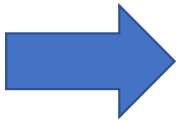
FDS

2021-04-26
백주엽

Possible ways

1. Anomaly detection  검토 대상

2. Binary classification (with up-sampling method)

+) graph model

Review of anomaly model

Anomaly based

A Unifying Review of Deep and Shallow Anomaly Detection

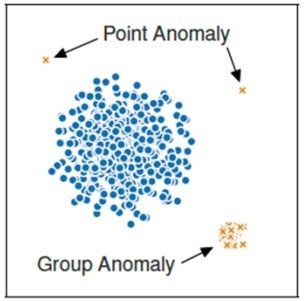
Lukas Ruff, Jacob R. Kauffmann, Robert A. Vandermeulen, Grégoire Montavon, Wojciech Samek, *Member, IEEE*, Marius Kloft*, *Senior Member, IEEE*, Thomas G. Dietterich*, *Member, IEEE*, Klaus-Robert Müller*, *Member, IEEE*.

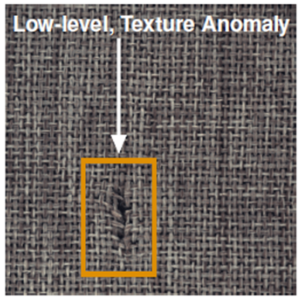
https://arxiv.org/pdf/2009.11732.pdf

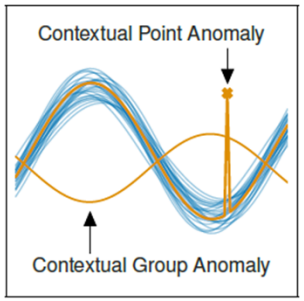
Definition

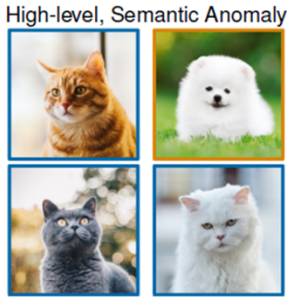
Anomaly(= novelty, outlier)

- An **anomaly** is an observation that **deviates considerably** from some **concept of normality**









Notation

- **P** : dist. of overall data
- **P+** : dist. of normal data
- **P-** : dist. of anomaly
- $\boldsymbol{A}$: set of anomaly

Main concept of AD

- **Main idea** : model learns normal cases and **deviations** from it **as anomaly**
- **Assumption1**) Data is **stationary**(but claim pattern may changes rapidly)
- **Assumption2**) P- is **uniform** dist. (anomaly occurs uniformly in feature space)

Dataset settings

Anomaly based

Unsupervised

- Most **common** way
- **Assumption**

  : $X \sim P$, where $P=P+$

- **Robust assumption** (ex. Robust AE)

  : $X \sim n*P+ + (1-n)*P-$, where $P = [P+, P-]$

Semi-supervised

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X} \quad \text{and} \quad (\tilde{\boldsymbol{x}}_1, \tilde{y}_1), \ldots, (\tilde{\boldsymbol{x}}_m, \tilde{y}_m) \in \mathcal{X} \times \mathcal{Y}$$

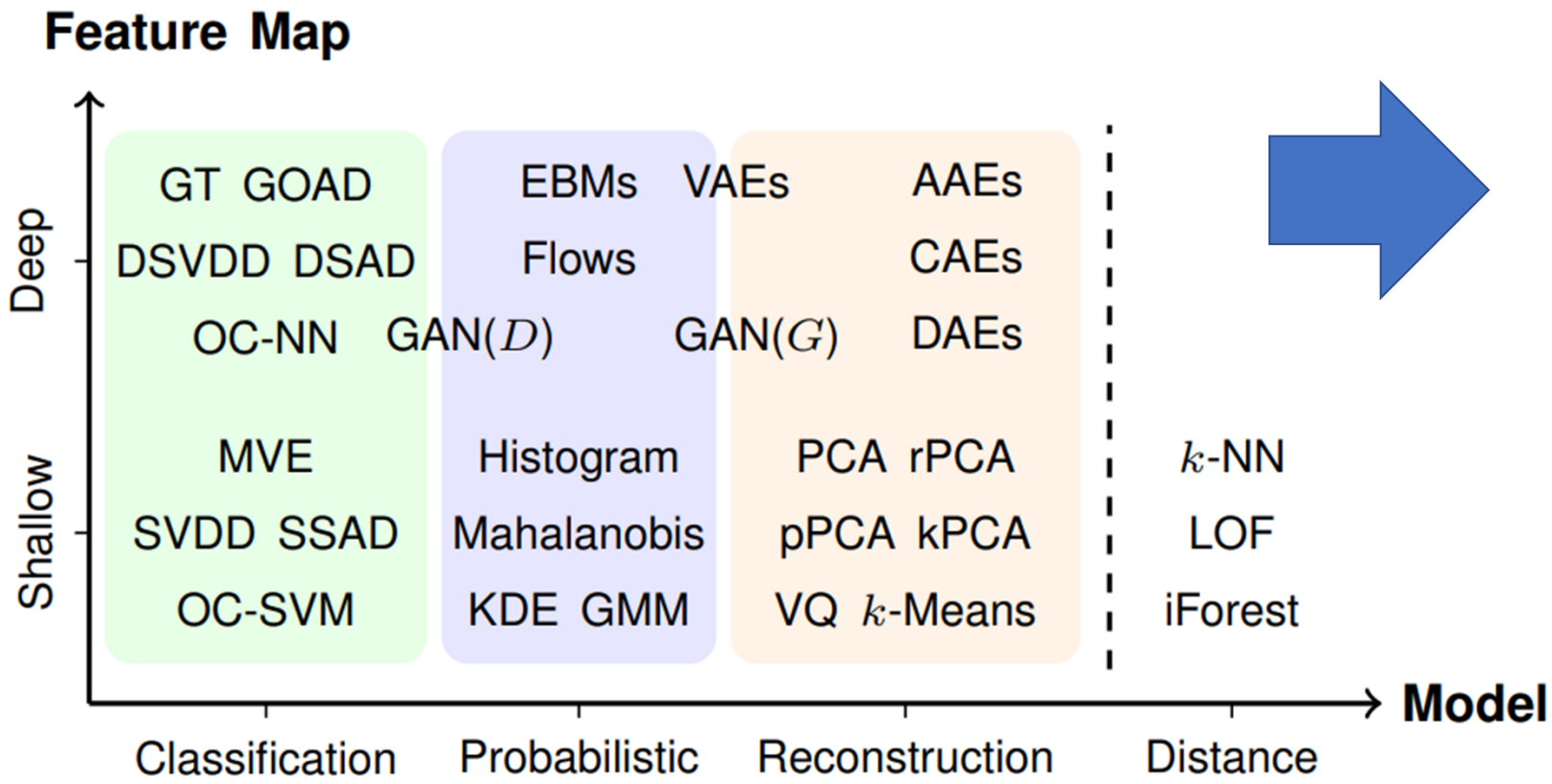
- 데이터 중 **일부**의 **label이 제공**되는 경우
- 명확한 anomaly의 경우에만 라벨을 제공
- Outlier exposure, **Deep SAD**

Supervised

- 자료의 모든 label이 제공되는 경우
- **Binary classification** 문제로 **환원**됨

Types of AD

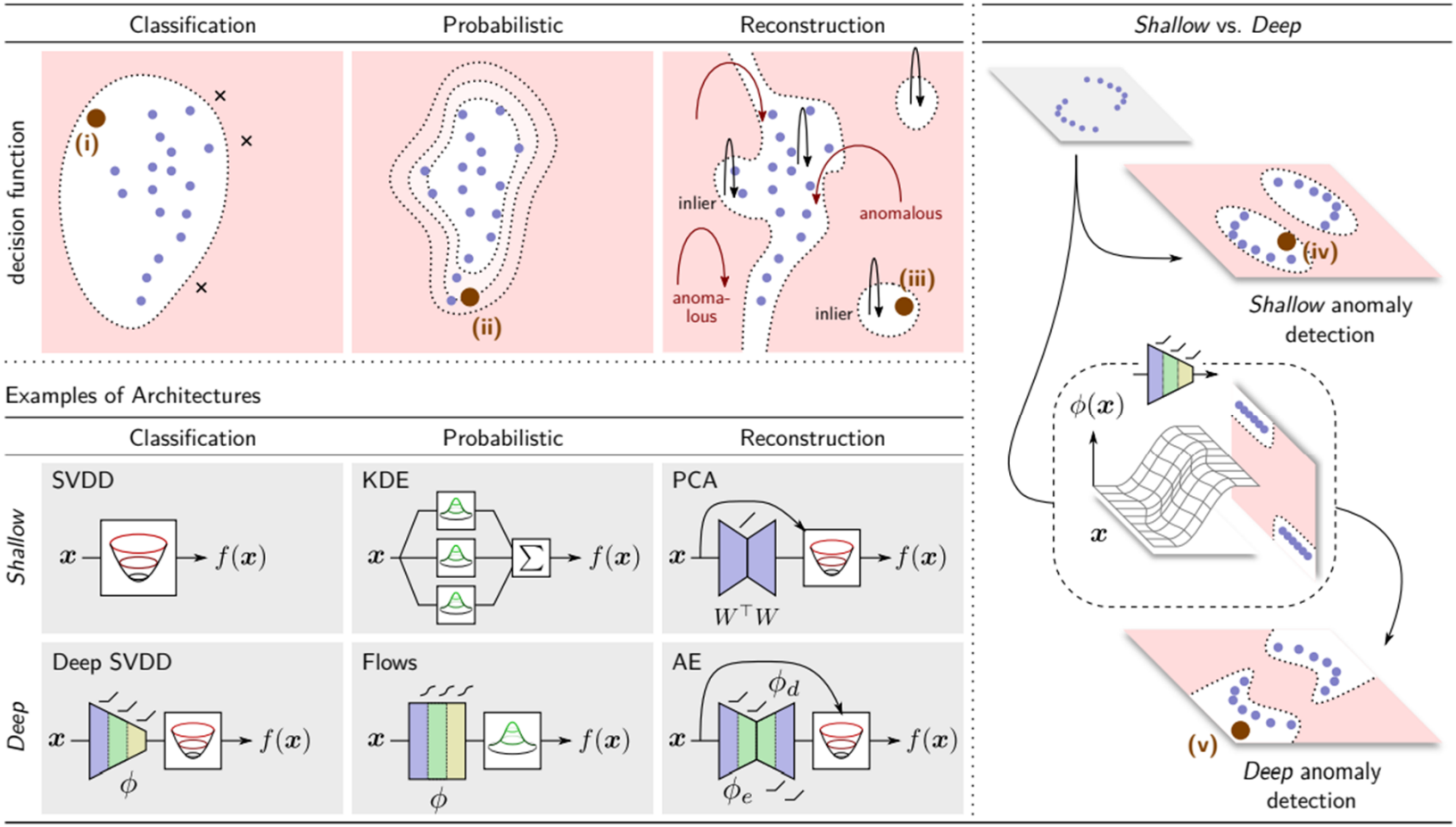
Types of AD



Which one is the best?? (장담x)

Anomaly based

Failure Detection



Full moon, small moon toy data

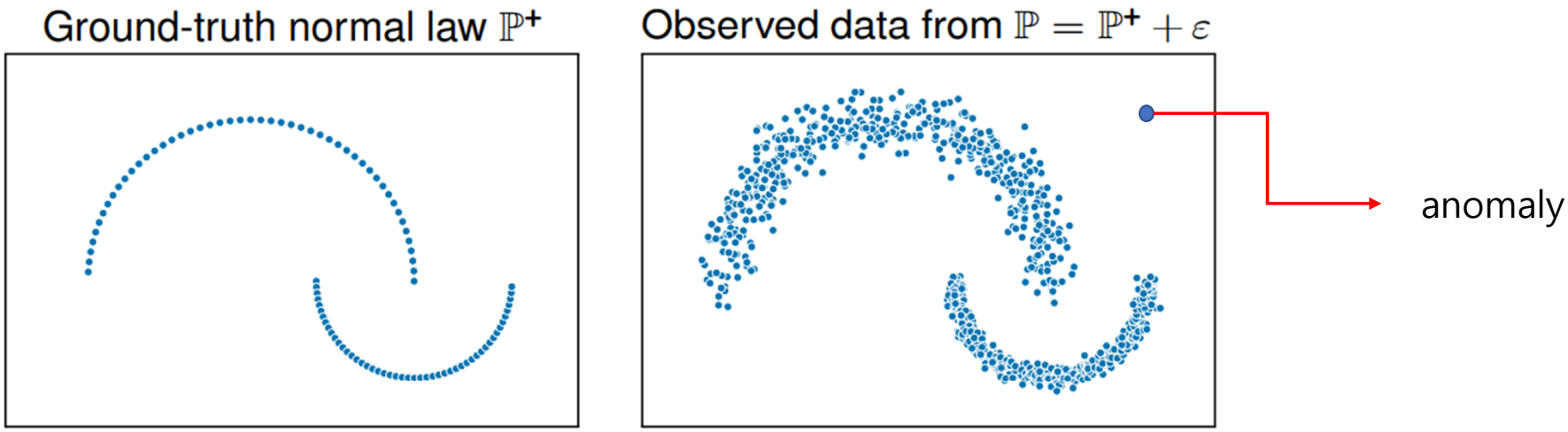
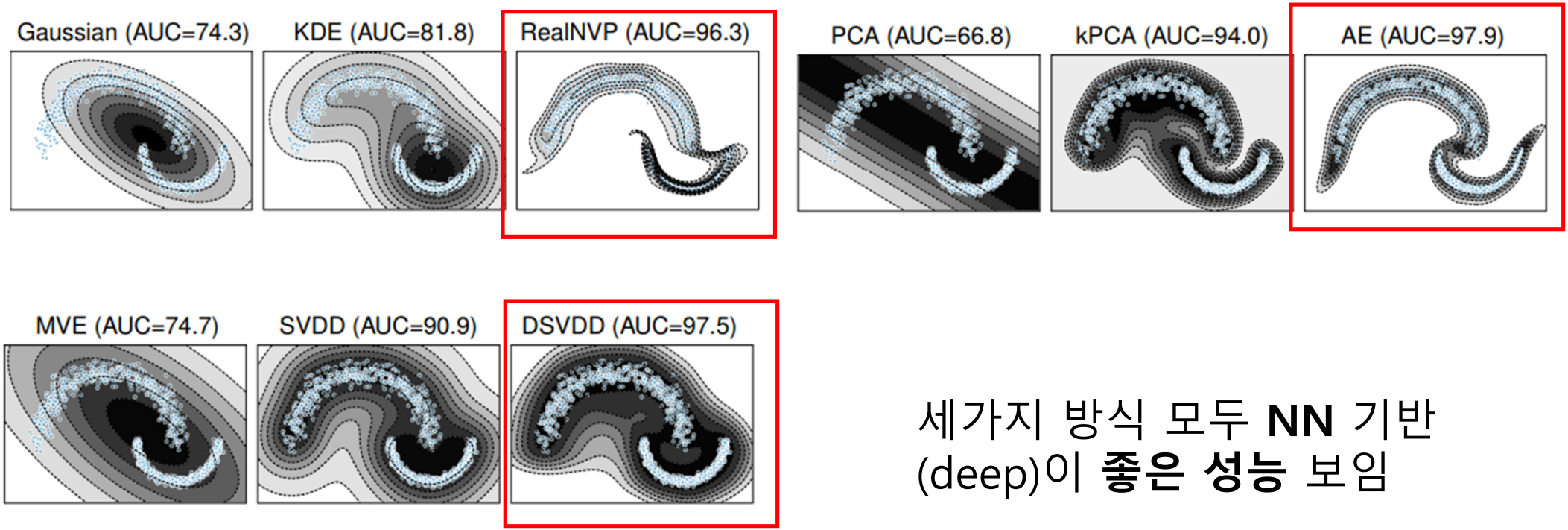


Fig. 4. A two-dimensional *Big Moon, Small Moon* toy example with real-valued ground-truth normal law $\mathbb{P}^+$ that is composed of two one-dimensional manifolds (bimodal, two-scale, non-convex). The unlabeled training data (n = 1,000, $m = 0$) is generated from $\mathbb{P} = \mathbb{P}^+ + \varepsilon$ which is subject to Gaussian noise ε. This toy data is non-hierarchical, context-free, and stationary. Anomalies are off-manifold points that may occur uniformly over the displayed range.

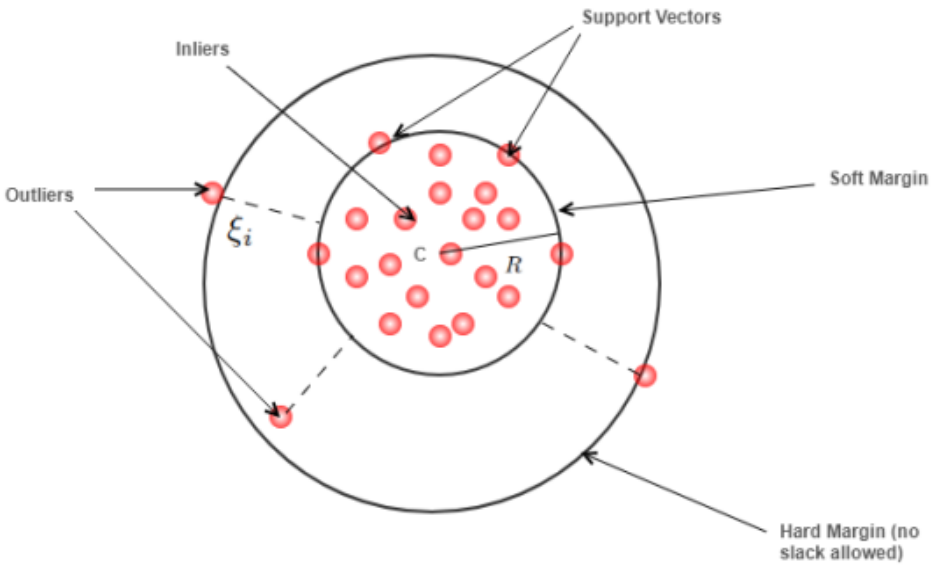
Result (참고용)



세가지 방식 모두 **NN** 기반 (deep)이 **좋은 성능** 보임

Classification model

One class SVM 원리



$$\min_{R,\boldsymbol{c},\boldsymbol{\xi}} \quad R^2 + \frac{1}{\nu n}\sum_i \xi_i$$

$$\text{s.t.} \quad \|\phi_k(\boldsymbol{x}_i) - \boldsymbol{c}\|^2_{\mathcal{F}_k} \leq R^2 + \xi_i,\ \xi_i \geq 0, \quad \forall i.$$

Outlier는 최대 ξ **거리 안에,**
Normal point는 **R 내에** 오게 끔 하는
최대한 **tight**한 **결정경계(원)**를 **학습**

Hypersphere = B(c, R)
υ : 원의 tight 함을 결정하는 hyper-parameter
F_k : kernel Hilbert space
(ϕ_k : X → F_k) : feature map

Anomaly based

Classification models

- One class SVM (or SVDD) with feature selected by human — Shallow
- Deep SVDD — Deep
- **Deep SAD (Deep SVDD with semi-supervised manner)** — Deep
- One class Neural Net — Deep

One class SVM (or SVDD) with feature selected by human

Objective :

$$\min_{R,\boldsymbol{c},\boldsymbol{\xi}} \quad R^2 + \frac{1}{\nu n}\sum_i \xi_i$$

$$\text{s.t.} \quad \|\phi_k(\boldsymbol{x}_i) - \boldsymbol{c}\|^2_{\mathcal{F}_k} \le R^2 + \xi_i,\ \xi_i \ge 0, \quad \forall i.$$

X : 880 features
(only normal cases)

→ Selected 42 features (**by human expert**) → **전문가에 의한 변수 선택!**

↓

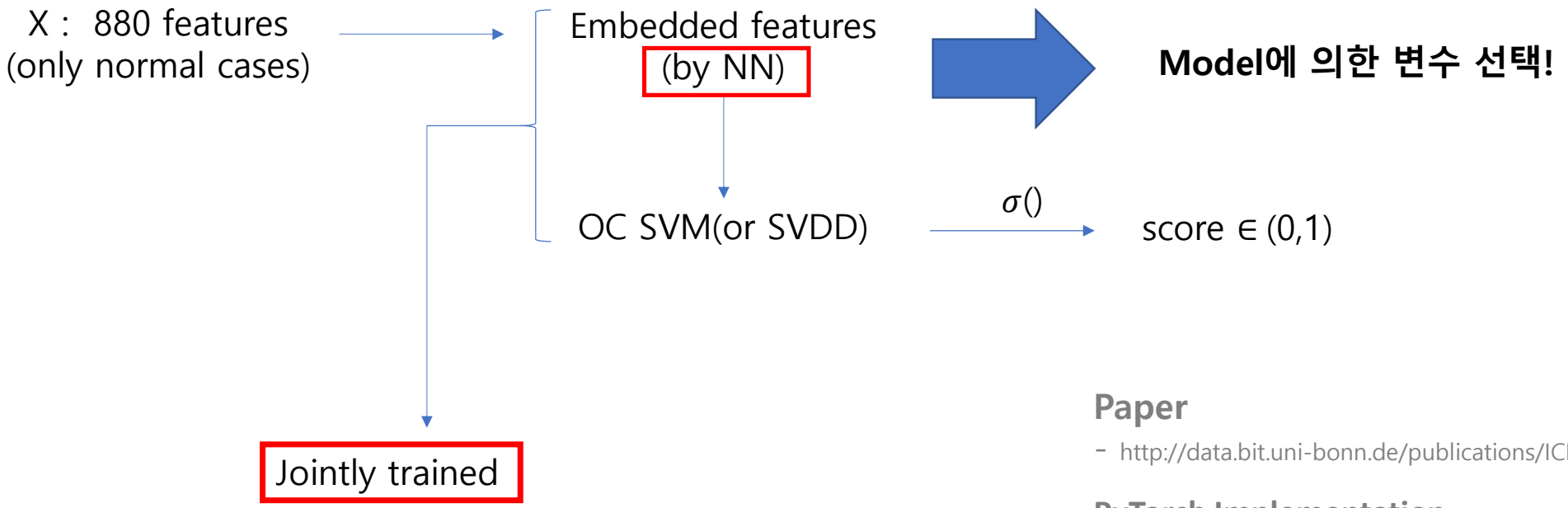
OC SVM(or SVDD) → $\sigma()$ → score ∈ (0,1)

Anomaly based

Deep SVDD

모든 점이 Center에 가까워지도록
단순화 해서 생각
-> 모든 점이 center에 몰리면 안되므로
W에 대한 restriction term을 둔다

Objective : $\min_{\mathcal{W}} \frac{1}{n}\sum_{i=1}^{n}\|\phi(\boldsymbol{x}_i;\mathcal{W})-\boldsymbol{c}\|^2+\frac{\lambda}{2}\sum_{\ell=1}^{L}\|\boldsymbol{W}^\ell\|_F^2.$



Paper

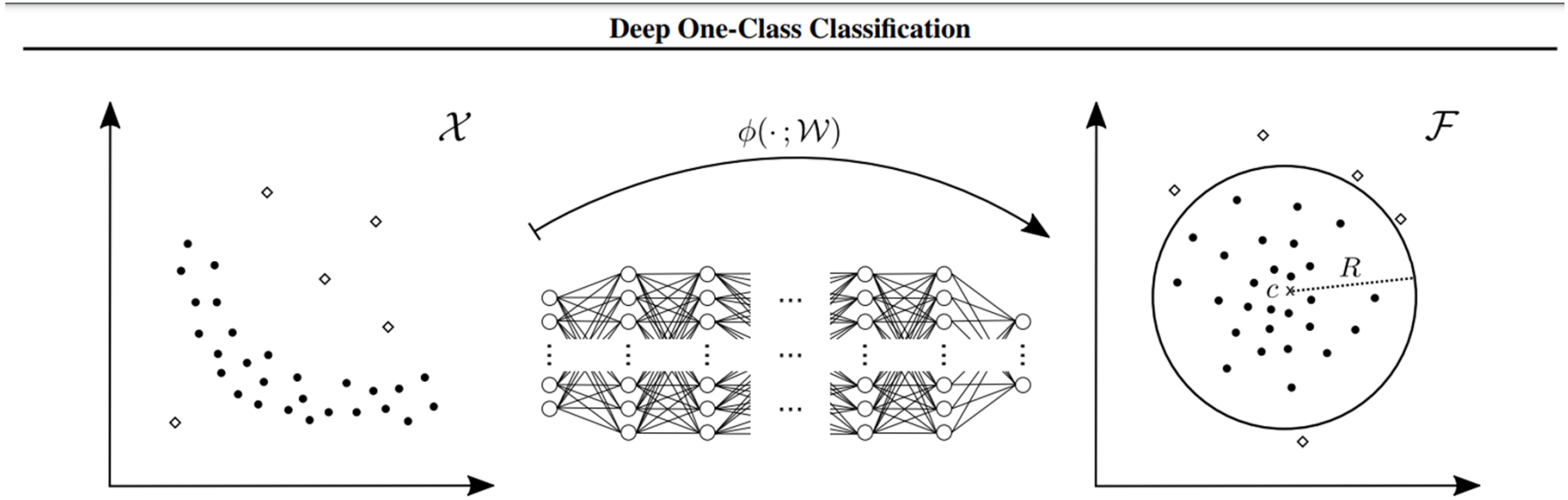
- http://data.bit.uni-bonn.de/publications/ICML2018.pdf

PyTorch Implementation

-https://github.com/lukasruff/Deep-SVDD-PyTorch

Anomaly based

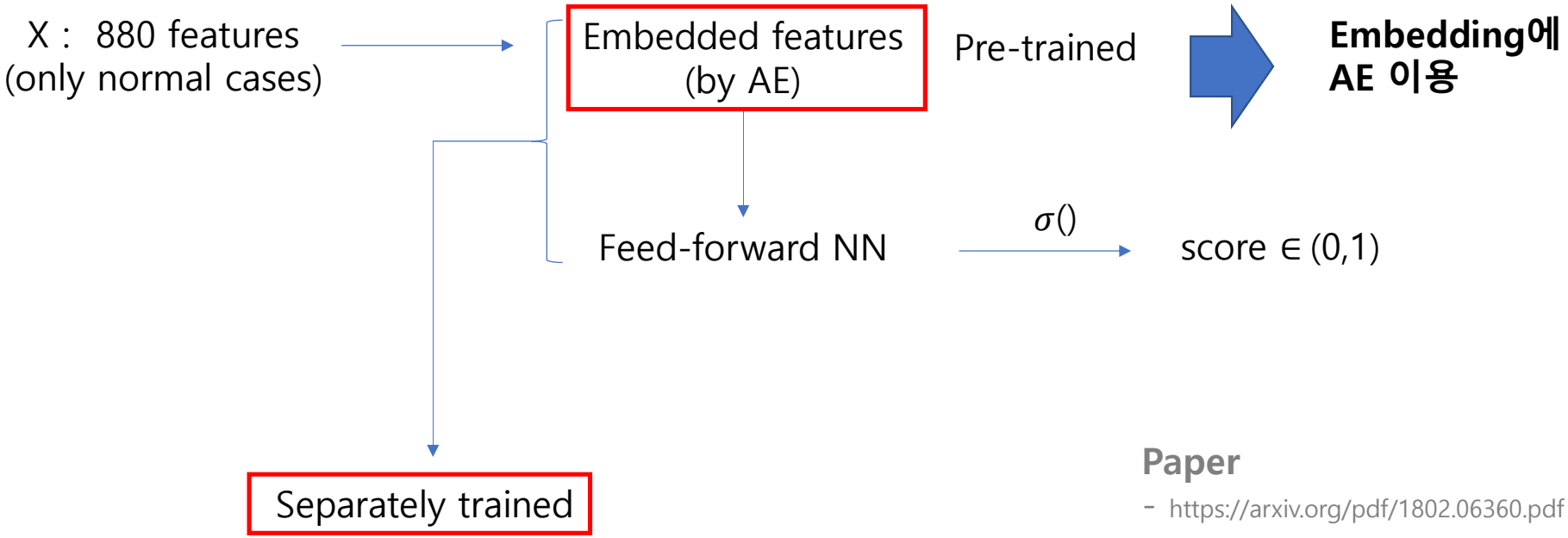
Deep SVDD



뷴류가 수월한 Latent **space**로 mapping 해주는 함수를 **NN 이용하여 찾는다**

One class Neural Net

Objective : $$\min_{w,V,r} \frac{1}{2}\|w\|_2^2 + \frac{1}{2}\|V\|_F^2 + \frac{1}{\nu}\cdot\frac{1}{N}\sum_{n=1}^{N}\max(0, r - \langle w, g(V\mathbf{X}_{n:})\rangle) - r$$



Paper

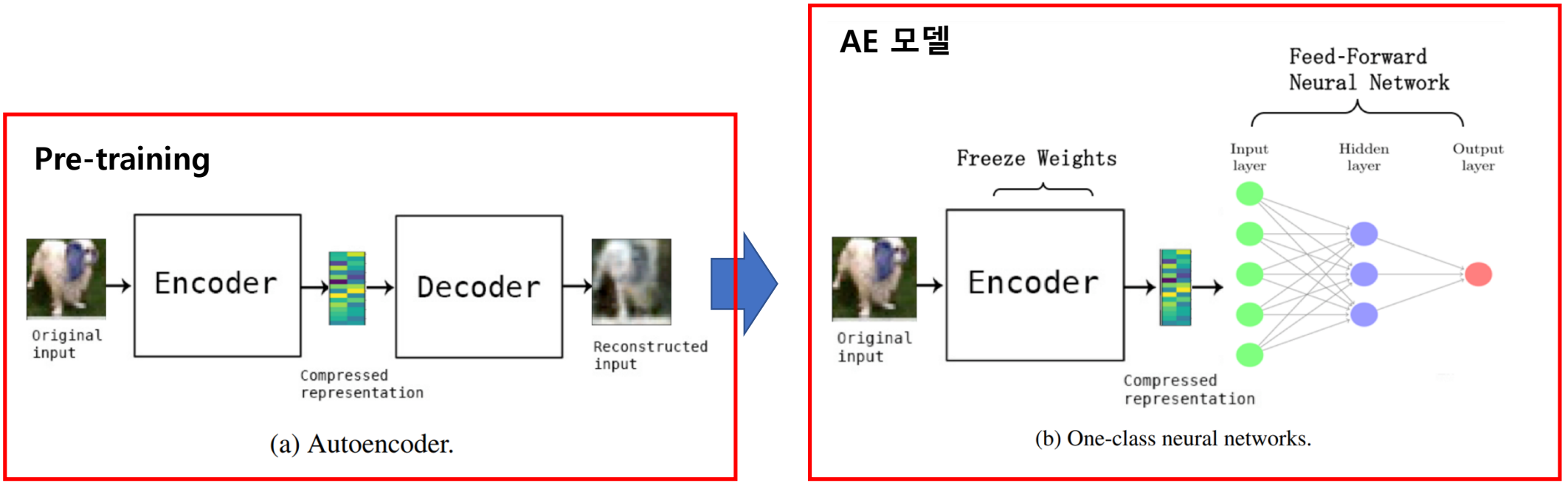
- https://arxiv.org/pdf/1802.06360.pdf

Keras Implementation

- https://github.com/raghavchalapathy/oc-nn

Anomaly based

One class Neural Net



AE 학습 후, encoder 딴을 **전이학습** 하여 AD에 **이용**

Anomaly based

Deep SAD

If outlier is close to C, loss gets large
If normal point is far from C, loss gets also large

Objective : $\min_{\mathcal{W}} \frac{1}{n+m}\sum_{i=1}^{n}\|\phi(\boldsymbol{x}_i;\mathcal{W})-\boldsymbol{c}\|^2 + \frac{\eta}{n+m}\sum_{j=1}^{m}\left(\|\phi(\tilde{\boldsymbol{x}}_j;\mathcal{W})-\boldsymbol{c}\|^2\right)^{\tilde{y}_j} + \frac{\lambda}{2}\sum_{\ell=1}^{L}\|\boldsymbol{W}^\ell\|_F^2.$

η : hyperparameter that controls balance between labeled and unlabeled
$\tilde{y}$: 1 if normal, -1 if anomaly

X : 880 features
Y : available label

라벨 이용하는 것이 특징

Embedded features (by AE)

Embedding에 AE 이용

OC SVM(or SVDD) → $\sigma()$ → score ∈ (0,1)

Separately trained

Paper

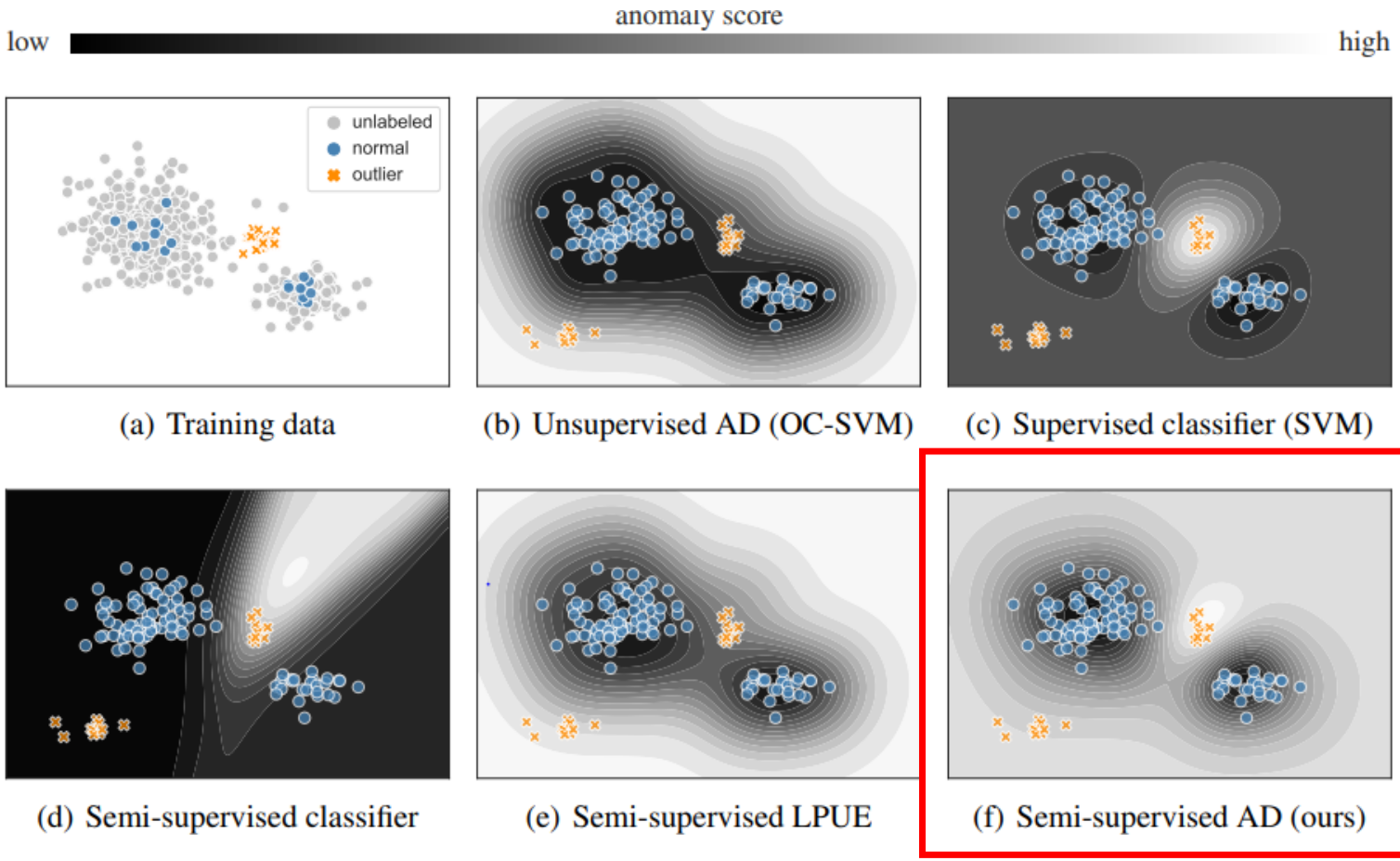
- http://data.bit.uni-bonn.de/publications/ICML2018.pdf

PyTorch Implementation

- https://github.com/lukasruff/Deep-SAD-PyTorch

Anomaly based

Deep SAD



unseen에도 좋은 성능

summary

	Deep/shallow	Use label	Main idea
OC-SVM	**Shallow**	**N**	**-**
Deep SVDD	Deep	N	NN feature extractor + SVDD + jointly learn
OC-neural net	Deep	N	AE + NN classifier
Deep SAD	**Deep**	**Y**	**AE** **+** **SVDD** **+** **Use available label**

Conclusion

- **OC – SVM 기반**의 Deep한 방법, shallow한 방법 **성능 평가** 후 적용 검토

- 후에 **GAN** 이용한 **Over-sampling** 방법도 **검토**

Next step

- 전처리 pipeline 구축
- Scikit learn 이용 baseline 설정
- GPU 서버 확보 후 deep한 방식 test
- 성능비교
- 최종 선정 모델 parameter 최적화