

Auto-Encoder

AILAB

Hanyang Univ.

Why?

- Data Compression
데이터 압축
- Data Visualization
데이터 가시화
- Curse of dimensionality
차원의 저주 해결
- Discovering most important features
가장 중요한 피쳐 찾기

Why?

- **Data Compression**

데이터 압축

- Data Visualization

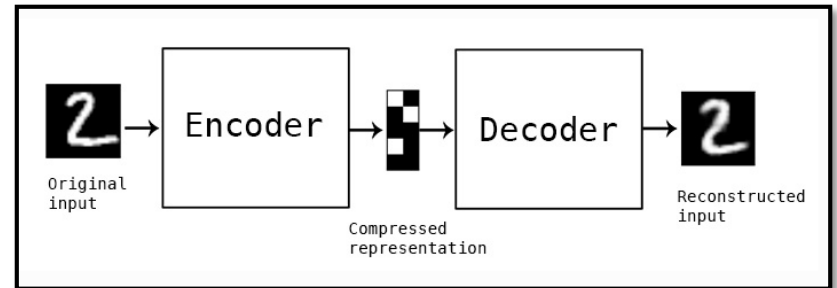
데이터 가시화

- Curse of dimensionality

차원의 저주 해결

- Discovering most important features

가장 중요한 피쳐 찾기



Why?

- Data Compression

데이터 압축

- **Data Visualization**

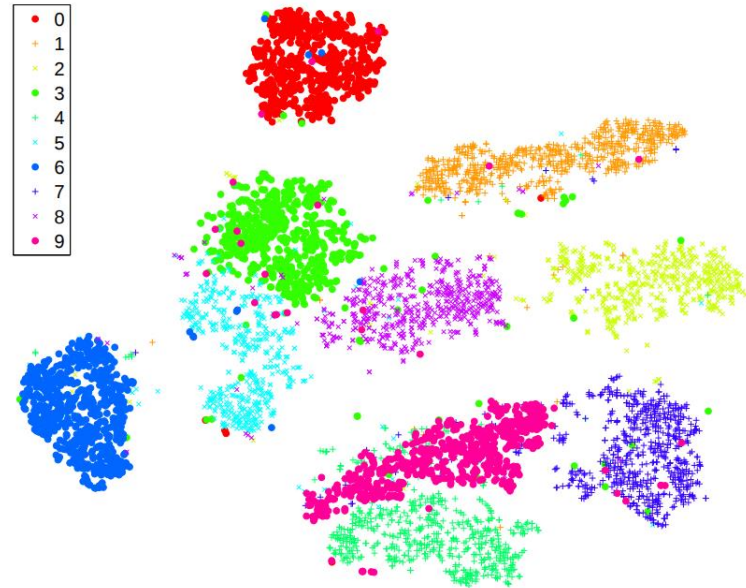
데이터 가시화

- Curse of dimensionality

차원의 저주 해결

- Discovering most important features

가장 중요한 피쳐 찾기



Why?

- Data Compression

데이터 압축

- Data Visualization

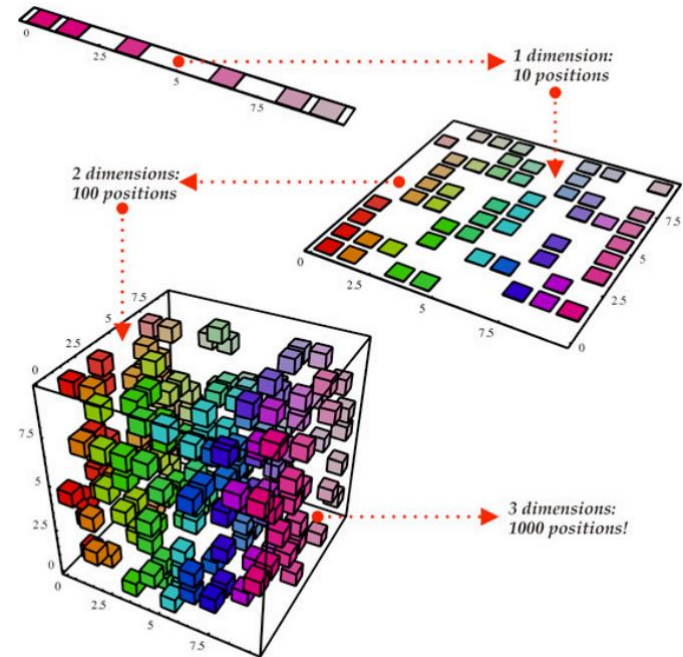
데이터 가시화

- **Curse of dimensionality**

차원의 저주 해결

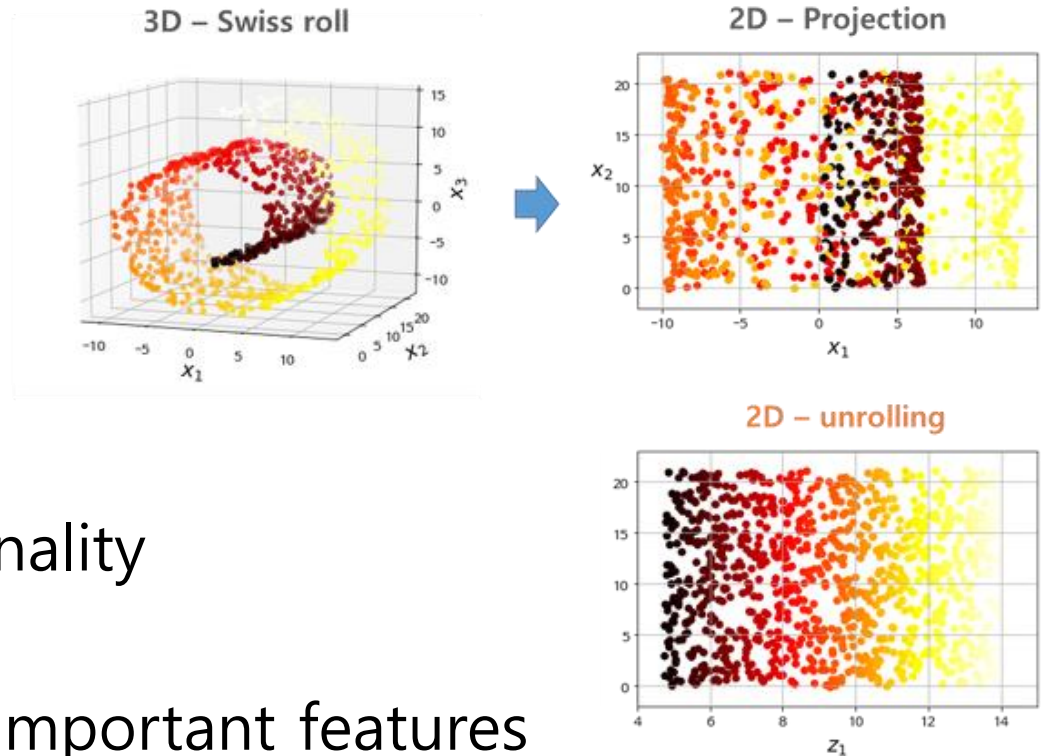
- Discovering most important features

가장 중요한 피쳐 찾기

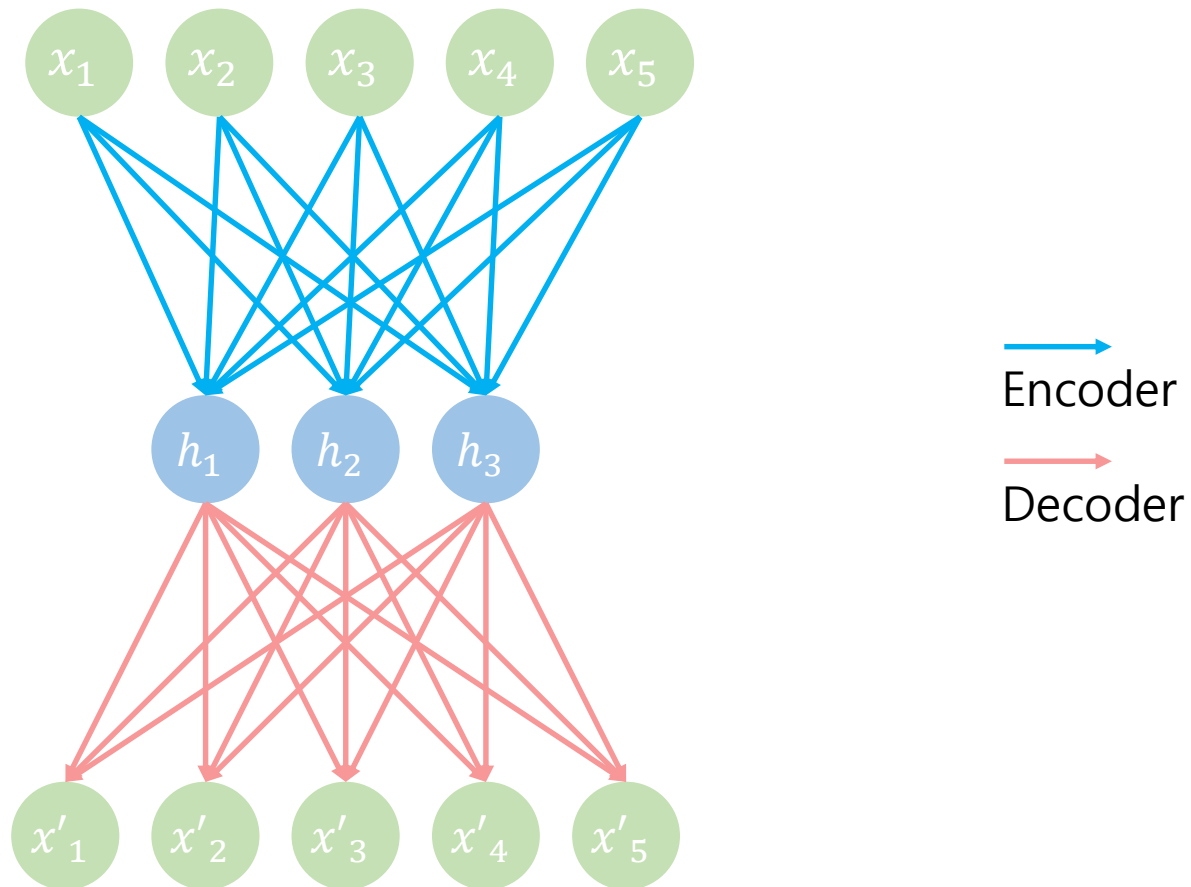


Why?

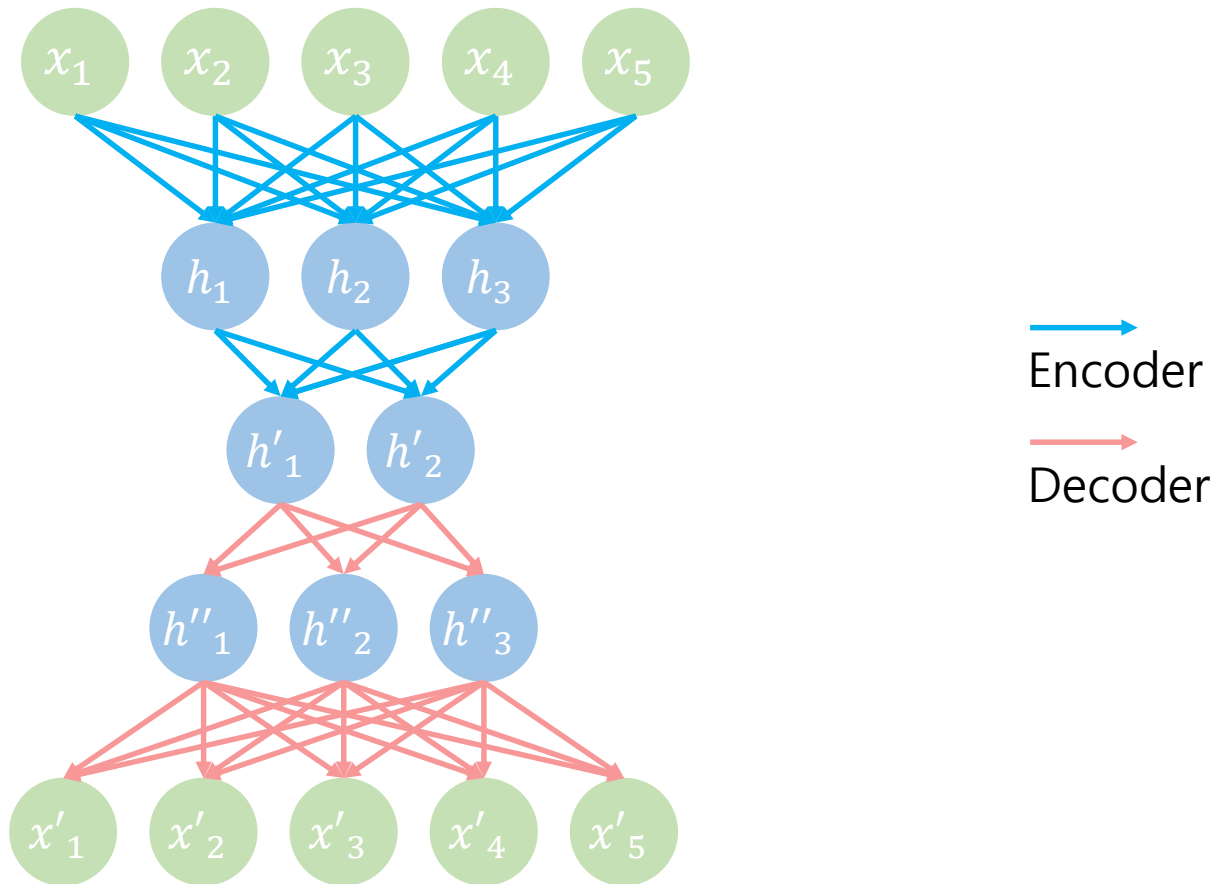
- Data Compression
데이터 압축
- Data Visualization
데이터 가시화
- Curse of dimensionality
차원의 저주 해결
- Discovering most important features
가장 중요한 피쳐 찾기



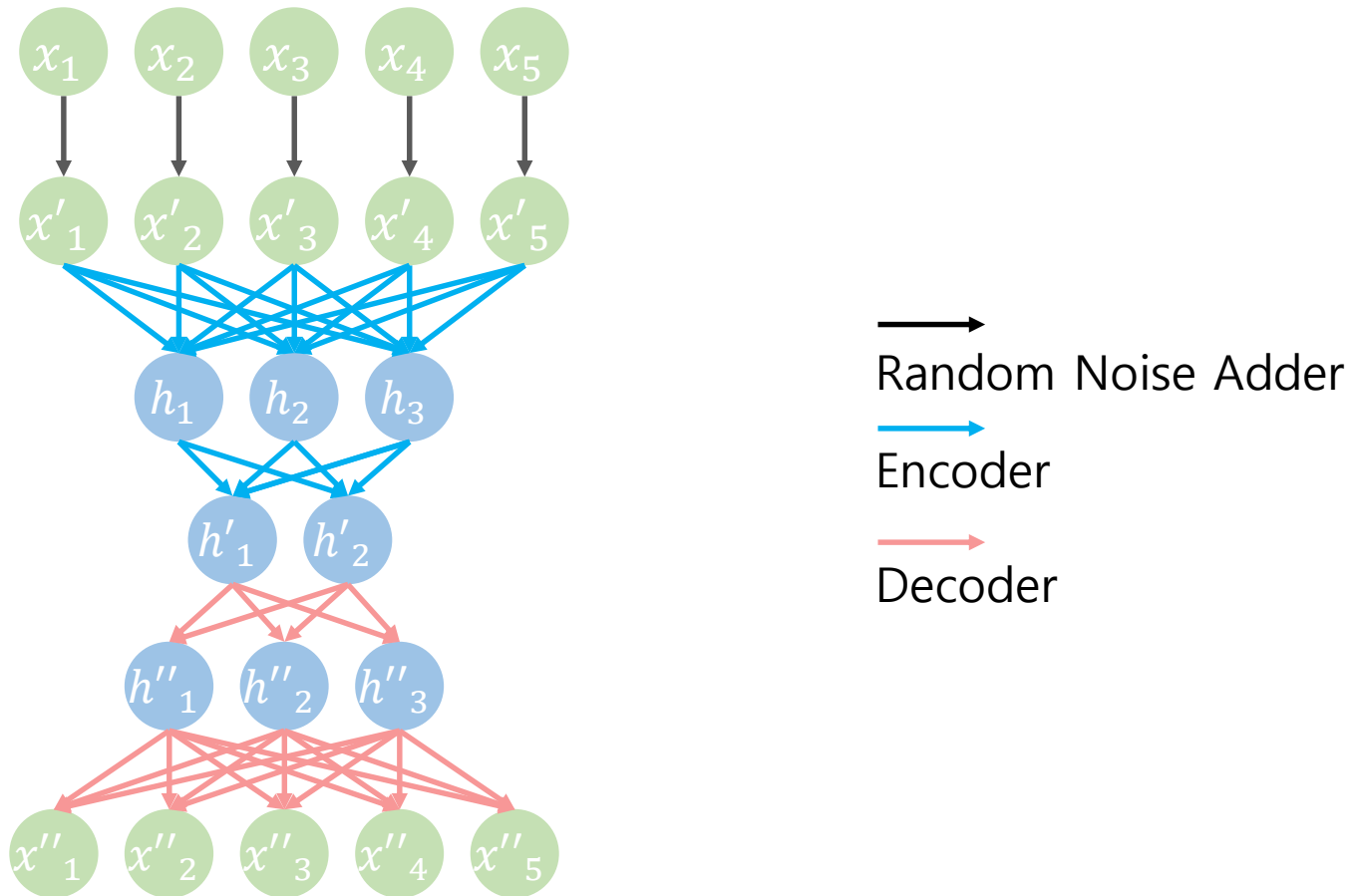
Auto-Encoder model



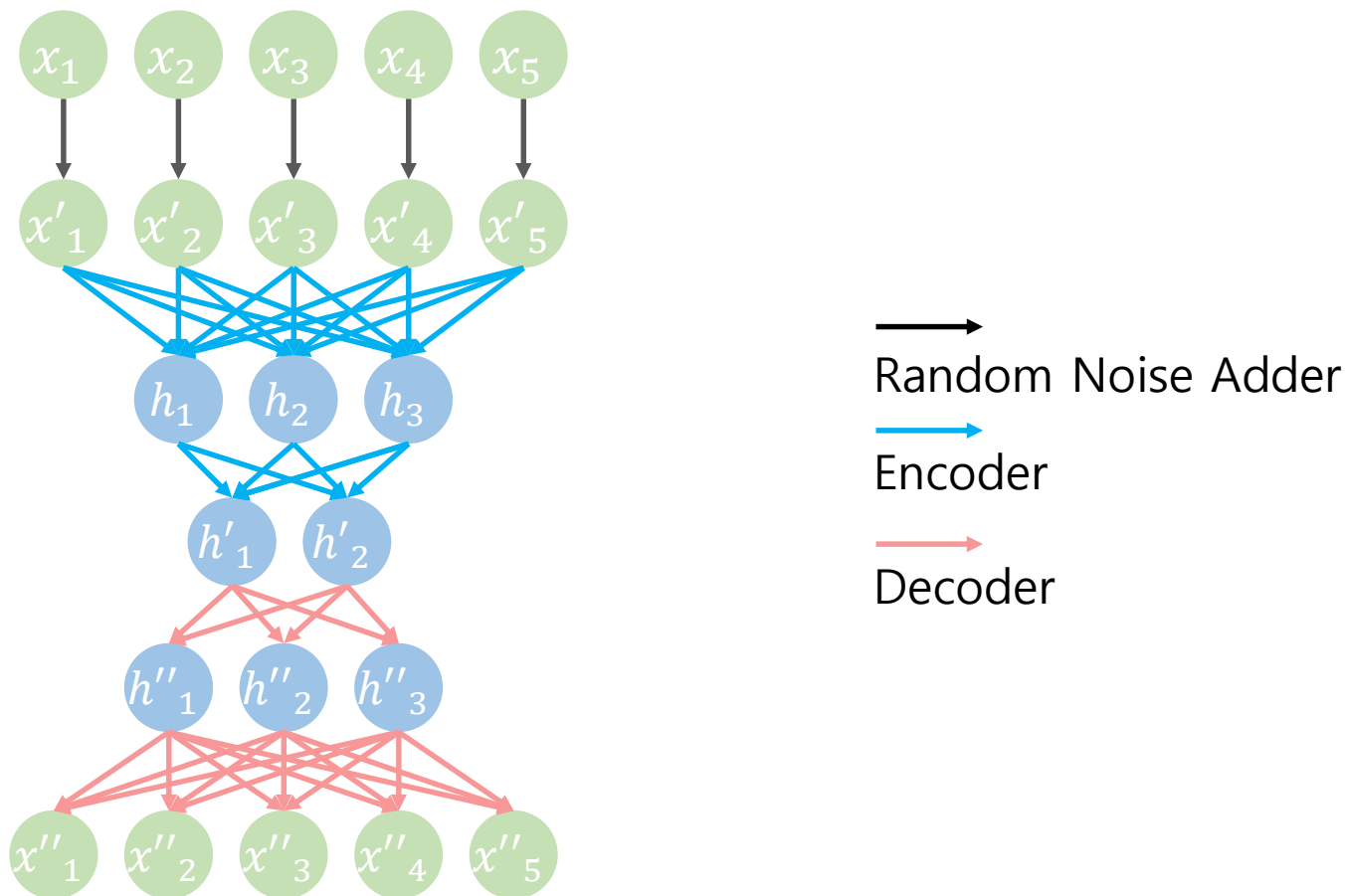
Stacked Auto-Encoder model



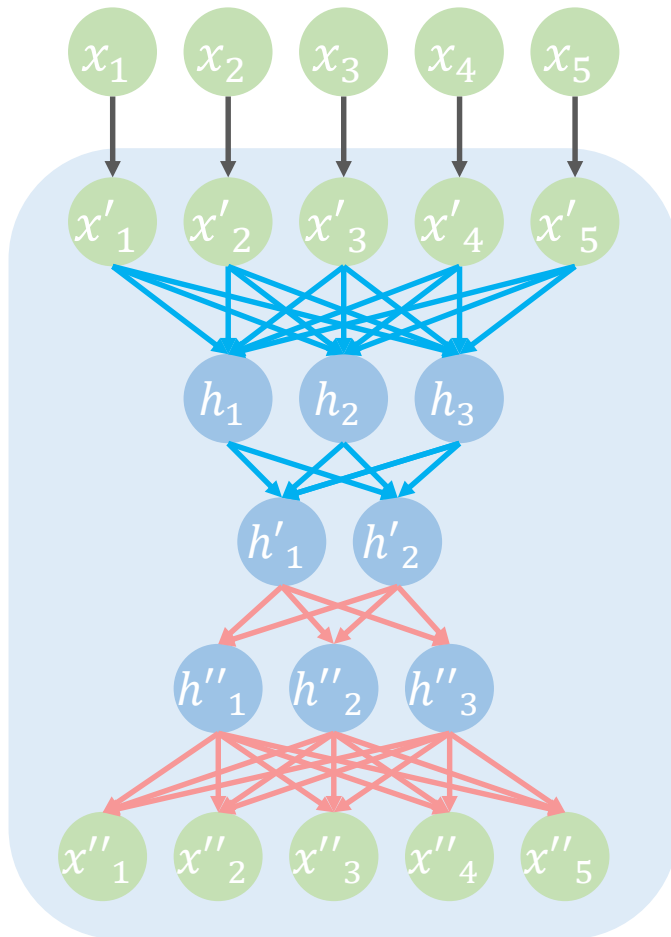
Denoising Auto-Encoder model



과제 : Denoising Auto-Encoder



Denoising Auto-Encoder model



InputSize : 784(28x28)

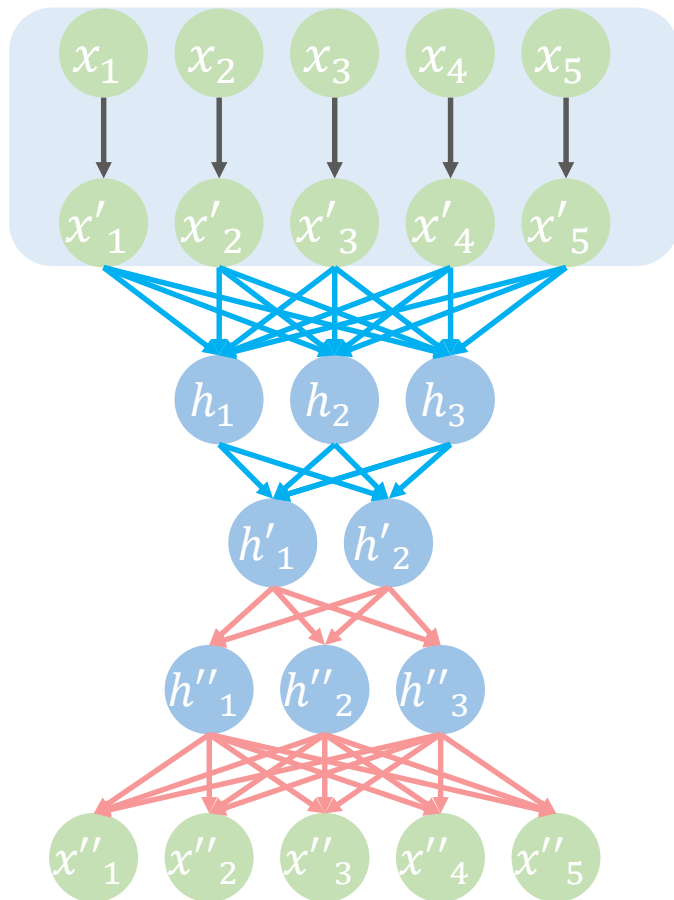
Encoder model (more than 2 layers)

Feature vector (64-d)

Decoder model (more than 2 layers)

OutputSize : 784(28x28)

Denoising Auto-Encoder model



Random Noise 생성

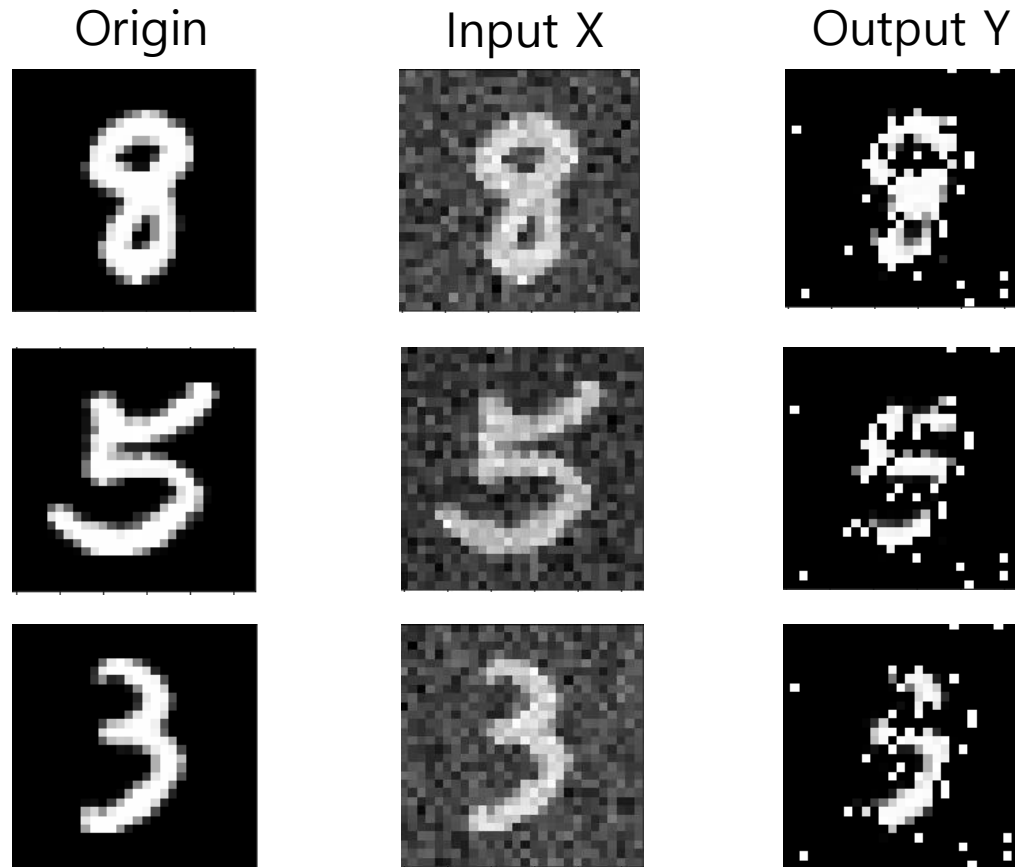
`batch_x_noise = batch_x + random noise`

#학습

`feed_dict={X:batch_x_noise, Y:batch_x}`

`sess.run(optimizer, feed_dict=feed_dict)`

Denoising Auto-Encoder Implement



과제

- 소스와 결과 캡처 GitLab에 제출
- 과제 기한 : **다음주 수요일 23:59** 까지
- 수업시간에 한 경우 바로 검사받고 **GitLab**에 제출
- GitLab 관련 사용법은 첨부 파일 확인