이상탐지(Anomaly Detection)

- 데이터 안에서 anomaly, outlier, abnormal과 같이 예상하지 못한 패턴(데이터 또는 이벤트)을 찾는 기법.
- cf. noise detection
  - 잡음은 분석 이전 단계에서 제거해야 할 성분.
  - 없어야 데이터 분석을 신뢰성 있게 할 수 있음.

학습데이타에 따른 분류

- supervised
  - 주어진 학습 데이터 셋에 정상 sample과 비정상 sample의 Data와 Label이 모두 존재하는 경우
  - 장점 : 양/불 판정 정확도가 높다.
  - 단점 : 비정상 sample을 취득하는데 시간과 비용이 많이 든다. / Class-Imbalance 문제를 해결 필요.
  - 분류 문제와 다른 점 - 매우 많은 정상 데이터에서 극소수의 비정상 데이터를 구별해야 함.
  - 해결 방법 : Data Augmentation(증강), Loss function 재설계, Batch Sampling 등.

- semi-supervised
  - 정상 sample만을 사용해서 모델을 학습.
  - 대표적인 알고리즘은 One-Class SVM이 있으며, 동작 원리는 정상 sample들을 둘러싸는 discriminative boundary를 설정하고, 이 boundary를 최대한 좁혀 boundary 밖에 있는 sample들을 모두 비정상으로 간주함.
  - 장점 : 정상 Sample만 있어도 학습이 가능.
  - 단점 : Supervised Learning 대비 정확도가 떨어진다.

학습데이타에 따른 분류

- unsupervised
  - Label이 존재하지 않을 때, 대부분의 데이터를 정상 Sample이라 가정하고 학습시키는 방식.
  - 장점 : Labeling과정이 필요하지 않다.
  - 단점 : 양/불 판정 정확도가 높지 않다. / hyper parameter에 매우 민감.

- AutoEncoder
  - Autoencoder의 code size (= latent variable의 dimension) 같은 hyper-parameter에 따라 전반적인 복원 성능이 좌우되기 때문에 양/불 판정 정확도가 Supervised Anomaly Detection에 비해 다소 불안정.
  - autoencoder에 넣어주는 input과 output의 차이를 어떻게 정의할 것인지(= 어떤 방식으로 difference map을 계산할지) 어떤 loss function을 사용해 autoencoder를 학습시킬지 등 여러 가지 요인에 따라 성능이 달라짐.

활용사례

- Cyber-Intrusion Detection:
  - 컴퓨터 시스템 상에 침입을 탐지하는 사례.
  - 주로 시계열 데이터를 다루며 RAM, file system, log file 등 일련의 시계열 데이터에 대해 이상치를 검출하여 침입을 탐지함.
- Fraud Detection:
  - 보험, 신용, 금융 관련 데이터에서 불법 행위를 검출하는 사례. 주로 표로 나타낸(tabular) 데이터
- Malware Detection:
  - Malware(악성코드)를 검출해내는 사례.
- Medical Anomaly Detection:
  - 의료 영상, 뇌파 기록 등의 의학 데이터에 대한 이상치 탐지 사례.
  - 주로 신호 데이터와 이미지 데이터를 다루며 X-ray, CT, MRI, PET 등 다양한 장비로부터 취득된 이미지를 다루기 때문에 난이도가 높음.
- Social Networks Anomaly Detection:
  - 주로 Text 데이터를 다루며 Text를 통해 스팸 메일, 비매너 이용자, 허위 정보 유포자 등을 검출함.
- Log Anomaly Detection:
  - 시스템이 기록한 log를 보고 실패 원인을 추적하는 사례.
  - 주로 Text 데이터를 다루며, failure message가 새로운 것이 계속 추가, 제외가 되는 경우에 딥러닝 기반 방법론을 사용하는 것이 효과적임.
- IoT Big-Data Anomaly Detection:
  - 사물 인터넷에 주로 사용되는 장치, 센서들로부터 생성된 데이터에 대해 이상치를 탐지하는 사례.
- Industrial Anomaly Detection:
  - 외관상에 발생하는 결함과, 장비의 고장 등의 비정상적인 sample 감지.

이상탐지에 사용되는 모델들

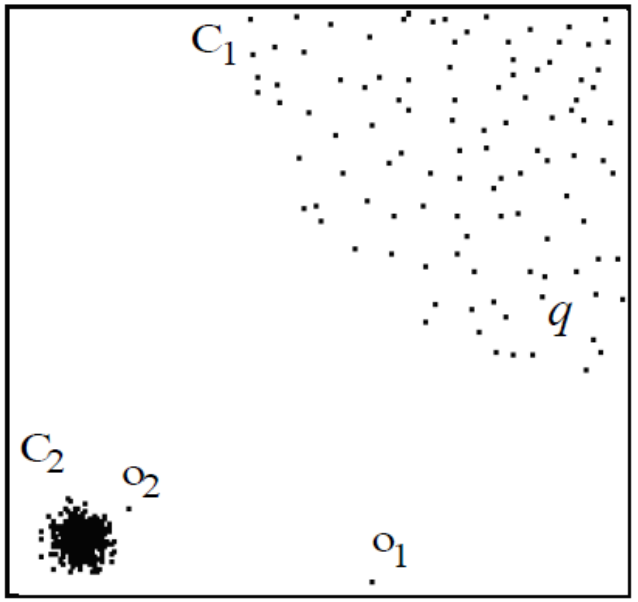
- 1. Model-based Methods
  - Isolation Forest : Tree based method로서 데이터를 분할 및 고립시켜 이상치를 탐지
  - 1-class SVM : 데이터가 존재하는 영역을 정의하여, 영역 밖의 데이터들은 이상치로 간주

- 2. Density/Distance-based Methods
  - Gaussian Mixture Model
  - k-Nearest Neighbours (kNN) method
  - LOF(Local Outlier Factors)

- 3. Reconstruction-based Methods
  - PCA(Principal Component Analysis) Method
  - Auto-Encoder based Method

- https://velog.io/@vvakki_/Anomaly-Detection%EC%9D%B4%EC%83%81%EC%B9%98-%ED%83%90%EC%A7%80%EB%9E%80

lof 모델 개념도



PyOD 소개

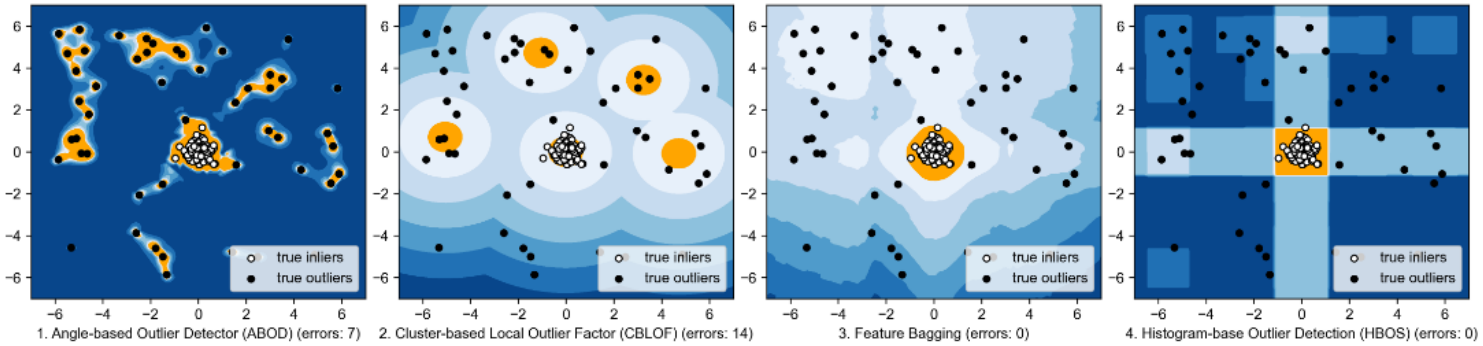
- Python Outlier Detection
- 다변량 데이터에서 이상치를 탐지하는 알고리즘 제공
- 40여개 이상의 발표된 이상탐지 알고리즘을 통일된 API로 제공
- https://github.com/yzhao062/pyod

5 줄 코드로 이상치 분석 완료

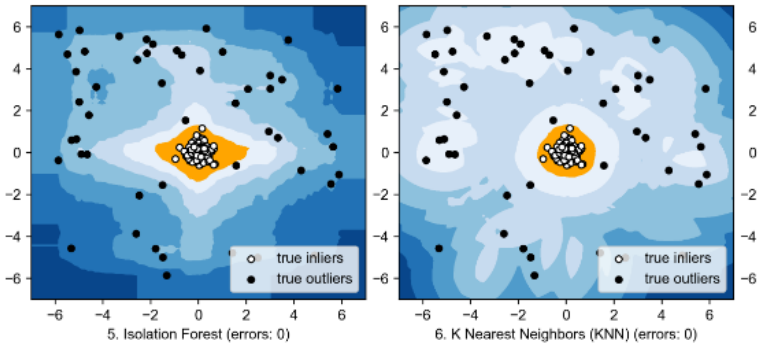
```
# train an ECOD detector
from pyod.models.ecod import ECOD
clf = ECOD()
clf.fit(X_train)

# get outlier scores
y_train_scores = clf.decision_scores_  # raw outlier scores on the train data
y_test_scores = clf.decision_function(X_test)  # predict raw outlier scores on test
```

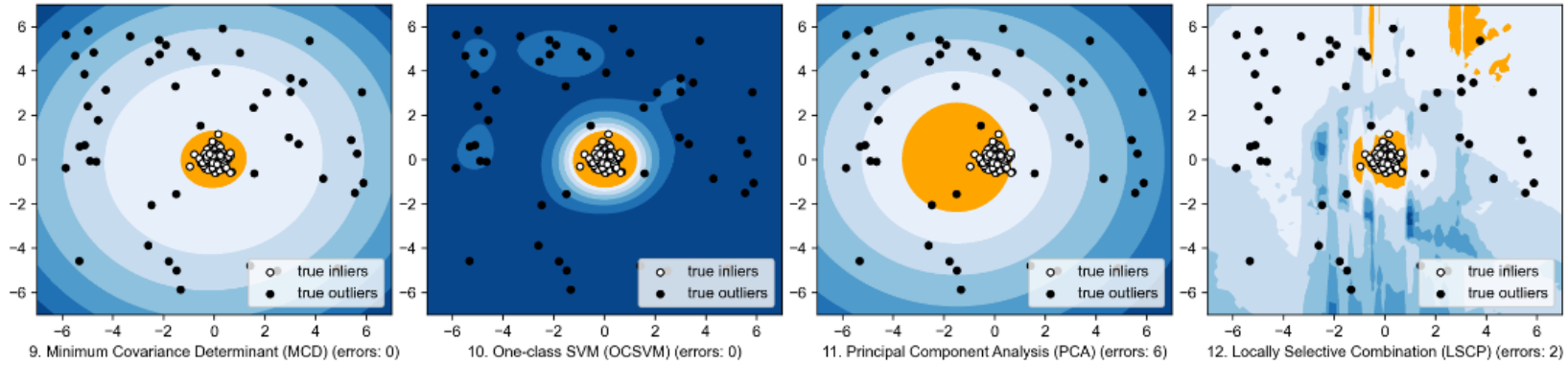
PyOD의 분석 결과



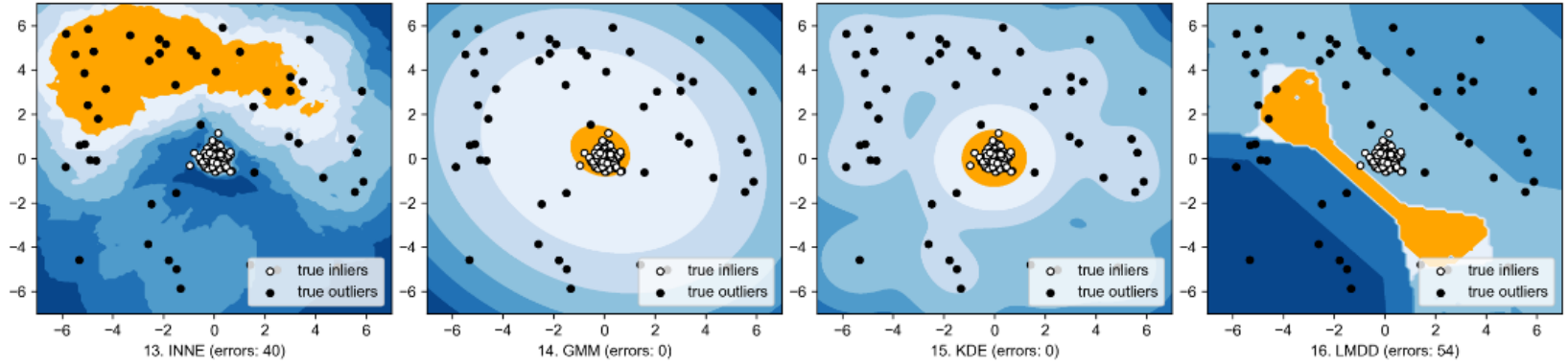
1. Angle-based Outlier Detector (ABOD) (errors: 7)
2. Cluster-based Local Outlier Factor (CBLOF) (errors: 14)
3. Feature Bagging (errors: 0)
4. Histogram-base Outlier Detection (HBOS) (errors: 0)



5. Isolation Forest (errors: 0)
6. K Nearest Neighbors (KNN) (errors: 0)



9. Minimum Covariance Determinant (MCD) (errors: 0)
10. One-class SVM (OCSVM) (errors: 0)
11. Principal Component Analysis (PCA) (errors: 6)
12. Locally Selective Combination (LSCP) (errors: 2)

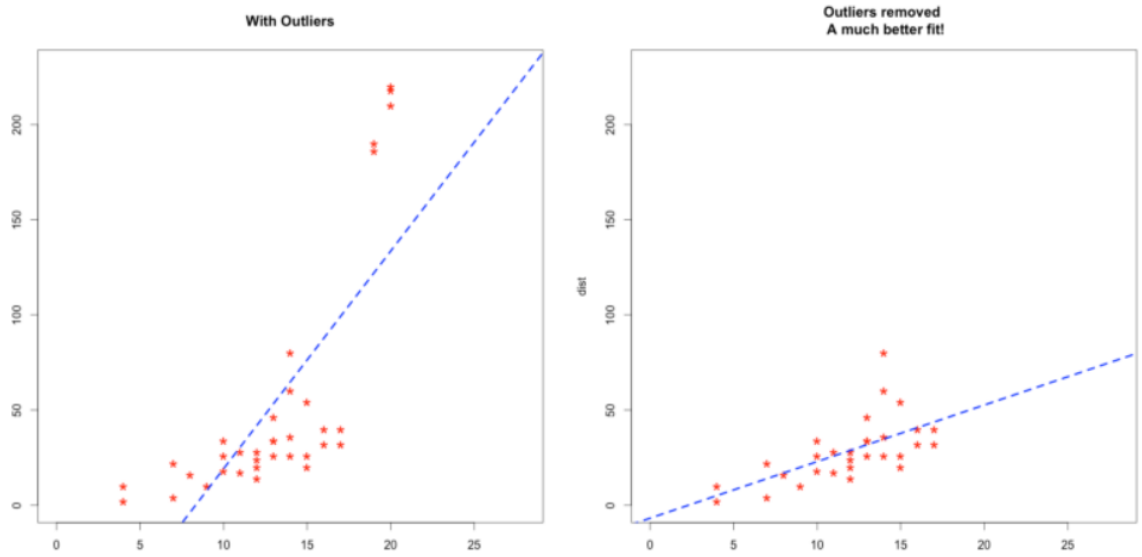


13. INNE (errors: 40)
14. GMM (errors: 0)
15. KDE (errors: 0)
16. LMDD (errors: 54)

PyOD 실습

- 1. pyod_BigMartSales.ipynb
- 원데이타의 목적 : 특정 마트(outlet)에서 어떤 상품이 얼마나 팔릴지 예측.
- 예측모델의 신뢰성을 높이기 위해 이상치를 먼저 제거하여 좀 더 질좋은 학습데이타 확보.
- 가격대별 판매량 데이터를 대상으로 이상치 추정.
- unsupervised 유형이며 통계모델에 따라 기준을 벗어난 데이타포인트를 이상치로 간주함.
- 실습환경
  - caret 가상환경 사용
  - $ pip install pyod combo

- 데이터
  - Big Mart Sales data
  - 1559 products across 10 stores in different cities.
  - ItemMRP : 상품의 최대 소매가격.
  - ItemOutletSales : 예측대상인 특정 점포에서의 판매량.



다변량 데이터에서의 이상치의 영향

Pycaret Anomaly Detection 실습

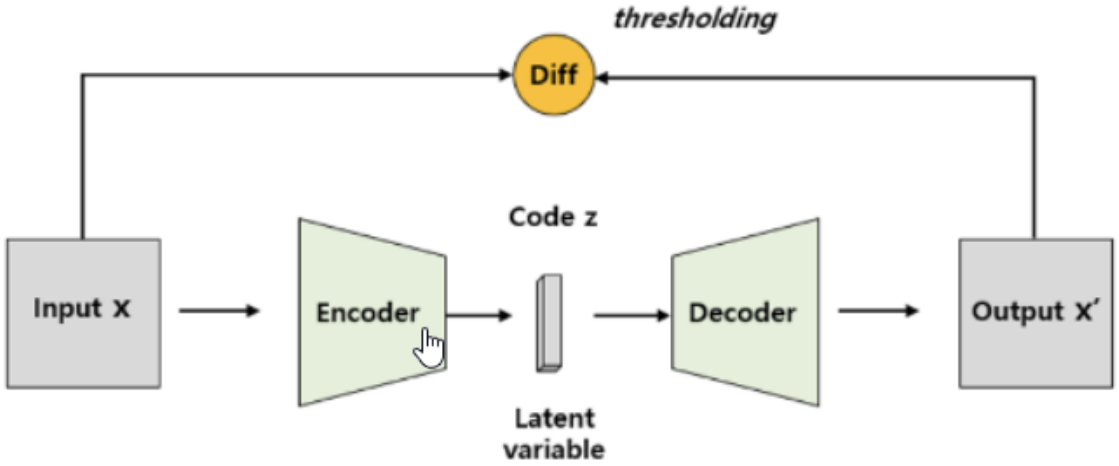
- 2. Anomaly Detection Tutorial.ipynb
- 원래 목적 : 8개의 클래스를 결정하는 단백질 요인을 판별.
- 이상치 탐지의 목적 : 목적을 달성하기 위해 좀 더 안정적인 데이터 확보.

- 데이터 개요
  - 정상 실험쥐 38마리, 다운증후군 실험쥐 34마리.
  - 각 실험쥐당 15 건의 단백질 측정 샘플로 구성됨.
  - 1080 개의 샘플(38마리 x 15 건 + 34마리 x 15건).
- 주요 특성
  - 2:78 : 단백질 발현값(?)
  - 79 : 유전자형(control / trisomic)
  - 80 : 처치 유형(saline / memantine)
  - 81 : behavoir (context-shock / shock-context)
  - 82 : class, 79~81 종합(c-CS-s, c-CS-m, c-SC-s, c-SC-m, t-CS-s, t-CS-m, t-SC-s, t-SC-m)

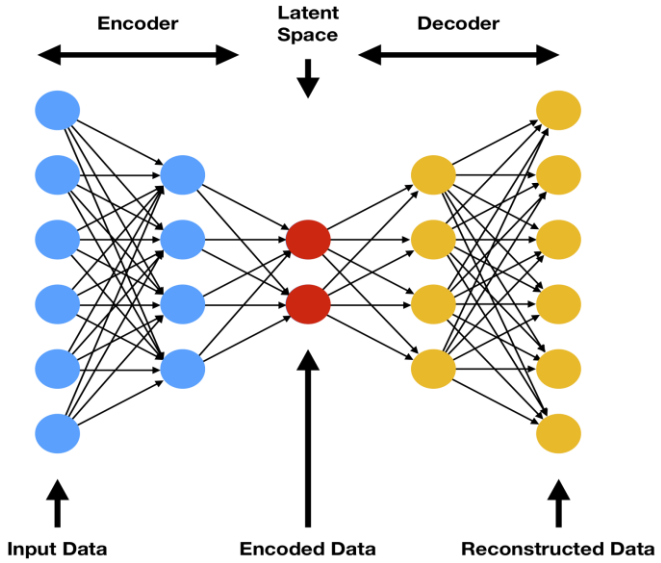
- 실습환경
  - caret 가상환경 사용

AutoEncoder

- 특징
  - Unsupervised Learning(?)
  - 입력과 출력층의 뉴런 수가 동일한 신경망 모델
  - Input Layer부터 Bottleneck layer 까지를 Encoder라 하며, 입력을 내부 표현으로 변환하는 과정.
  - Bottleneck layer 부터 Output Layer 까지를 Decoder라 하며, 내부 표현을 출력으로 변환하는 과정.
  - 가운데 Bottleneck layer를 기점으로 나비모양의 대칭구조.
  - Bottleneck layer는 Latent Variable, Feature, Hidden representation 로도 칭함.



손실함수 : $\frac{1}{N}\sum_{i=1}^{N}(X_i - \hat{X}_i)^2$



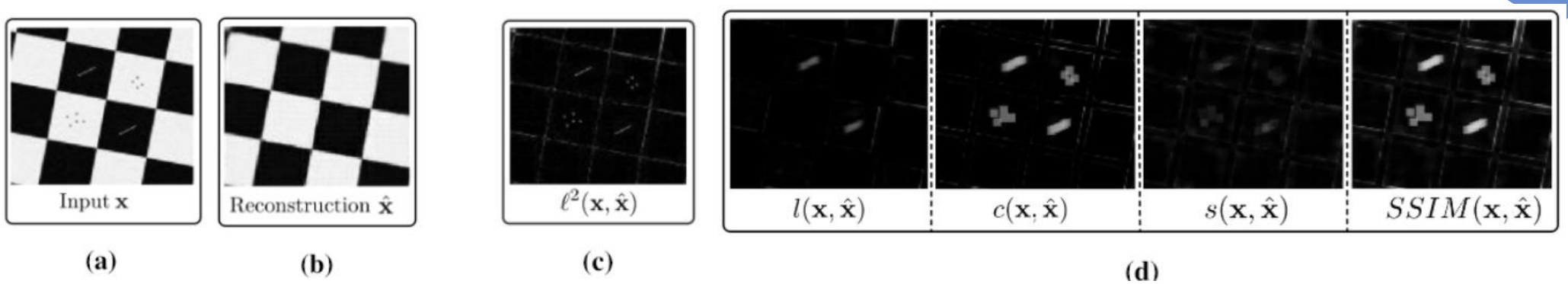
AutoEncoder의 작동

- 작동원리
  - 오토인코더의 특성이 입력 데이터의 가장 중요한 특징을 학습하는 것이므로, noise가 아닌 중요 특징에 대해서만 학습.
  - Loss function은 입력과 출력의 차이(MSE)로 계산. 즉, 입력과 출력이 동일해지도록 학습.
  - 비정상 sample이 정상 sample에 비해 복원 loss(=MSE)가 큼.
  - 복원 loss가 일정 threshold를 넘으면 비정상, 그렇지 않으면 정상으로 판정.
  - 입력층과 출력층 사이 hidden layer의 dimension은 hyper parameter로, 몇 층을 쌓을지, 몇 개의 뉴런으로 할지는 성능과 목적에 따라 지정하면서 튜닝.

- MNIST 의 경우
  - 숫자가 존재하지 않는 대부분의 pixel들은 굳이 기억하지 않을 것임.
  - 숫자정보가 존재하는 pixel들만 좀 더 효율적으로 기억하도록 학습.

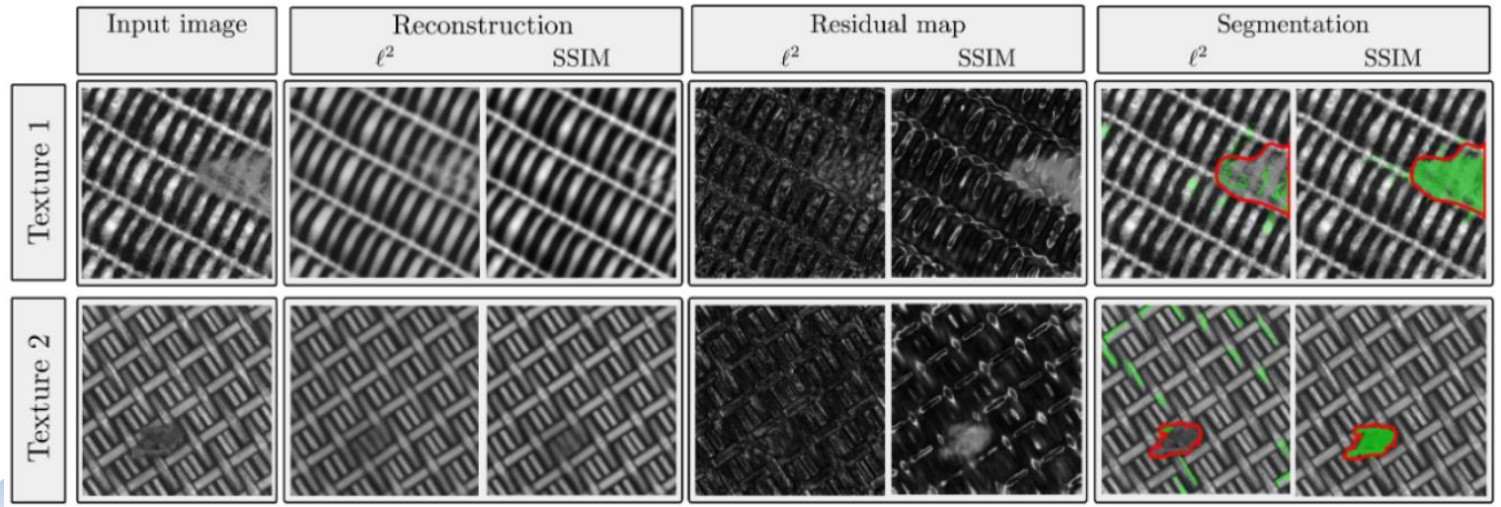
- 병목 구간이 최적보다 클 경우
  - 불필요한 입력특성도 기억하므로 비정상 데이터의 변별력이 떨어질 것임.
  - 입력 샘플과 같은 크기로 병목구간 설정하면 identity function이 되어 입력을 그대로 복사할 것이므로,
  - 학습 과정에서 보지 못했던 비정상 샘플이 주어져도 정상으로 인식할 것임.

AutoEncoder 적용 예



(a) 원본 이미지 (b) 복원 이미지 (c) l^2 잔차맵 (d) SSIM 잔차맵

직물의 이상탐지 사례



Improving Unsupervised Defect Segmentation by Applying Structural Similarity To Autoencoders
Paul Bergmann, Sindy Lowe, Michael Fauser, David Sattlegger, and Carsten Steger feb, 2019

AutoEncoder 실습

- 3. TF_autoencoder.ipynb
  - autoencoder 구조를 적용한 다양한 응용
  - 가상환경 : autokeras 가상환경 사용

- 예제 1 – 원본복원
  - 데이터 : fashion-MNIST
  - autoencoder를 통하여 차원축소(784 -> 64) –> 복원(64 -> 784)
  - 원본 이미지 복원성능 확인

- 예제 2 – 잡음 제거
  - 데이터 : fashion-MNIST
  - 원본 데이타에 잡음 추가
  - conv2D layer 를 적용하여 autoencoder 구현
  - 잡음 섞인 이미지에서 원본 이미지 복원성능 확인

- 예제 3 – 이상감지
  - 심전도 데이터로부터 이상발생시 감지
  - 데이터 : 140개 데이터 포인트로 구성된 5,000 개의 심전도 데이터
  - 정상 레이블 데이터로 훈련.
  - 이상 데이타로 테스트시 복원 데이타와의 차이를 이용해 이상여부 감지.