데이터의 차원 축소, 근데 시각화를 곁들인

발표자: 신중현

0. Intro

컴퓨터는 세상을 어떻게 바라볼까?

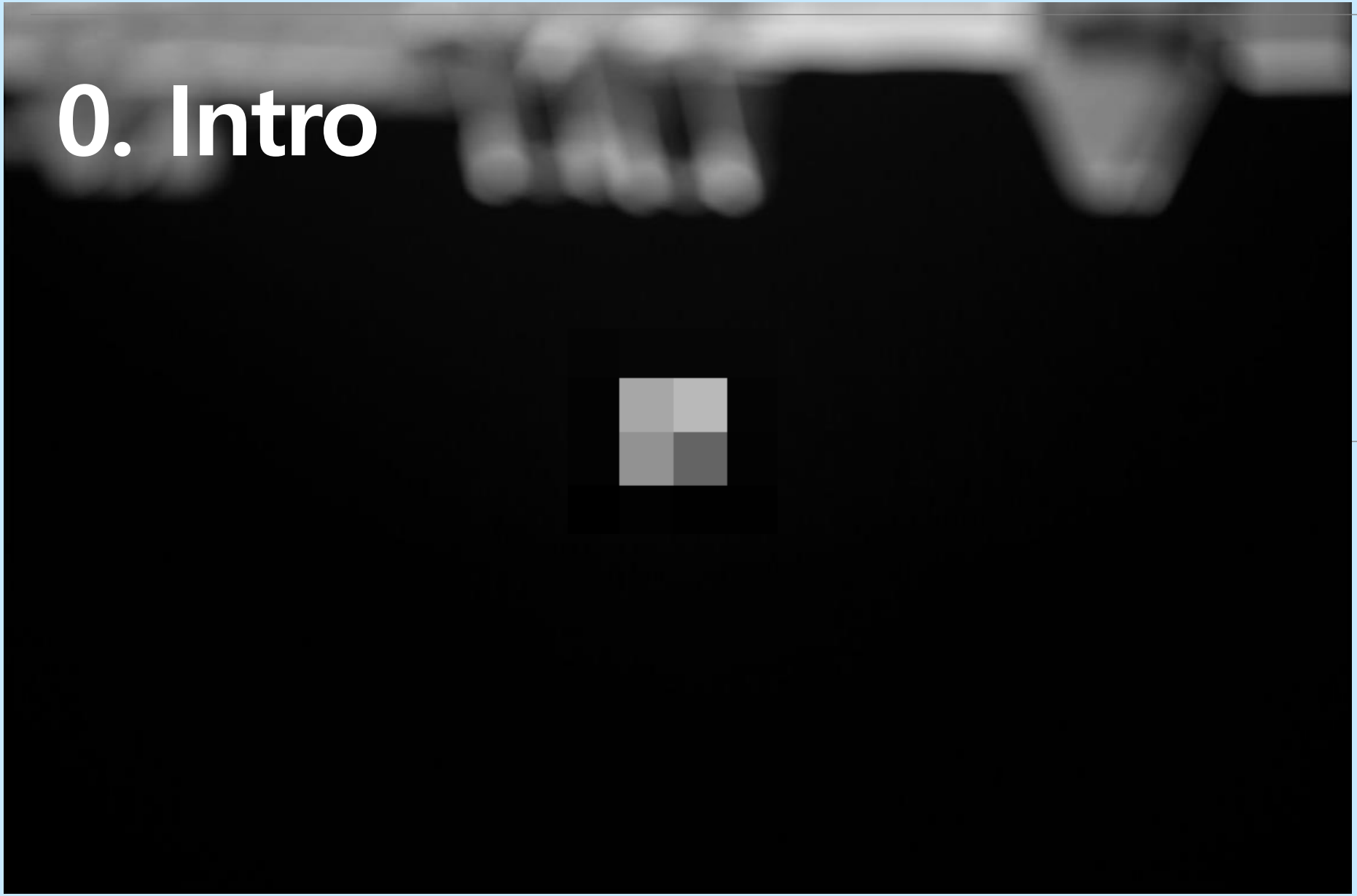
0. Intro

컴퓨터는 세상을 어떻게 바라볼까?

0. Intro

컴퓨터는 세상을 어떻게 바라볼까?

0. Intro

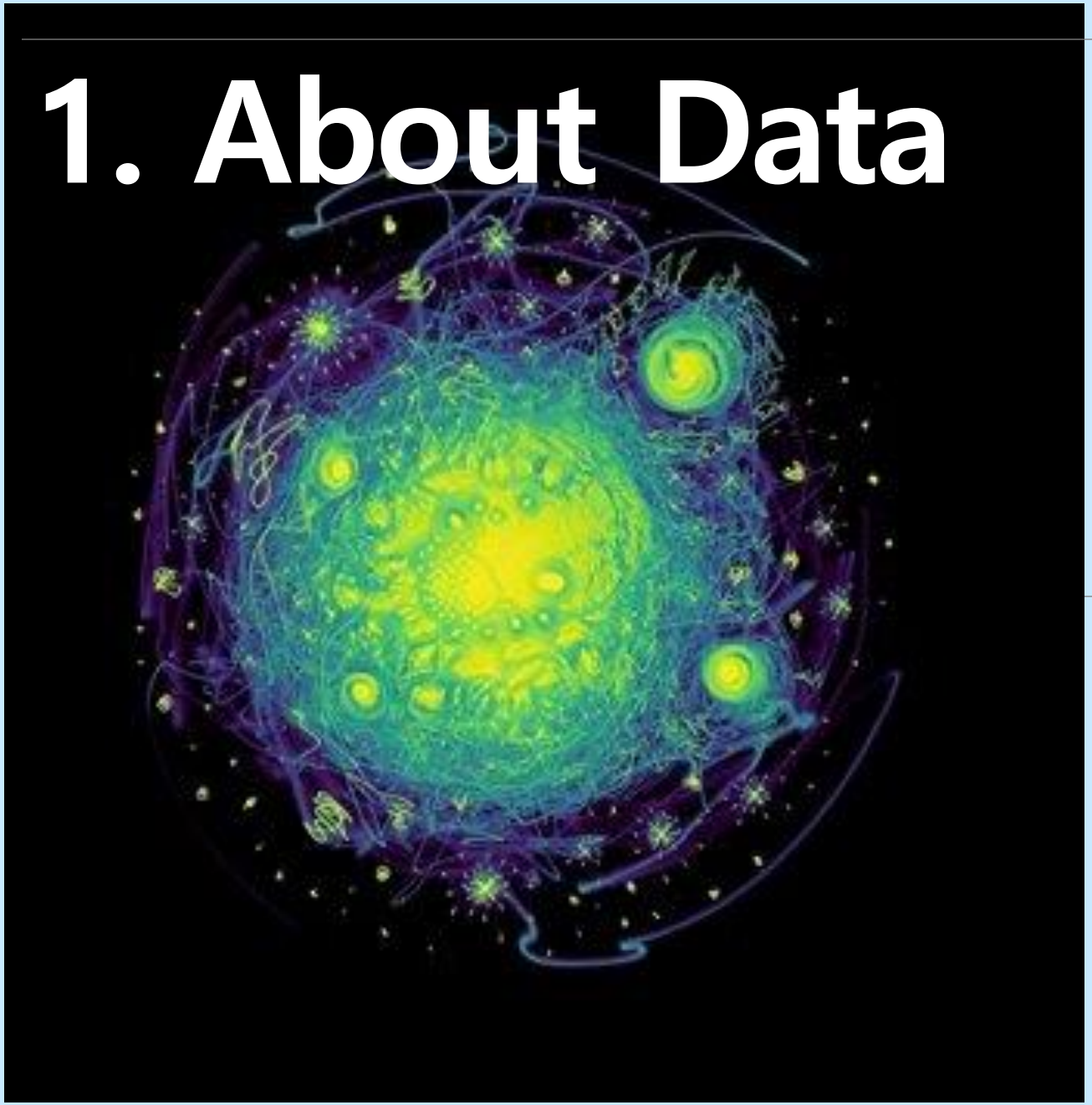


차원 축소란 무엇일까요?

'latent feature '를 찾는 것

INDEX

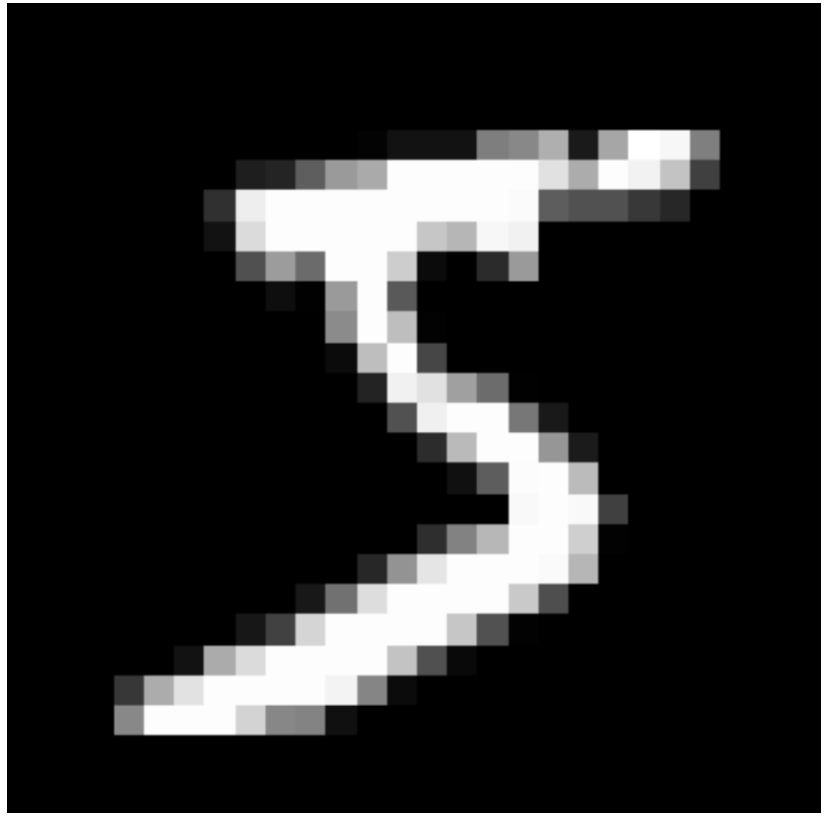
1. About Data



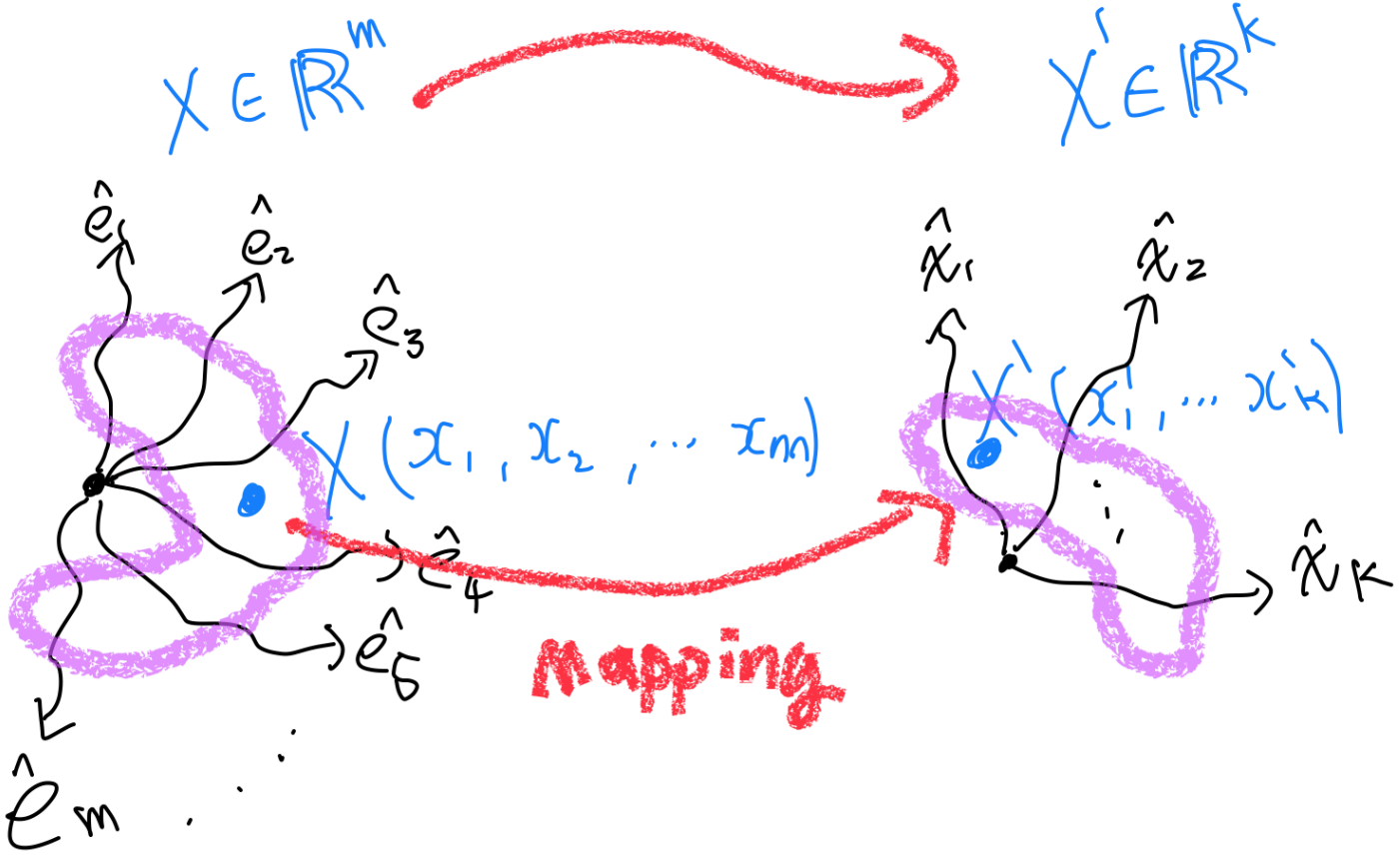
차원이란 무엇일까요?

데이터의 차원이란?

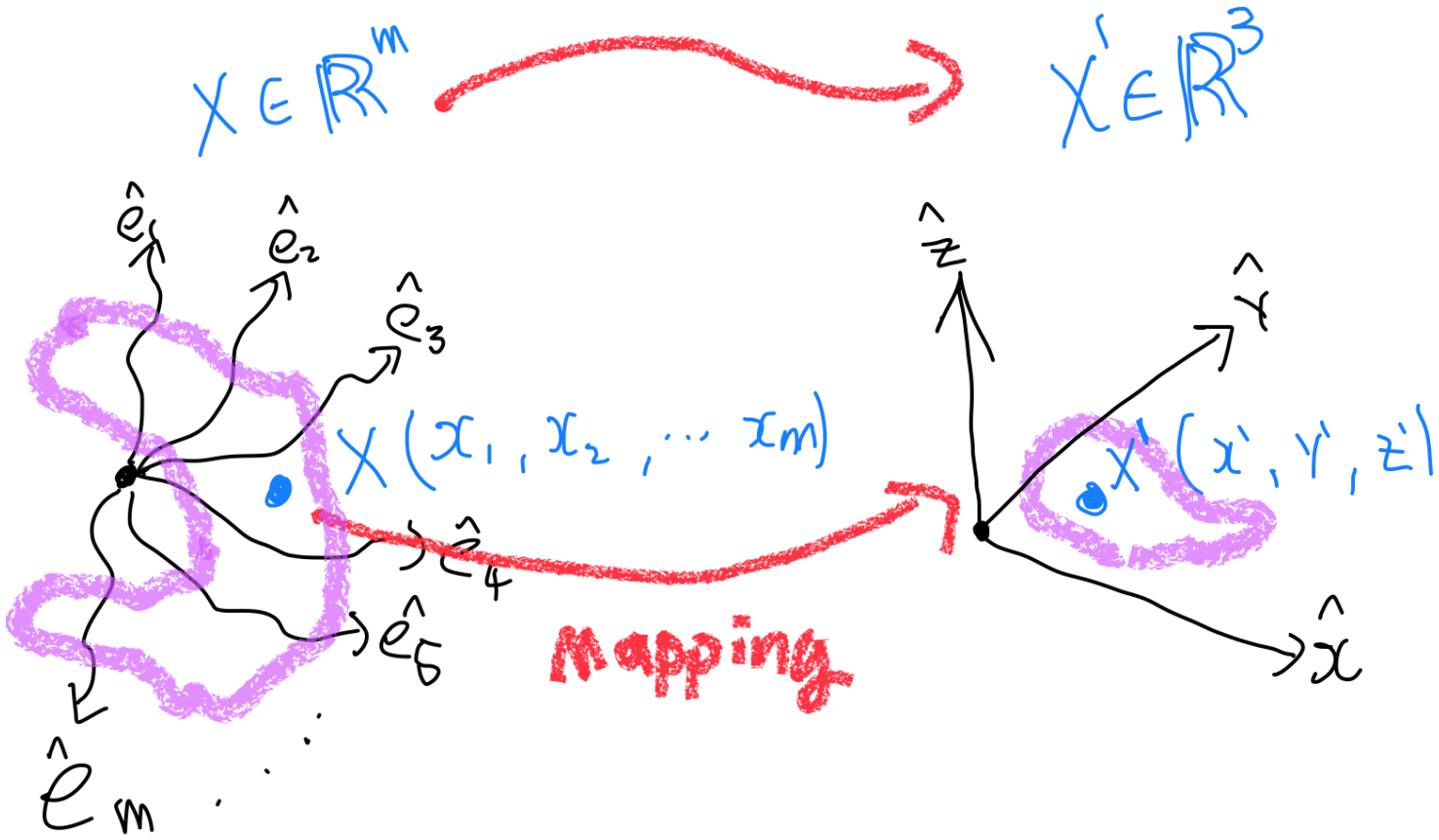
데이터에 대해서 생각해보면



데이터에 대해서 생각해보면

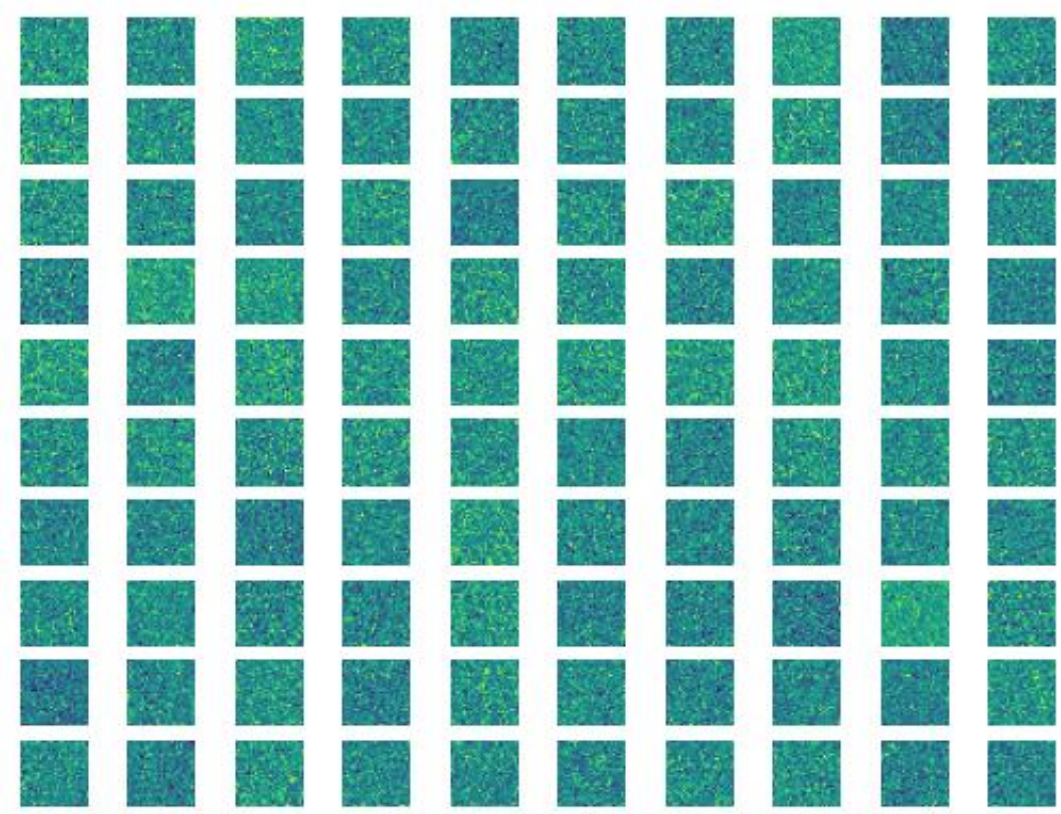


데이터에 대해서 생각해보면

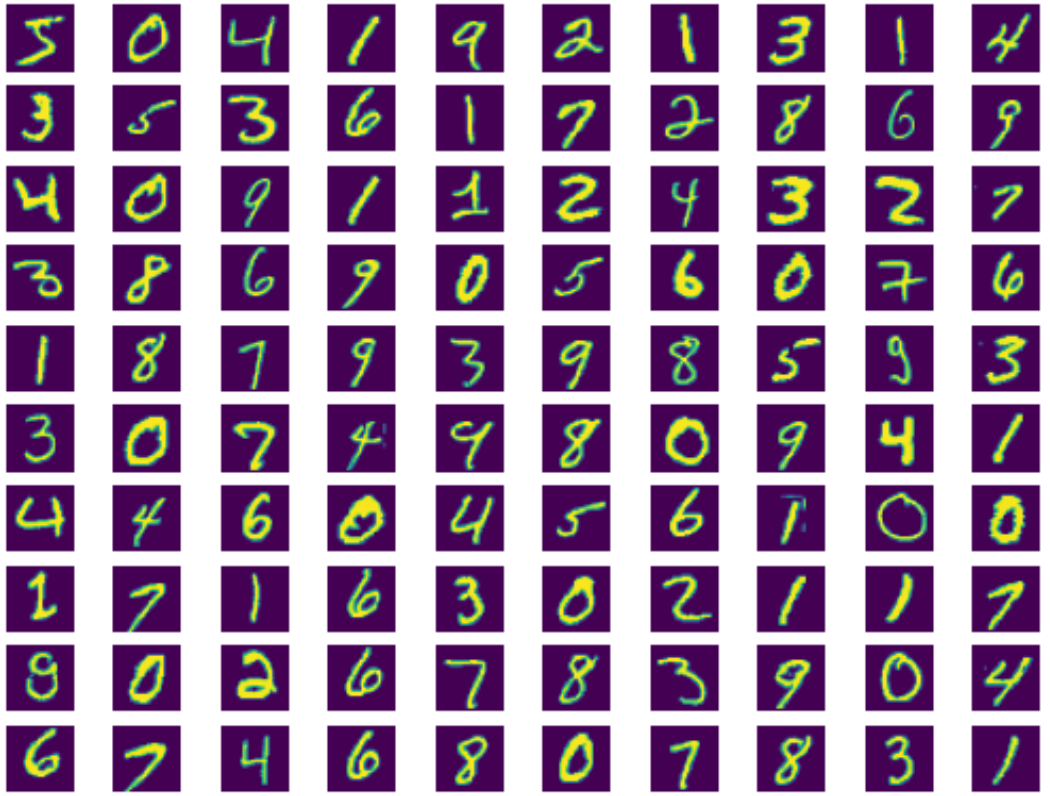


데이터에 대해서 생각해보면

임의의 데이터



손글씨 데이터

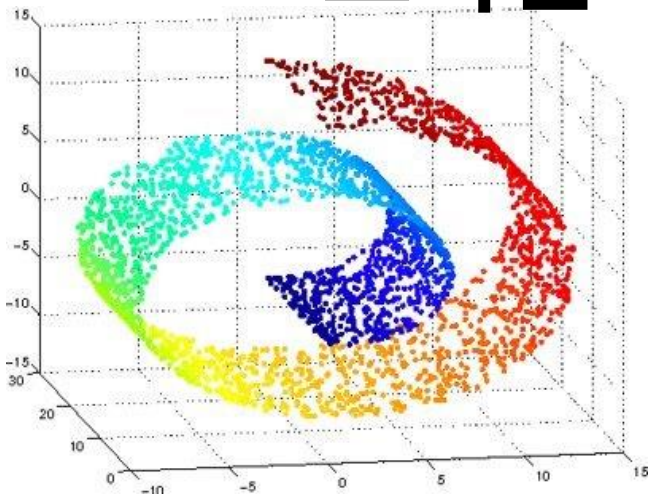


28x28의 크기의 mnist data도 천문학적인 경우의 수를 가집니다.

보통의 경우에는 왼쪽처럼 노이즈 데이터이고, 오른쪽은 매우 특별한 분포라고 생각 할 수 있습니다.

Manifold hypothesis (assumptions)

고차원 데이터를 저차원으로 만들 수 있을까?



고차원의 데이터의 밀도는 낮지만, 이들의 집합을 포함하는 저차원의 매니폴드가 있다.

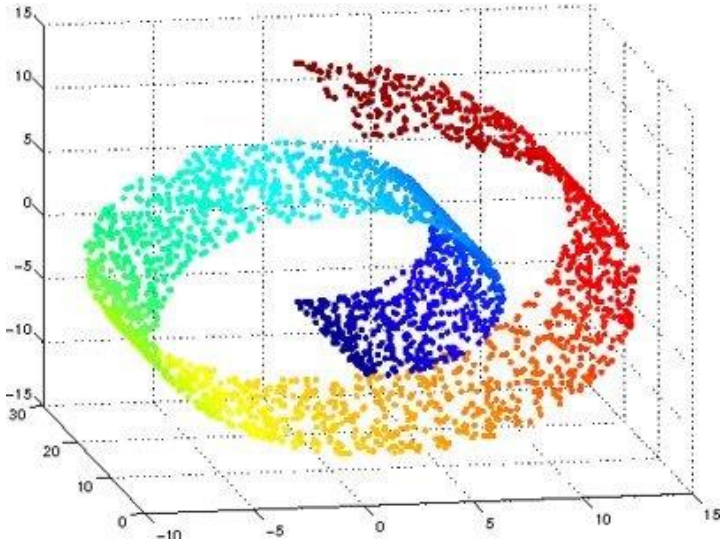
이 저차원의 매니폴드를 벗어나는 순간 급격히 밀도는 낮아진다.

- A d dimensional manifold $\mathcal{M}$ is embedded in an m dimensional space, and there is an explicit mapping $f : \mathcal{R}^d \rightarrow \mathcal{R}^m$ where $d \leq m$
- We are given samples $x_i \in \mathcal{R}^m$ with noise

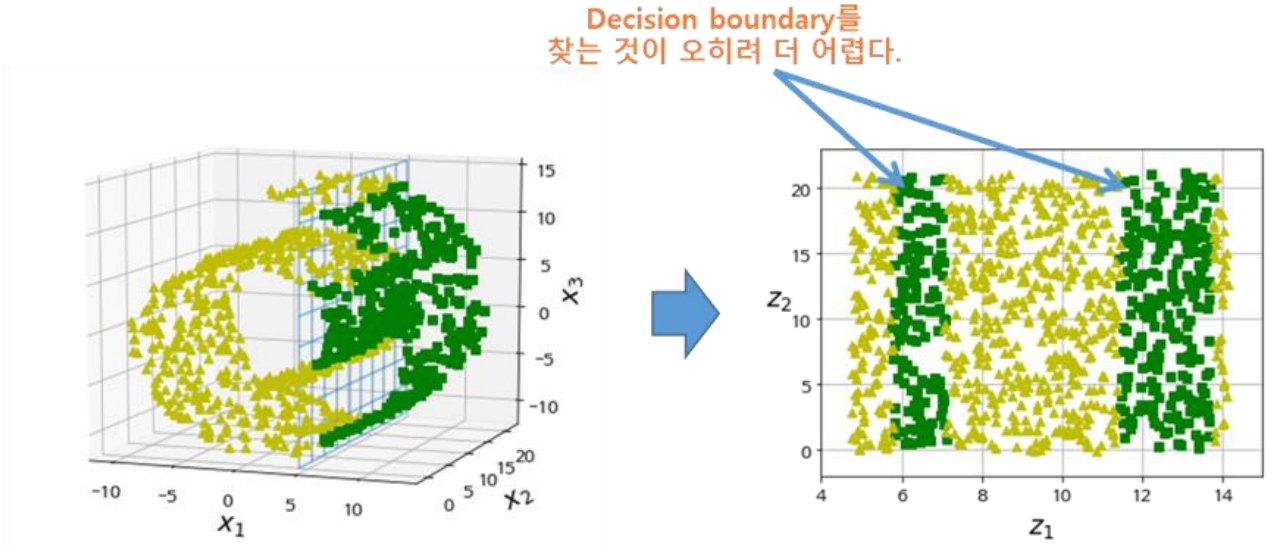
$$x_i = f(\tau_i) + \epsilon_i$$

- $f(\cdot)$ is called embedding function, m is the extrinsic dimension, d is the intrinsic dimension or the dimension of the latent space
- Finding $f(\cdot)$ or τ_i from the given x_i is called manifold learning
- We assume $p(\tau)$ is smooth, is distributed uniformly, and noise is small $\rightarrow$ Manifold Hypothesis

어떤 방법을 써야 할까?

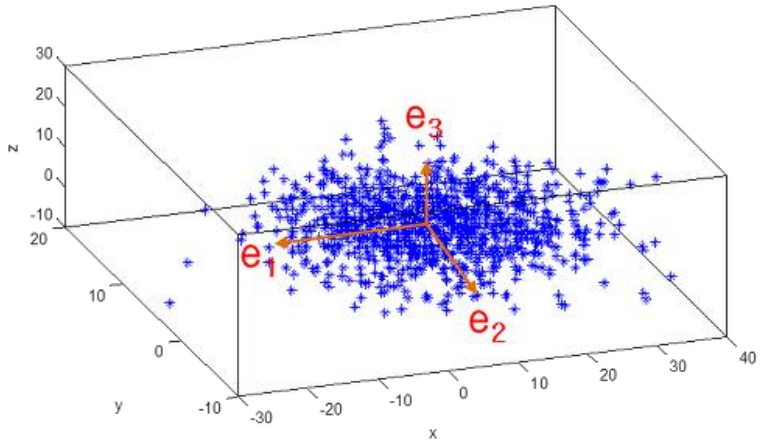


선형 차원 축소 부적합



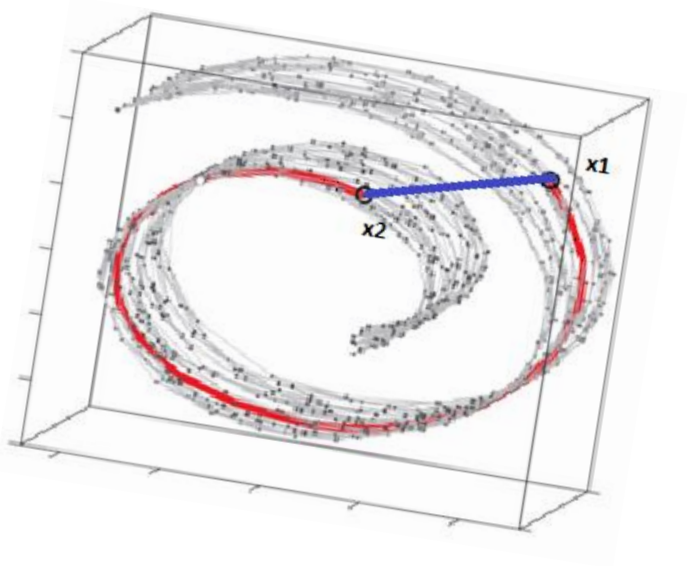
비선형 차원 축소 부적합

차원 축소를 위한 방법



선형 차원 축소

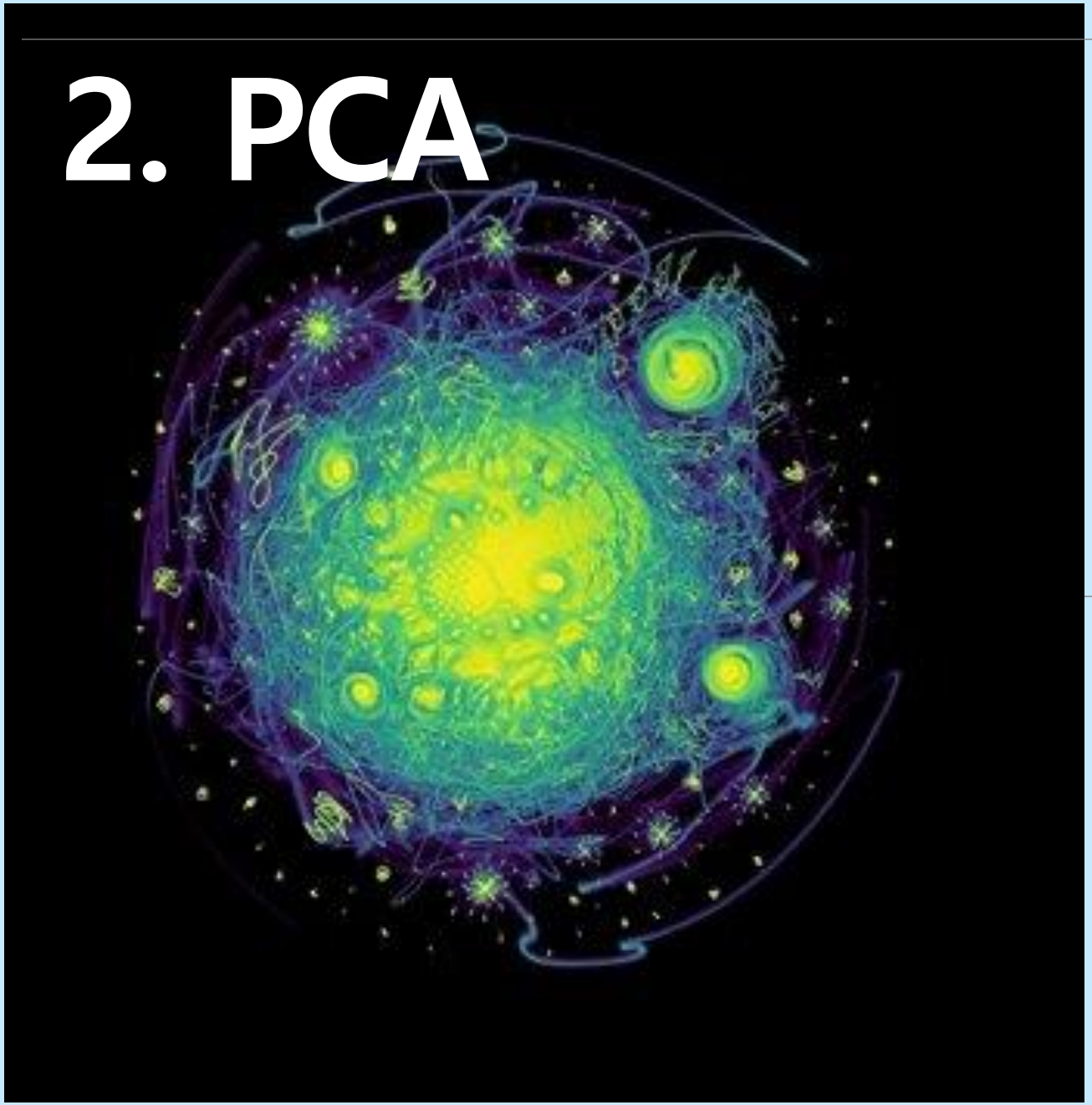
선형 차원 축소는 주로 주성분 분석(Principal Component Analysis, PCA)을 활용합니다. PCA는 가장 널리 사용되는 선형 차원 축소 방법으로, 주어진 데이터의 공분산 구조를 이용하여 주요한 정보를 유지하면서 새로운 축으로 데이터를 투영합니다.



비선형 차원 축소

비선형 차원 축소는 주로 t-분포 확률적 이웃 임베딩(t-Distributed Stochastic Neighbor Embedding, t-SNE)과 오토인코더(Autoencoder, AE) 등을 사용합니다. t-SNE는 고차원 데이터의 군집 구조를 보존하는 2차원 또는 3차원 시각화를 위해 사용되며, AE는 비선형 데이터의 잠재 표현을 학습하여 차원 축소와 생성 모델링을 수행합니다.

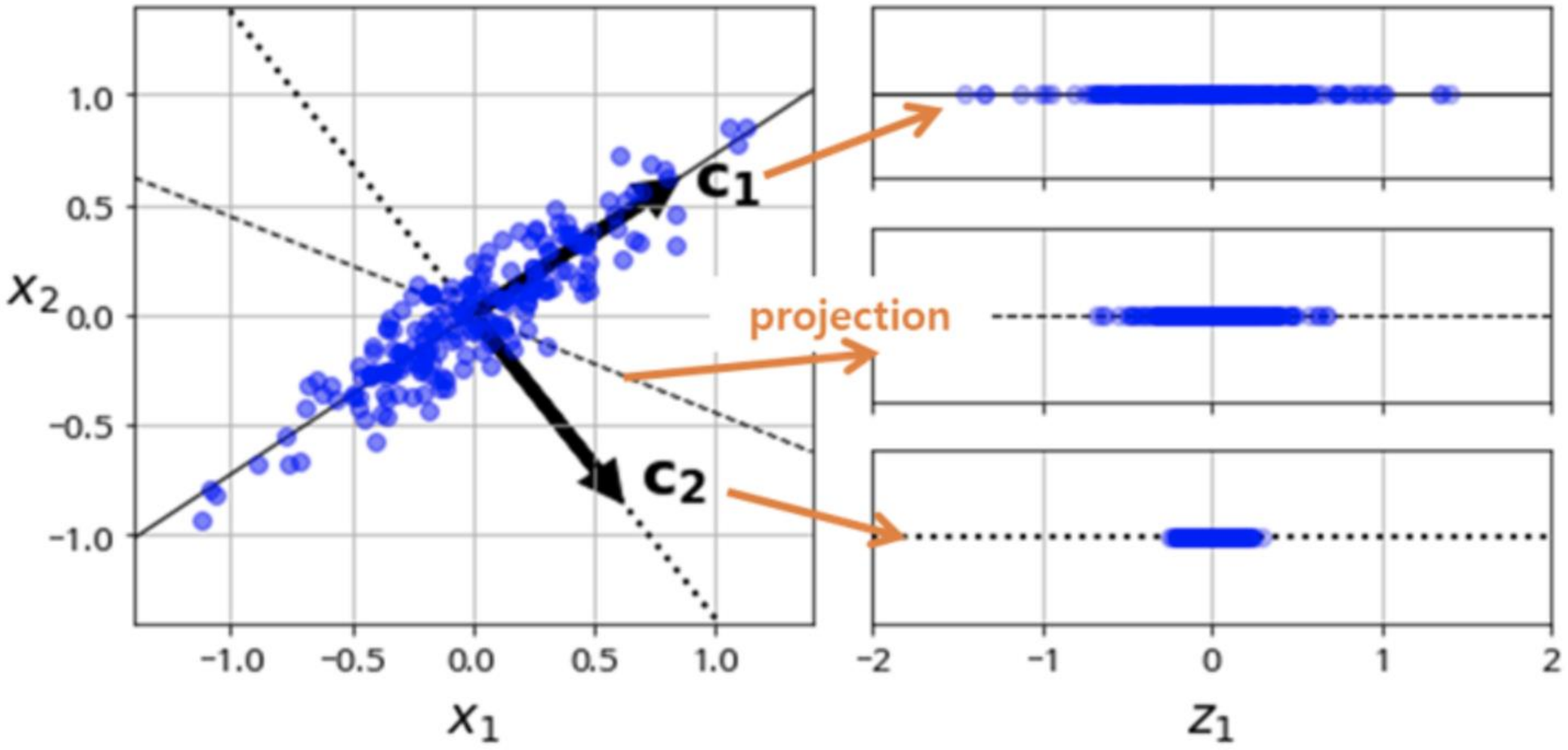
2. PCA



(Principal component analysis, 주성분 분석)
PCA란 무엇일까요?

데이터의 분산을 가장 큰 방향으로 찾고, 그 방향을 주성분으로 사용하는 데이터 축소 기법입니다.

PCA



PCA Assumption

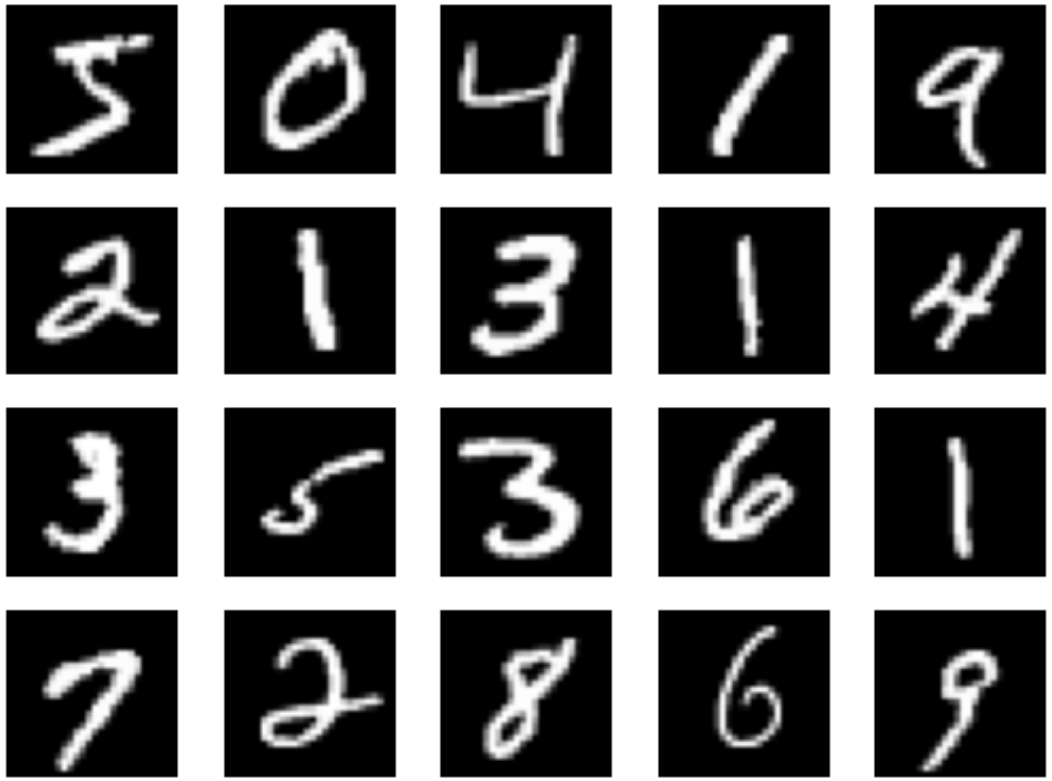
1. independent assumption (독립성)

- 많은 통계 검정에서는 관측치가 독립적이라고 가정합니다. 즉, 데이터 집합의 두 관측치가 서로 관련되어 있거나 어떤 식으로든 서로 영향을 미치지 않습니다.

2. Linearity (선형성)

- re-expressing the data as a *linear combination* of its basis vectors.

Dataset

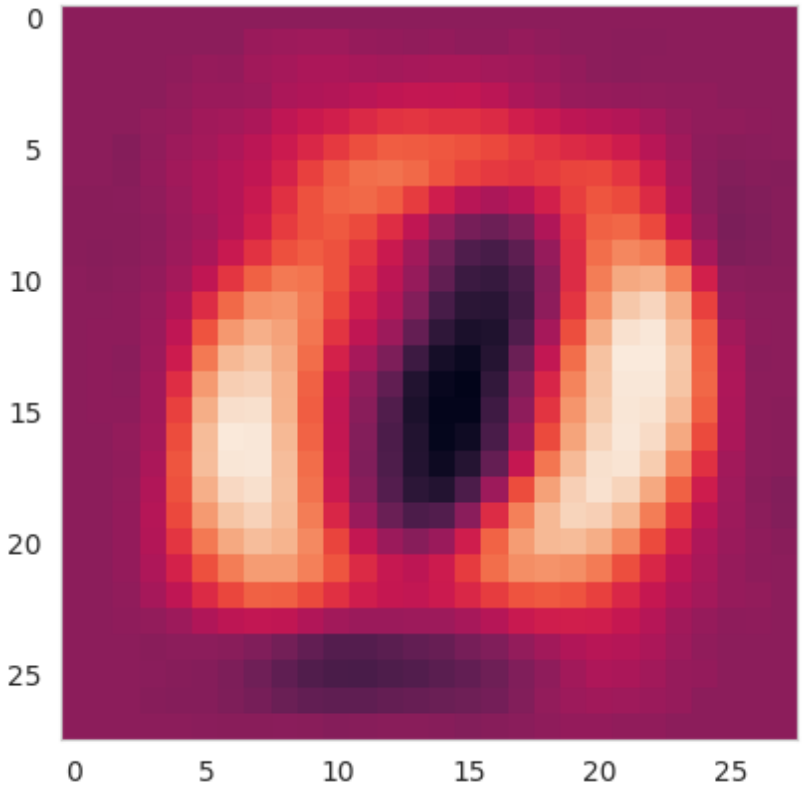


MNIST

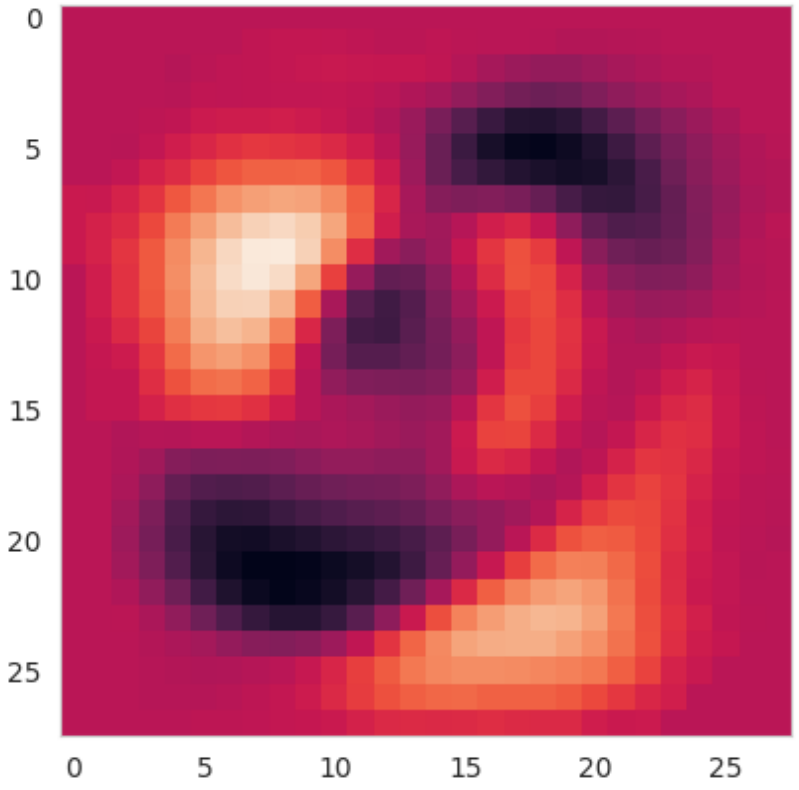
AT&T Laboratories Cambridge : olivetti face datasets

데이터에 대해서 생각해보면

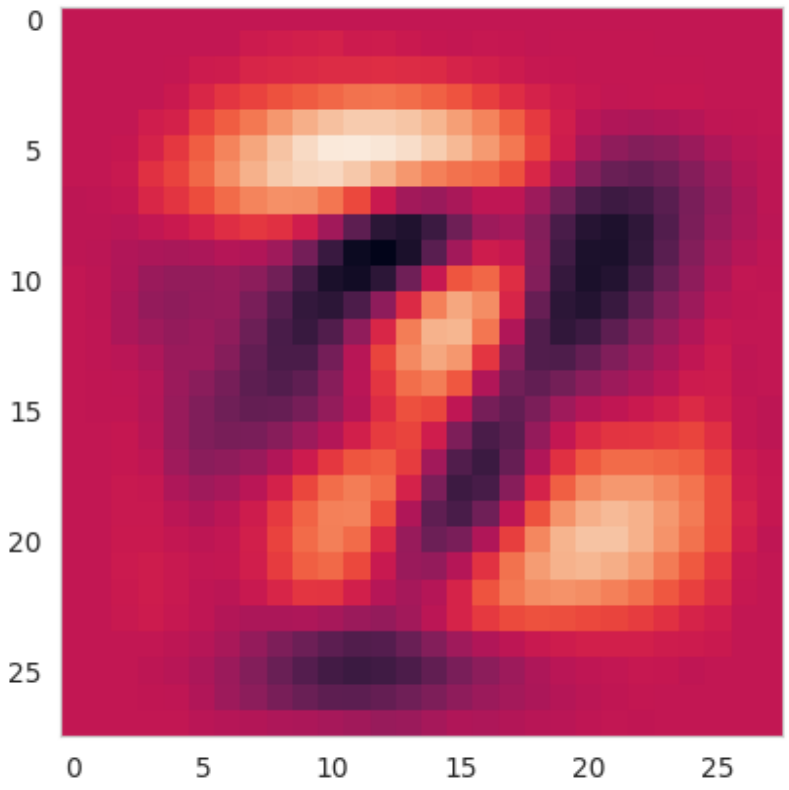
MNIST eigen vector 시각화



New basis 1

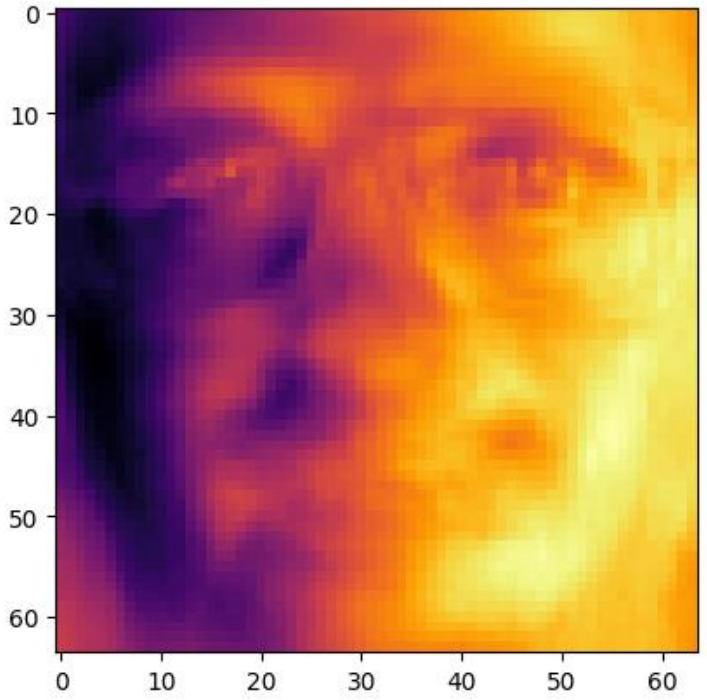


New basis 2

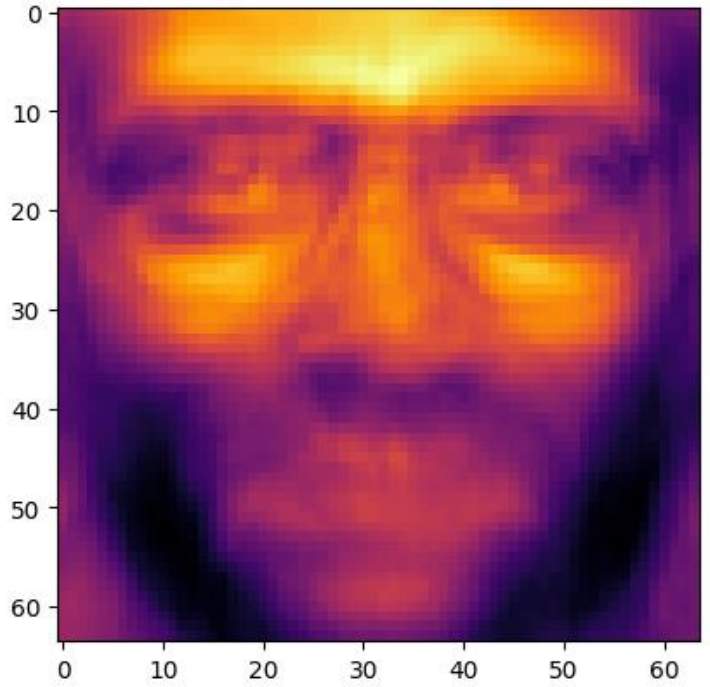


New basis 3

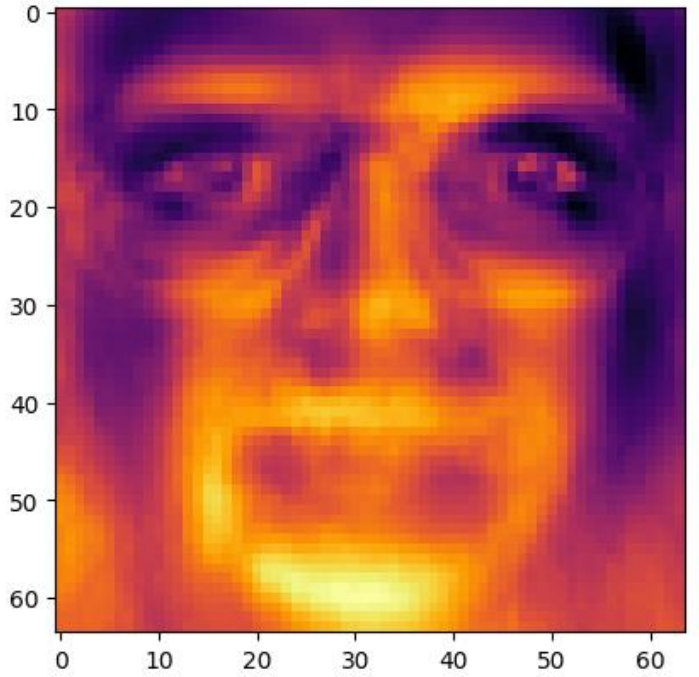
Face data eigen vector 시각화



New basis 1

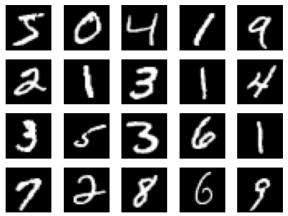


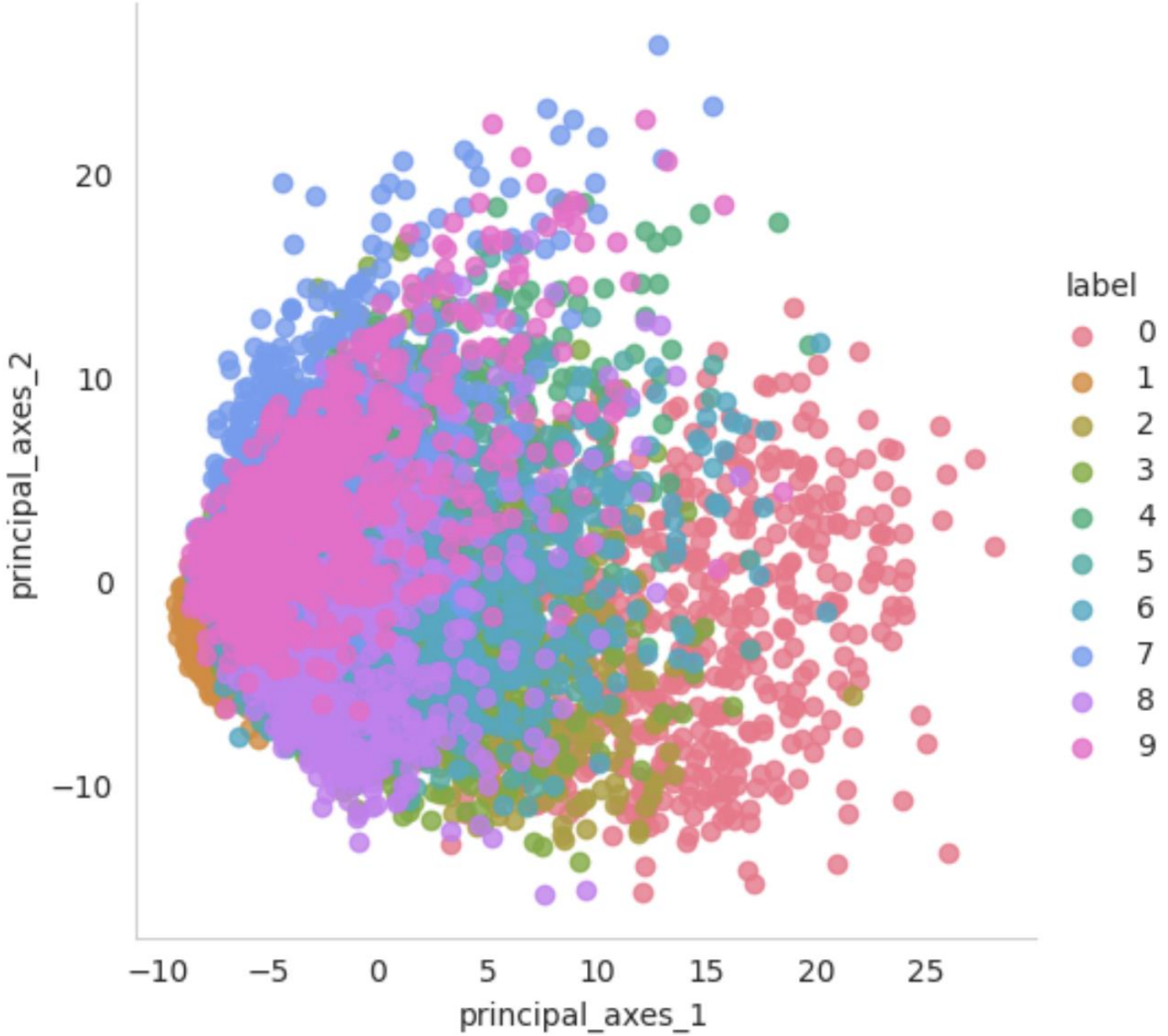
New basis 2



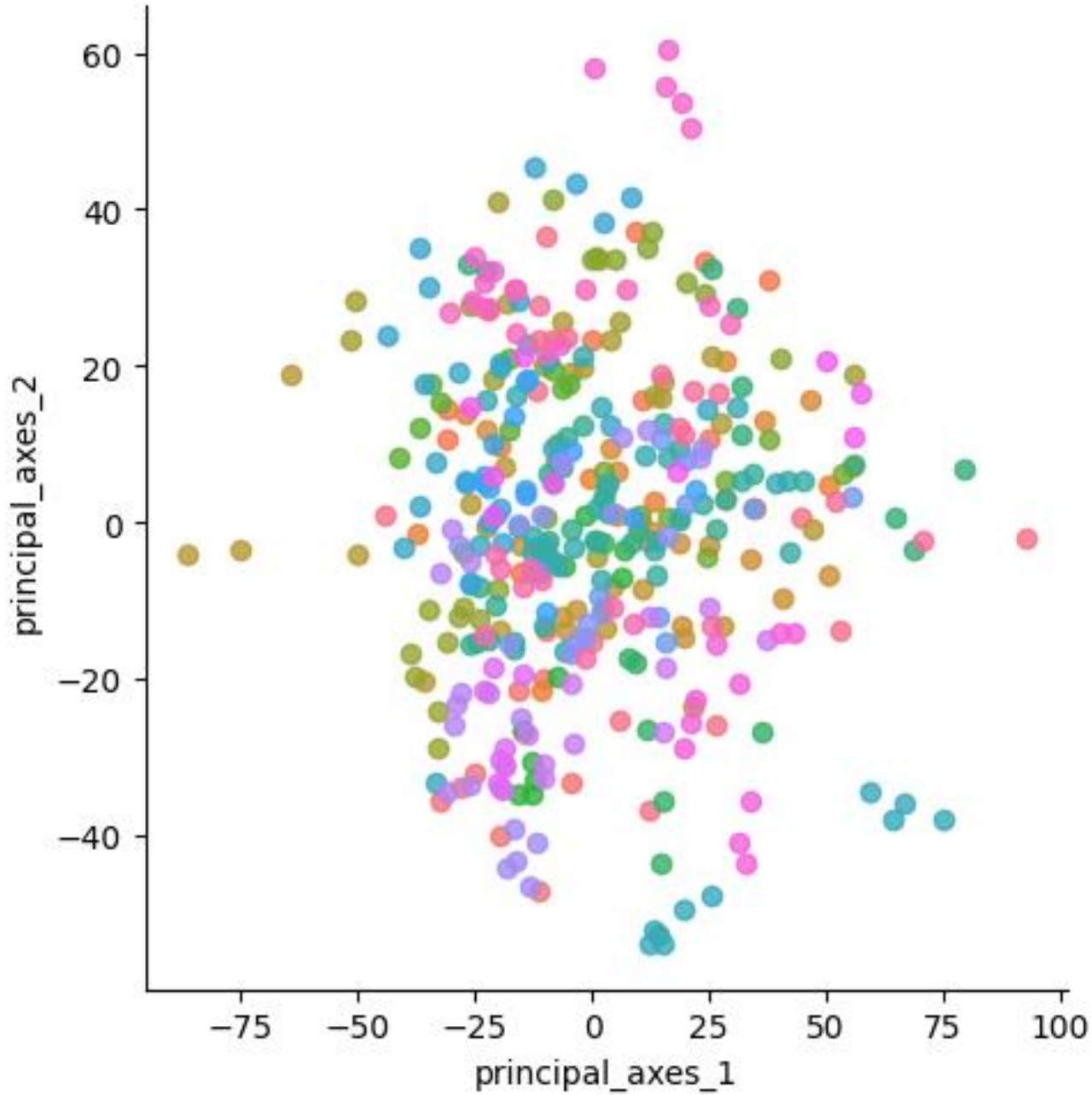
New basis 3

2. PCA

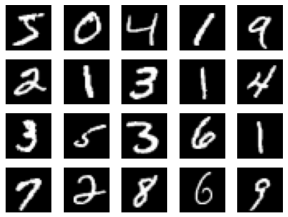


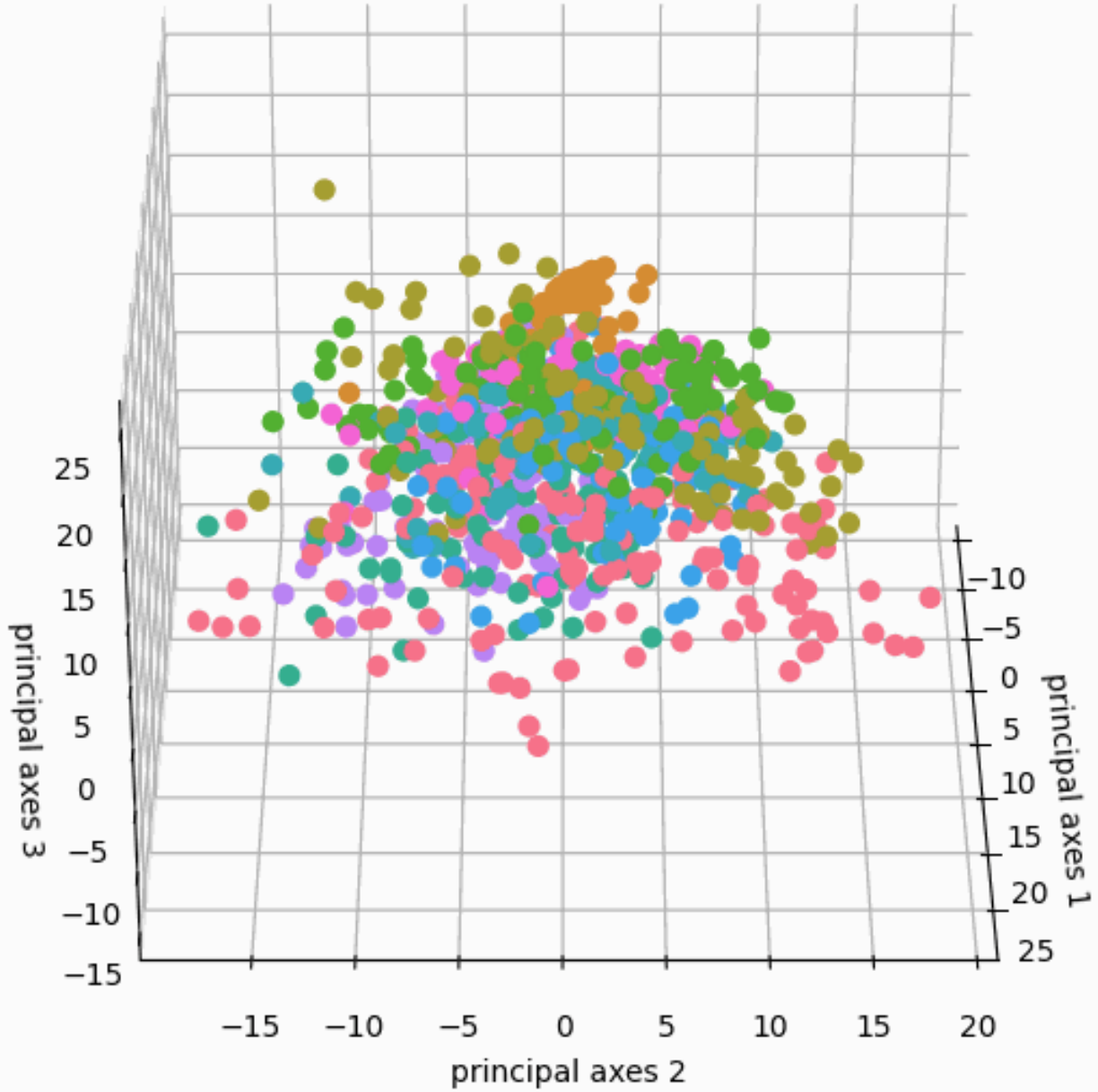

principal_axes_2
20
10
0
−10
−10
−5
0
5
10
15
20
25
principal_axes_1
label
0
1
2
3
4
5
6
7
8
9

2. PCA

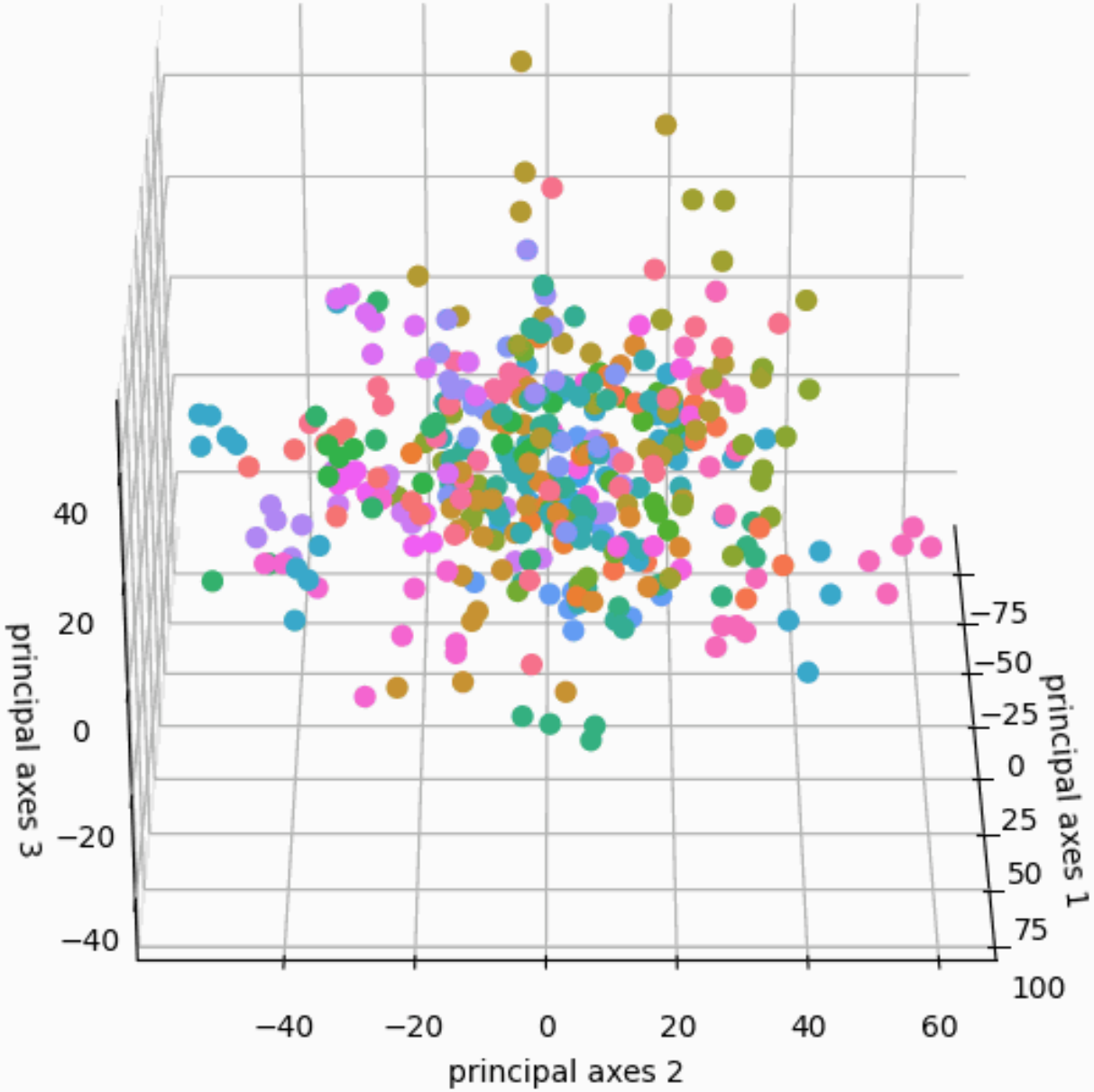

60
40
20
0
−20
−40
principal_axes_2
−75
−50
−25
0
25
50
75
100
principal_axes_1

2. PCA



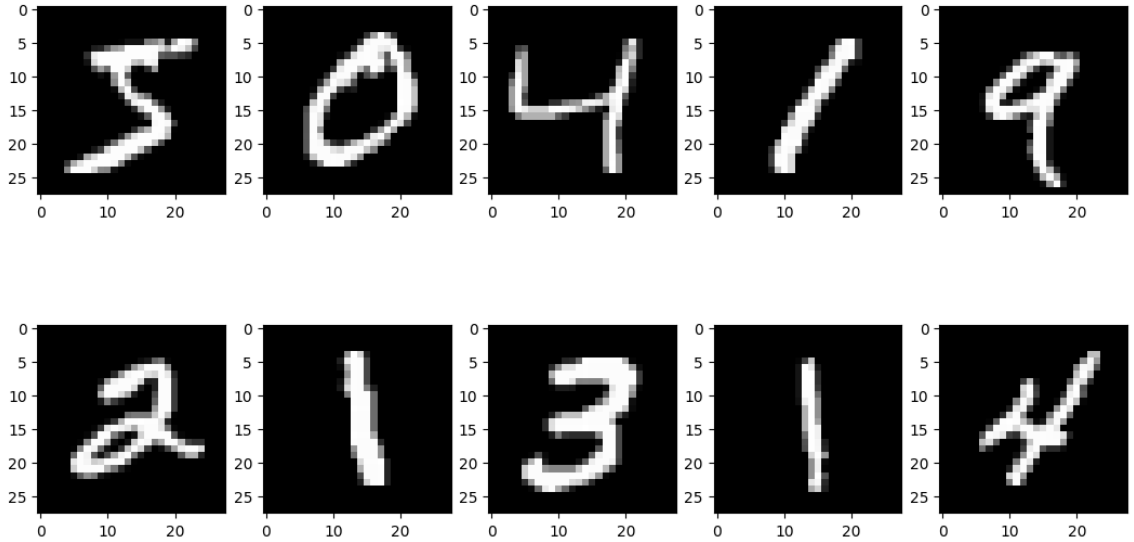

principal axes 3
principal axes 2
principal axes 1
25
20
15
10
5
0
−5
−10
−15
−15
−10
−5
0
5
10
15
20
−10
−5
0
5
10
15
20
25

2. PCA

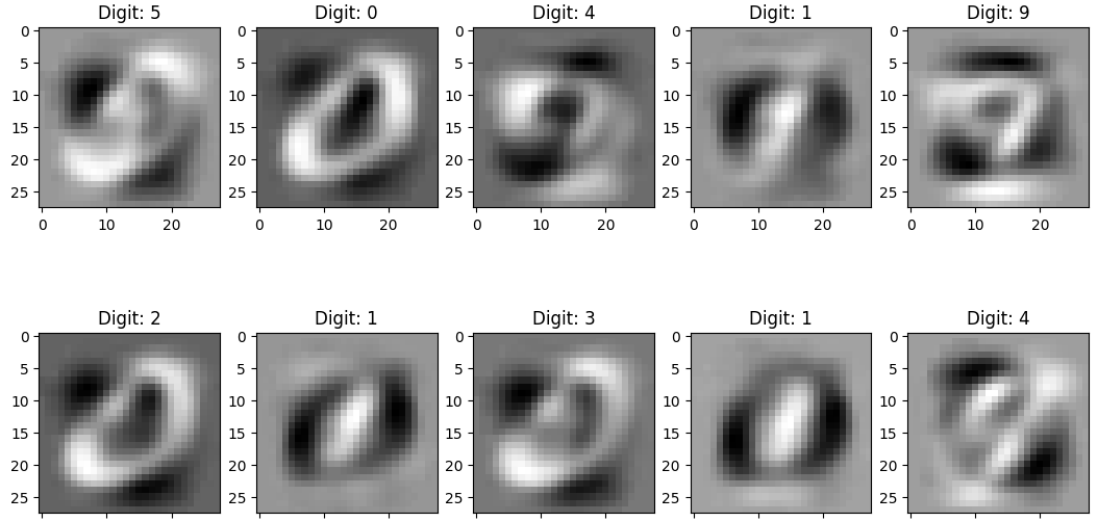

principal axes 3
40
20
0
−20
−40
principal axes 2
−40
−20
0
20
40
60
principal axes 1
−75
−50
−25
0
25
50
75
100

2. PCA

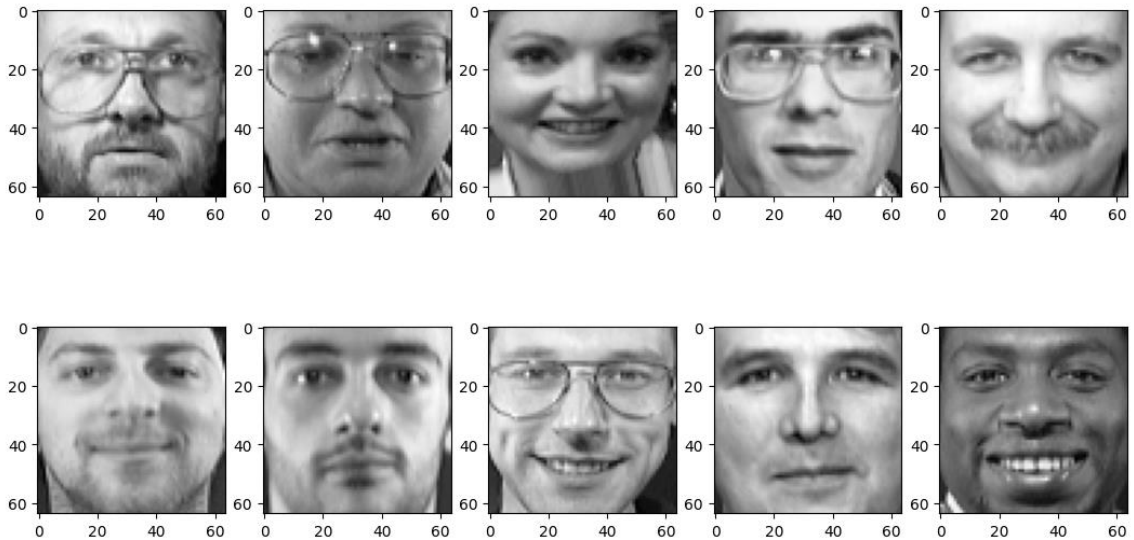
Original data



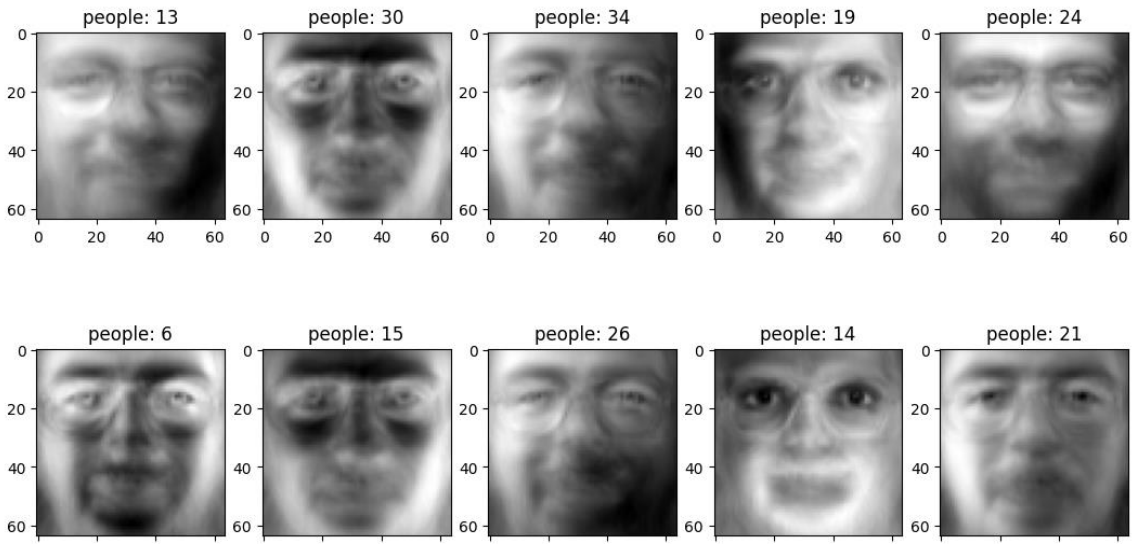
Reconstruction (3Dim)



Original data

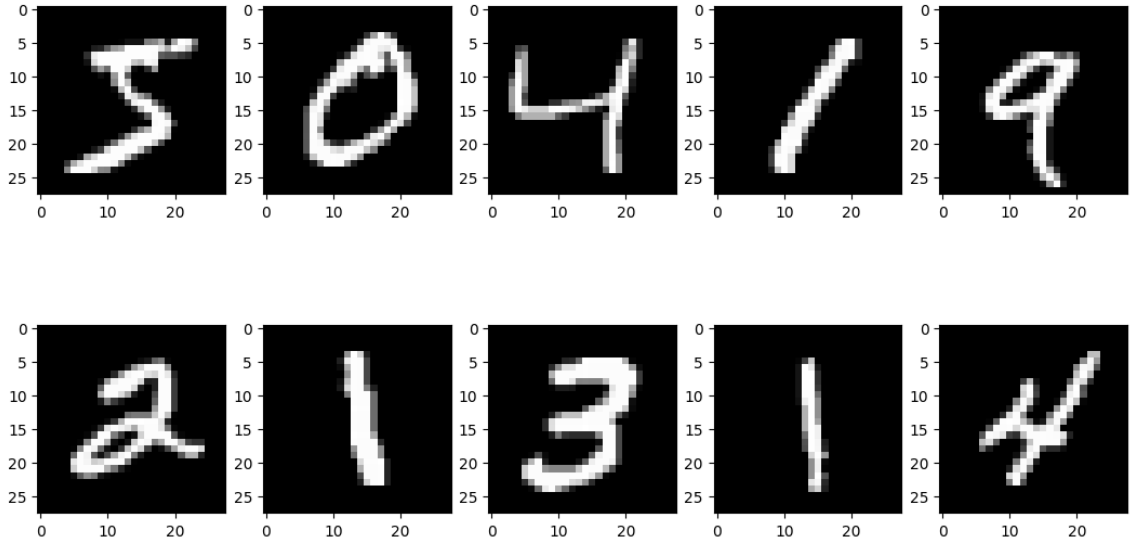


Reconstruction (3Dim)

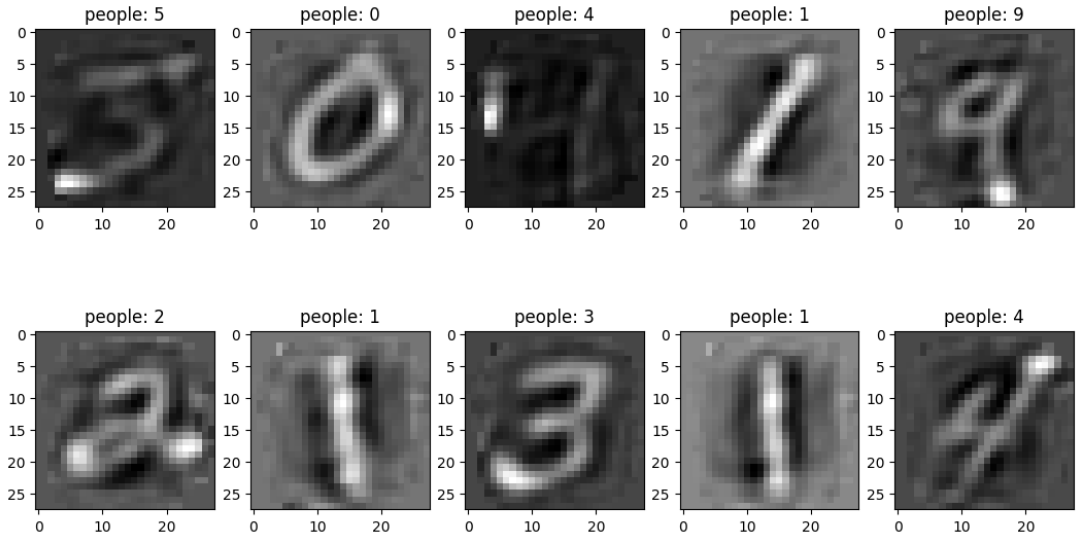


2. PCA

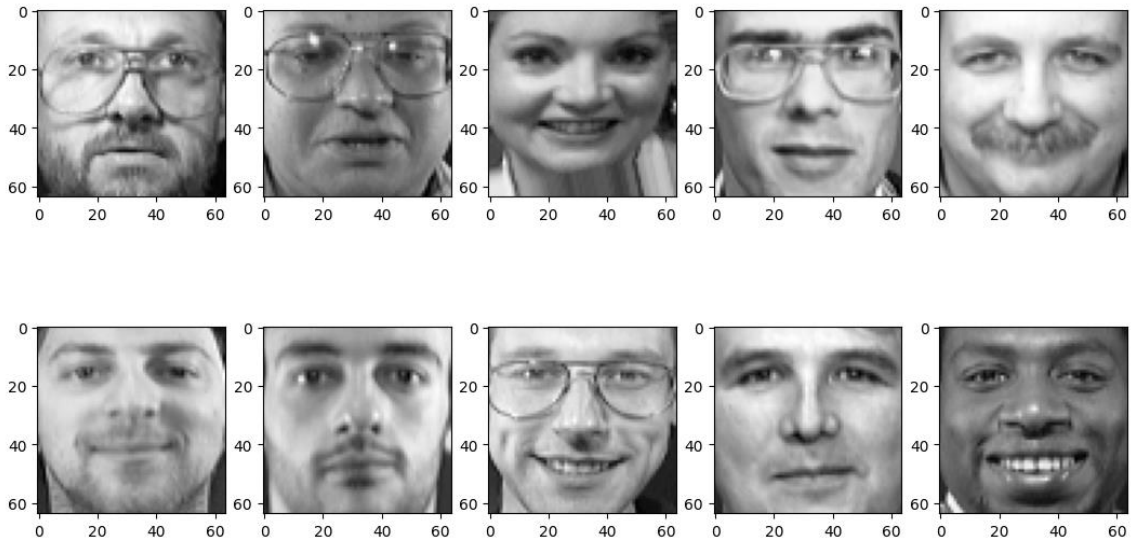
Original data



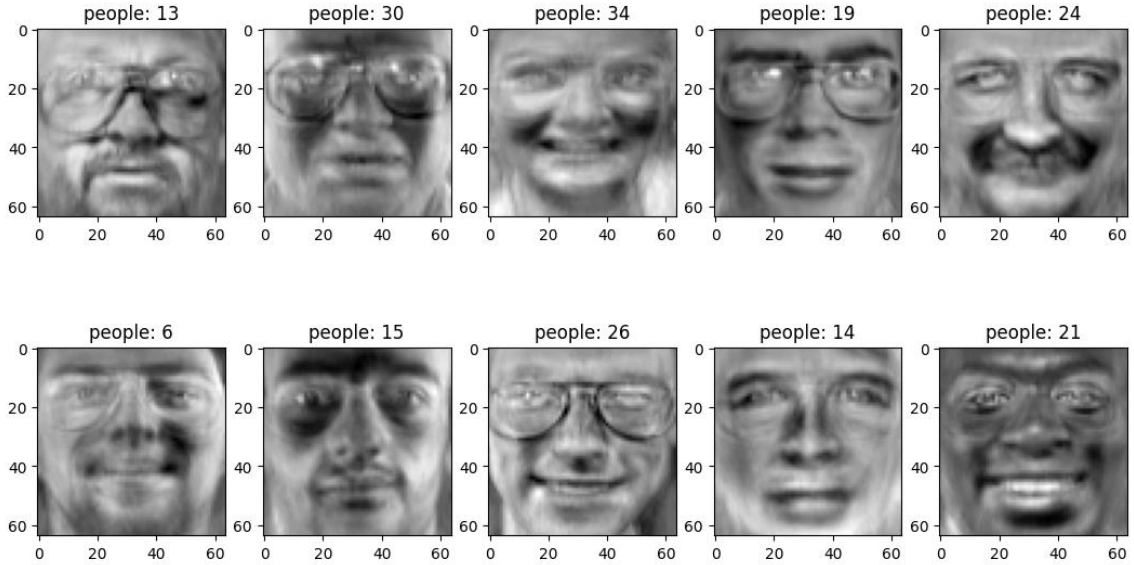
Reconstruction (77Dim)



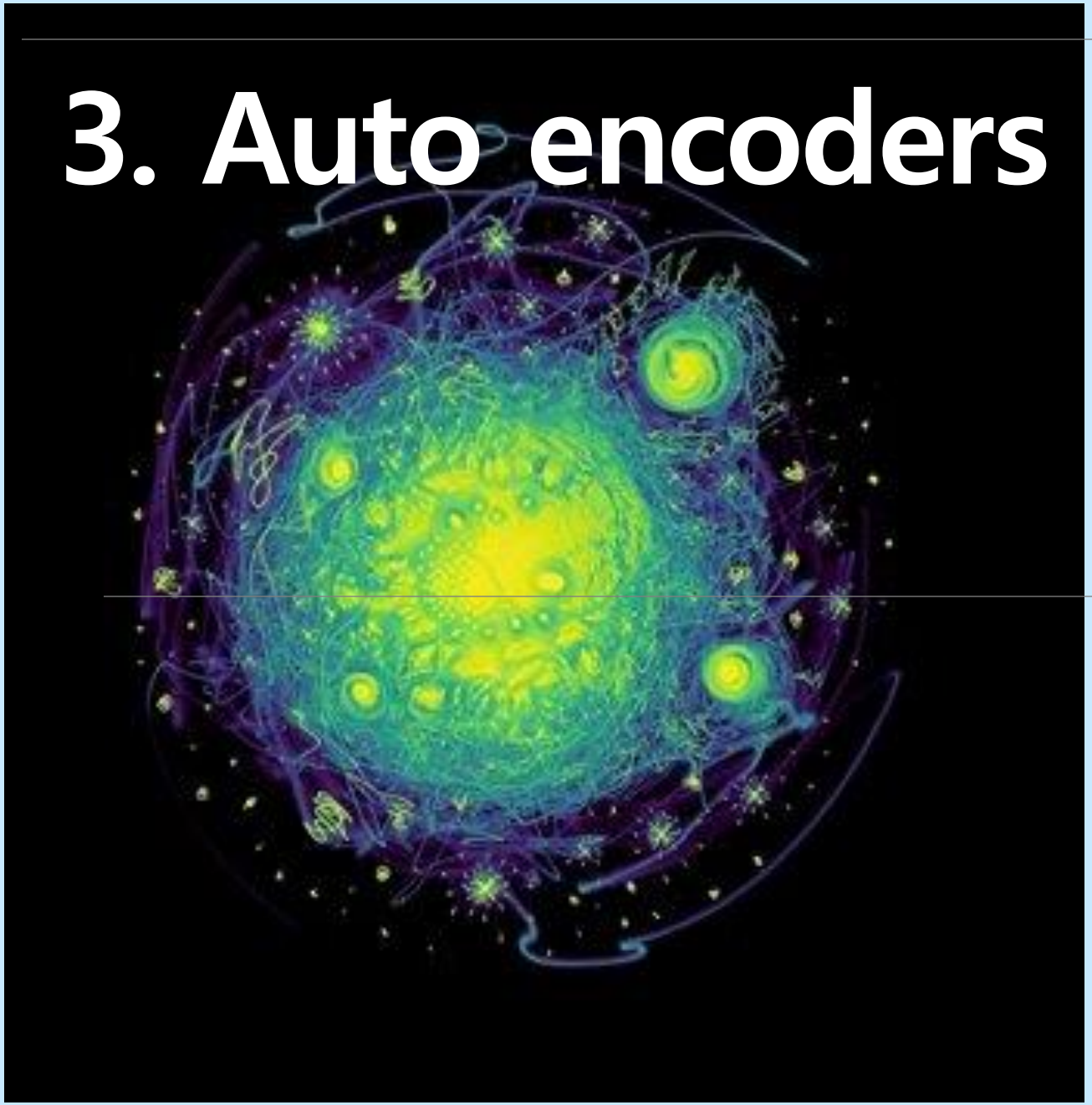
Original data



Reconstruction (77Dim)



3. Auto encoders



Auto encoders 란 무엇일까요?

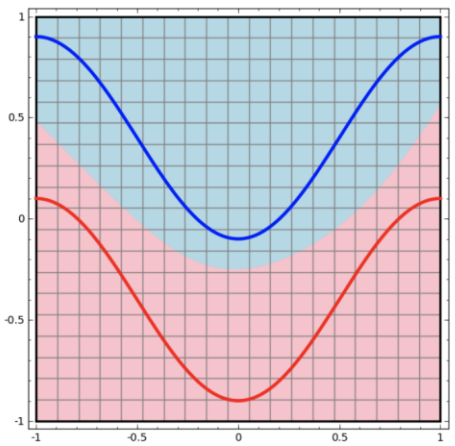
Autoencoder는 Encoder와 Decoder로 이루어져 있습니다. Encoder는 고차원의 데이터를 저차원의 공간으로 압축하고, Decoder는 저차원의 공간에서 고차원의 데이터를 복원합니다

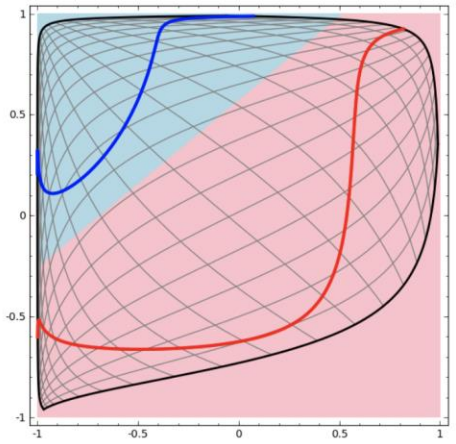
01.

Hidden layer

(Affine transformation , Activation function)를 통해

데이터가 선형적으로 분리 될 수 있도록 표현을 학습한다.



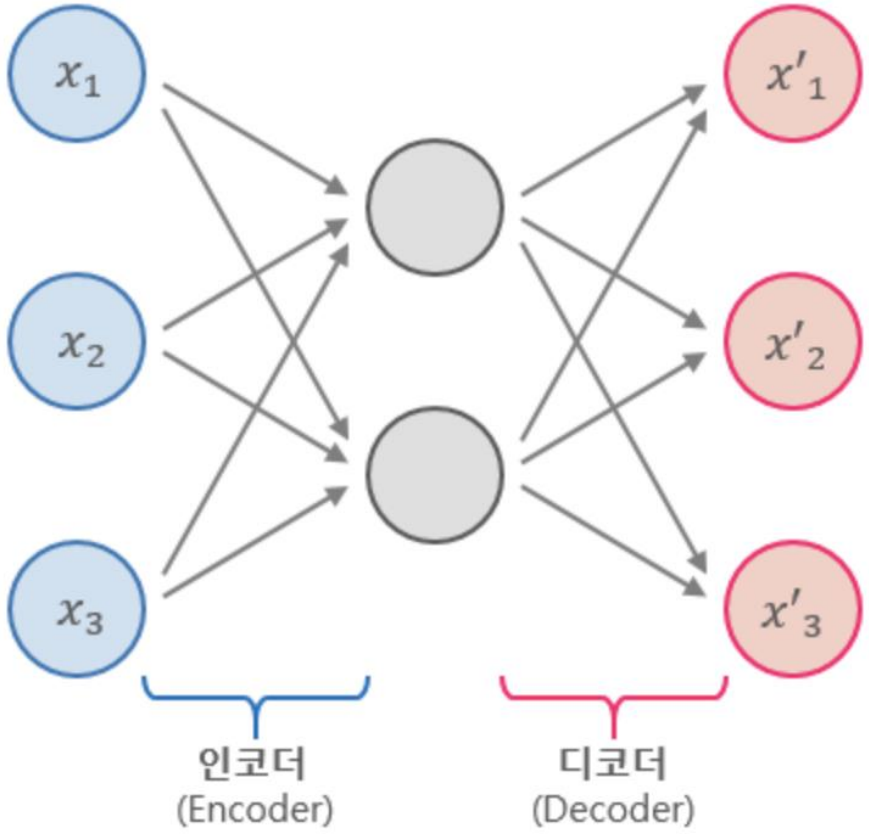


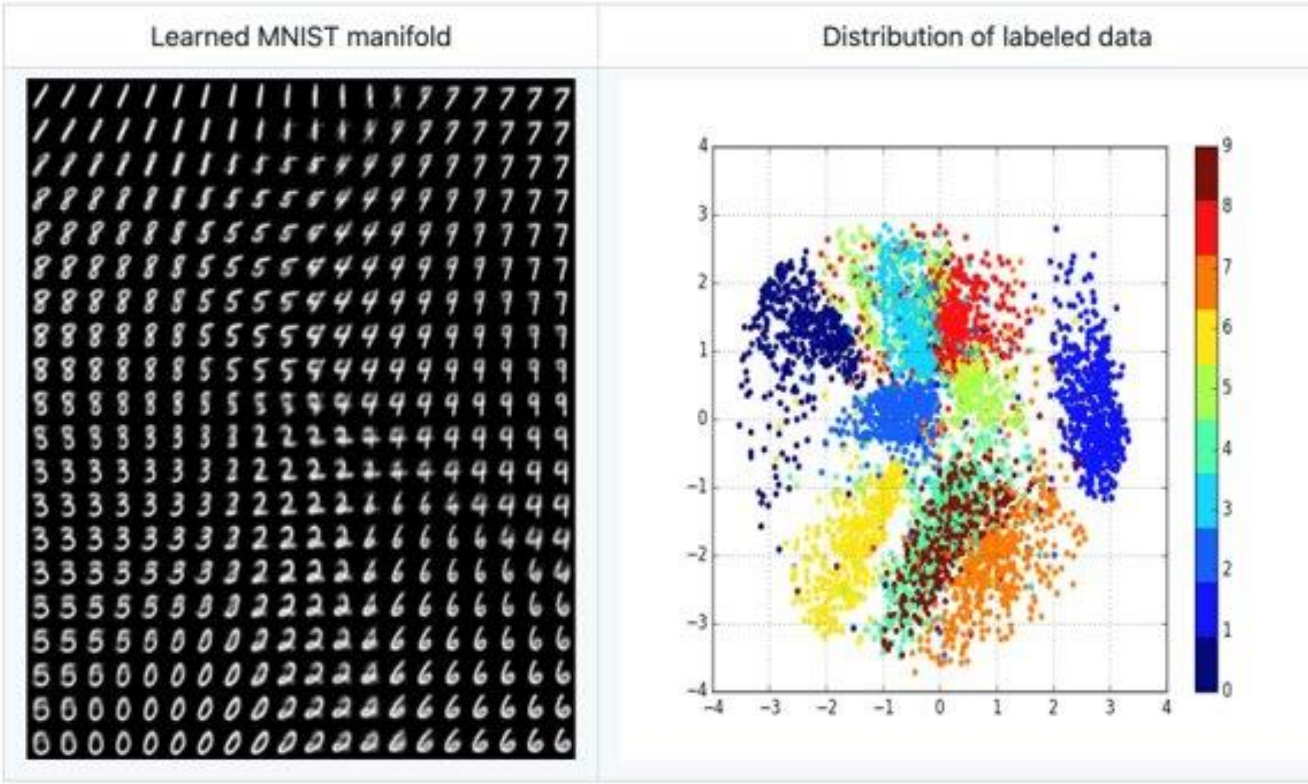
02.

여러 번 반복하면, 네트워크는 점차적으로 데이터의 표현을 조정하고 분리 가능한 구조를 학습 => 적절한 manifold 형성

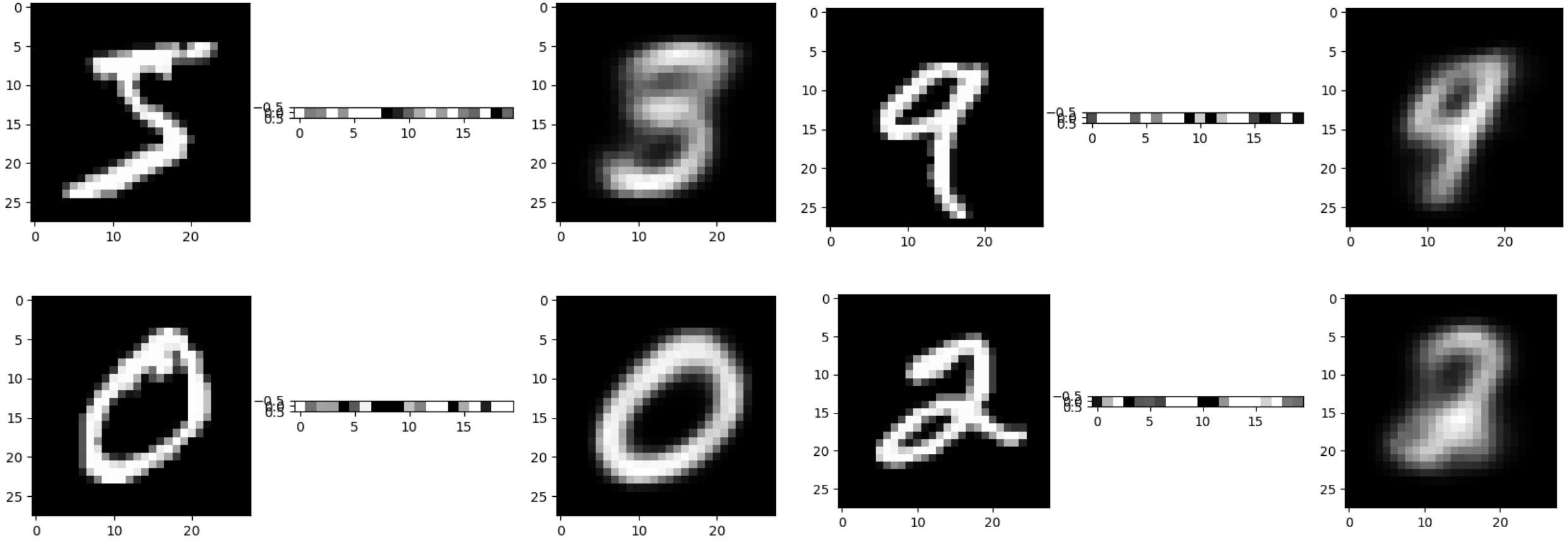
하지만 모든 데이터에 대해 선형적인 분리가 가능한 manifold를 항상 찾을 수 있는 것은 아님. 데이터의 복잡성과 차원 수, 분포 등에 따라 적절한 manifold의 형태가 달라질 수 있습니다. 따라서 적절한 manifold를 찾기 위해서는 여러 가지 다양한 모델과 알고리즘을 시도해보고 결과를 평가해야 합니다.

Auto encoder 구조

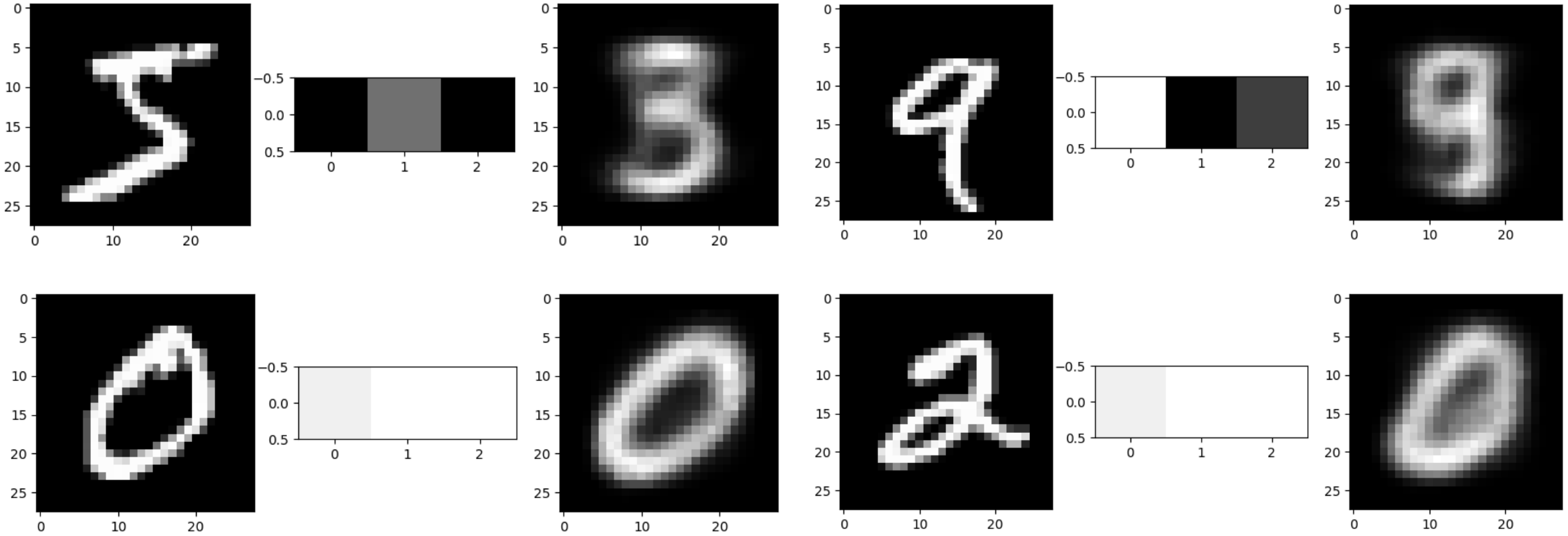


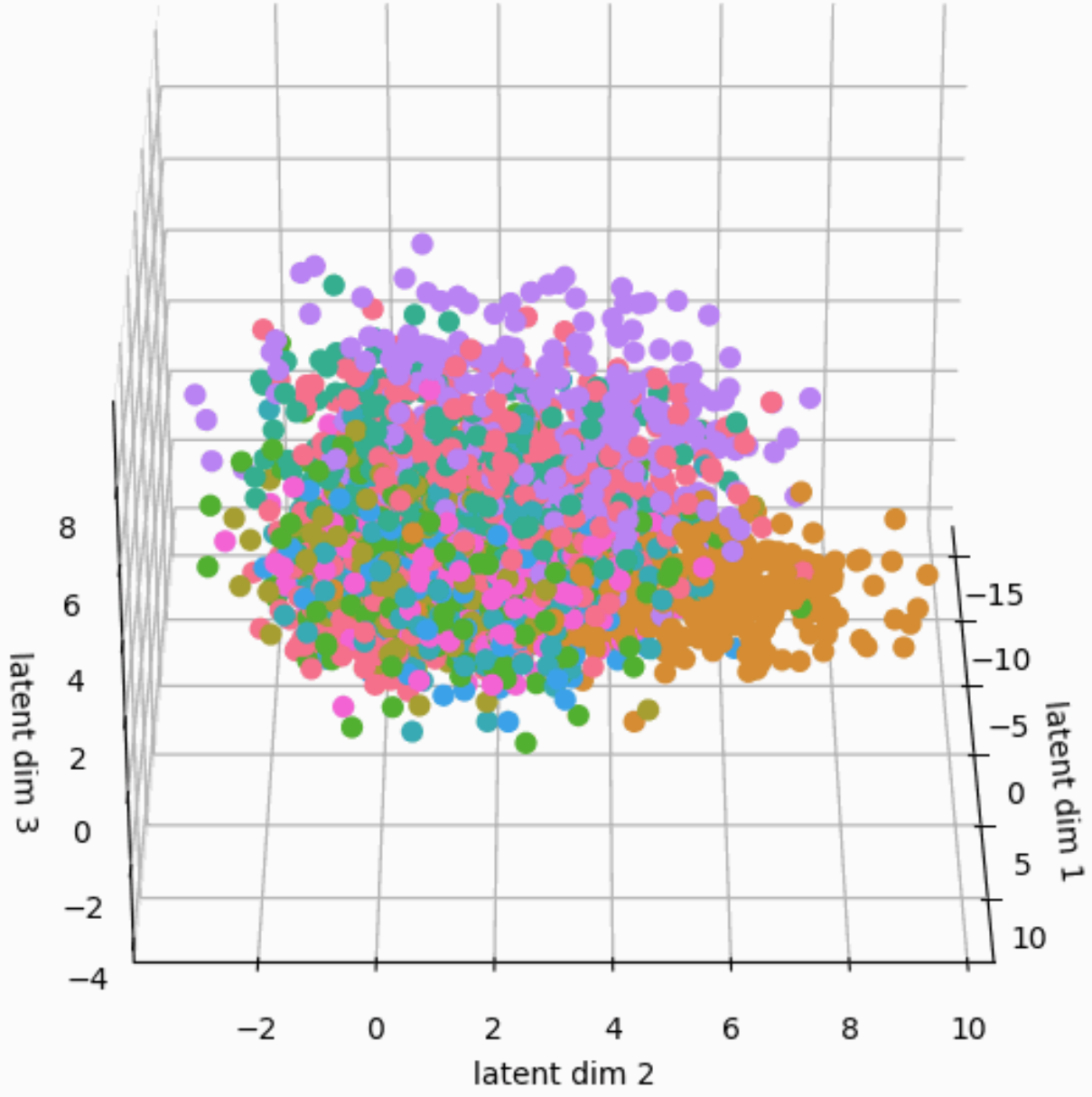


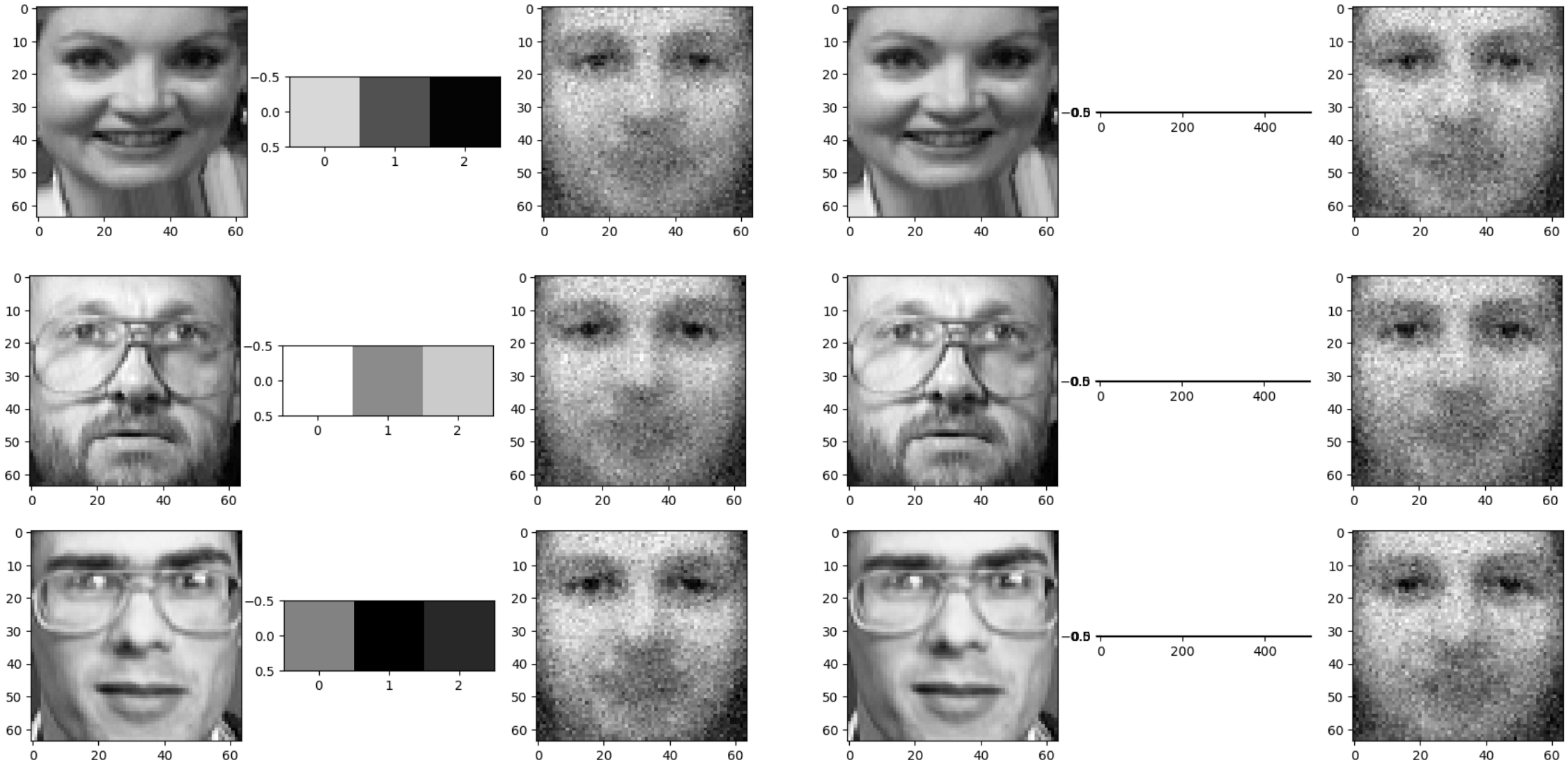
Autoencoder



Autoencoder

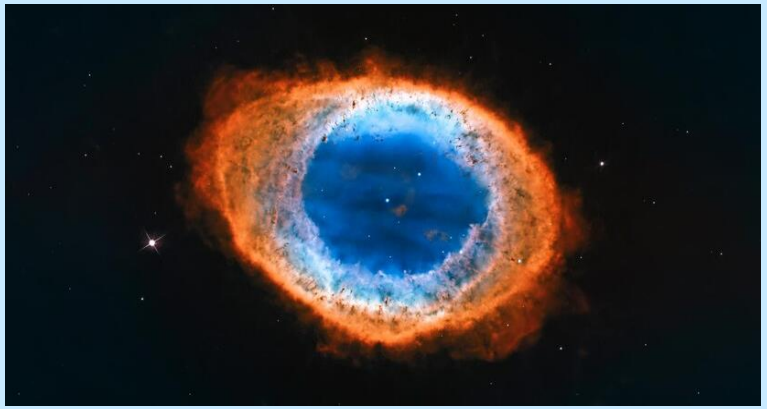






End of Document

Thank You.



Reference

https://colah.github.io/posts/2014-10-Visualizing-MNIST/

https://arxiv.org/abs/1312.6114

https://arxiv.org/abs/1404.1100

https://github.com/ExcelsiorCJH/Hands-On-ML/

https://www.slideshare.net/NaverEngineering/ss-96581209

https://scikit-learn.org/stable/modules/manifold.html#manifold

https://www.youtube.com/@aailabkaist6236/playlists

https://www.youtube.com/@naverd2848/playlists

https://proceedings.neurips.cc/paper/2002/hash/6150ccc6069bea6b5716254057a194ef- Abstract.html

https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf?fbcl