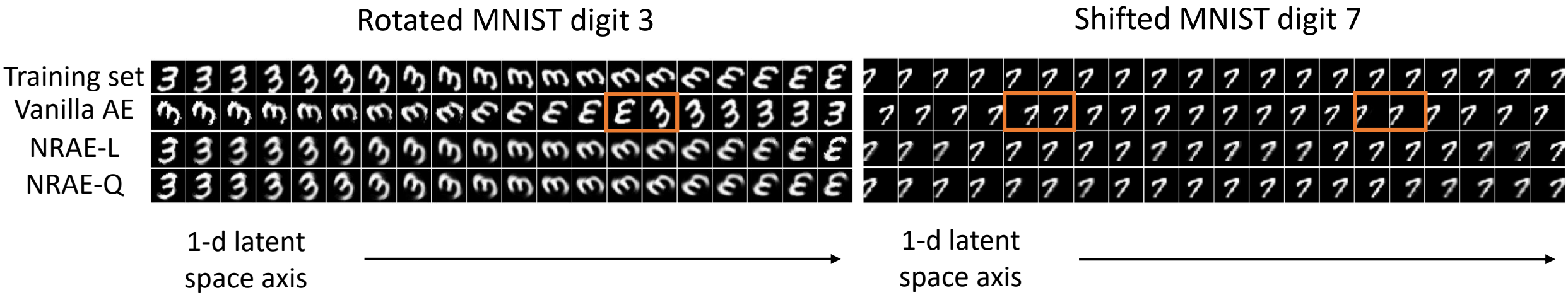
NRAE-L (ours)



NRAE-Q (ours)

Given a set of corrupted observed data, the NRAE (for both linear and quadratic approximations) produces smooth decoded manifolds

Learning the Correct Local Connectivity



NRAE learns the correct order of the set of rotated/shifted images, and can generate continuously varying images (unlike the vanilla AE, where discontinuities are marked by orange boxes).

Manifold Learning Performance

We compare manifold learning performance (measured by the test set reconstruction error) of NRAE-L and NRAE-Q against the following baseline algorithms:

- AE (Rumelhart *et al*., 1986)
- VAE (D. P. Kingma *et al*., ICLR 2014)
- WAE (I. Tolstikhin *et a*l., ICLR 2018)
- DAE (P. Vincent *et al*., JMLR 2010)
- CAE (S. Rifai *et al*., ICML 2011)
- GRAE (Andres F. Duque *et al*., ICBDB 2020)
- SPAE (Xingyu Chen *et al*., TNNLS 2021)

Manifold Learning Performance

Table 1) The test set reconstruction errors of the standard image datasets. The size column indicates the dataset size (small or large). **The best and second-best results are colored red and blue, respectively (lower-the-better).**

Dataset	Size	AE	VAE	WAE	DAE	CAE	GRAE	SPAE	NRAE-L	NRAE-Q
MNIST	S	0.01002	0.01091	0.01009	0.00999	0.00998	0.01004	0.00989	0.00953	0.00968
	L	0.00688	0.00756	0.00690	0.00684	0.00692	0.00696	0.00694	0.00649	0.00683
FMNIST	S	0.01485	0.01652	0.01428	0.01446	0.01319	0.01331	0.01363	0.01289	0.01277
	L	0.01118	0.01235	0.01106	0.01099	0.01052	0.01060	0.01065	0.01060	0.01044
KMNIST	S	0.03267	0.03234	0.03283	0.03280	0.03279	0.03206	0.03268	0.03071	0.03021
	L	0.02844	0.02963	0.02776	0.02814	0.02762	0.02753	0.02732	0.02564	0.02602
Omniglot	S	0.03038	0.03627	0.03078	0.03068	0.02714	0.02967	0.02889	0.02668	0.02631
	L	0.02704	0.03192	0.02728	0.02696	0.02567	0.02648	0.02644	0.02578	0.02539
SVHN	S	0.00320	0.00420	0.00320	0.00369	0.00273	0.00317	0.00307	0.00202	0.00192
	L	0.00174	0.00204	0.00190	0.00177	0.00178	0.00173	0.00175	0.00148	0.00147
CIFAR10	S	0.01466	0.01620	0.01431	0.01427	0.01208	0.01452	0.01504	0.00768	0.00691
	L	0.00960	0.01123	0.00863	0.00900	0.00755	0.00832	0.00898	0.00629	0.00587
CIFAR100	S	0.01465	0.01713	0.01463	0.01484	0.01369	0.01391	0.01477	0.00765	0.00717
	L	0.01015	0.01064	0.00951	0.00862	0.00842	0.00910	0.00912	0.00678	0.00635
CELEBA	S	0.00780	0.00937	0.00830	0.00782	-	0.00814	0.00861	0.00608	0.00747
	L	0.00613	0.00646	0.00630	0.00590	-	0.00595	0.00665	0.00563	0.00565

Conclusion

- We propose the NRAE, a graph-based autoencoder that resolves the overfitting and wrong local connectivity issues.
- The decoder approximation is more important.
- The implementation is remarkably easy and simple.
- Experimental results show that manifold learning performance is significantly improved.

More in the Paper

- Diverse experimental results
- Relation to local polynomial regression
- Convergence to the vanilla autoencoder
- Graph construction and kernel design
- Special case: linear decoder function

…and many more!

Thank you for listening!

Contact: yhlee@robotics.snu.ac.kr
Code: https://github.com/Gabe-YHLee/NRAE-public