



# Keras (3)

고려대학교 INI Lab

**AutoEncoder**

# Encoder

기존의 데이터를 주어진 알고리즘을 통해 변환, 혹은 암호화(코딩)하여 새로운 결과물을 생성하는 작업

변환값 설정

파일 포맷 (형태) MP4 - MP4 파일 (PC용)

**동영상 코덱**

☒ 비디오 사용

비디오 코덱 (Video Codec) **XviD MPEG-4 Video**

비트레이트 (Bit Rate) **1200**

비디오 크기 조정 **화질**

☐ 원본 ☒ **화질** ☐ 원본

☐ 원본 비율유지하면서 사이즈 변경 (빈공간 검은색)  
☐ 비율 많이 다를 때 자동 잘라내기

비디오 프레임레이트 조정 **프레임** **23.970**

필터 ☐ 상하반전 ☐ 좌우반전

회전 ☒ 회전안함 ☐ 시계 방향 90도 ☐ 시계 반대 방향 90도

**오디오**

☒ 오디오 사용 ☒ 스테레오 ☐ 모노 3배

오디오 코덱 (Audio Codec) MP3 2배

비트레이트 (Bit Rate) 128 기준 100%

샘플링 레이트 (Sampling Rate) **음질** **44100** 볼륨 조절 100% 0%

VBR 설정 **cbr**

**자막**

☒ 자막 사용

위치 95 언어 cp949

크기 4

테두리 두께 2

테두리 변질 2

자막 지연 0 초

폰트 경로 C:\Program Files (x86)\WCacaoEncoderWmr 찾기

☒ 속도 변경 1 x 배속 예: 1.0 미만 원본속도, 2.5 미만 2.5배 빠름

☐ 프레임 중복 허용으로 안전성/호환성 향상

결과물 파일명 형태 {ori\_name}, {base\_ext}

원본파일이름 = {ori\_name}

출력포맷의 기본확장자 = {base\_ext}

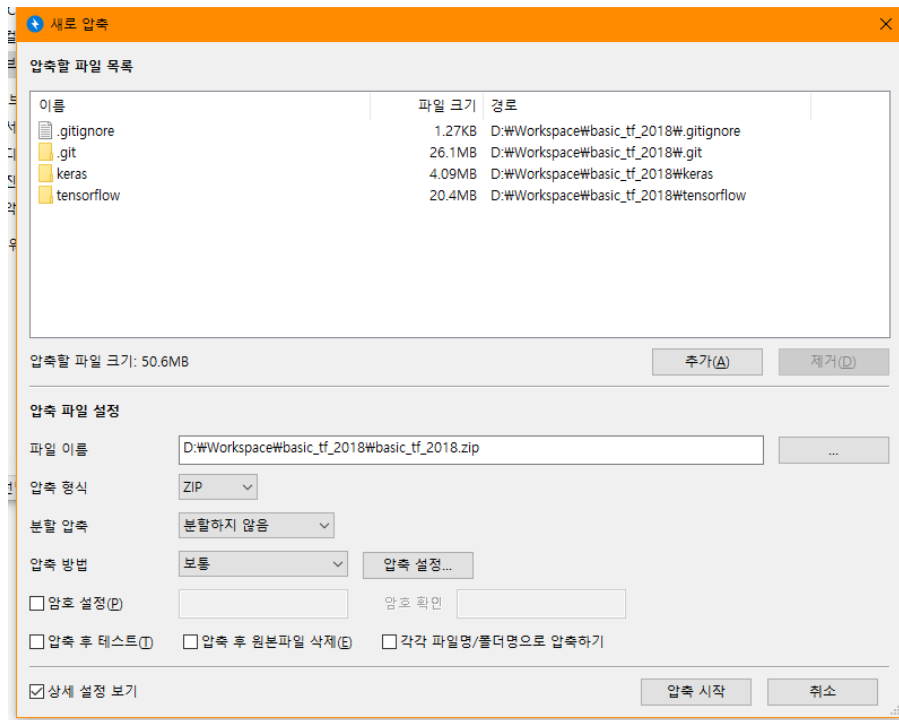
추가 명령어 **men** **encoder**

추가 명령어

적용 취소

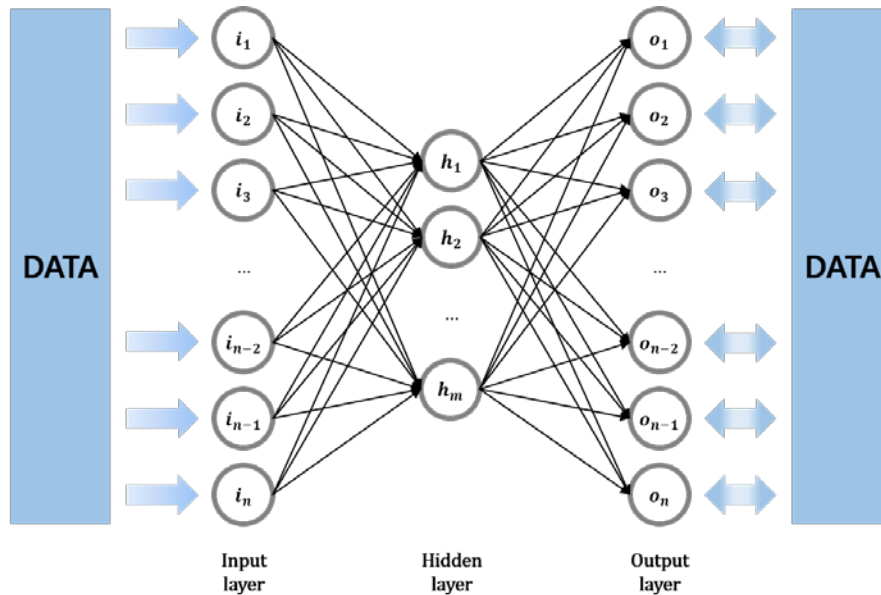
# Encoder

기존의 데이터를 주어진 알고리즘을 통해 변환, 혹은 암호화(코딩)하여 새로운 결과물을 생성하는 작업



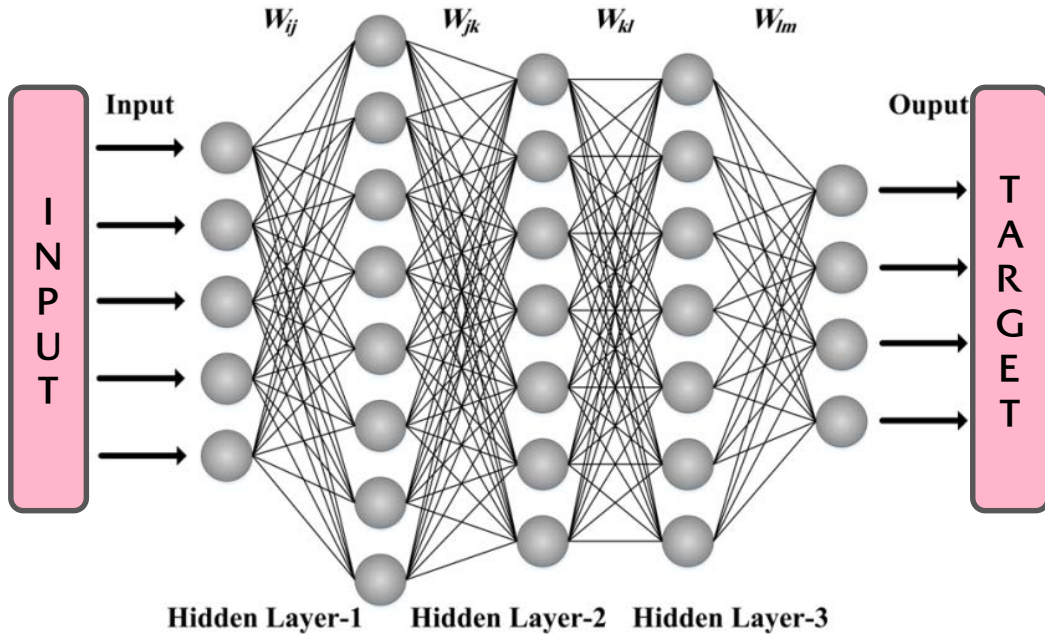
# AutoEncoder

스스로 결과물을 생성하는 과정(알고리즘)을 학습하는 기계학습 모델



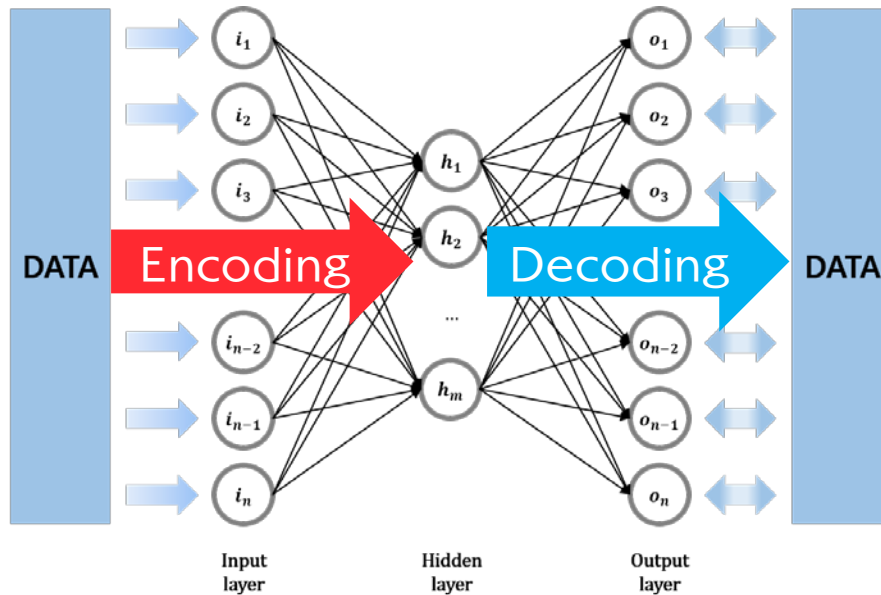
# Review FFNN

AutoEncoder의 Target은?

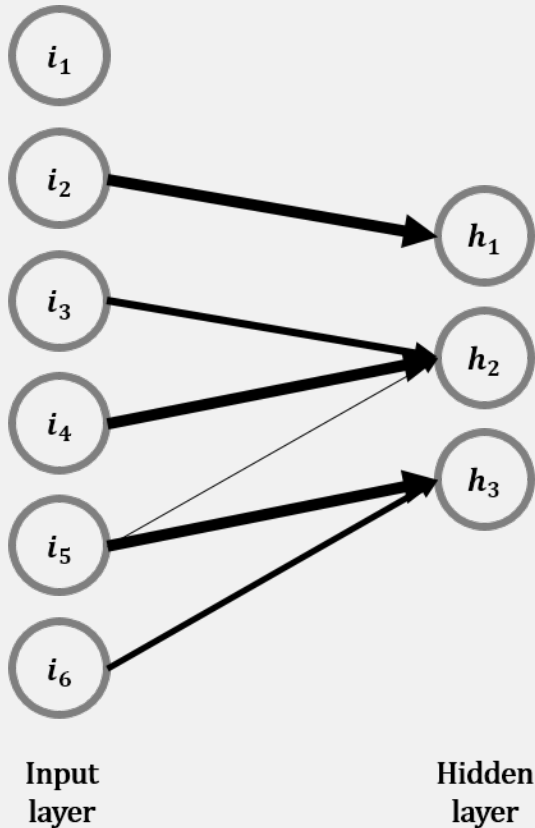


# AutoEncoder

Target을 input 데이터와 동일하게 놓고, 인코딩 작업을 수행한 후,  
이를 디코딩 과정으로 원래 데이터로 복원하는 과정을 통해 학습을 수행



# Why AutoEncoder?



데이터를 재생산하는 과정을 통해, 고차원의 데이터를 저차원의 데이터로 변환(손실 압축)

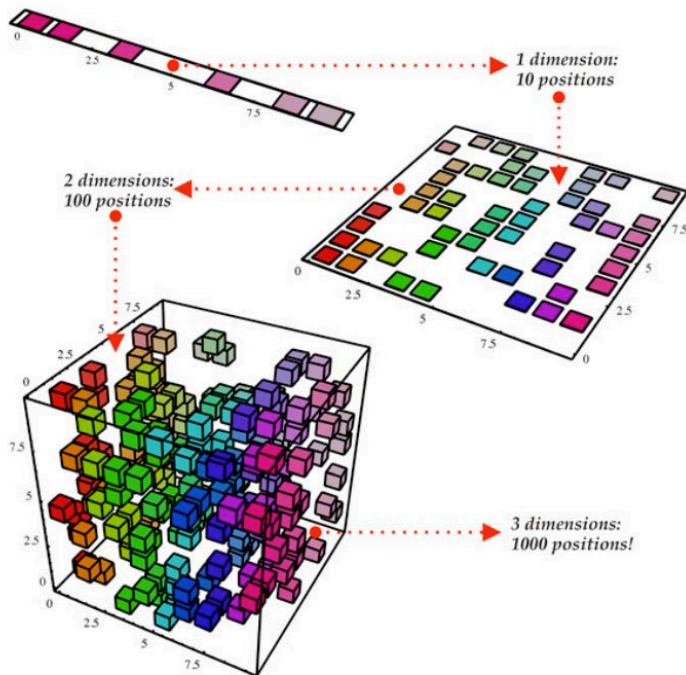
1. 데이터의 차원을 줄여서 불필요한 데이터를 제거하고
2. 데이터의 숨겨진 특징을 찾아내는 것
3. Manifold Learning

▶ PreTraining



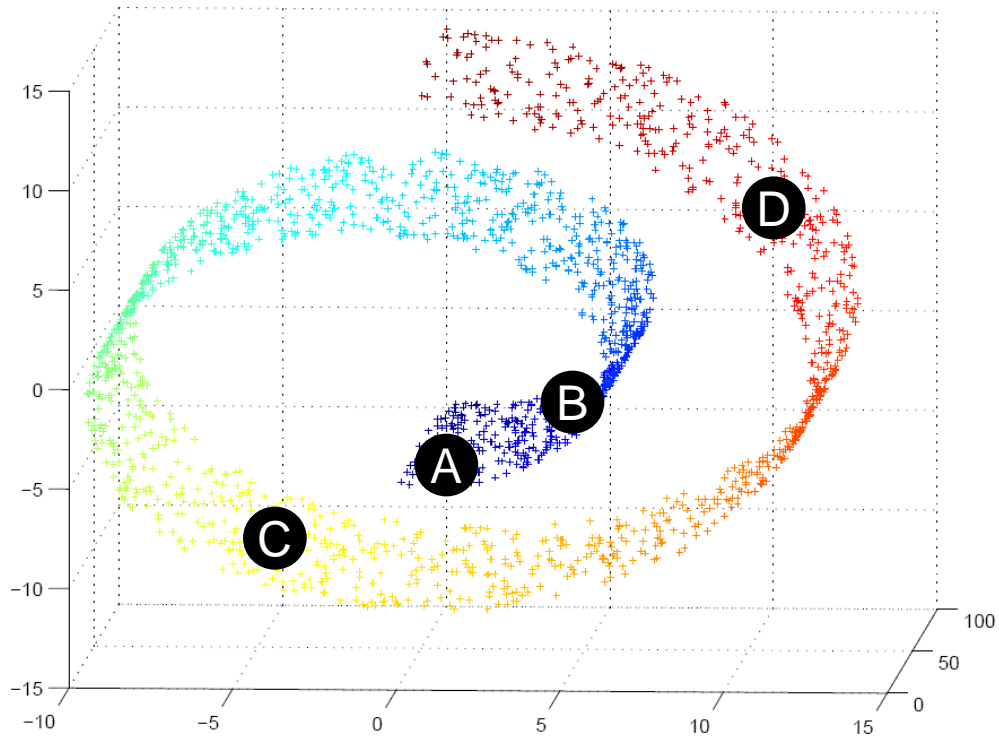
# Curse of Dimensionality

차원이 커질 수록, 해당 차원에 표현되는 데이터를 제대로 학습하기 위해 필요한 데이터의 양이 기하급수적으로 증가



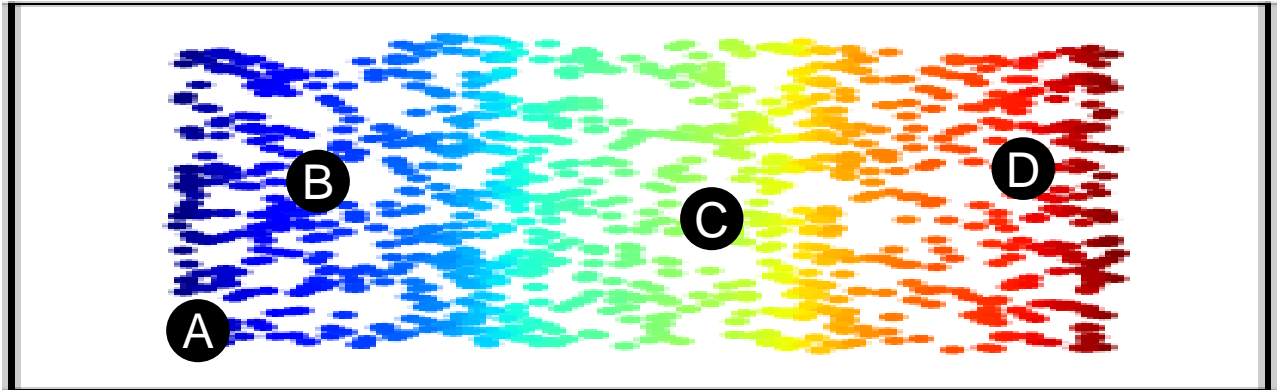
# Manifold

두 점 사이의 거리 혹은 유사도가 근거리에서는 유의미하지만 원거리에서는 그렇지 않은 공간



# Manifold

두 점 사이의 거리 혹은 유사도가 근거리에서는 유의미하지만 원거리에서는 그렇지 않은 공간

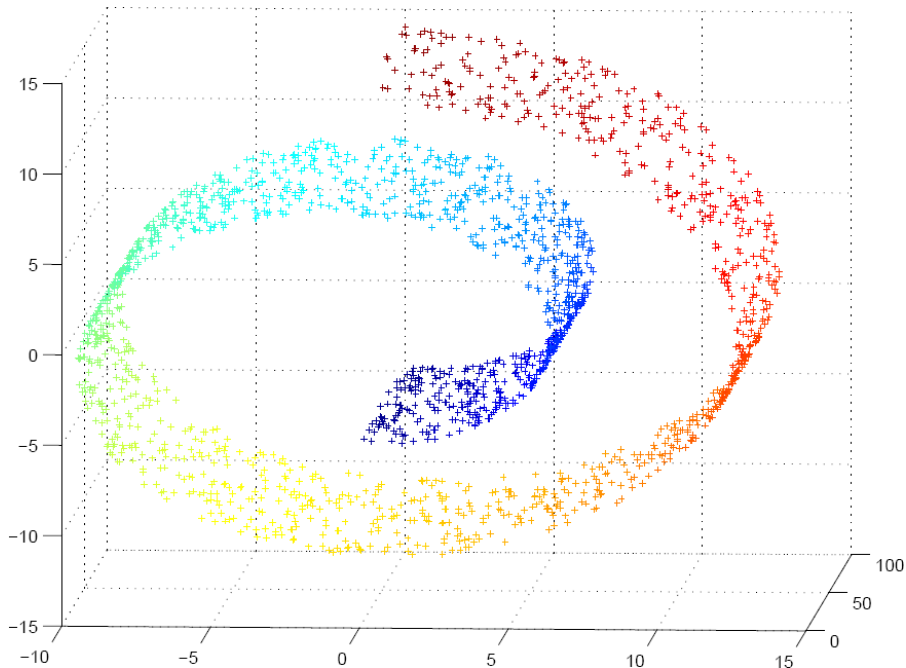


# Manifold Hypothesis

고차원의 데이터는 차원이 크지만, 고차원 데이터 분포를 표현하는 Manifold가 존재

대부분의 데이터는 Manifold 상에 존재하며, 이를 벗어나면 밀도가 낮아짐

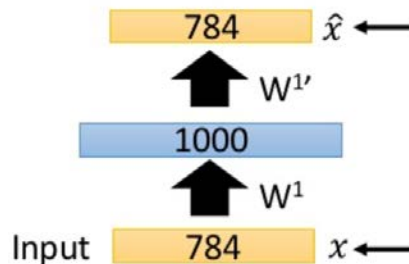
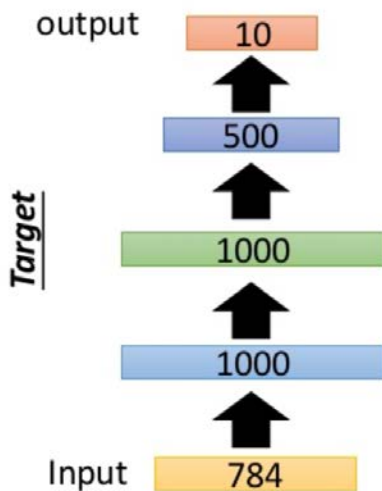
▶ 고차원 데이터를 저차원의 Manifold로 표현하면, 차원의 저주를 해결할 수 있을 것이다.



# PreTraining

정답 값인 label data를 이용하여 network를 training 시키기 이전에, 먼저 입력 값을 이용하여 network를 선행학습.

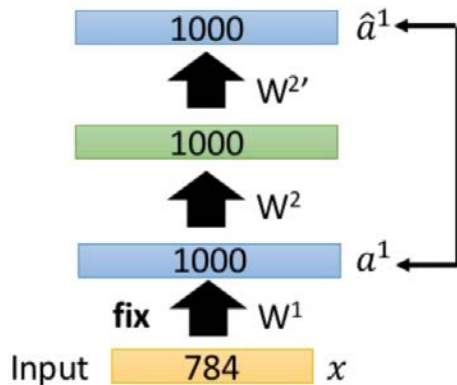
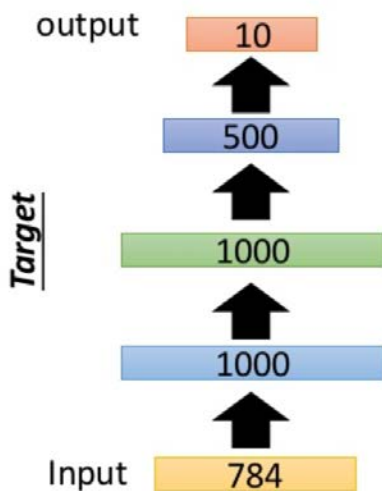
- ▶ 입력 데이터의 sparsity를 줄이기 위하여



# PreTraining

정답 값인 label data를 이용하여 network를 training 시키기 이전에, 먼저 입력 값을 이용하여 network를 선행학습.

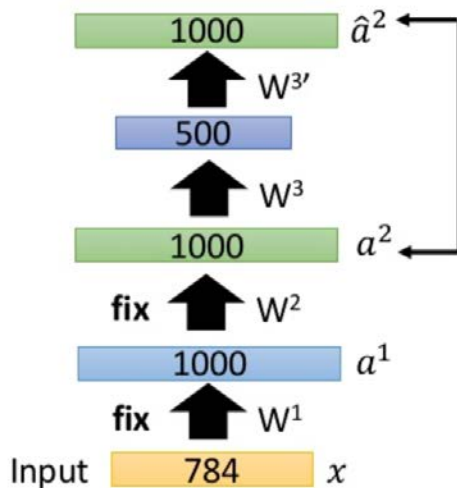
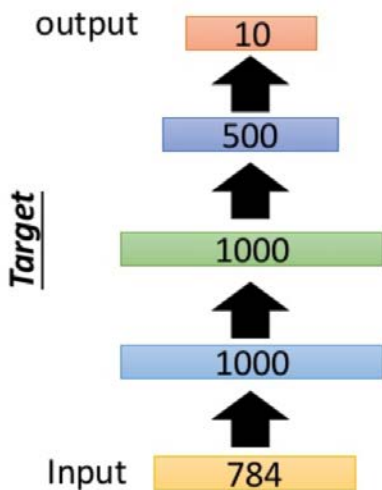
- ▶ 입력 데이터의 sparsity를 줄이기 위하여



# PreTraining

정답 값인 label data를 이용하여 network를 training 시키기 이전에, 먼저 입력 값을 이용하여 network를 선행학습.

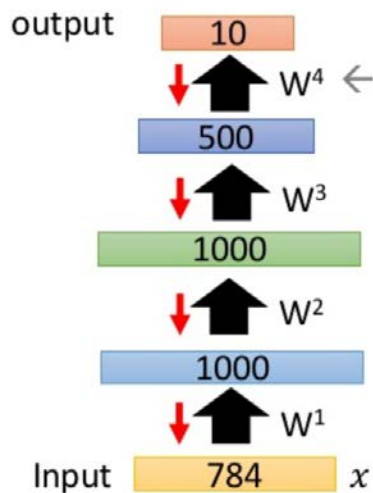
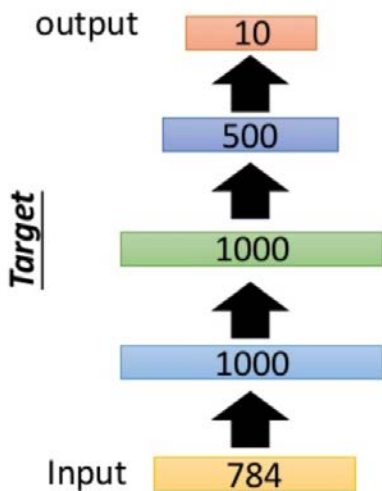
- ▶ 입력 데이터의 sparsity를 줄이기 위하여



# PreTraining

정답 값인 label data를 이용하여 network를 training 시키기 이전에, 먼저 입력 값을 이용하여 network를 선행학습.

- ▶ 입력 데이터의 sparsity를 줄이기 위하여

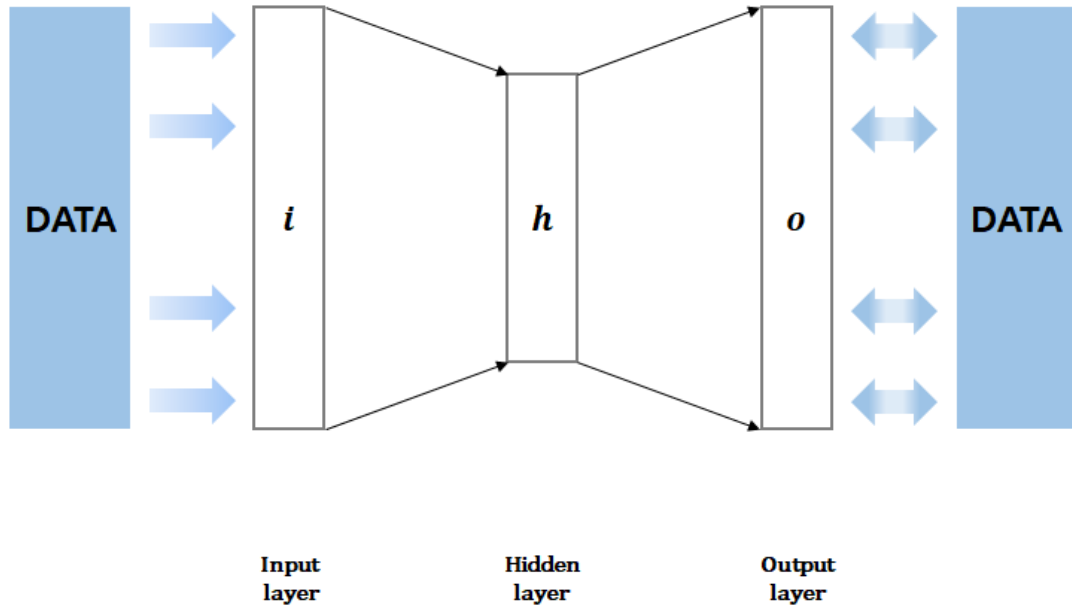




# Type of AutoEncoder

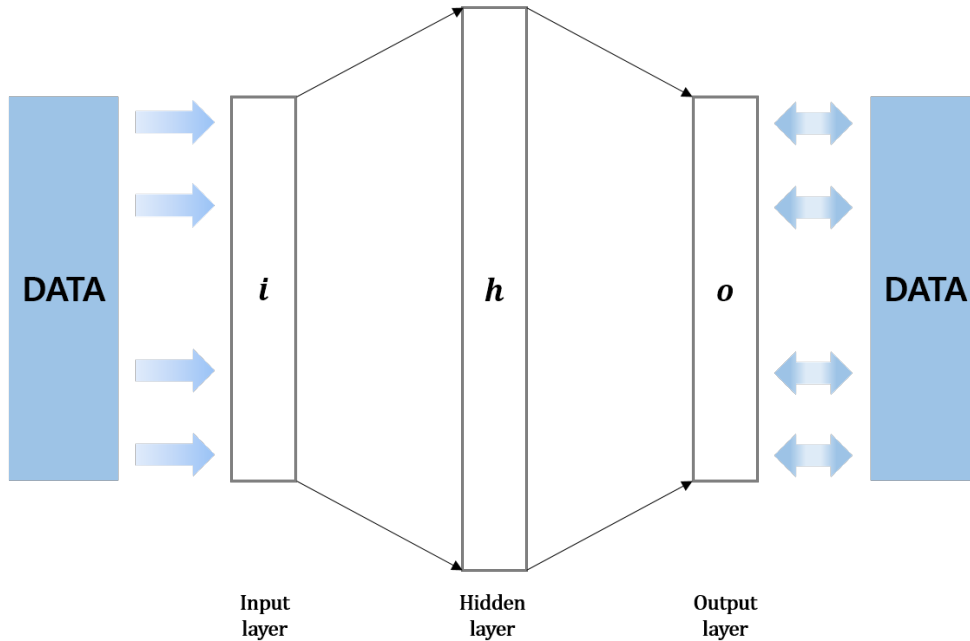
## Undercomplete Autoencoder

▶ 실제 데이터보다 불완전(undercomplete)하게 데이터를 인코딩하는 작업을 수행하여 데이터의 sparsity를 해소



# Type of AutoEncoder

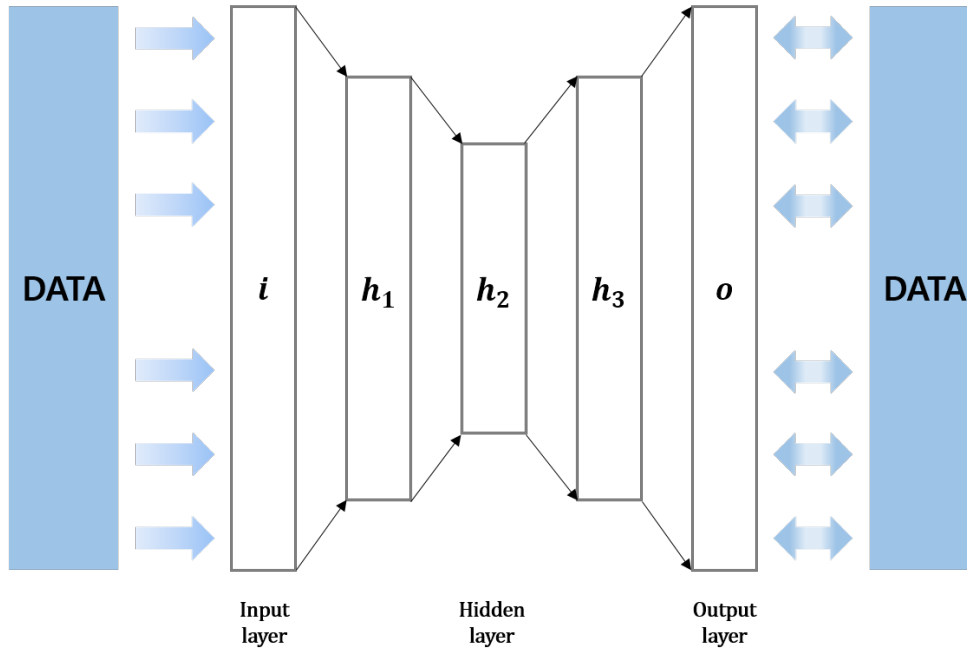
## Overcomplete Autoencoder



# Type of AutoEncoder

## Stacked Autoencoder

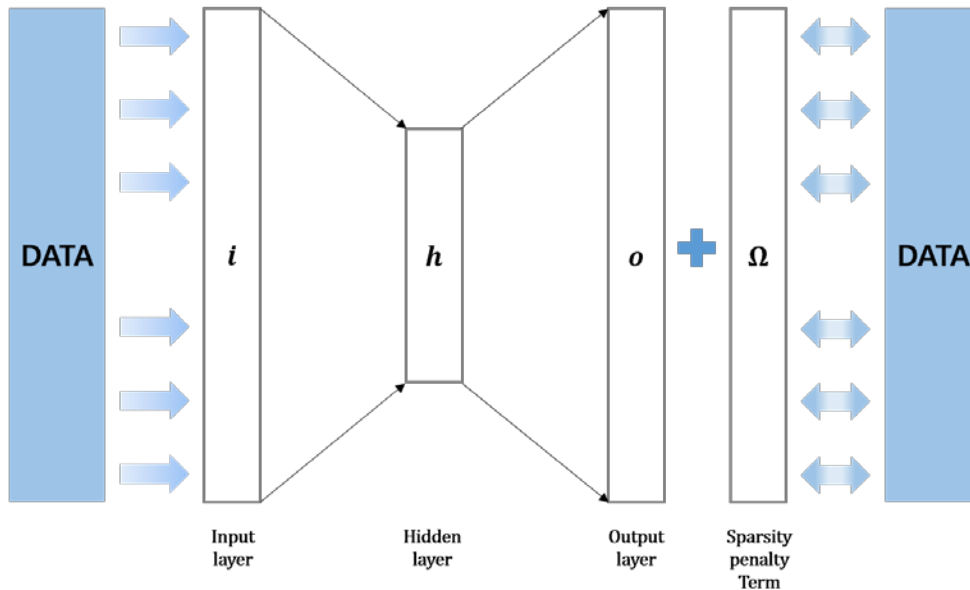
- ▶ 인코딩 작업을 단계적으로 수행. 좀 더 정교하게 데이터를 encoding, decoding 할 수 있음



# Type of AutoEncoder

## Sparse Autoencoder

▶ 데이터 분포도를 표현하는 sparsity penalty term을 계산하여, 데이터의 sparsity를 줄이는 방향으로 학습 유도



# Type of AutoEncoder

## Denoising Autoencoder

▶ 입력값에 의도적으로 noise 데이터를 추가하여 모델을 학습시킴. Noise에 tolerant한 일반화된 모델을 학습하는 것이 가능

