



Anomaly Detection

강필성

고려대학교 산업경영공학부

Bflysoft & WIGO AI LAB

AGENDA

01 이상치 탐지: 개요

02 밀도 기반 이상치 탐지 기법

03 모델 기반 이상치 탐지 기법

이상치 탐지

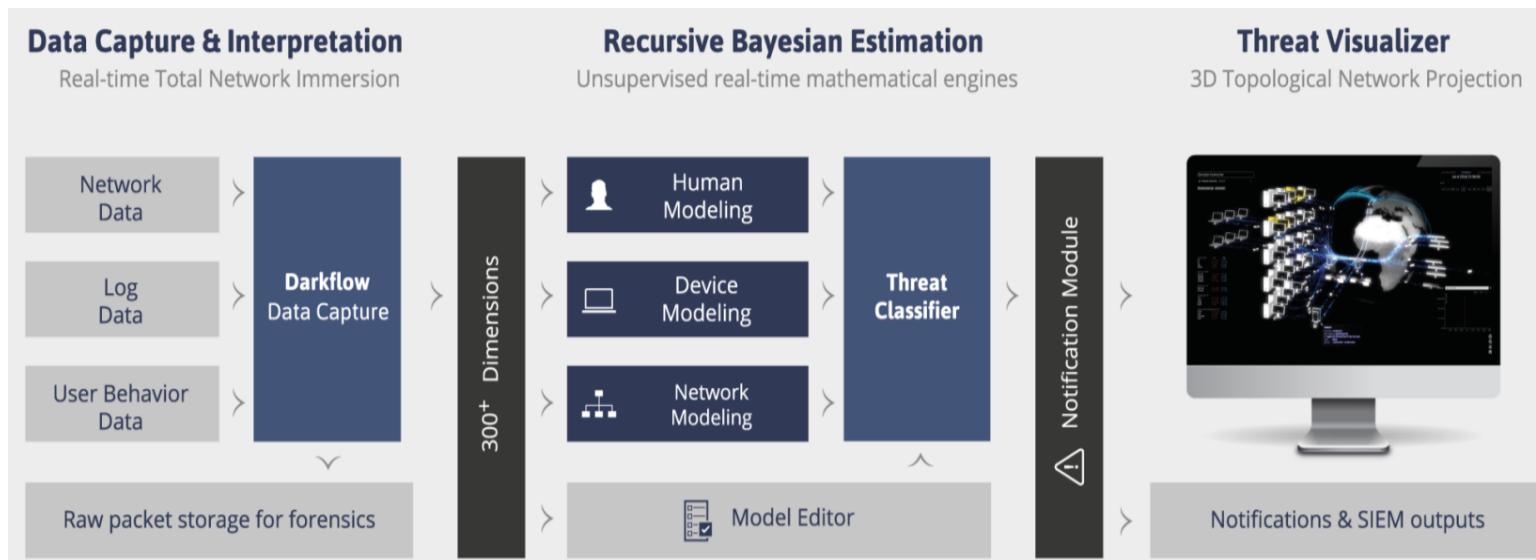
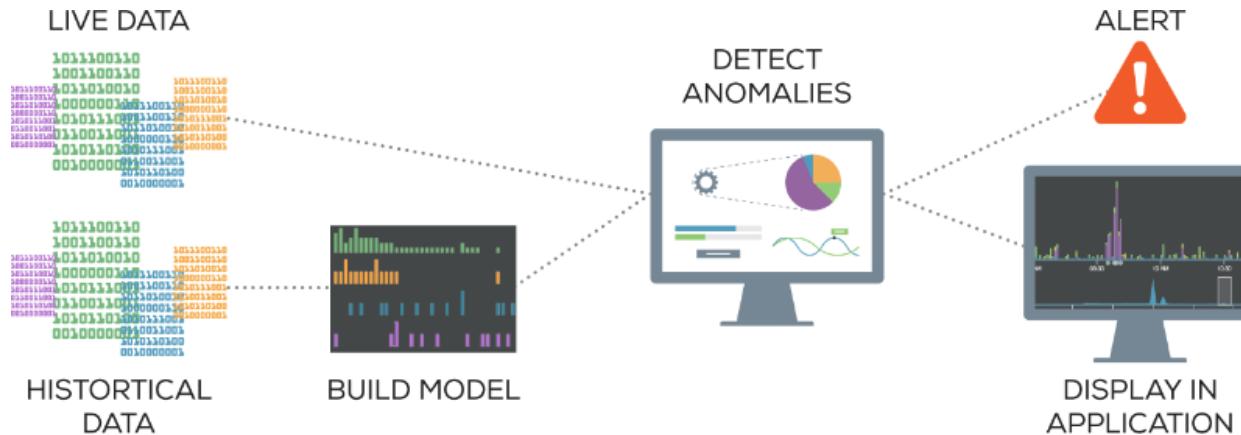
- 이상치 데이터(Novel Data)란?

“Observations that deviate so much from other observations as to arouse suspicions that they were generated by a different mechanism (Hawkins, 1980)”
“Instances that their true probability density is very low (Harmeling et al., 2006)”

- ✓ 이상치 데이터는 노이즈 데이터와는 다름
 - 노이즈는 측정 과정에서의 무작위성(randomness)에 기반함
 - 노이즈는 이상치탐지 전에 제거되는 것이 일반적임
- ✓ 이상치 데이터는 적지만 중요한 데이터임
 - 정상적인 데이터를 생성하는 매커니즘을 위반하여 생성됨

이상치 탐지

- 이상치 탐지 사례: 시스템 로그 기반 이상행위 탐지



이상치 탐지

Data Preparation

ADFA-LD Dataset

SySCALL Trace



265	104	265	104	3	175
104	142	3	3	3	104
265	104	142	142	175	
146	142	146	142	265	3
175	175	142	142	175	
119	265	142	146	265	
146	119	142	146	142	
142	142	142	146	104	
265	3	119	3	265	119
146	146	146	265	146	
142	142	146	142	119	

Vectorization

Doc2vec

Classifier

Average/Concatenate

Paragraph Matrix

Paragraph id

RNN-AutoEncoder

Encoder Bi-RNN

Forward Encoder

Backward Encoder

x_1

x_{t-1}

x_t

b_1

b_2

b_t

b_{t+1}

b_{t+2}

b_{t+3}

b_{t+4}

b_{t+5}

b_{t+6}

b_{t+7}

b_{t+8}

b_{t+9}

b_{t+10}

b_{t+11}

b_{t+12}

b_{t+13}

b_{t+14}

b_{t+15}

b_{t+16}

b_{t+17}

b_{t+18}

b_{t+19}

b_{t+20}

b_{t+21}

b_{t+22}

b_{t+23}

b_{t+24}

b_{t+25}

b_{t+26}

b_{t+27}

b_{t+28}

b_{t+29}

b_{t+30}

b_{t+31}

b_{t+32}

b_{t+33}

b_{t+34}

b_{t+35}

b_{t+36}

b_{t+37}

b_{t+38}

b_{t+39}

b_{t+40}

b_{t+41}

b_{t+42}

b_{t+43}

b_{t+44}

b_{t+45}

b_{t+46}

b_{t+47}

b_{t+48}

b_{t+49}

b_{t+50}

b_{t+51}

b_{t+52}

b_{t+53}

b_{t+54}

b_{t+55}

b_{t+56}

b_{t+57}

b_{t+58}

b_{t+59}

b_{t+60}

b_{t+61}

b_{t+62}

b_{t+63}

b_{t+64}

b_{t+65}

b_{t+66}

b_{t+67}

b_{t+68}

b_{t+69}

b_{t+70}

b_{t+71}

b_{t+72}

b_{t+73}

b_{t+74}

b_{t+75}

b_{t+76}

b_{t+77}

b_{t+78}

b_{t+79}

b_{t+80}

b_{t+81}

b_{t+82}

b_{t+83}

b_{t+84}

b_{t+85}

b_{t+86}

b_{t+87}

b_{t+88}

b_{t+89}

b_{t+90}

b_{t+91}

b_{t+92}

b_{t+93}

b_{t+94}

b_{t+95}

b_{t+96}

b_{t+97}

b_{t+98}

b_{t+99}

b_{t+100}

b_{t+101}

b_{t+102}

b_{t+103}

b_{t+104}

b_{t+105}

b_{t+106}

b_{t+107}

b_{t+108}

b_{t+109}

b_{t+110}

b_{t+111}

b_{t+112}

b_{t+113}

b_{t+114}

b_{t+115}

b_{t+116}

b_{t+117}

b_{t+118}

b_{t+119}

b_{t+120}

b_{t+121}

b_{t+122}

b_{t+123}

b_{t+124}

b_{t+125}

b_{t+126}

b_{t+127}

b_{t+128}

b_{t+129}

b_{t+130}

b_{t+131}

b_{t+132}

b_{t+133}

b_{t+134}

b_{t+135}

b_{t+136}

b_{t+137}

b_{t+138}

b_{t+139}

b_{t+140}

b_{t+141}

b_{t+142}

b_{t+143}

b_{t+144}

b_{t+145}

b_{t+146}

b_{t+147}

b_{t+148}

b_{t+149}

b_{t+150}

b_{t+151}

b_{t+152}

b_{t+153}

b_{t+154}

b_{t+155}

b_{t+156}

b_{t+157}

b_{t+158}

b_{t+159}

b_{t+160}

b_{t+161}

b_{t+162}

b_{t+163}

b_{t+164}

b_{t+165}

b_{t+166}

b_{t+167}

b_{t+168}

b_{t+169}

b_{t+170}

b_{t+171}

b_{t+172}

b_{t+173}

b_{t+174}

b_{t+175}

b_{t+176}

b_{t+177}

b_{t+178}

b_{t+179}

b_{t+180}

b_{t+181}

b_{t+182}

b_{t+183}

b_{t+184}

b_{t+185}

b_{t+186}

b_{t+187}

b_{t+188}

b_{t+189}

b_{t+190}

b_{t+191}

b_{t+192}

b_{t+193}

b_{t+194}

b_{t+195}

b_{t+196}

b_{t+197}

b_{t+198}

b_{t+199}

b_{t+200}

b_{t+201}

b_{t+202}

b_{t+203}

b_{t+204}

b_{t+205}

b_{t+206}

b_{t+207}

b_{t+208}

b_{t+209}

b_{t+210}

b_{t+211}

b_{t+212}

b_{t+213}

b_{t+214}

b_{t+215}

b_{t+216}

b_{t+217}

b_{t+218}

b_{t+219}

b_{t+220}

b_{t+221}

b_{t+222}

b_{t+223}

b_{t+224}

b_{t+225}

b_{t+226}

b_{t+227}

b_{t+228}

b_{t+229}

b_{t+230}

b_{t+231}

b_{t+232}

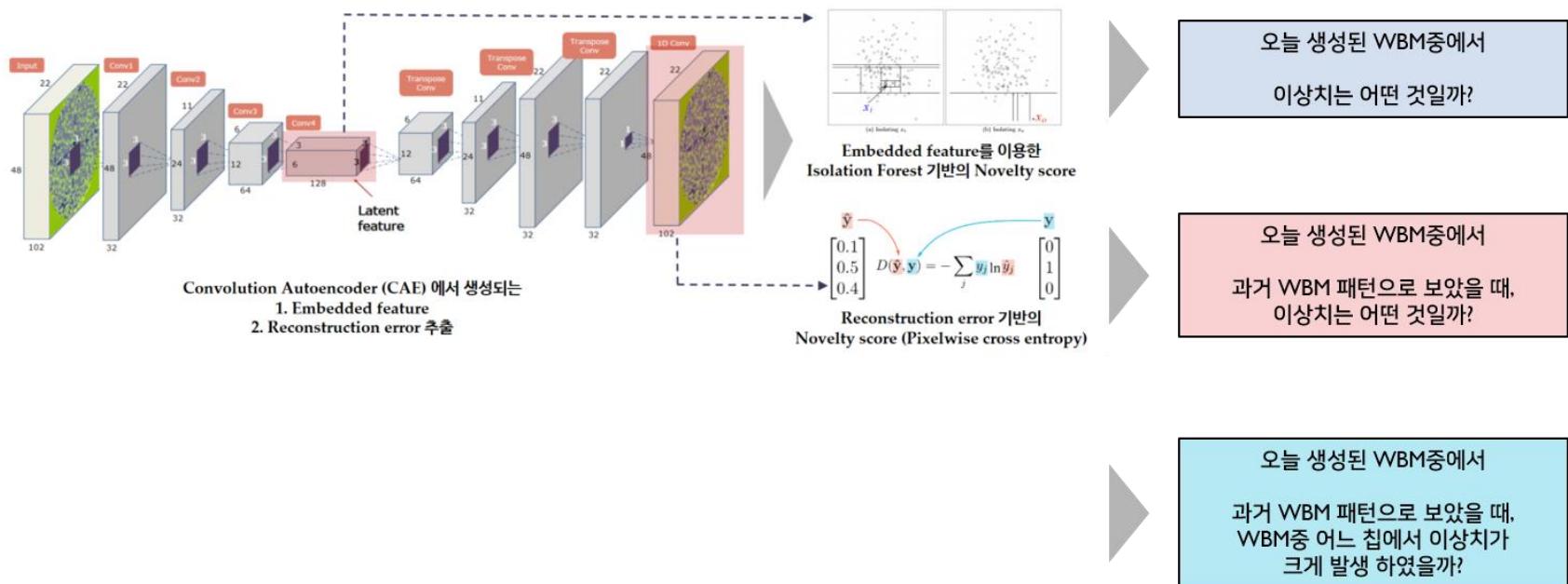
b_{t+233}

b_{t+234}

이상치 탐지

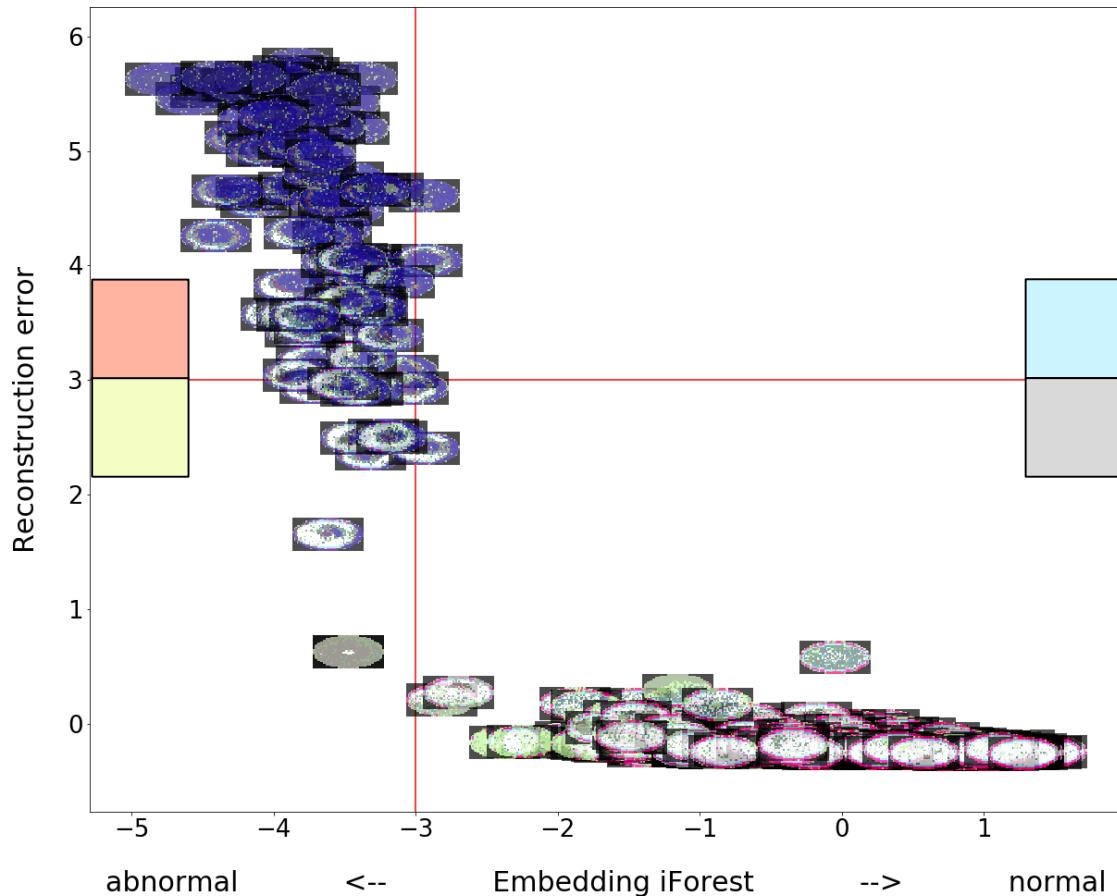
• 이상치 탐지 사례

- 오늘 생산된 웨이퍼들 중에서 WBM 관점에서 특이한 웨이퍼를 판별할 수 있는가?
- 오늘 생산된 웨이퍼들 중에서 과거 생산된 웨이퍼들과는 다른 WBM을 가지는 웨이퍼들을 파악할 수 있는가?
- 과거와 다른 WBM 패턴으로 판별될 경우, 어떤 칩/다이가 판별에 큰 영향을 미쳤는지 알 수 있는가?



이상치 탐지

- 이상치 탐지 사례



Quadrant 1

1. 의미 : 과거 패턴 기준 이상치, 오늘 생성 기준 정상군
2. 규칙 : IF > -3 & Recon ≥ 3

Quadrant 2

1. 의미 : 과거 패턴 기준 이상치, 오늘 생성 기준 이상치
2. 규칙 : IF ≤ -3 & Recon ≥ 3

Quadrant 3

1. 의미 : 과거 패턴 기준 정상군, 오늘 생성 기준 이상치
2. 규칙 : IF > -3 & Recon < 3

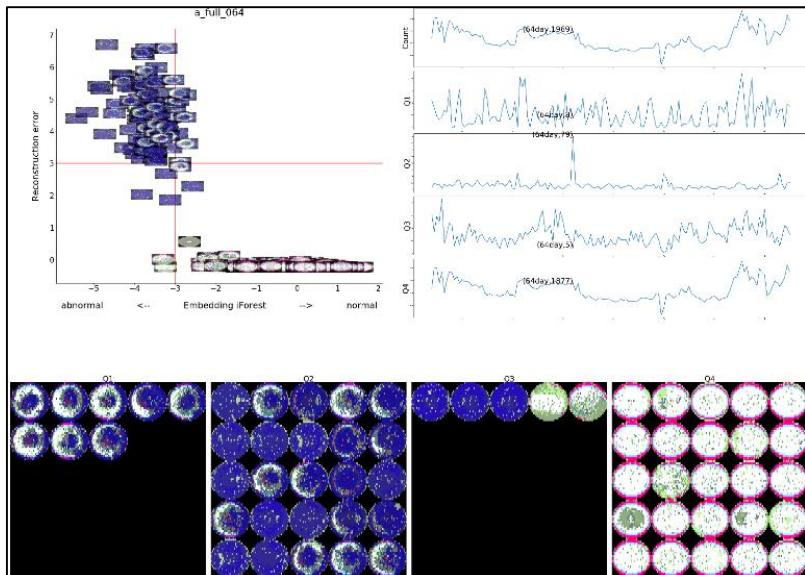
Quadrant 4

1. 의미 : 과거 패턴 기준 정상군, 오늘 생성 기준 정상군
2. 규칙 : IF ≤ -3 & Recon < 3

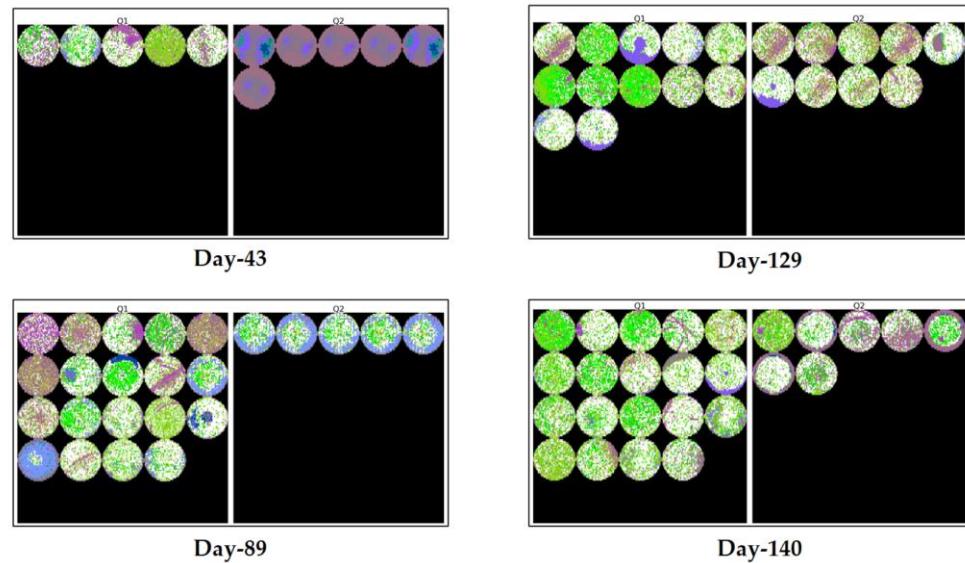
이상치 탐지

- 이상치 탐지 사례

- ✓ 이상치는 상대적인 개념이므로 날마다 이상치로 판별되는 WBM이 달라질 수 있음



< 연구에서 생성한 시각화 도구 >



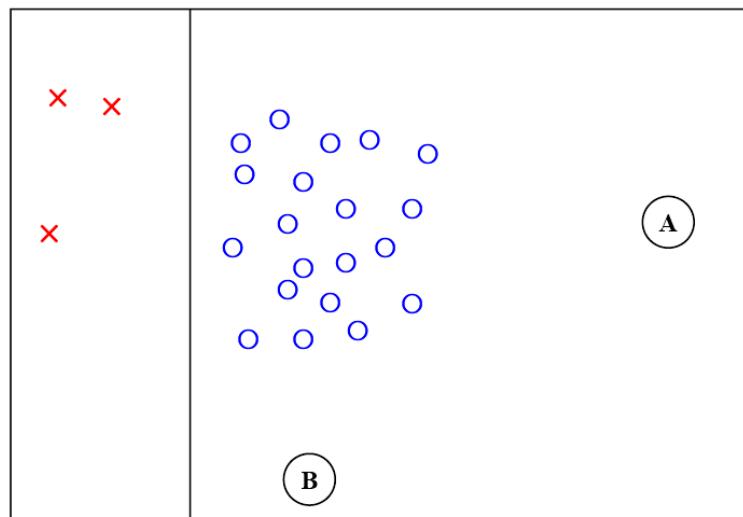
< 날마다 다른 패턴으로 생성되는 이상치 WBM >

이상치 탐지

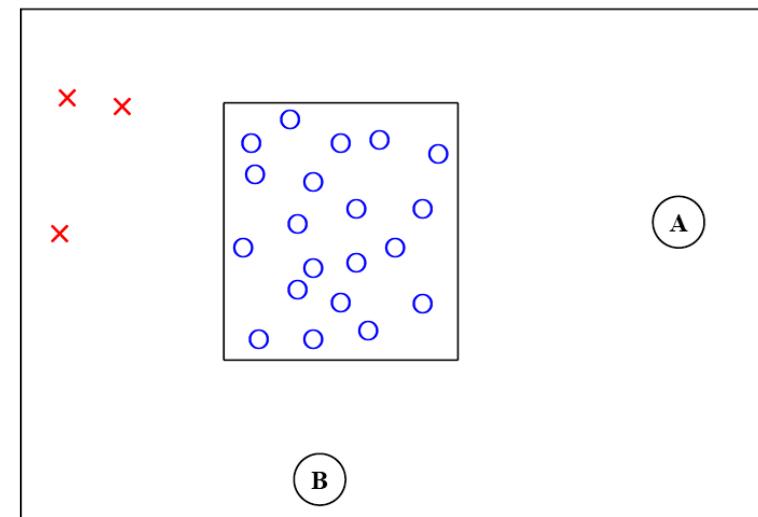
- 이상치 탐지 기법의 적용 분야

- ✓ 대부분의 데이터가 한 범주에 속하며 극소수의 데이터만 다른 범주에 속하는 문제
 - 제조업 공정에서의 불량 탐지
 - 신용카드 사기 거래 탐지
 - 통신망의 불법적인 사용 등

- 분류(Classification) 문제와 이상치 탐지 문제의 차이



Binary classification

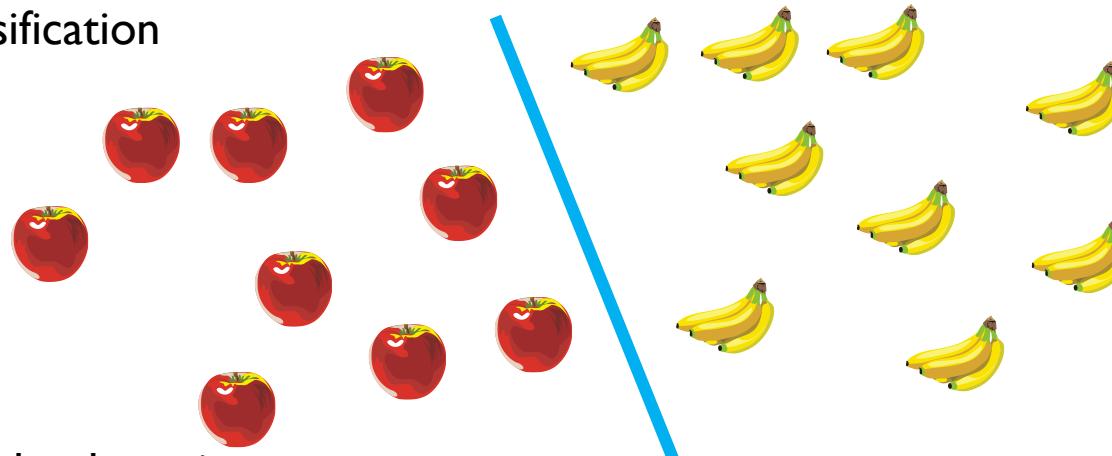


Novelty detection

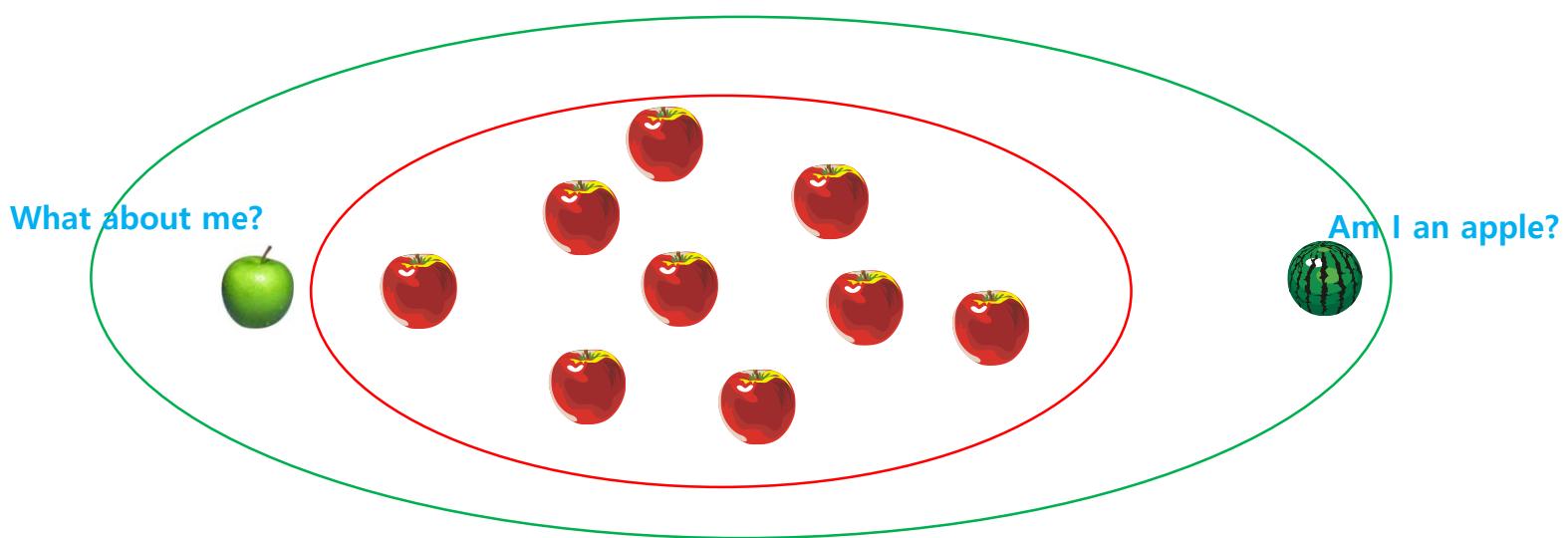
이상치 탐지

- 분류와 이상치 탐지가 데이터로부터 학습하는 방식의 차이

✓ Classification



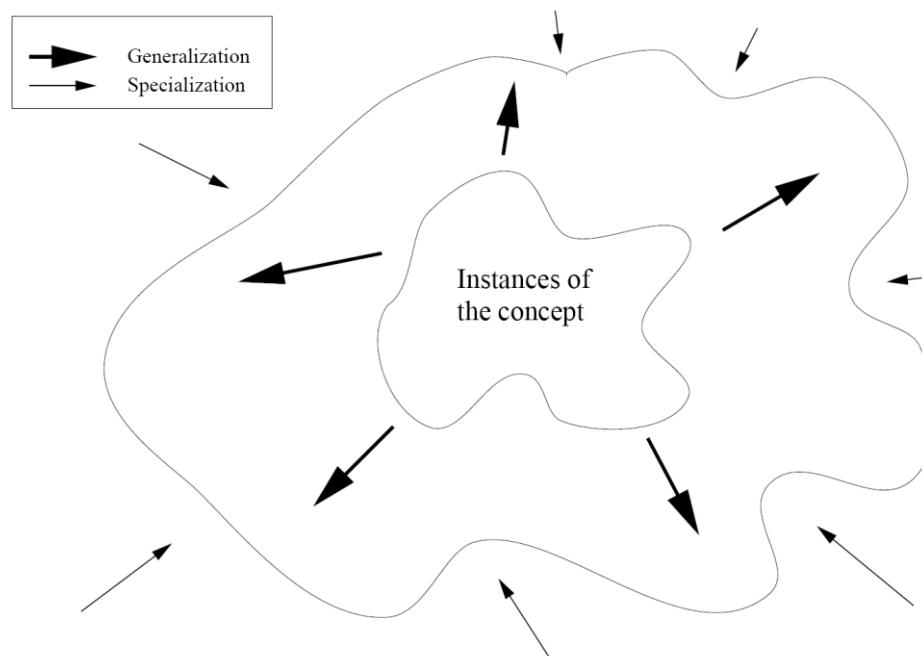
✓ Novelty detection



이상치 탐지

- 일반화(Generalization)와 특수화(Specialization)

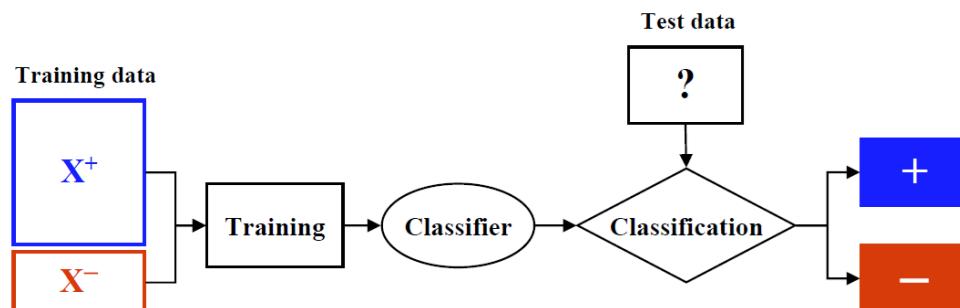
- ✓ 일반화: 주어진 데이터로부터 정상 범주의 개념을 확장해 가는 것
- ✓ 특수화: 주어진 데이터로부터 정상 범주의 개념을 좁혀 가는 것
- ✓ 일반화에 치중할 경우 이상치 데이터 판별이 어렵게 되며, 특수화에 치중할 경우 과적 합의 위험(빈번한 false alarm)에 빠질 수 있음



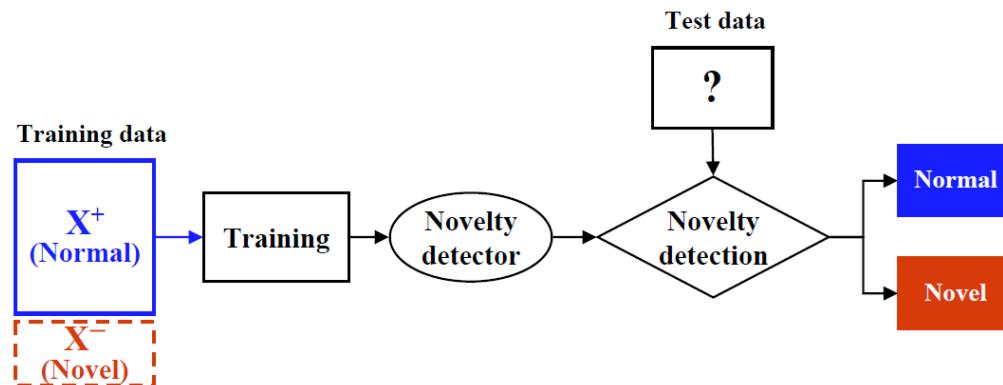
이상치 탐지: 접근 방법

- 가정

- ✓ 주어진 데이터에는 다수의 정상(normal) 데이터와 매우 적은 수의 비정상(novel, abnormal) 데이터가 혼재되어 있음
- ✓ 분류 문제



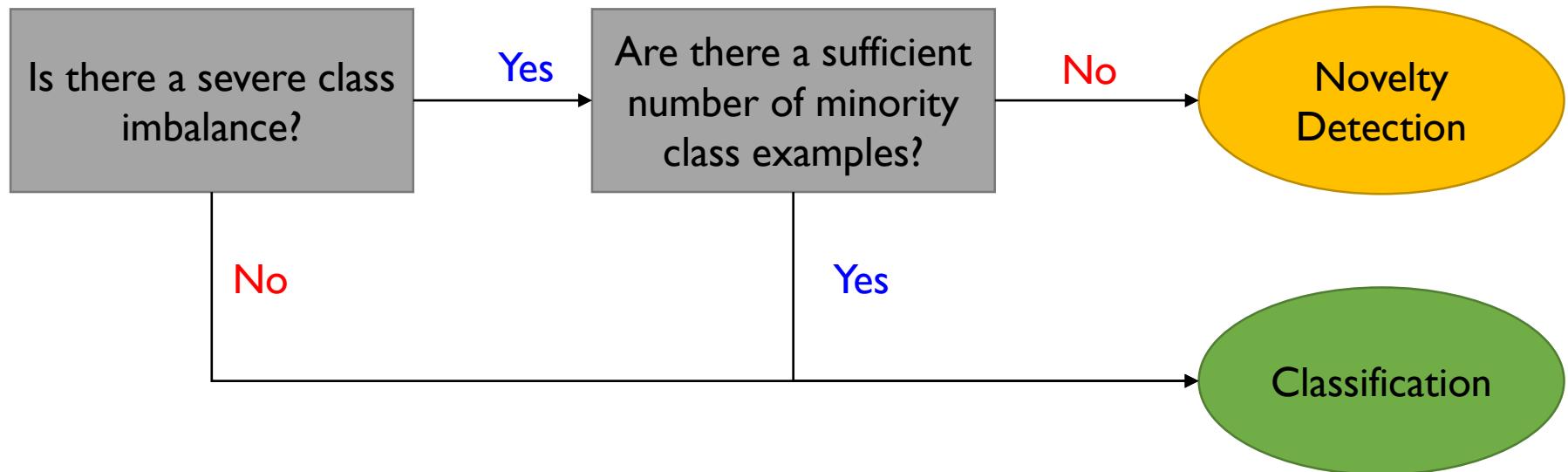
- ✓ 이상치 탐지 문제



이상치 탐지 vs. 분류

- Classification vs. Novelty Detection

✓ Which one to use?



이상치 탐지: 평가 지표

- 이상치 탐지 방법론 평가

- ✓ 이상치 알고리즘에 대한 결과물이 산출되면 이를 바탕으로 최종적으로는 이상치인지 아닌지에 대한 판별을 해야 함
- ✓ 실제 데이터에서는 이러한 cut-off에 따라 혼동 행렬이 달라지기 때문에 이상치 탐지 방법론의 성능 역시 달라짐

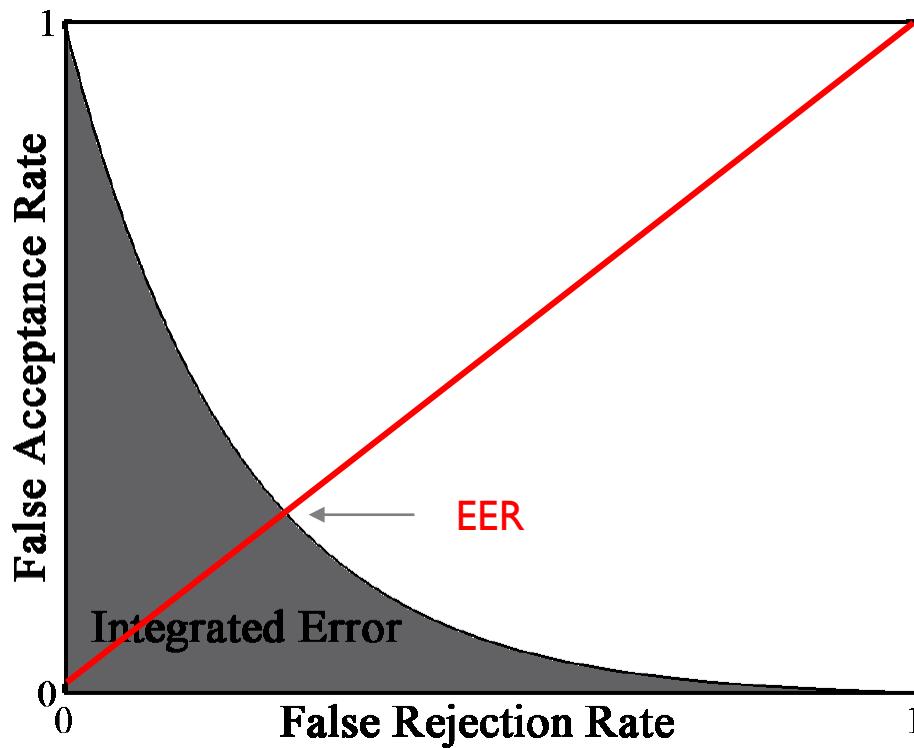
혼동 행렬		모델에 의한 예측	
		불량	정상
실제값	불량	True Positive	False Negative
	정상	False Positive	True Negative

- ✓ 이상치 탐지 방법론 자체의 정합성을 평가할 때는 Cut-off에 영향을 받지 않는 성능 평가 지표가 필요함

이상치 탐지: 평가 지표

- 이상치 탐지 방법론 평가 지표

Metric	Description
False Rejection Rate (FRR, 오탐)	원래 정상인데 이상치로 잘못 판별한 비율
False Acceptance Rate (FAR, 누락)	원래 이상치인데 정상으로 잘못 판별한 비율
Equal Error Rate (EER)	FRR과 FAR이 같아지는 지점
Integrated Error (IE)	The area under the FRR-FAR curve (= 1-AUROC)



AGENDA

01 이상치 탐지: 개요

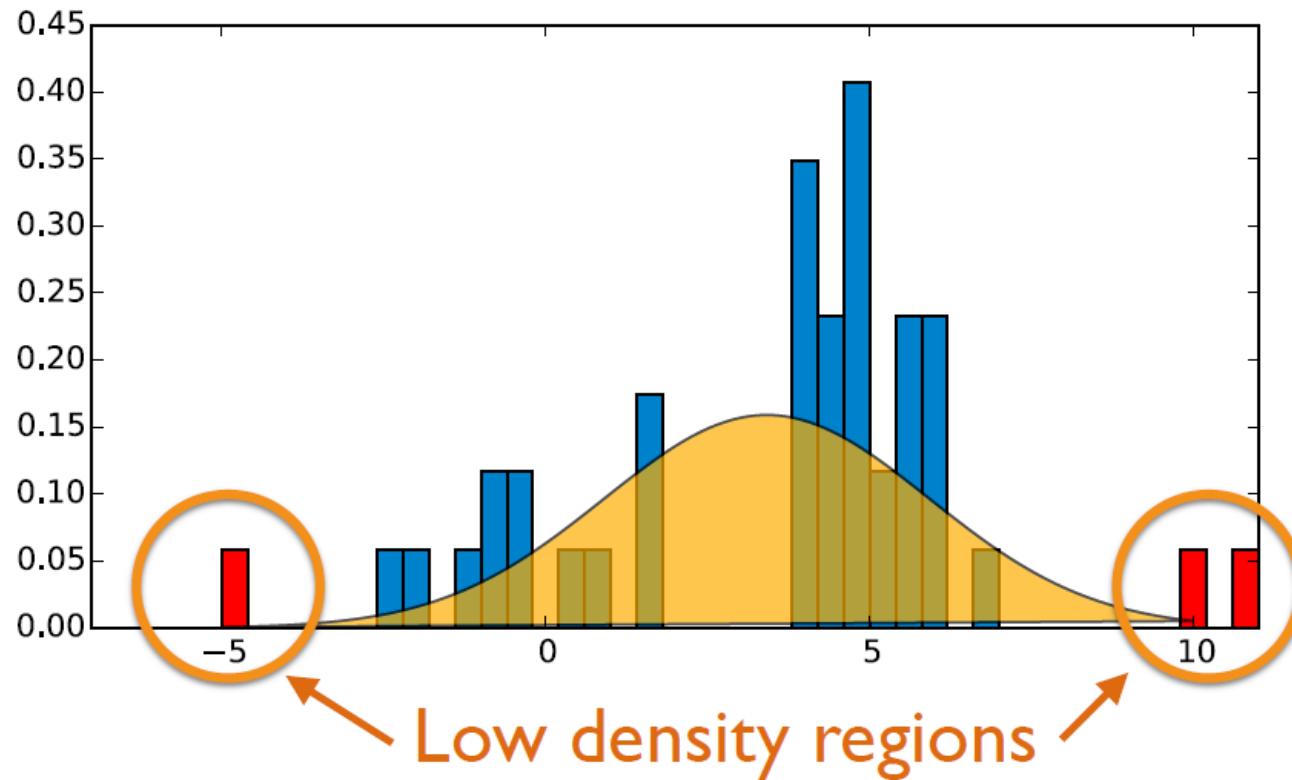
02 밀도 기반 이상치 탐지 기법

03 모델 기반 이상치 탐지 기법

밀도 기반 이상치 탐지 기법

- 목적

- ✓ 주어진 데이터를 바탕으로 각 객체들이 생성될 확률을 추정
- ✓ 새로운 데이터가 생성될 확률이 낮을 경우 이상치로 판단

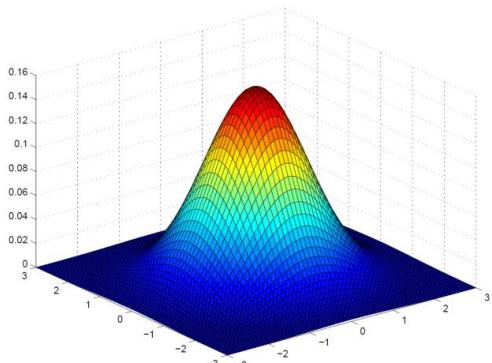


밀도 기반 이상치 탐지 기법

- 목적

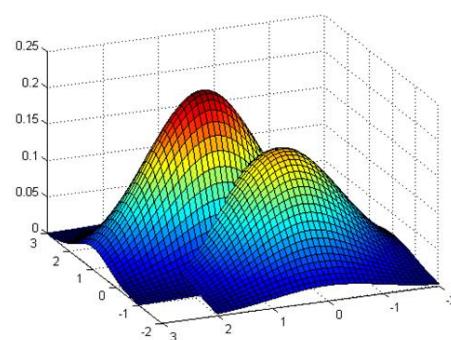
- ✓ 주어진 데이터를 바탕으로 각 객체들이 생성될 확률을 추정
- ✓ 새로운 데이터가 생성될 확률이 낮을 경우 이상치로 판단

Gaussian Density Estimation



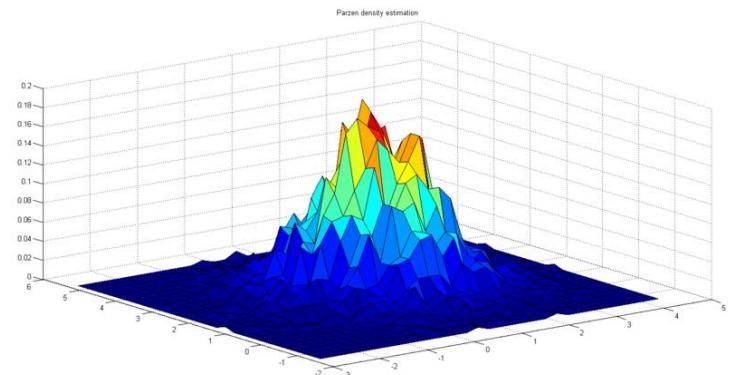
Number of modals
 $= 1$

Mixture of Gaussian Density Estimation



$| <$
Number of modals
 $< \text{Number of instances}$

Kernel Density Estimation

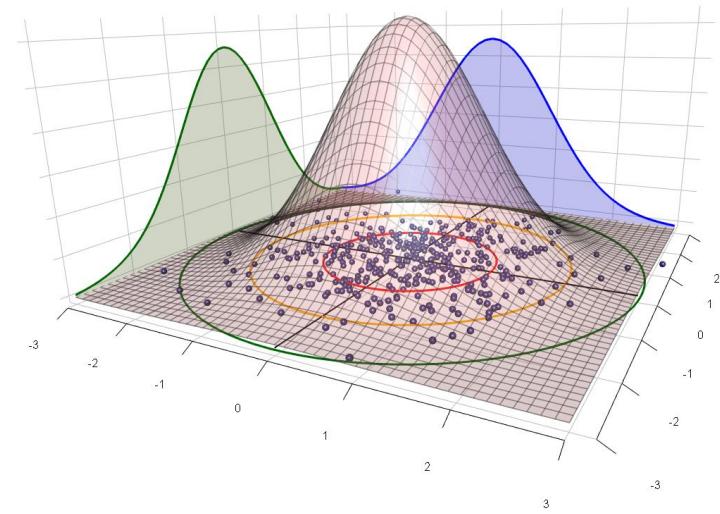
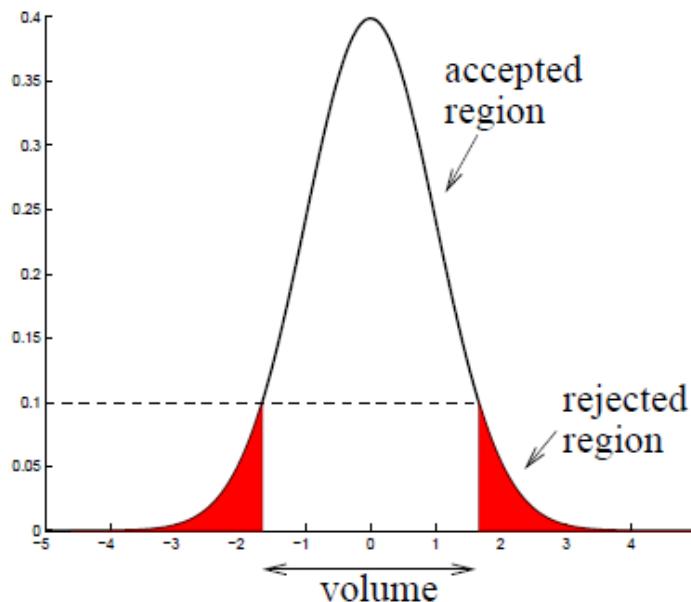


Number of modals
 $= \text{Number of instances}$

밀도 기반 이상치 탐지 기법: Gauss

- Gaussian Density Estimation

- ✓ 모든 데이터가 하나의 가우시안(정규) 분포로부터 생성됨을 가정



$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left[\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right]$$

$$\boldsymbol{\mu} = \frac{1}{n} \sum_{\mathbf{x}_i \in \mathbf{X}^+} \mathbf{x}_i \quad (\text{mean vector}), \quad \Sigma = \frac{1}{n} \sum_{\mathbf{x}_i \in \mathbf{X}^+} (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T \quad (\text{covariance matrix})$$

밀도 기반 이상치 탐지 기법: Gauss

- Gaussian Density Estimation

- ✓ 모든 데이터가 하나의 가우시안(정규) 분포로부터 생성됨을 가정
- ✓ 학습: 주어진 정상 데이터들을 통해 가우시안 분포의 평균 벡터와 공분산 행렬을 추정
- ✓ 테스트: 새로운 데이터에 대하여 생성 확률을 구하고 이 확률이 낮을수록 이상치에 가까운 것으로 판정함
- ✓ 장점
 - 추정이 간단하며 학습 시간이 짧음
 - 적절한 기준치(cut-off)를 분포로부터 정할 수 있음
 - 각 변수의 측정 단위에 영향을 받지 않음

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}|^{1/2}} \exp \left[\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right]$$

밀도 기반 이상치 탐지 기법: Gauss

- Gaussian Density Estimation

- ✓ 1차원 데이터에 대한 학습 예시
- ✓ 우리가 추정해야 하는 파라미터: 평균 μ and 분산 σ^2
- ✓ 정상 데이터 생성 확률을 최대화 하도록 평균과 분산을 계산하는 것이 가능
(Analytically tractable)

$$L = \prod_{i=1}^N P(x_i | \mu, \sigma^2) = \prod_{i=1}^N \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2}\right)$$

$$\log L = -\frac{1}{2} \sum_{i=1}^N \frac{(x_i - \mu)^2}{\sigma^2} - \frac{N}{2} \log(2\pi\sigma^2)$$

밀도 기반 이상치 탐지 기법: Gauss

- Gaussian Density Estimation

- ✓ 최대 우도 추정법을 이용하여 해를 찾을 수 있음 ($\gamma = 1/\sigma^2$)

$$\log L = -\frac{1}{2} \sum_{i=1}^N \gamma(x_i - \mu)^2 - \frac{N}{2} \log(2\pi) + \frac{N}{2} \log(\gamma)$$

$$\frac{\partial \log L}{\partial \mu} = \gamma \sum_{i=1}^N (x_i - \mu) = 0 \rightarrow \mu = \frac{1}{N} \sum_{i=1}^N x_i$$

$$\frac{\partial \log L}{\partial \gamma} = -\frac{1}{2} \sum_{i=1}^N (x_i - \mu)^2 + \frac{N}{2\gamma} = 0 \rightarrow \sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$$

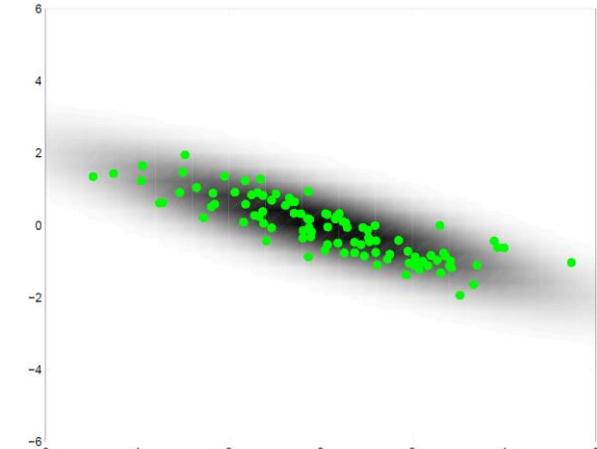
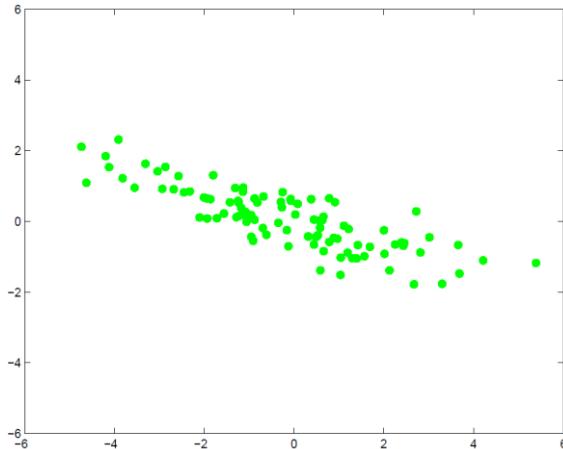
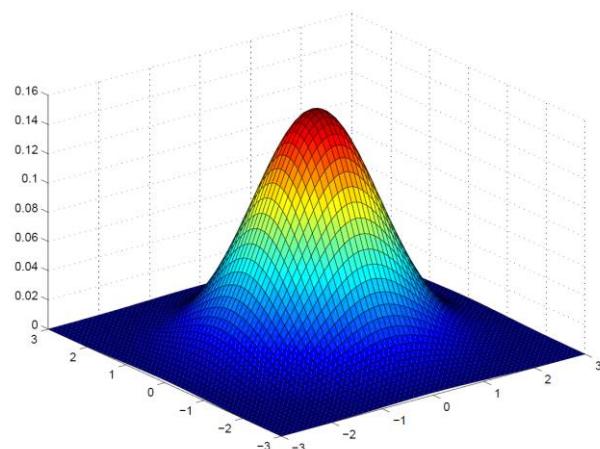
밀도 기반 이상치 탐지 기법: Gauss

- Gaussian Density Estimation

- ✓ 일반적인 다변량 데이터에 대해서도 동일한 방식으로 평균 벡터와 공분산 행렬을 구하는 것이 가능함

$$\boldsymbol{\mu} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i$$

$$\boldsymbol{\Sigma} = \frac{1}{N} \sum_{i=1}^N (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T$$



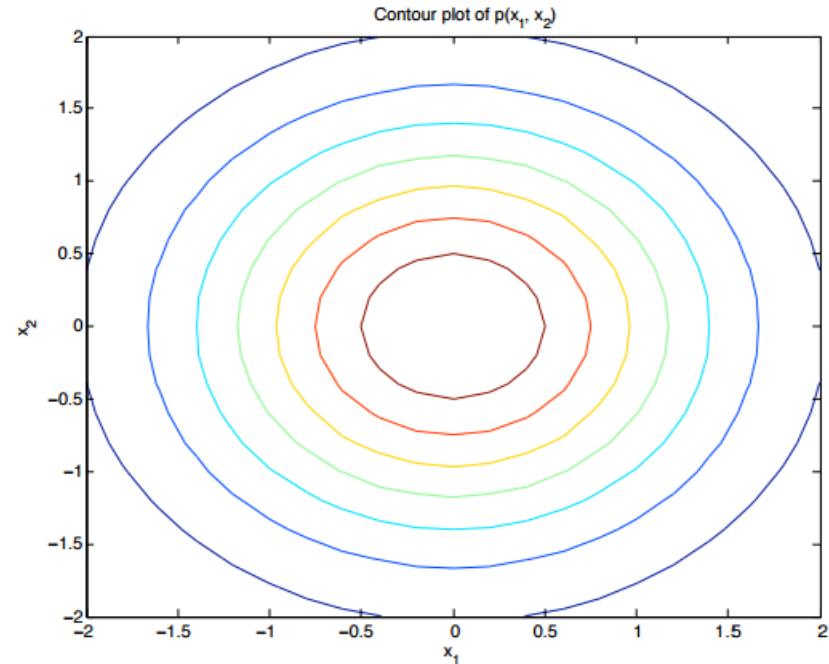
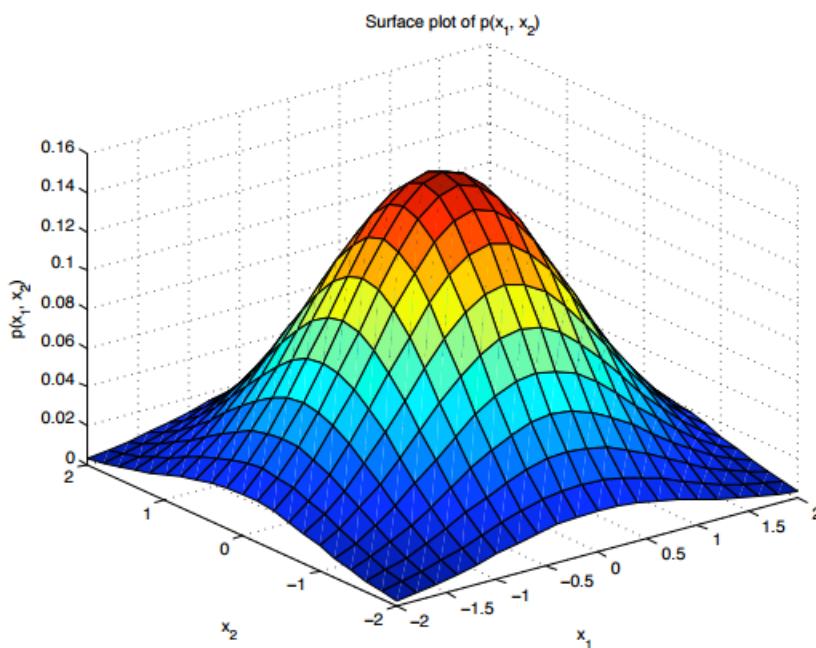
밀도 기반 이상치 탐지 기법: Gauss

- Gaussian Density Estimation

- ✓ 다변량 데이터에서 공분산행렬 조건에 따른 분포의 모양

Spherical

$$\Sigma = \sigma^2 \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}$$



(a) Spherical Gaussian (diagonal covariance, equal variances)

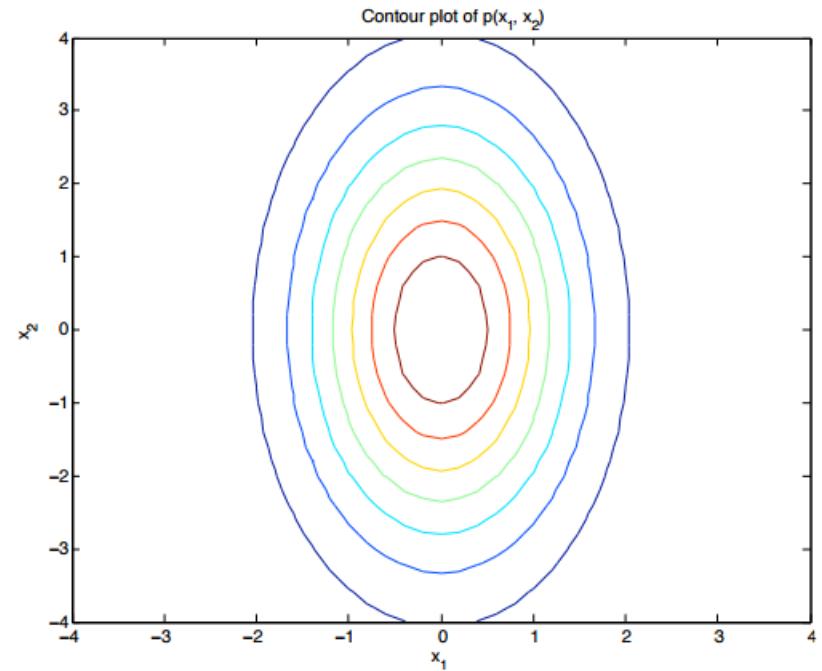
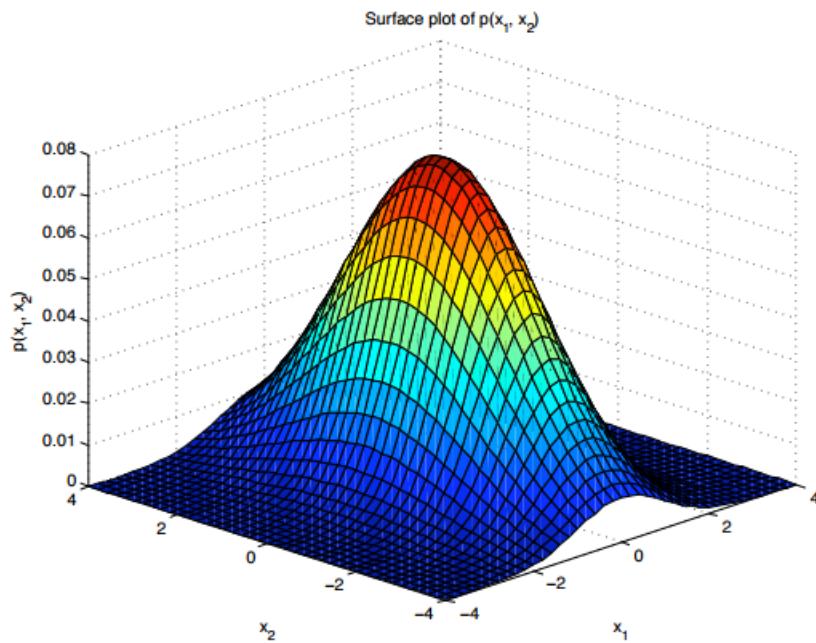
밀도 기반 이상치 탐지 기법: Gauss

- Gaussian Density Estimation

- ✓ 다변량 데이터에서 공분산행렬 조건에 따른 분포의 모양

Diagonal

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_d^2 \end{bmatrix}$$



(b) Gaussian with diagonal covariance matrix

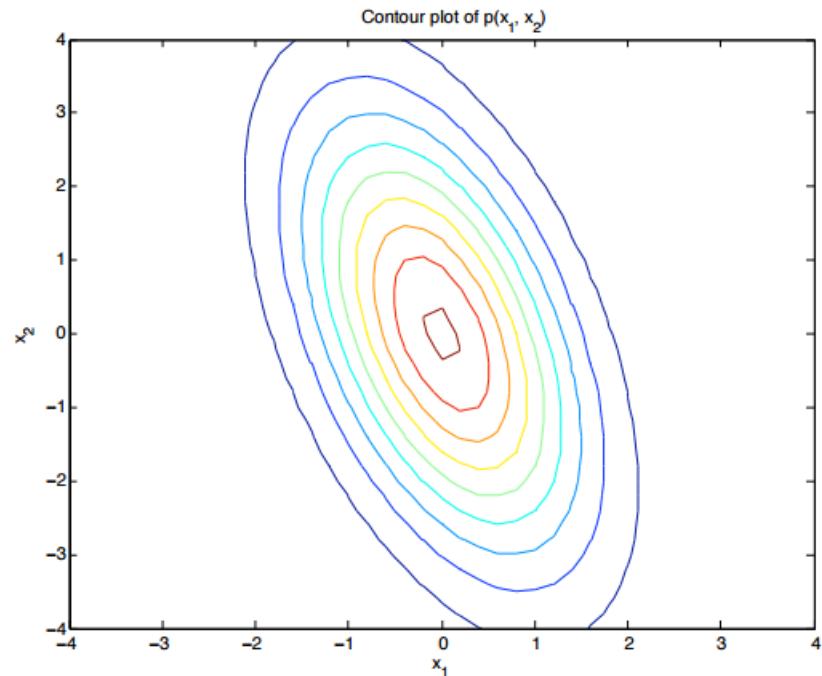
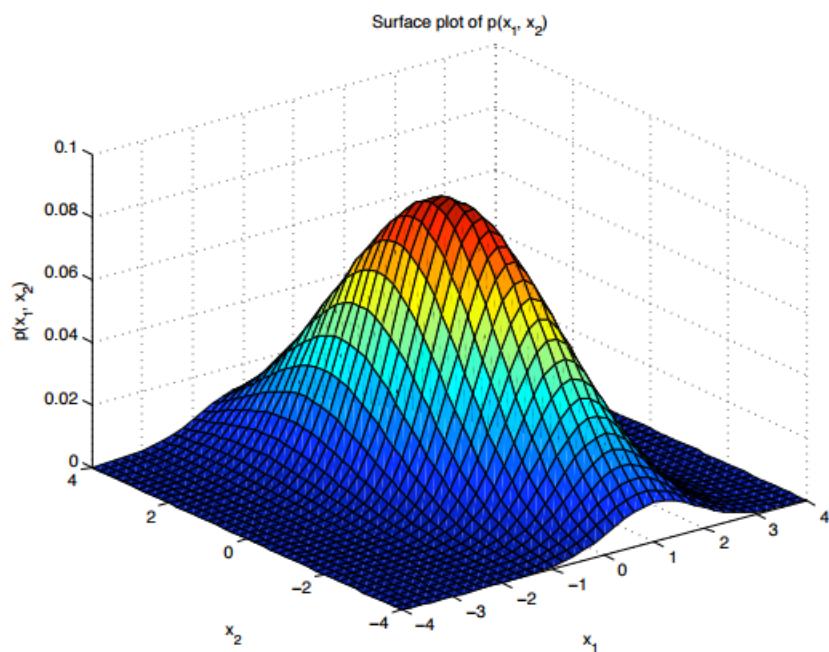
밀도 기반 이상치 탐지 기법: Gauss

- Gaussian Density Estimation

- ✓ 다변량 데이터에서 공분산행렬 조건에 따른 분포의 모양

Full

$$\Sigma = \begin{bmatrix} \sigma_{11} & \cdots & \sigma_{1d} \\ \vdots & \ddots & \vdots \\ \sigma_{d1} & \cdots & \sigma_{dd} \end{bmatrix}$$



(c) Gaussian with full covariance matrix

밀도 기반 이상치 탐지 기법: MoG

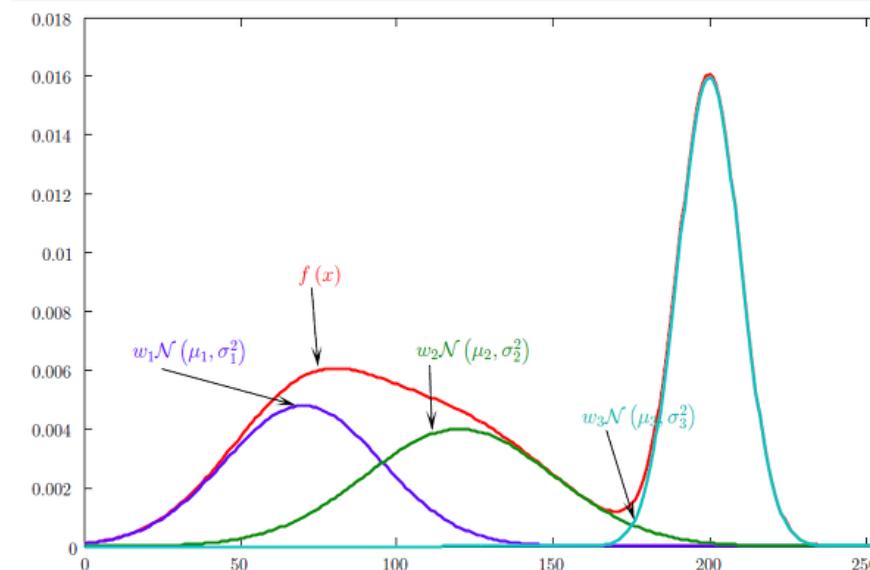
- Mixture of Gaussian (MoG) Density Estimation

- ✓ Gaussian Density Estimation

- 데이터의 분포에 대해 매우 강한 가정을 가지고 있음 → Unimodal Gaussian

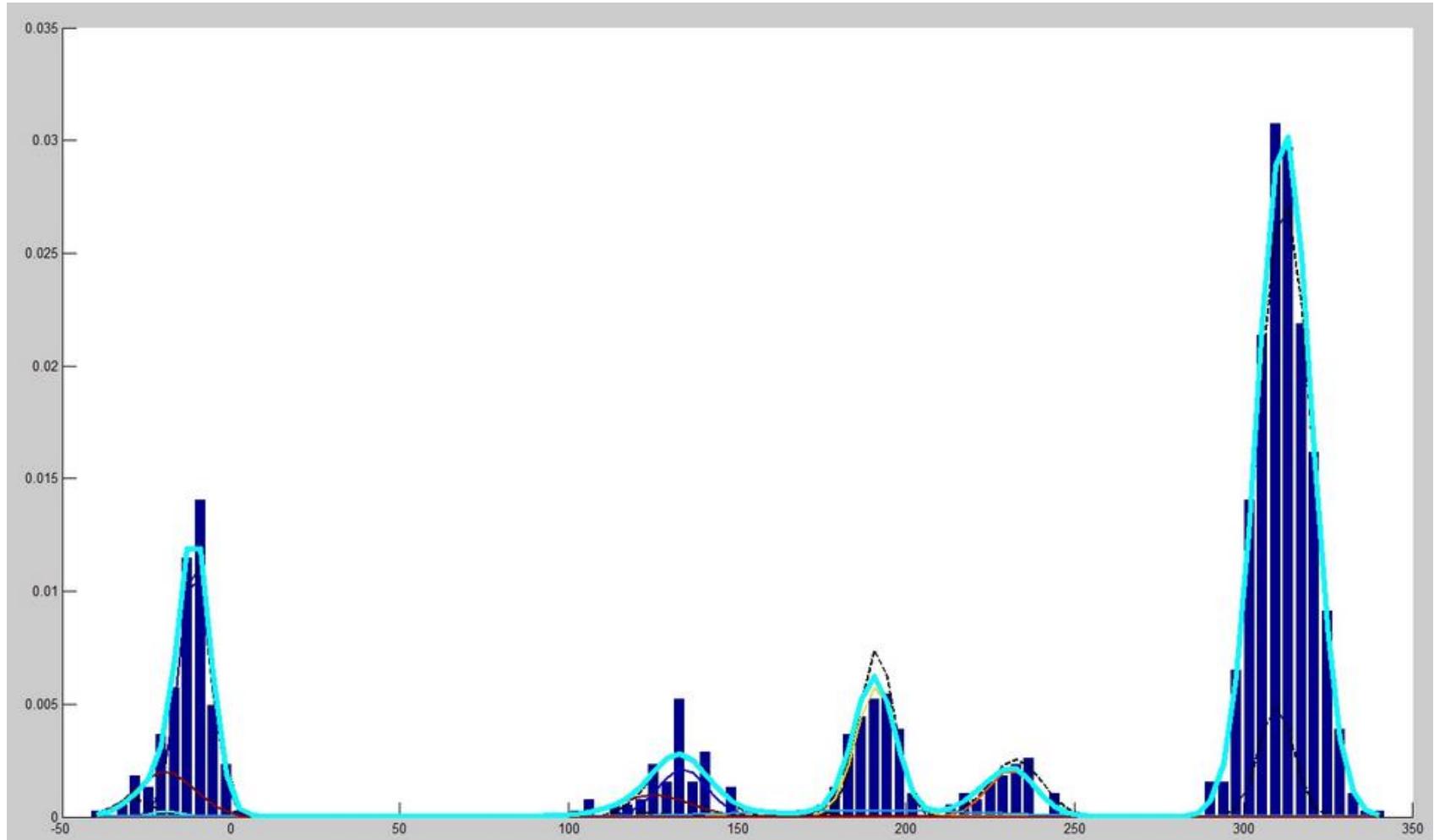
- ✓ MoG

- 데이터는 여러 개의 가우시안 분포의 혼합으로 이루어져 있음을 허용
 - 이 가우시안 분포들의 선형 결합으로 전체 데이터의 분포를 표현



밀도 기반 이상치 탐지 기법: MoG

- Mixture of Gaussian (MoG) Density Estimation



밀도 기반 이상치 탐지 기법: MoG

- Mixture of Gaussian (MoG) Density Estimation
 - ✓ Probability of an instance belonging to the normal class

$$p(\mathbf{x}|\lambda) = \sum_{m=1}^M w_m g(\mathbf{x}|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$$

- ✓ Distribution of each Gaussian model

$$g(\mathbf{x}|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_m|^{1/2}} \exp \left[\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_m)^T \boldsymbol{\Sigma}_m^{-1} (\mathbf{x} - \boldsymbol{\mu}_m) \right]$$

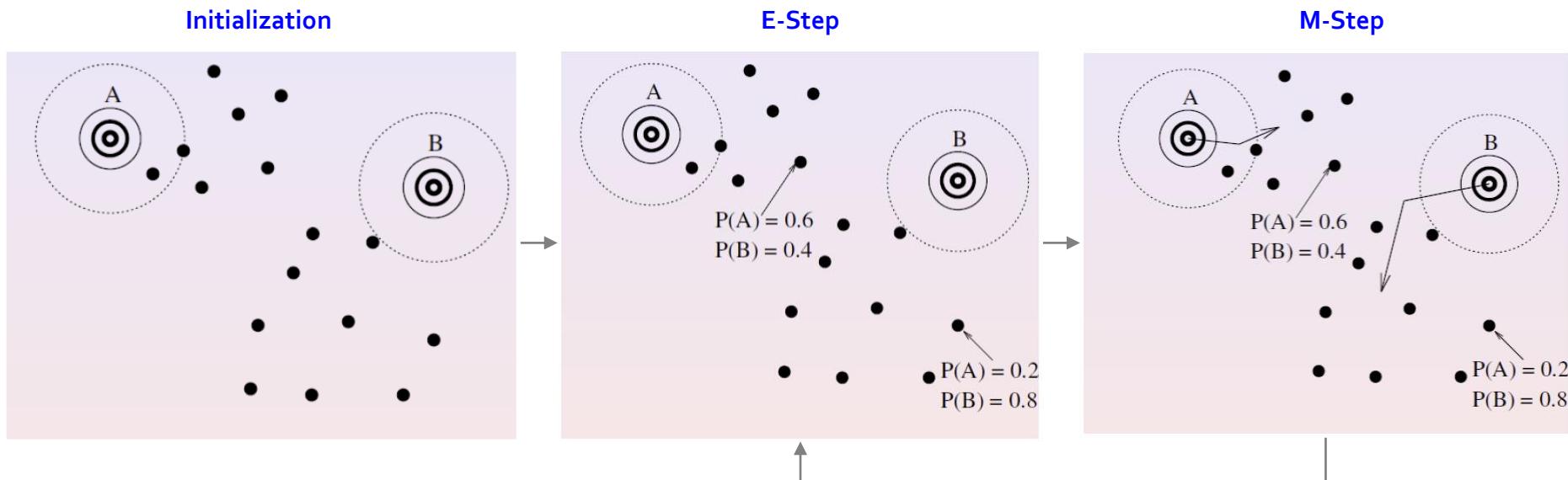
$$\lambda = \{w_m, \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m\}, \quad m = 1, \dots, M$$

밀도 기반 이상치 탐지 기법: MoG

- Mixture of Gaussian (MoG) Density Estimation

- ✓ Expectation-Maximization Algorithm

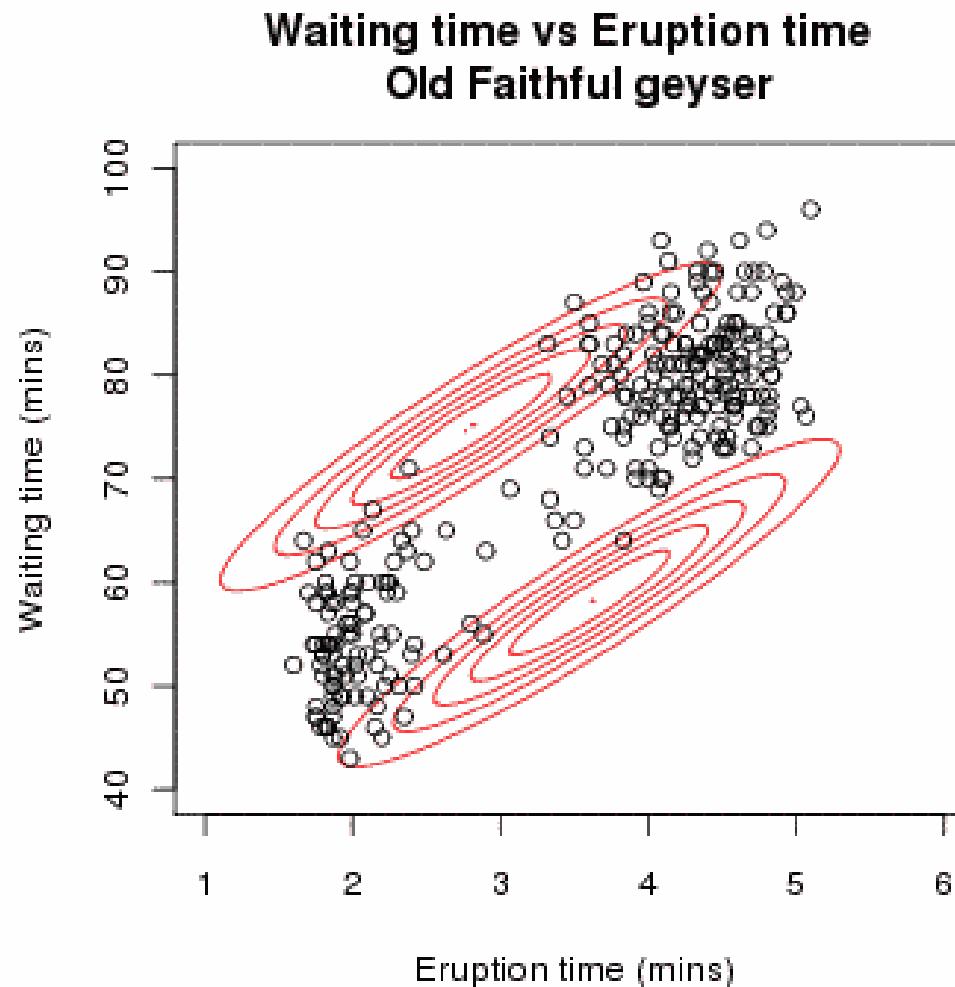
- E-Step: Given the current estimate of the parameters, compute the conditional probabilities
- M-Step: Update the parameters to maximize the expected likelihood found in the E-Step



밀도 기반 이상치 탐지 기법: MoG

- Mixture of Gaussian (MoG) Density Estimation

✓ EM 알고리즘 예시



밀도 기반 이상치 탐지 기법: MoG

- Mixture of Gaussian (MoG) Density Estimation

✓ Expectation

$$p(m|\mathbf{x}_i, \lambda) = \frac{w_m g(\mathbf{x}_i | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)}{\sum_{k=1}^M w_k g(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$$

✓ Maximization

- Mixture weight

$$w_m^{(new)} = \frac{1}{N} \sum_{i=1}^N p(m|\mathbf{x}_i, \lambda)$$

- Means and variances

$$\boldsymbol{\mu}_m^{(new)} = \frac{\sum_{i=1}^N p(m|\mathbf{x}_i, \lambda) \mathbf{x}_i}{\sum_{i=1}^N p(m|\mathbf{x}_i, \lambda)}, \quad \sigma_m^{2(new)} = \frac{\sum_{i=1}^N p(m|\mathbf{x}_i, \lambda) \mathbf{x}_i^2}{\sum_{i=1}^N p(m|\mathbf{x}_i, \lambda)} - \boldsymbol{\mu}_m^{2(new)}$$

밀도 기반 이상치 탐지 기법: MoG

- Mixture of Gaussian (MoG) Density Estimation

- ✓ 공분산 행렬의 형태에 따른 MoG의 모양

Spherical

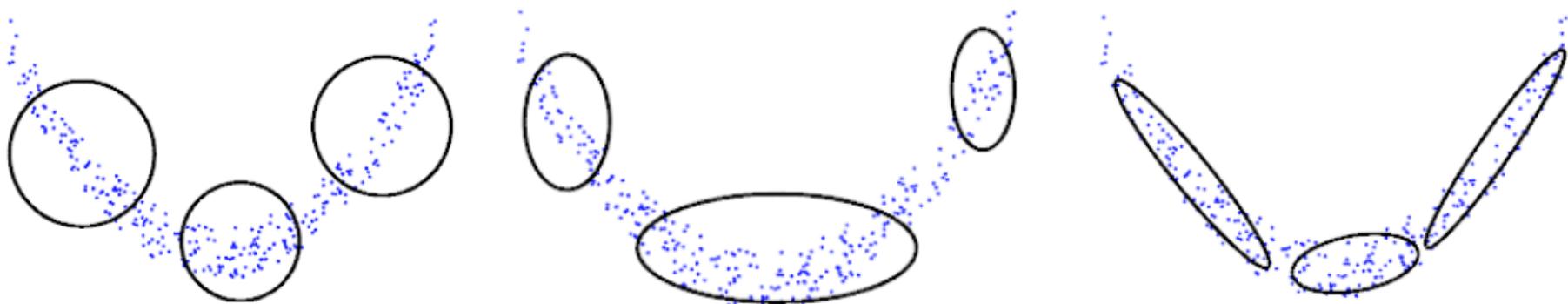
$$\Sigma = \sigma^2 \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}$$

Diagonal

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_d^2 \end{bmatrix}$$

Full

$$\Sigma = \begin{bmatrix} \sigma_{11} & \cdots & \sigma_{1d} \\ \vdots & \ddots & \vdots \\ \sigma_{d1} & \cdots & \sigma_{dd} \end{bmatrix}$$



- Less precise
- Very efficient to compute

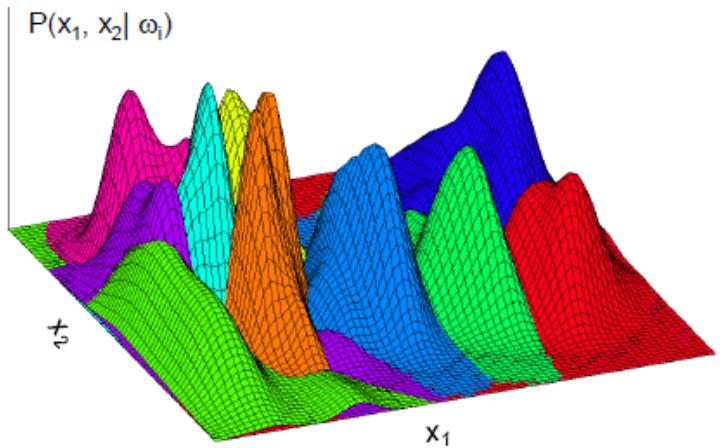
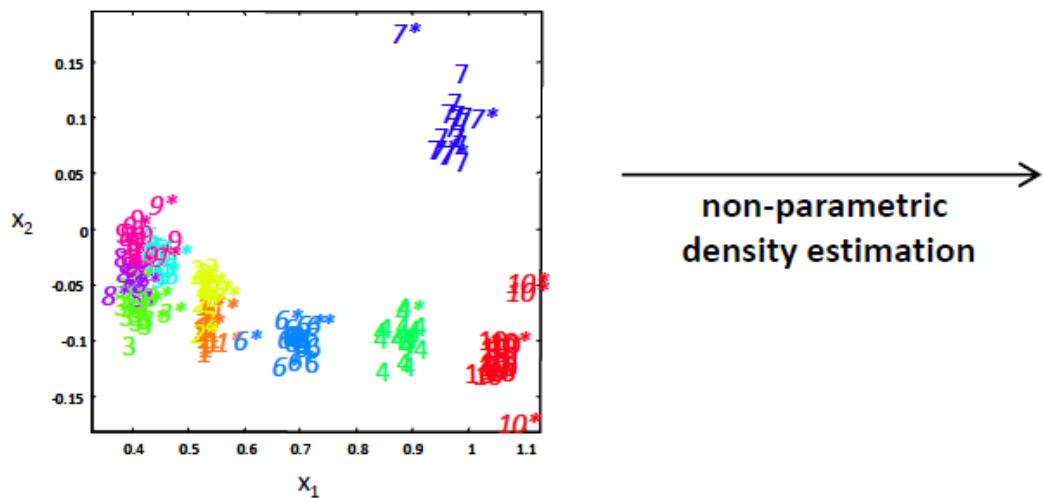
- More precise
- Efficient to compute

- Very precise
- Less efficient to compute

밀도 기반 이상치 탐지 기법: Parzen Window

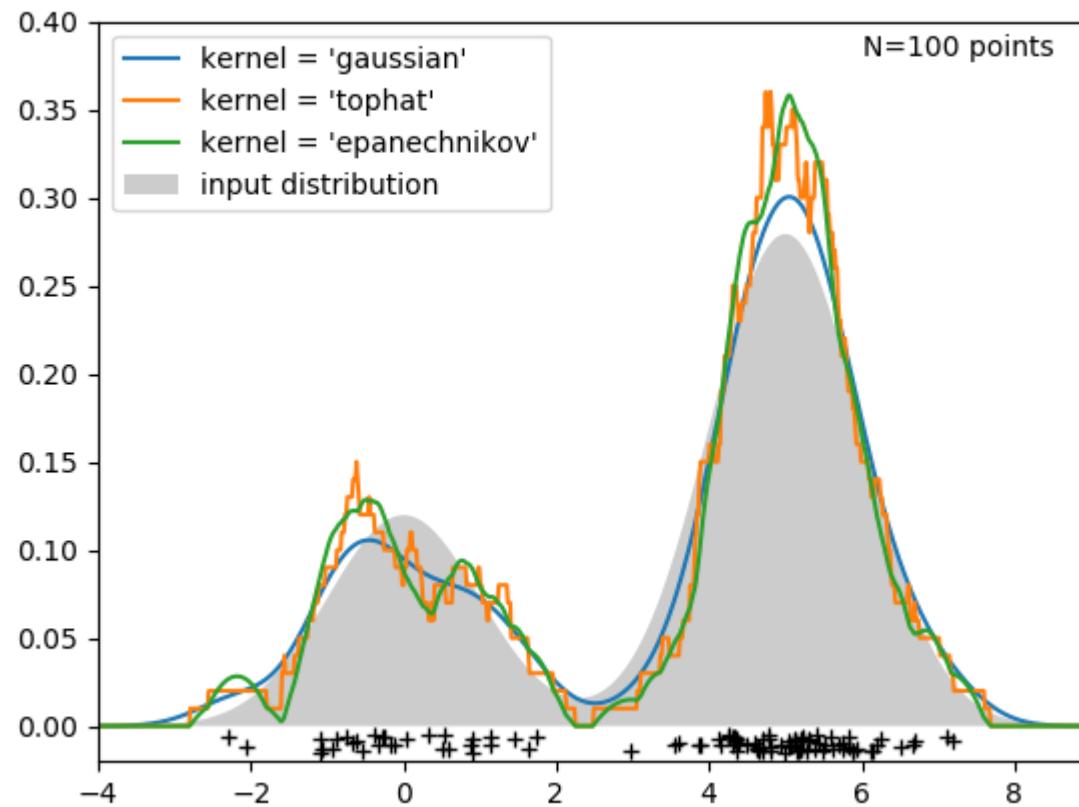
- Kernel-density Estimation

- ✓ 데이터가 특정한 분포(예: 가우시안)을 갖는다는 가정 없이 주어진 데이터로부터 주변부의 밀도를 추정하는 방식



밀도 기반 이상치 탐지 기법: Parzen Window

- Kernel-density Estimation: 1-D Example



밀도 기반 이상치 탐지 기법: Parzen Window

- Kernel-density Estimation (Optional)

- ✓ 확률밀도함수 $p(x)$ 로부터 생성된 벡터 x 가 주어진 영역 R 내부에 속할 확률

$$P = \int_R p(x') dx'$$

- ✓ N 개의 벡터 $\{x^1, x^2, \dots, x^n\}$ 가 동일한 분포로부터 생성되었을 때 그 중에서 k 개가 영역 R 내부에 속할 확률

$$P(k) = \binom{N}{k} P^k (1 - P)^{N-k}$$

- ✓ 이항 분포의 성질에 의해 k/N 의 기대값과 분산은 다음과 같이 계산됨

$$E\left[\frac{k}{N}\right] = P, \quad Var\left[\frac{k}{N}\right] = \frac{P(1 - P)}{N}$$

밀도 기반 이상치 탐지 기법: Parzen Window

- Kernel-density Estimation (Optional)

- ✓ N 이 무한대에 접근하게 되면 앞 식에서 분산은 0에 가까워짐
- ✓ 그렇다면 해당 영역에 대한 발생 확률은 앞 식의 기대값으로 표현할 수 있음

$$P \cong \frac{k}{N}$$

- ✓ 영역 R 의 크기가 $p(x)$ 의 변화가 없을 정도로 매우 작다고 가정을 하면 다음이 성립

$$P = \int_R p(x')dx' \cong p(x)V$$

- V 는 R 영역의 Volume
- ✓ 앞의 두 결과물을 결합하면 다음이 성립

$$P = \int_R p(x')dx' \cong p(x)V = \frac{k}{N}, \quad p(x) = \frac{k}{NV}$$

밀도 기반 이상치 탐지 기법: Parzen Window

- Kernel-density Estimation (Optional)

$$p(x) = \frac{k}{NV}, \quad \text{where } \left\{ \begin{array}{l} V: \text{volume surrounding } x \\ N: \text{the total number of examples} \\ k: \text{the number of examples inside } V \end{array} \right\}$$

- ✓ 확률밀도의 추정은 샘플 수 N 이 클수록, 영역의 볼륨 V 가 작을수록 정확해짐
- ✓ 실제 상황에서는 데이터의 수는 고정되어 있으므로 적절한 V 를 찾는 문제로 귀결
 - 영역 R 내부에 충분한 데이터가 포함되도록 커야 하며,
 - 영역 R 내부에서는 $p(x)$ 의 변동이 없다는 가정을 뒷받침해줄 수 있도록 작아야 함
- ✓ V 를 고정시키고 k 를 찾는 것이 커널 밀도 추정의 목적임

밀도 기반 이상치 탐지 기법: Parzen Window

- Kernel-density Estimation (Optional)

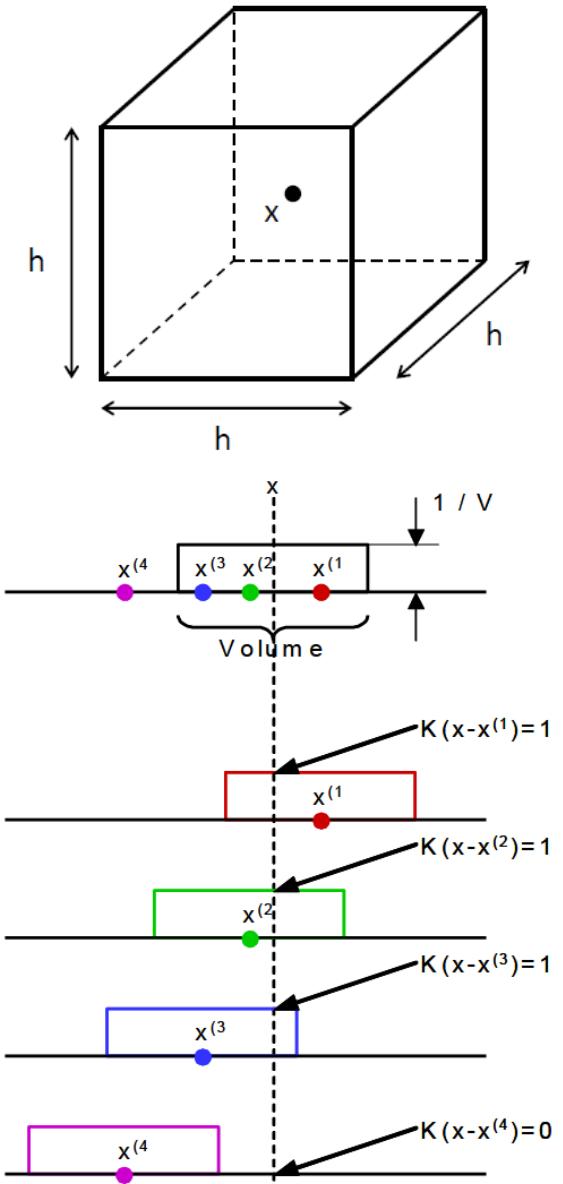
- ✓ Parzen Window Density Estimation

- k개의 객체를 포함하는 영역 \mathbf{x} 를 중심으로 하며 각 면의 길이가 h 인 Hypercube로 정의
 - 이의 볼륨 V 는 h^d 로 정의됨 (d 는 차원 수)
 - Kernel function을 다음과 같이 정의

$$K(u) = \begin{cases} 1 & |u_j| < \frac{1}{2} \forall j = 1, \dots, d \\ 0 & otherwise \end{cases}$$

$$k = \sum_{i=1}^N K\left(\frac{\mathbf{x}^i - \mathbf{x}}{h}\right)$$

$$p(x) = \frac{1}{Nh^d} \sum_{i=1}^N K\left(\frac{\mathbf{x}^i - \mathbf{x}}{h}\right)$$



밀도 기반 이상치 탐지 기법: Parzen Window

- Kernel-density Estimation (Optional)

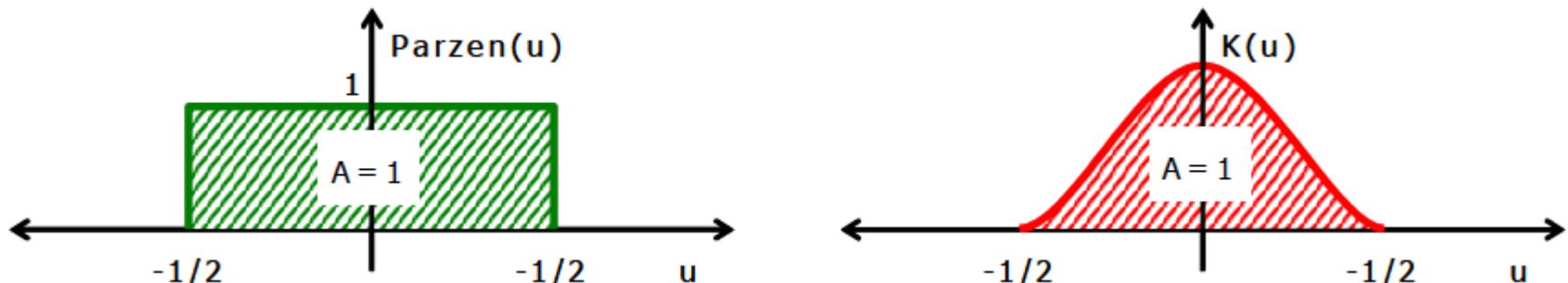
- ✓ $K(u)$ 의 단점

- 불연속적인 추정, Hypercube 내에 있는 객체들에 대한 동등한 가중치

- ✓ Smooth kernel function

$$P = \int_R K(x) dx = 1$$

- Commonly use a radially symmetric and unimodal pdf, such as Gaussian



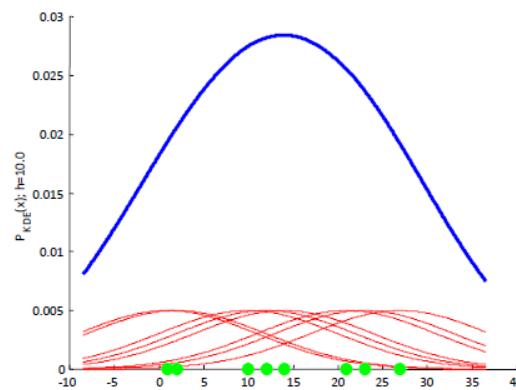
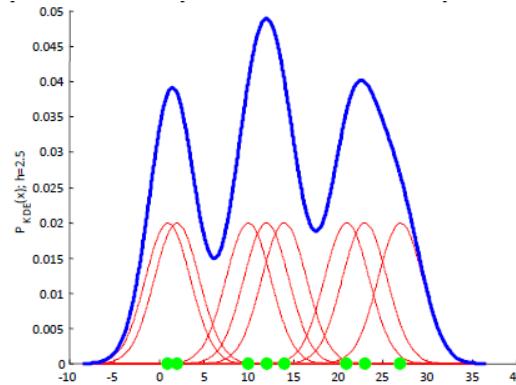
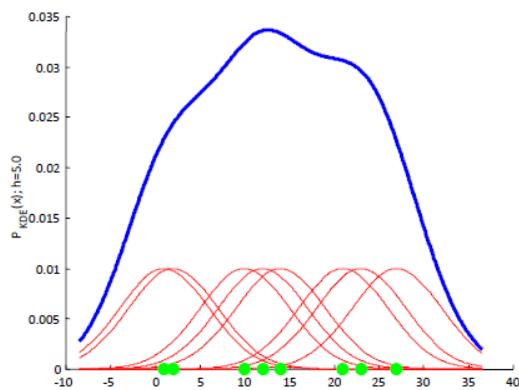
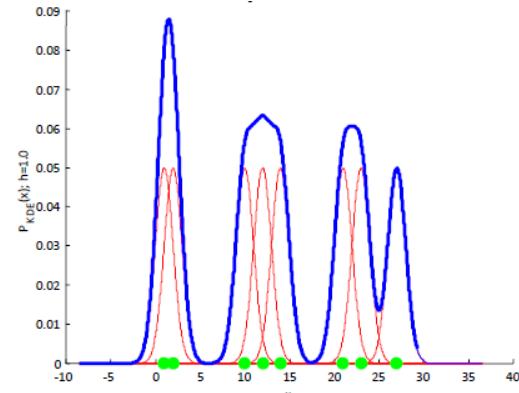
$$p(x) = \frac{1}{N} \sum_{i=1}^N K\left(\frac{\mathbf{x}^i - \mathbf{x}}{h}\right)$$

밀도 기반 이상치 탐지 기법: Parzen Window

- Kernel-density Estimation (Optional)

- ✓ Smoothing parameter (bandwidth) h

- A large h will over-smooth the density distribution
 - A small h will result in a spiky density distribution

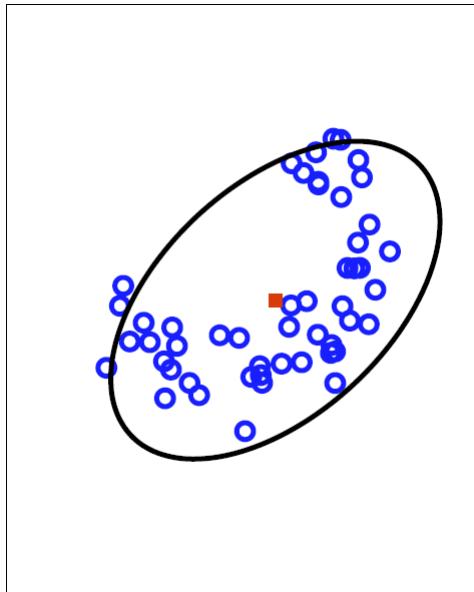


밀도 기반 이상치 탐지 기법: Parzen Window

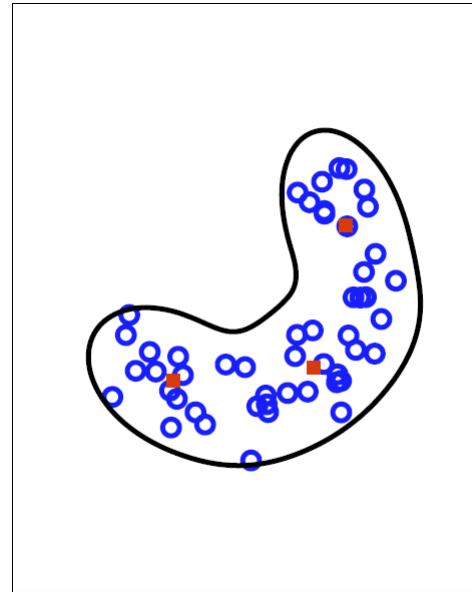
- Kernel Density Estimation

- ✓ The smoothing parameter h can be optimized through EM algorithm

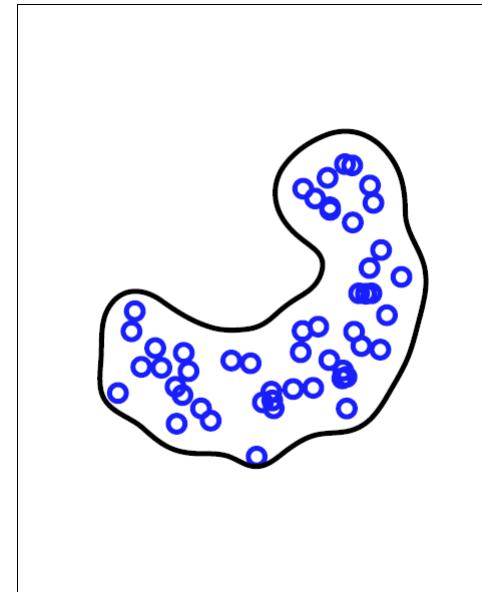
Gaussian density estimation



Mixture of Gaussian



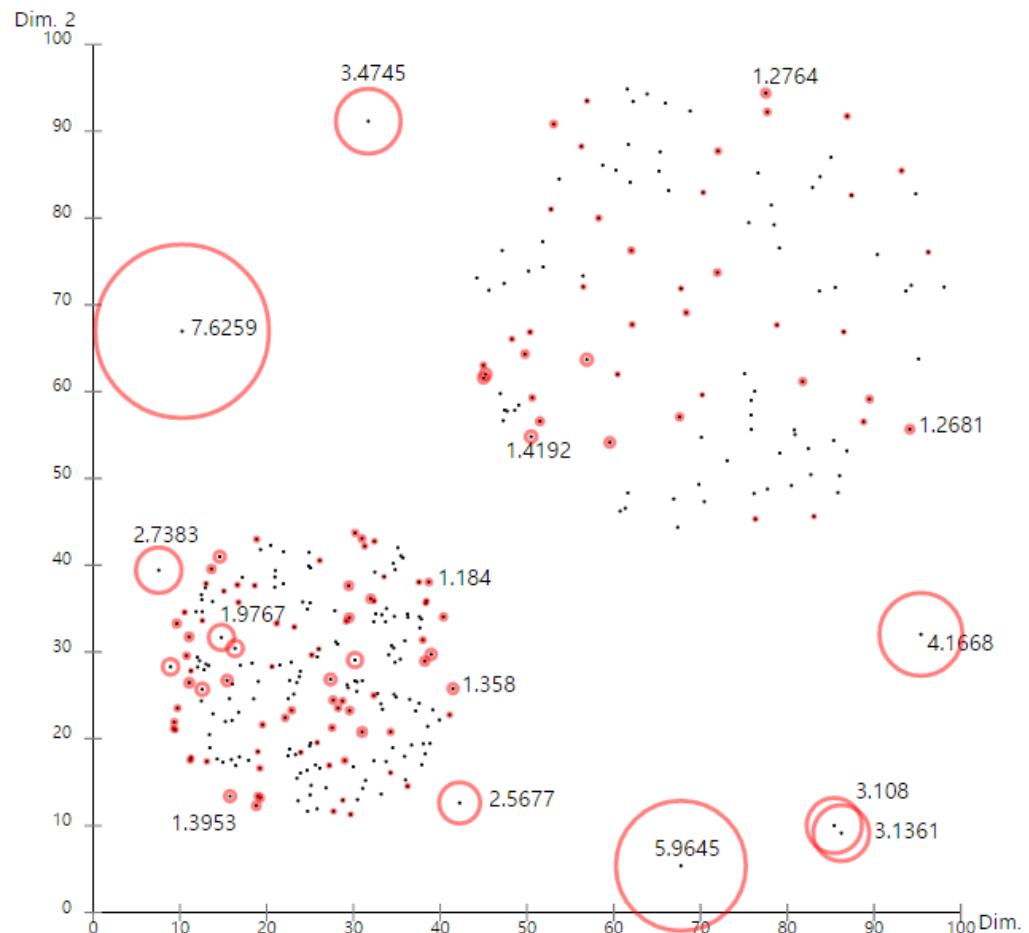
Parzen window with Gaussian Kernel



밀도 기반 이상치 탐지 기법: LOF

- Local Outlier Factor (LOF)

- ✓ 목적: 이상치 스코어를 산출할 때, 주변부 데이터의 밀도를 고려하고자 함



밀도 기반 이상치 탐지 기법: LOF

- LOF 알고리즘

- ✓ **Definition I:** k-distance of an object p

- 임의의 양의 정수 k에 대해서 the k-distance of object p ($k\text{-distance}(p)$)는 다음 두 조건을 만족하는 데이터셋 D의 두 점 p와 o의 거리 $d(p,o)$ 로 정의됨
 - D에 속하는 개체 중 p를 제외하고 최소한 k개의 개체 o'에 대해서 $d(p,o') \leq d(p,o)$ 를 만족
 - D에 속하는 개체 중 p를 제외하고 최대 k-1개의 개체 o'에 대해서 $d(p,o') < d(p,o)$ 를 만족
 - 이는 단순히 동률을 고려한 k번째 이웃까지의 거리로 생각할 수 있음

1st	2nd	3rd	4th	5th	6th	7th	3-distance
1	2	3	3	3	4	5	3
1	2	2	2	3	4	5	2
1	1	1	1	2	3	4	1

밀도 기반 이상치 탐지 기법: LOF

- LOF 알고리즘

- ✓ **Definition 2:** k-distance neighborhood of an object p

- 개체 p의 k-distance가 Definition 1과 같이 주어질 때, k-distance neighborhood of p 는 p에서부터 k-distance보다 멀지 않은 거리에 있는 모든 개체들의 집합을 의미함

$$N_k(p) = \{q \in D \setminus \{p\} \mid d(p, q) \leq k - \text{distance}(p)\}$$

- Example

1st	2nd	3rd	4th	5th	6th	7th	$k - \text{distance}(3)$	$N_3(p)$
1	2	3	3	3	4	5	3	5
1	2	2	2	3	4	5	2	4
1	1	1	1	2	3	4	1	4

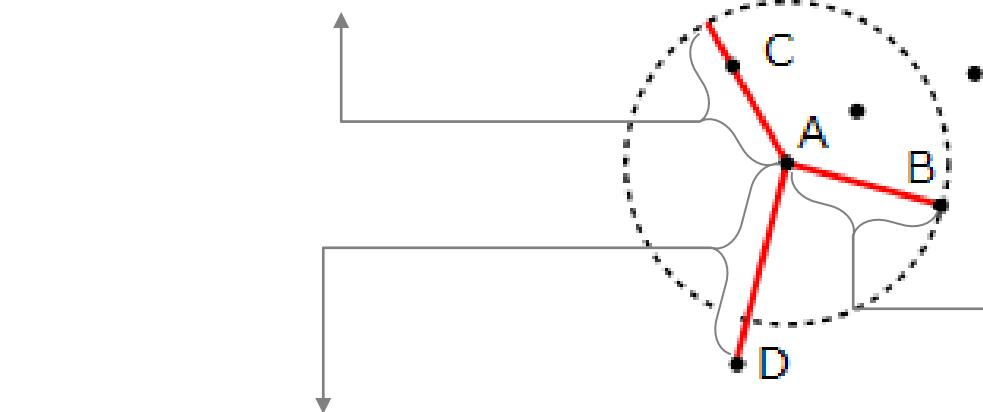
밀도 기반 이상치 탐지 기법: LOF

- LOF 알고리즘

- ✓ **Definition 3:** reachability distance

- $reachability-distance_k(p, o) = \max\{k - distance(o), d(p, o)\}$
 - Examples

$reachability-distance_3(C, A)$



$reachability-distance_3(D, A)$

$$3 - distance(A) \\ = reachability-distance_3(B, A)$$

밀도 기반 이상치 탐지 기법: LOF

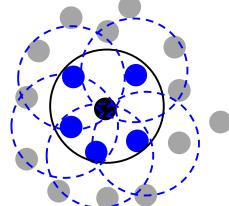
- LOF 알고리즘

- ✓ **Definition 4:** local reachability density of an object p

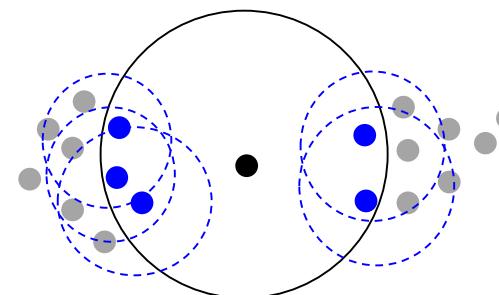
$$lrd_k(p) = \frac{|N_k(p)|}{\sum_{o \in N_k(p)} \text{reachability} - \text{distance}_k(p, o)}$$

- Case 1: 개체 p의 주변에 높은 밀도로 다른 개체들이 존재하는 경우 위 식의 분모인 $lrd_k(p)$ 는 작게 되어 $lrd_k(p)$ 는 큰 값을 갖는다.
- Case 2: 개체 p가 두 개의 높은 밀도를 갖는 군집 사이의 밀도가 낮은 공간에 위치하게 되면 위 식의 분모인 $lrd_k(p)$ 가 커지게 되고 $lrd_k(p)$ 는 작은 값을 갖는다.

Case 1



Case 2



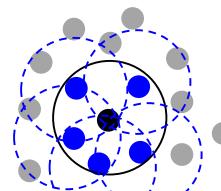
밀도 기반 이상치 탐지 기법: LOF

- LOF 알고리즘

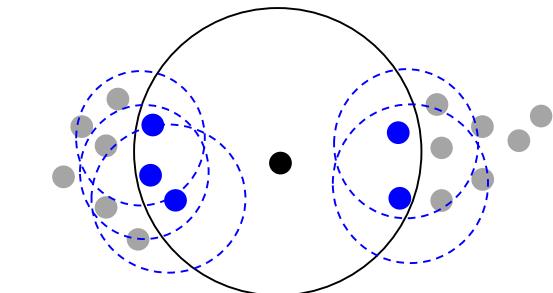
✓ **Definition 5:** local outlier factor of an object p

$$LOF_k(p) = \frac{\sum_{o \in N_k(p)} \frac{lrd_k(o)}{lrd_k(p)}}{|N_k(p)|} = \frac{1}{lrd_k(p)} \frac{\sum_{o \in N_k(p)} lrd_k(o)}{|N_k(p)|}$$

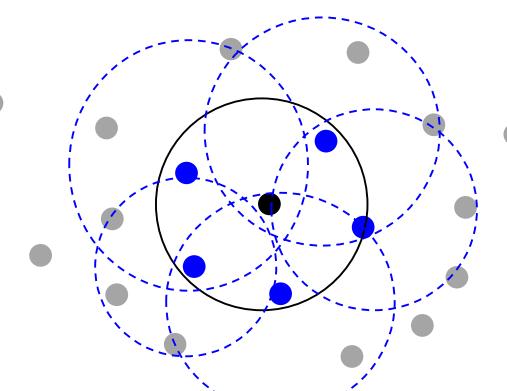
Case 1



Case 2



Case 3



● : p

● : o

○ : k -distance of p

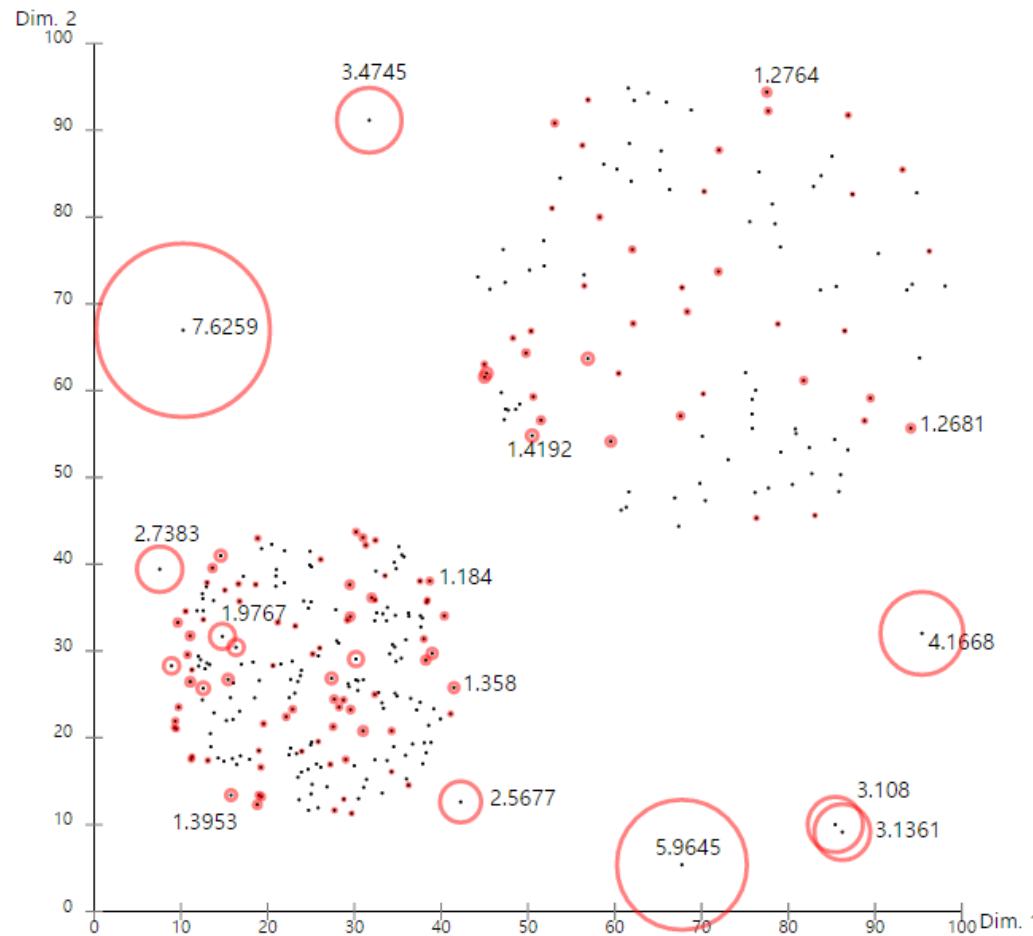
○ : k -distance of o

Case	$lrd_k(p)$	$lrd_k(o)$	$LOF_k(p)$
Case 1	Large	Large	Small
Case 2	Small	Large	Large
Case 3	Small	Small	Small

밀도 기반 이상치 탐지 기법: LOF

- LOF 알고리즘

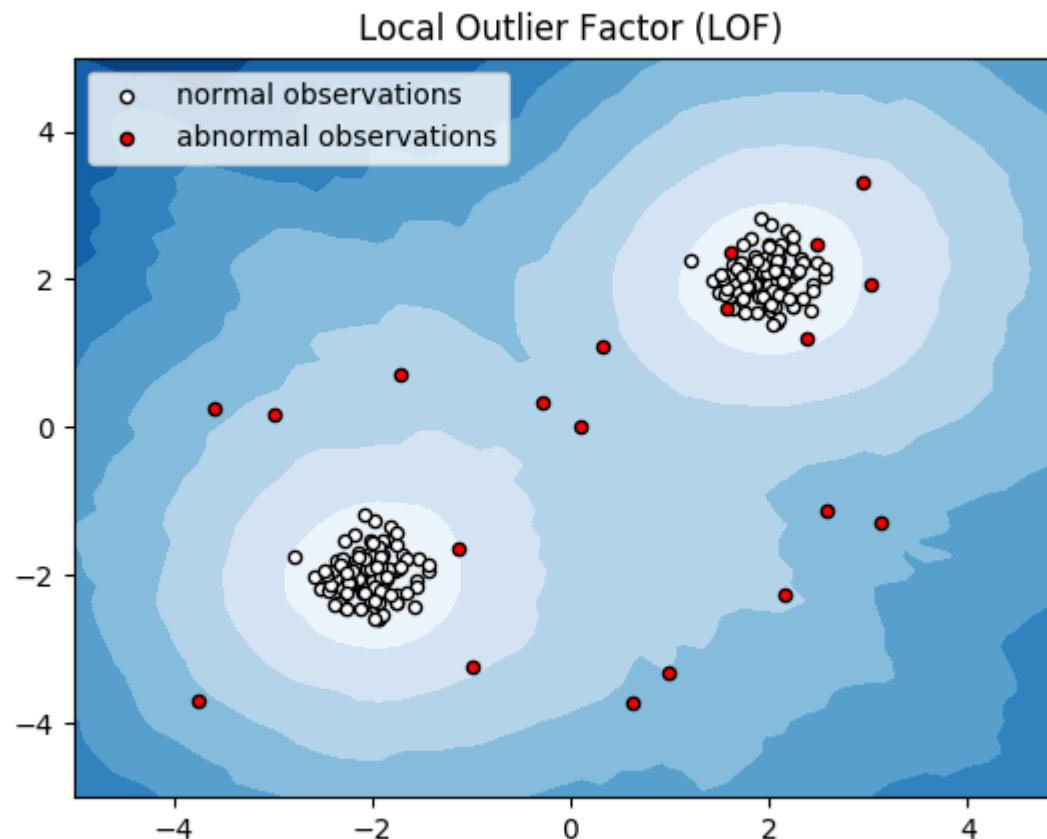
- ✓ LOF의 결과물은 각 개체들의 주변 밀도를 고려한 이상치 스코어임



밀도 기반 이상치 탐지 기법: LOF

- Local Outlier Factors (LOF)

- ✓ LOF contour plot



AGENDA

01 이상치 탐지: 개요

02 밀도 기반 이상치 탐지 기법

03 모델 기반 이상치 탐지 기법

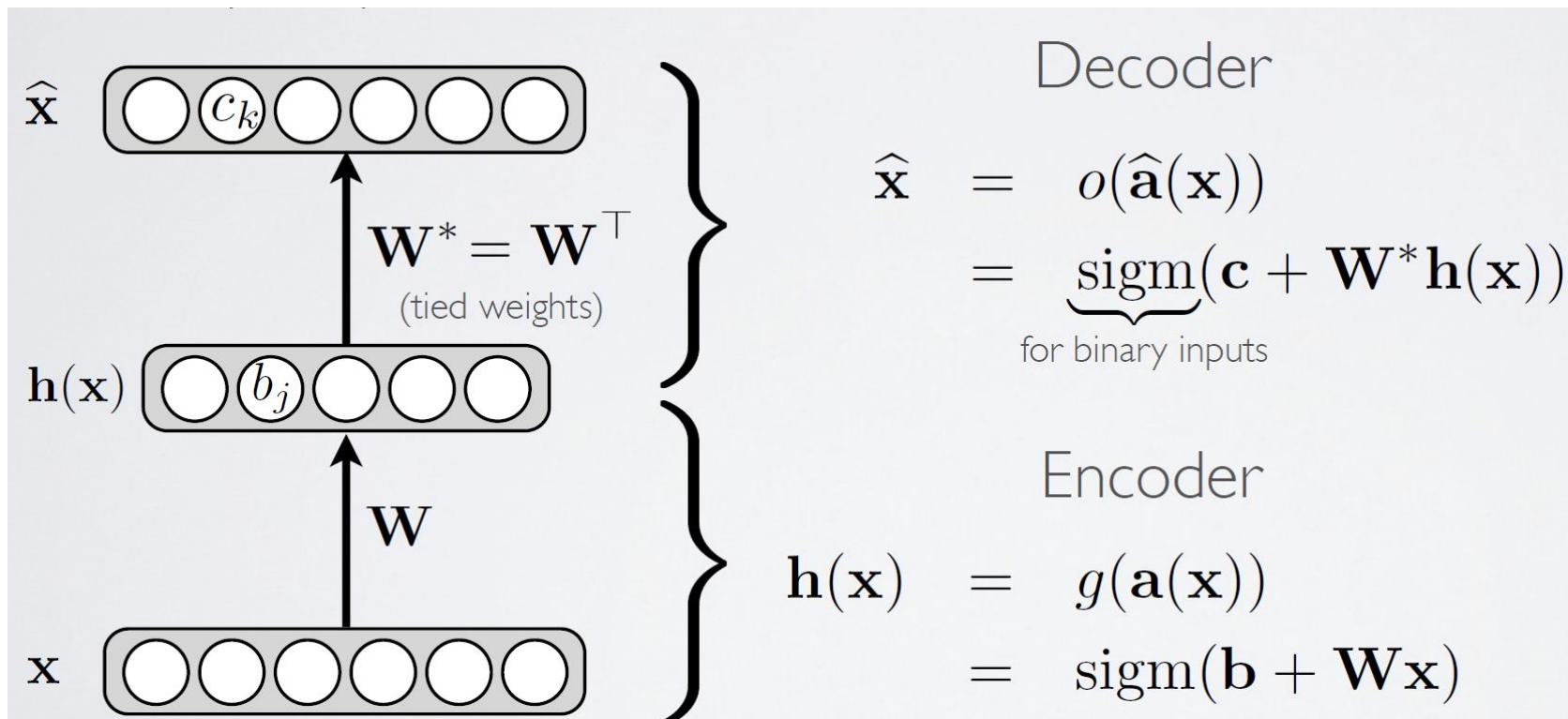
모델 기반 이상치 탐지 기법: AE

- Auto-Encoder (Auto-Associative Neural Network)

✓ 입력과 출력이 동일한 인공 신경망 구조

- Loss function:

$$l(f(\mathbf{x})) = \frac{1}{2} \sum_k (\hat{x}_k - x_k)^2$$



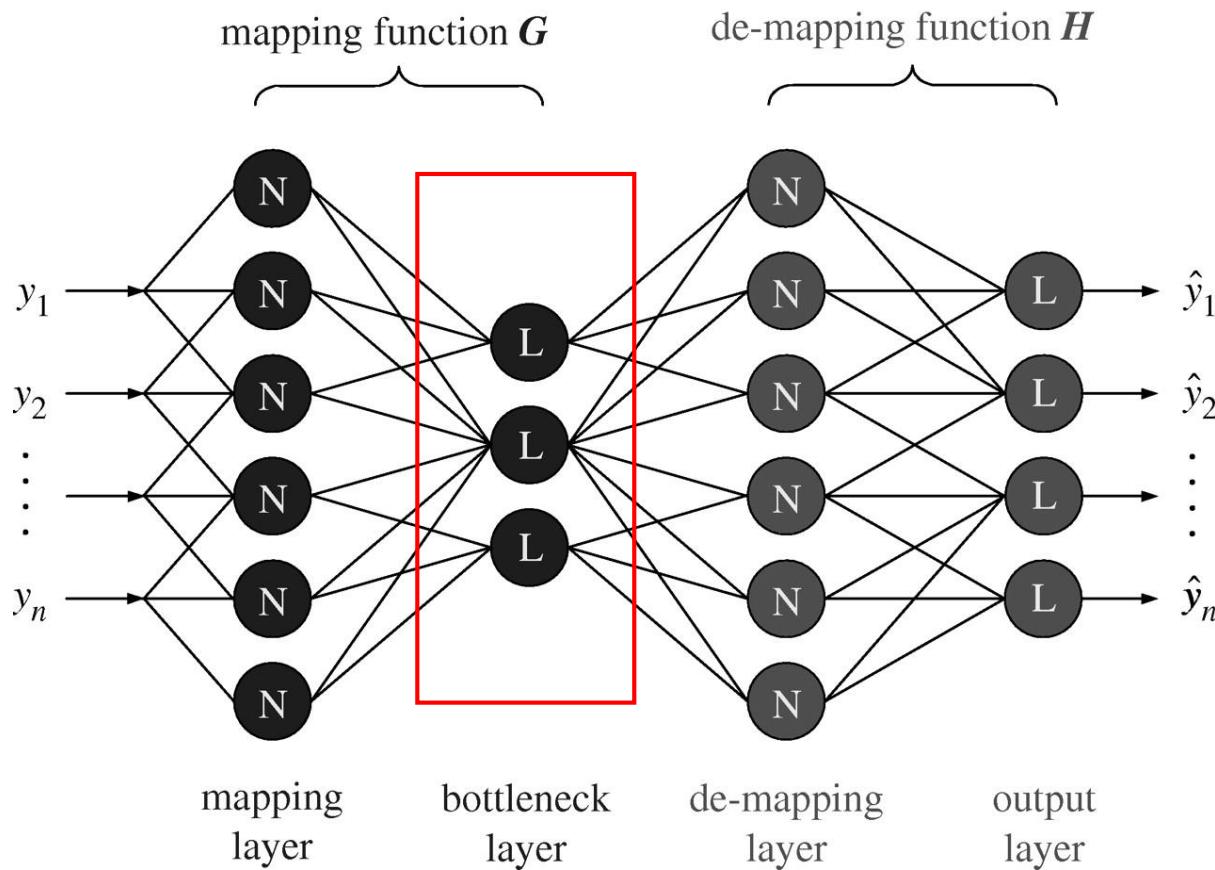
모델 기반 이상치 탐지 기법: AE

- Auto-Encoder (Auto-Associative Neural Network)
 - ✓ 입력과 출력이 동일한 인공 신경망 구조
 - ✓ 정상 데이터들에 대한 학습이 충분히 되어 있을 경우
 - 정상 데이터는 자기 자신을 잘 복제할 수 있는 신경망이 학습이 되나,
 - 이상치 데이터는 학습 기회가 적어서 상대적으로 복제를 잘 못할 것을 가정하는 모형

모델 기반 이상치 탐지 기법: AE

- Auto-Encoder (Auto-Associative Neural Network)

- ✓ 반드시 입력 변수의 수보다 은닉 노드의 수가 더 적은 은닉 층이 있어야 함
 - 이 층에서 정보의 축약이 이루어짐



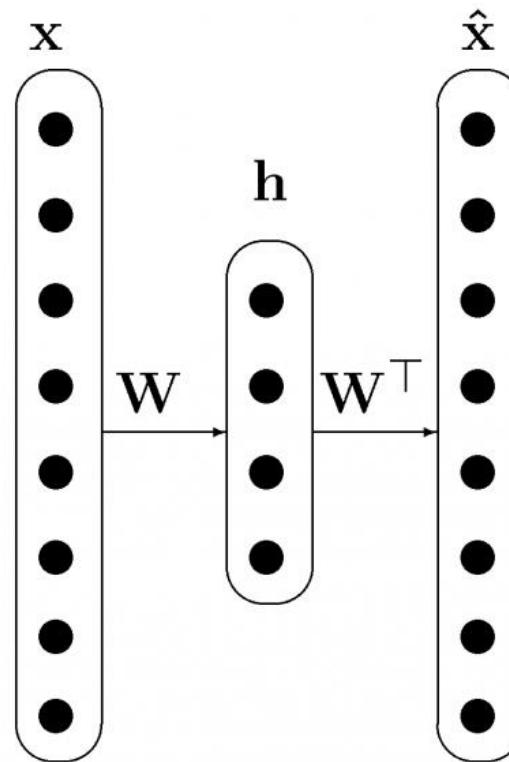
모델 기반 이상치 탐지 기법: AE

- Auto-Encoder (Auto-Associative Neural Network) 예시

✓ 숫자 2를 학습시키는 오토 인코더 → 5를 입력으로 제공하면 5가 산출되지 않을 가능성이 높음 (Loss가 큼) → 이 입력은 이상치로 판별하자!



(a) Input



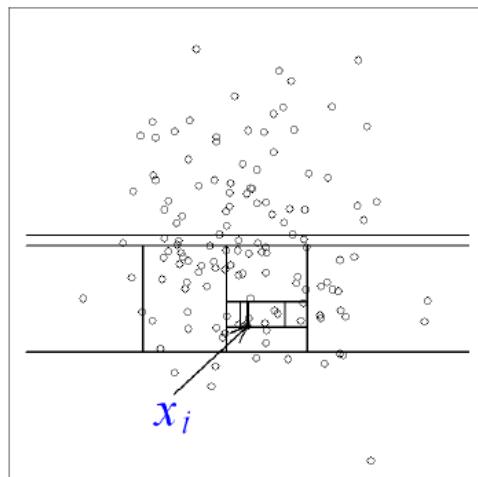
(b) Neural Encoding



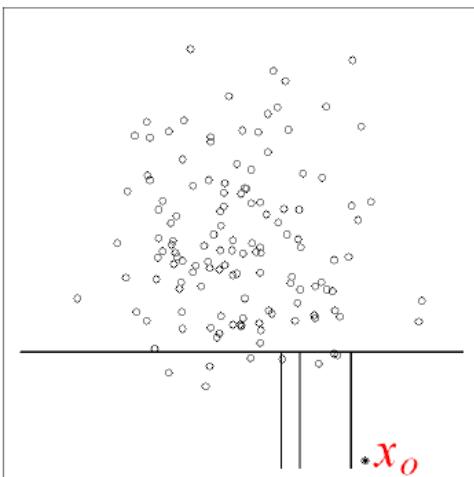
(c) Reconstruction

모델 기반 이상치 탐지 기법: IF

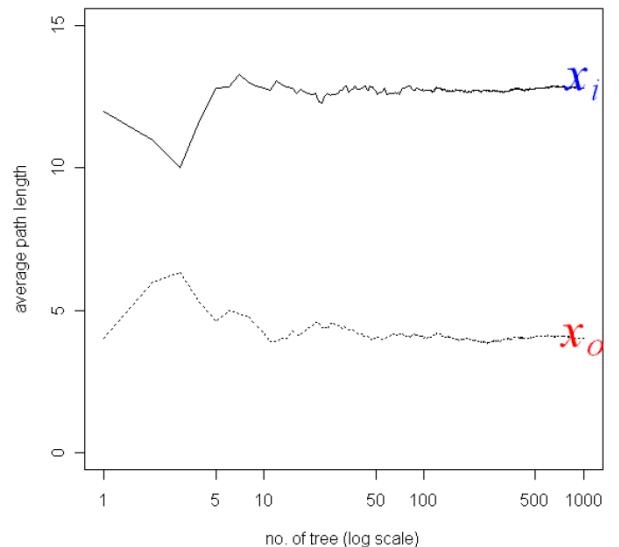
- Isolation Forest (I-Forest)
 - ✓ Motivation: Few and Different
 - 소수 범주(이상치)는 개체수가 적음
 - 소수 범주 데이터는 정상 범주 데이터와는 특정 속성 값이 많이 다를 가능성이 높음
- 하나의 객체를 고립(isolation)시키는 Tree를 생성해보자
 - ✓ 정상 데이터라면 고립시키는데 많은 분기(split)가 필요
 - ✓ 이상치 데이터라면 상대적으로 적은 분기(split)만으로 고립 가능



(a) Isolating x_i



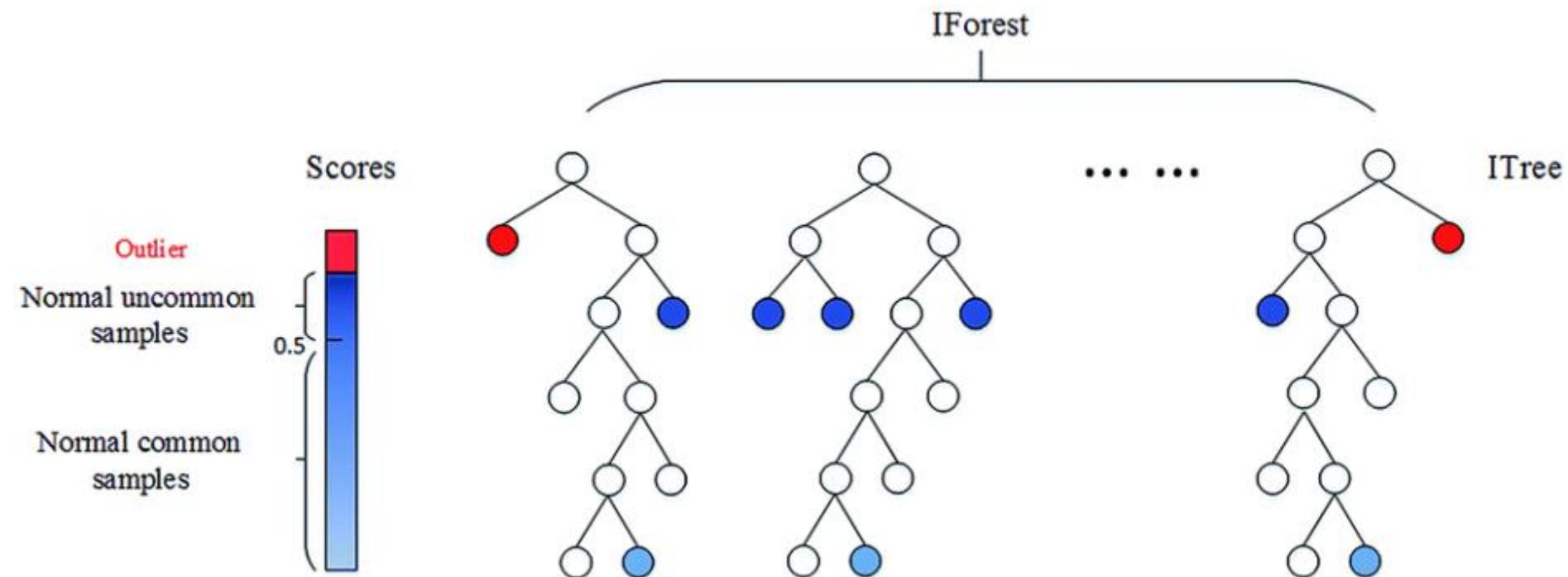
(b) Isolating x_o



모델 기반 이상치 탐지 기법: IF

- I-Forest

- ✓ 객체를 고립시킬 때까지 몇 번이나 분기(split)를 했는지에 대한 정보로 이상치 점수를 부여할 수 있지 않을까?



모델 기반 이상치 탐지 기법: IF

- Isolation Tree (iTree)
 - ✓ n 개의 객체로 구성된 샘플 데이터 X 는 임의의 속성 q 의 임의의 기준점 p 를 사용하여 다음 조건을 만족할 때까지 재귀적으로 분기를 수행
 - Tree가 사전에 설정한 최대 깊이에 도달
 - 영역 X 안에 단 하나의 객체만 존재
 - 영역 X 안에 존재하는 객체들이 모두 같은 입력 변수 값들을 가짐
- Path Length (경로 길이)
 - ✓ 객체 x 의 경로 길이 $h(x)$ 는 Root node로부터 x 가 속한 말단 노드까지 도달하기 위해 거쳐간 edge의 수로 정의됨
 - $h(x)$ 는 평균 기대 path length $c(n)$ 을 사용하여 다음과 같이 정규화 가능

$$c(n) = 2H(n - 1) - \frac{2(n - 1)}{n}, \quad H(i) = \ln(i) + 0.5772156649 \text{ (Euler's constant)}$$

모델 기반 모델 기반 이상치 탐지 기법: IF 탐지 기법

- 이상치 스코어

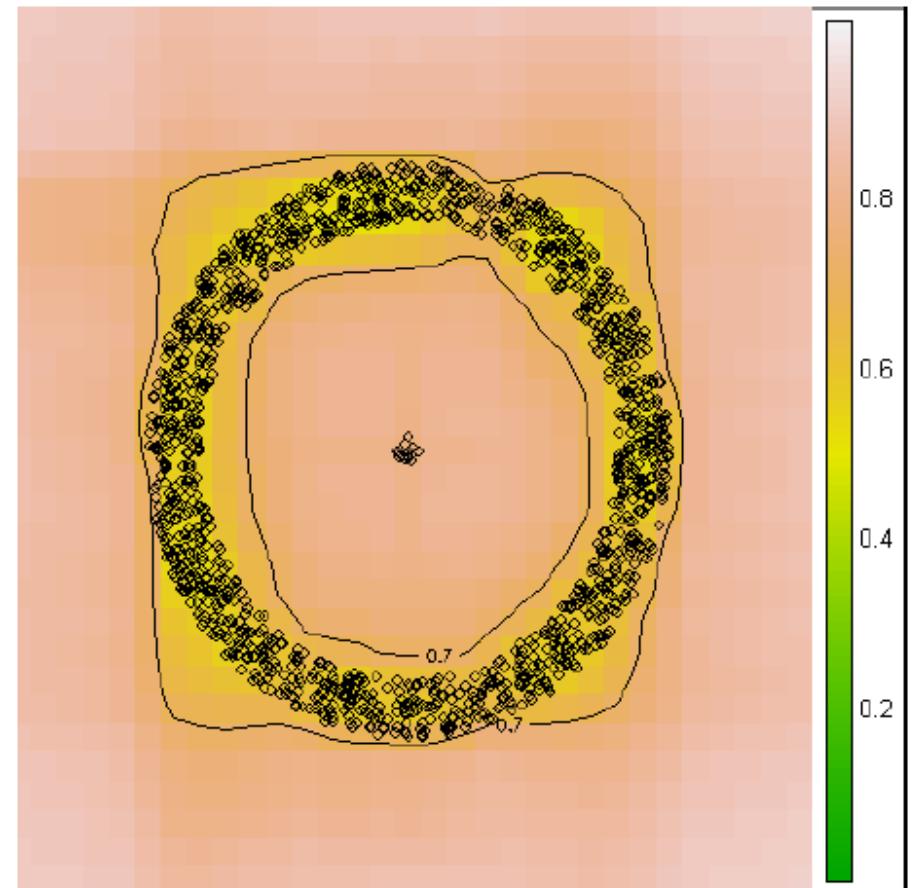
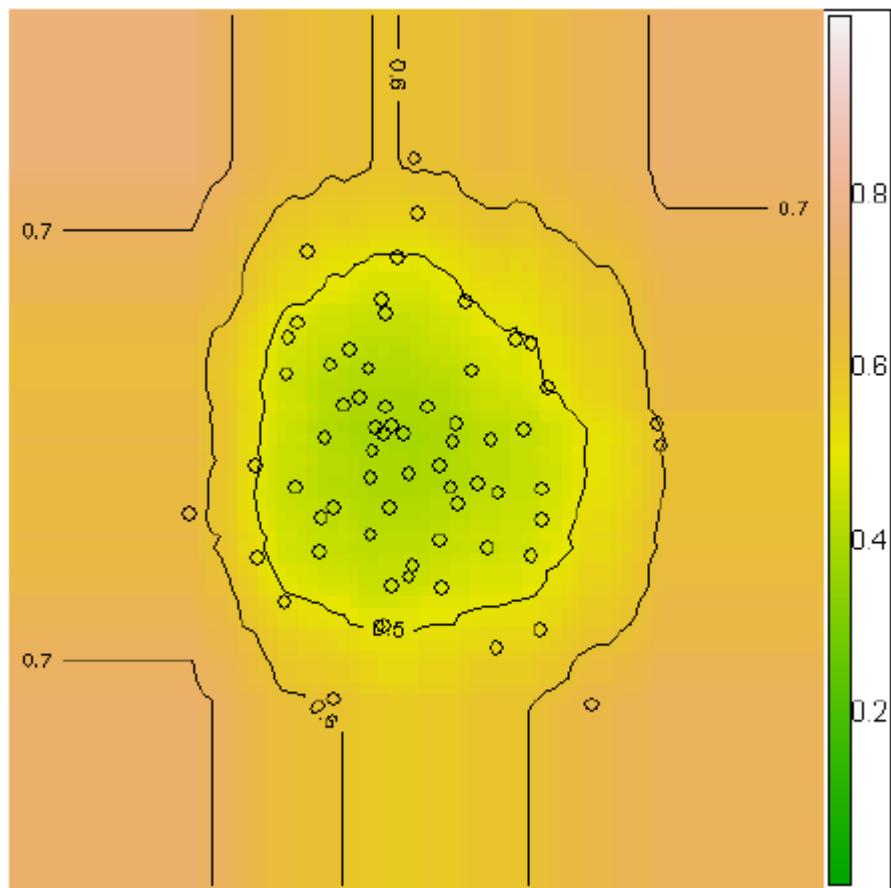
- ✓ 객체 x 의 이상치 스코어 s 는 다음과 같이 정의됨

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}}$$

- When $E(h(x)) \rightarrow c(n)$, $s \rightarrow 0.5$
 - When $E(h(x)) \rightarrow 0$, $s \rightarrow 1$
 - When $E(h(x)) \rightarrow n - 1$, $s \rightarrow 0$
- ✓ 즉, Tree에서 path length가 짧을수록 이상치 스코어는 1에 가까워지고, path length가 길수록 이상치 스코어는 0에 가까워짐

모델 기반 이상치 탐지 기법: IF

- 이상치 점수 등고선 예시



모델 기반 이상치 탐지 기법: IF

- I-Forest 학습

- ✓ 데이터셋을 무작위로 샘플링
- ✓ iTree 구축
- ✓ Path length 계산

Algorithm 1 : $iForest(X, t, \psi)$

Inputs: X - input data, t - number of trees, ψ - subsampling size

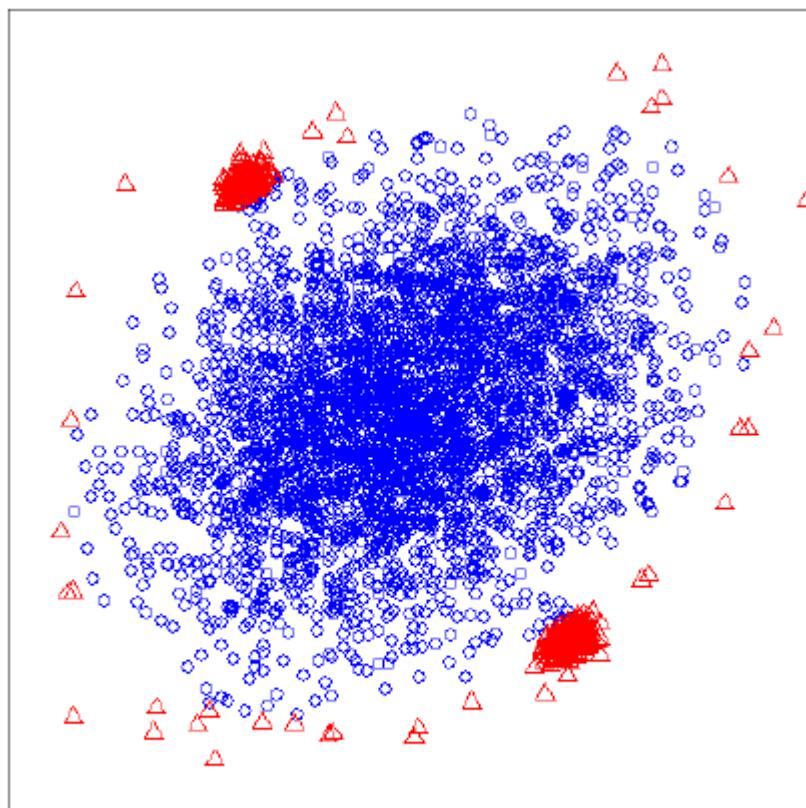
Output: a set of t iTrees

- 1: **Initialize Forest**
 - 2: **for** $i = 1$ to t **do**
 - 3: $X' \leftarrow sample(X, \psi)$
 - 4: $Forest \leftarrow Forest \cup iTree(X')$
 - 5: **end for**
 - 6: **return** $Forest$
-

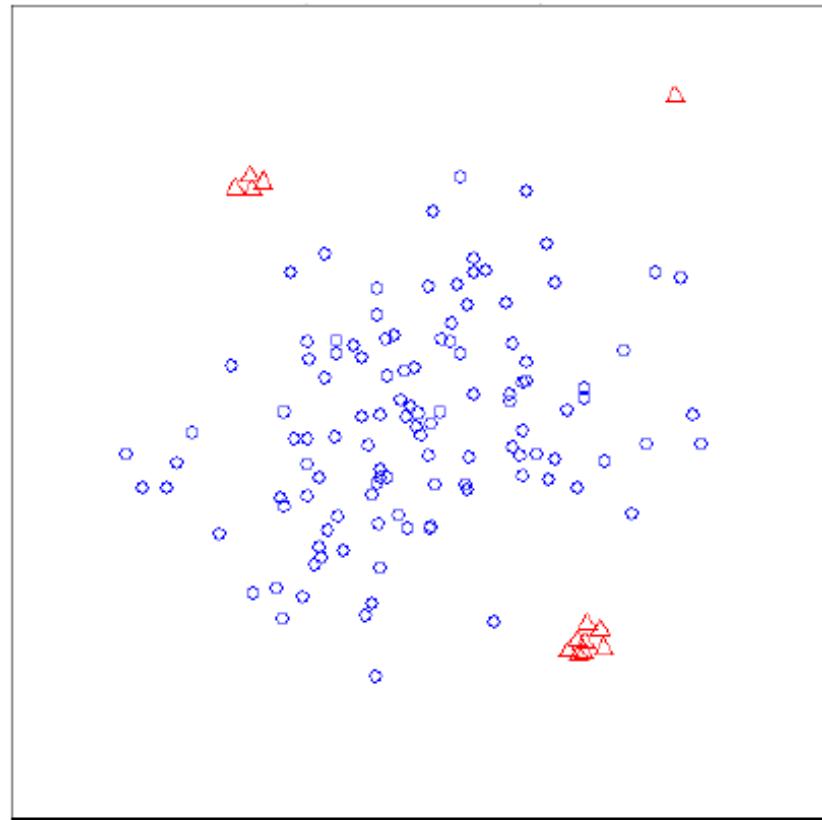
모델 기반 이상치 탐지 기법: IF

- I-Forest 학습

- ✓ 데이터셋을 무작위로 샘플링: iTree 하나 생성에 256개 정도의 샘플 데이터면 충분



(a) Original sample
(4096 instances)



(b) Sub-sample
(128 instances)

모델 기반 이상치 탐지 기법: IF

- I-Forest 학습

- ✓ iTree 구축

Algorithm 2 : $iTree(X')$

Inputs: X' - input data

Output: an $iTree$

```
1: if  $X'$  cannot be divided then
2:   return  $exNode\{Size \leftarrow |X'|\}$ 
3: else
4:   let  $Q$  be a list of attributes in  $X'$ 
5:   randomly select an attribute  $q \in Q$ 
6:   randomly select a split point  $p$  between the  $max$  and  $min$  values of attribute
     $q$  in  $X'$ 
7:    $X_l \leftarrow filter(X', q < p)$ 
8:    $X_r \leftarrow filter(X', q \geq p)$ 
9:   return  $inNode\{Left \leftarrow iTree(X_l),$ 
10:       $Right \leftarrow iTree(X_r),$ 
11:       $SplitAtt \leftarrow q,$ 
12:       $SplitValue \leftarrow p\}$ 
13: end if
```

모델 기반 이상치 탐지 기법: IF

- I-Forest 학습

- ✓ Path length 계산

Algorithm 3 : $\text{PathLength}(x, T, hlim, e)$

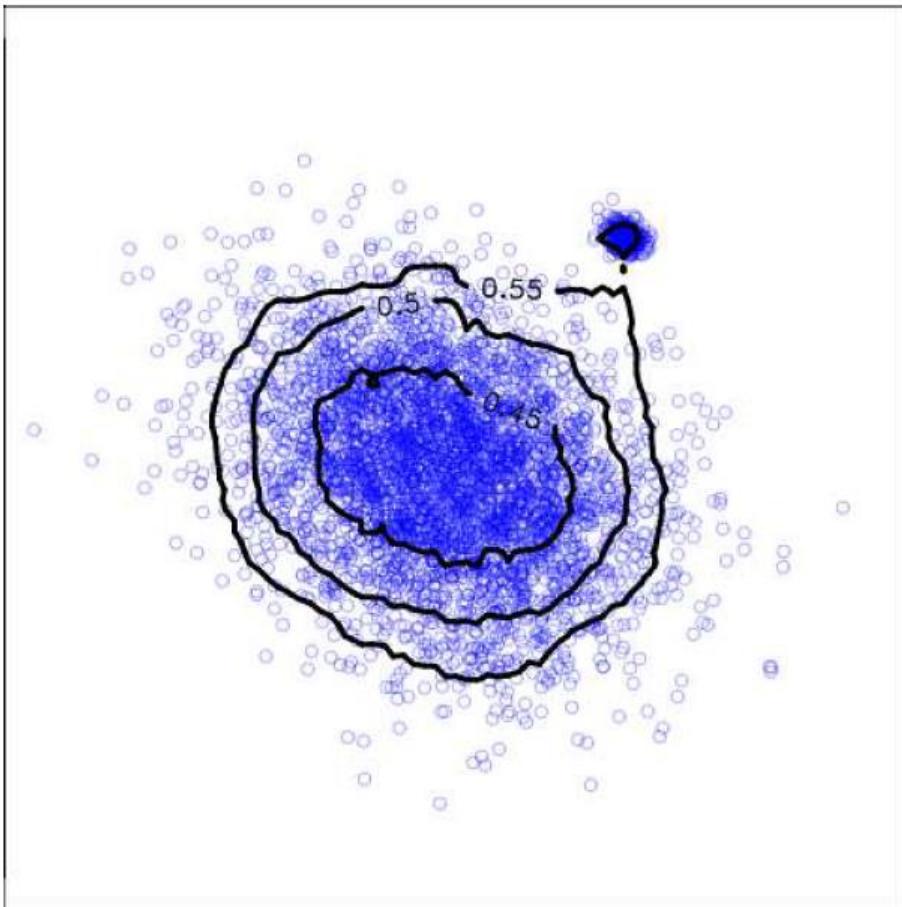
Inputs : x - an instance, T - an i Tree, $hlim$ - height limit, e - current path length;
to be initialized to zero when first called

Output: path length of x

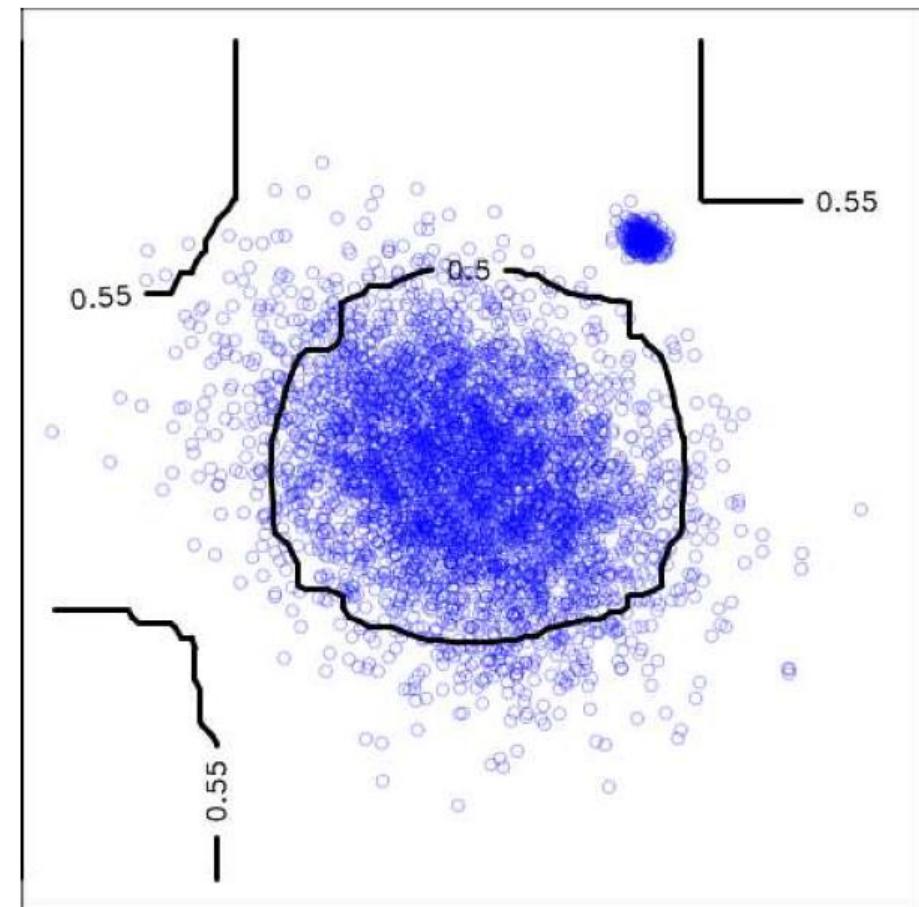
- 1: **if** T is an external node or $e \geq hlim$ **then**
- 2: **return** $e + c(T.\text{size})$ { $c(\cdot)$ is defined in Equation 1}
- 3: **end if**
- 4: $a \leftarrow T.\text{splitAtt}$
- 5: **if** $x_a < T.\text{splitValue}$ **then**
- 6: **return** $\text{PathLength}(x, T.\text{left}, hlim, e + 1)$
- 7: **else** { $x_a \geq T.\text{splitValue}$ }
- 8: **return** $\text{PathLength}(x, T.\text{right}, hlim, e + 1)$
- 9: **end if**

모델 기반 이상치 탐지 기법: IF

- 개별 iTree의 최대 깊이 (height limit)을 다르게 제한할 때의 효과



(a) $hlim = 6$,

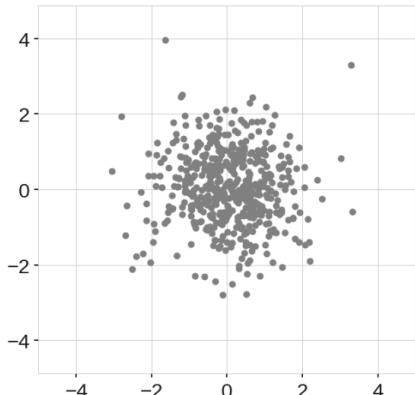


(b) $hlim = 1$,

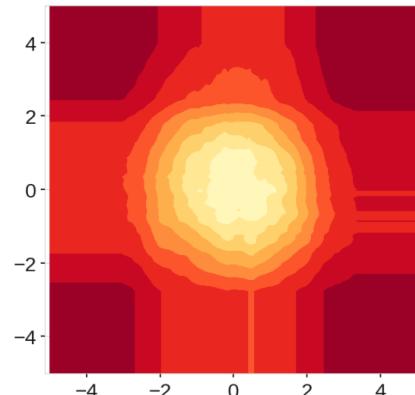
모델 기반 이상치 탐지 기법: Extended IF

Hariri et al. (2018)

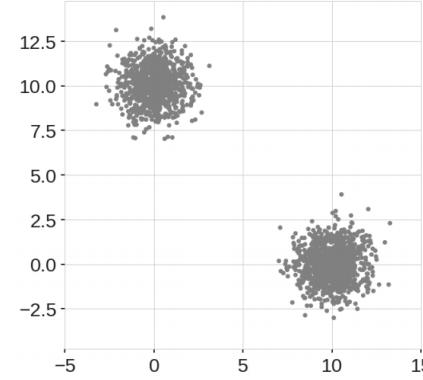
- Motivation



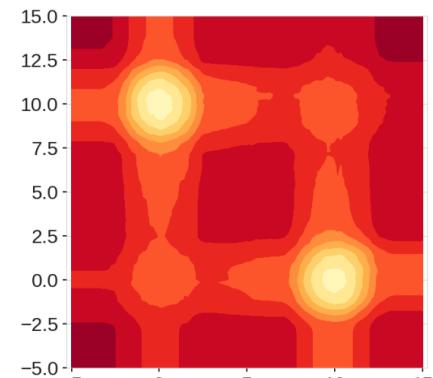
(a) Normally Distributed Data



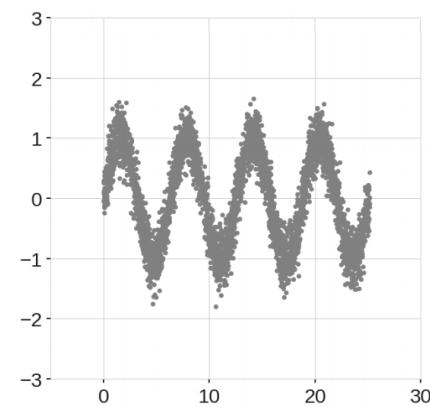
(b) Anomaly Score Map



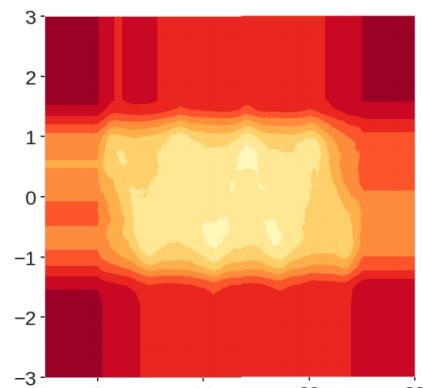
(a) Two normally distributed clusters



(b) Anomaly Score Map



(a) Sinusoidal data points with Gaussian noise.



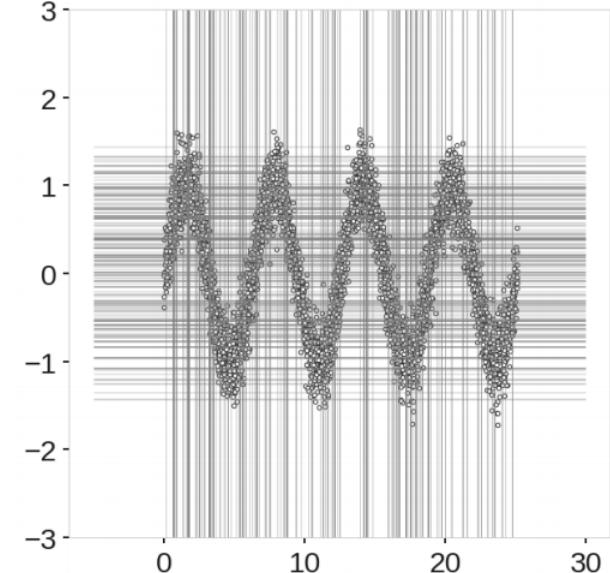
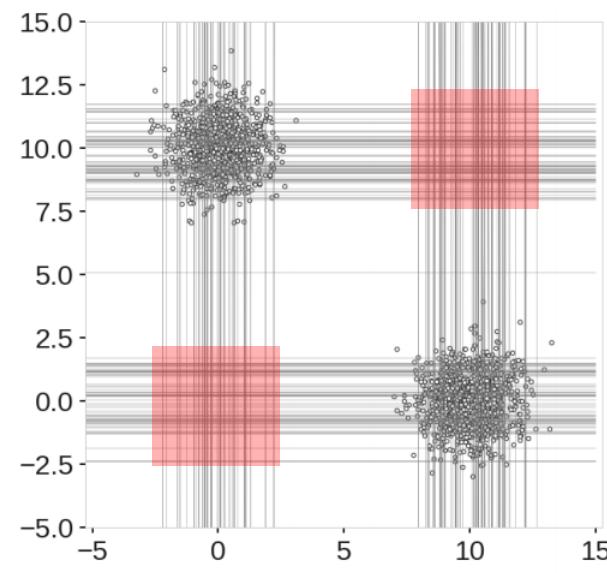
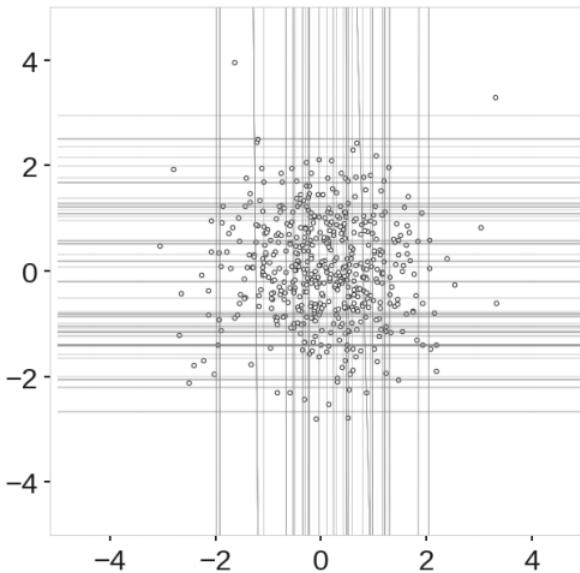
(b) Anomaly Score Map

Standard Isolation Forest
cannot work well for this dataset

모델 기반 이상치 탐지 기법: Extended IF

- Contribution

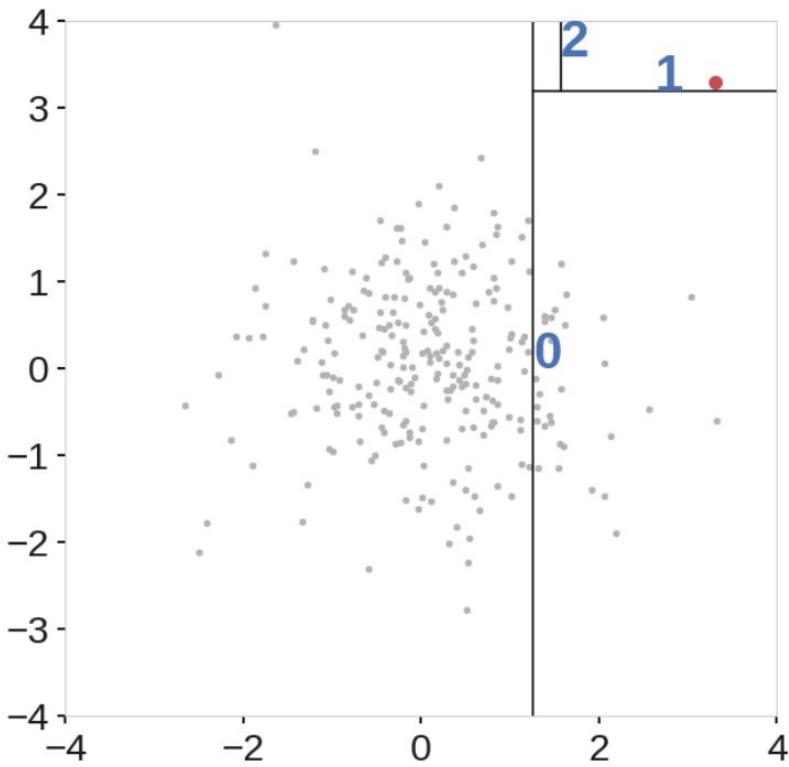
But as we have seen, **the branch cuts are always either horizontal or vertical**, and this **introduces a bias and artifacts in the anomaly score map**. There is no fundamental reason in the algorithm that requires this restriction, and so at each branching point, we can select **a branch cut that has a random “slope”**.



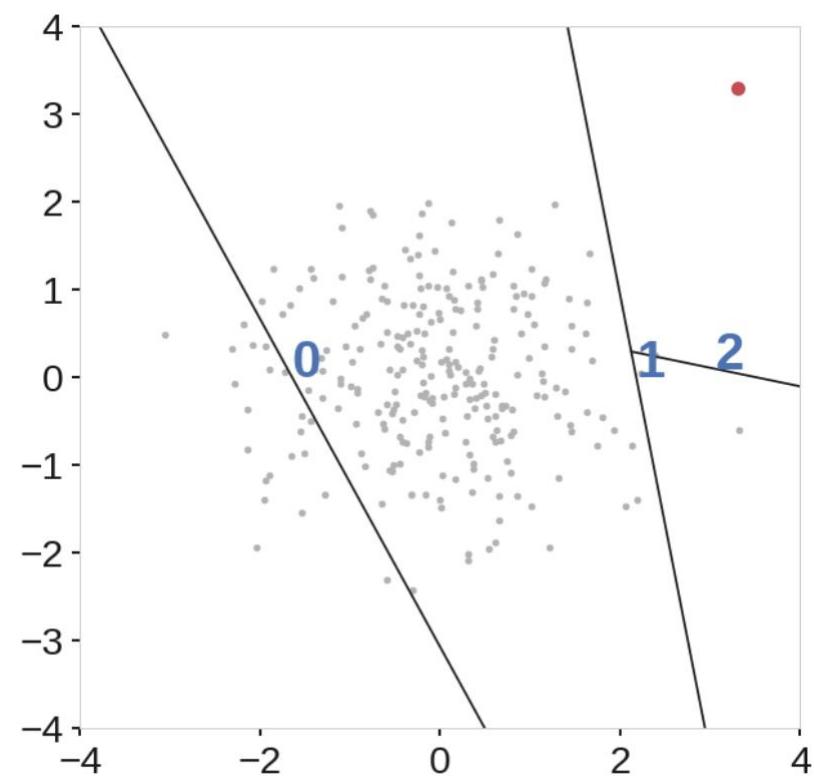
모델 기반 이상치 탐지 기법: Extended IF

- Illustrative example

Standard Isolation Forest

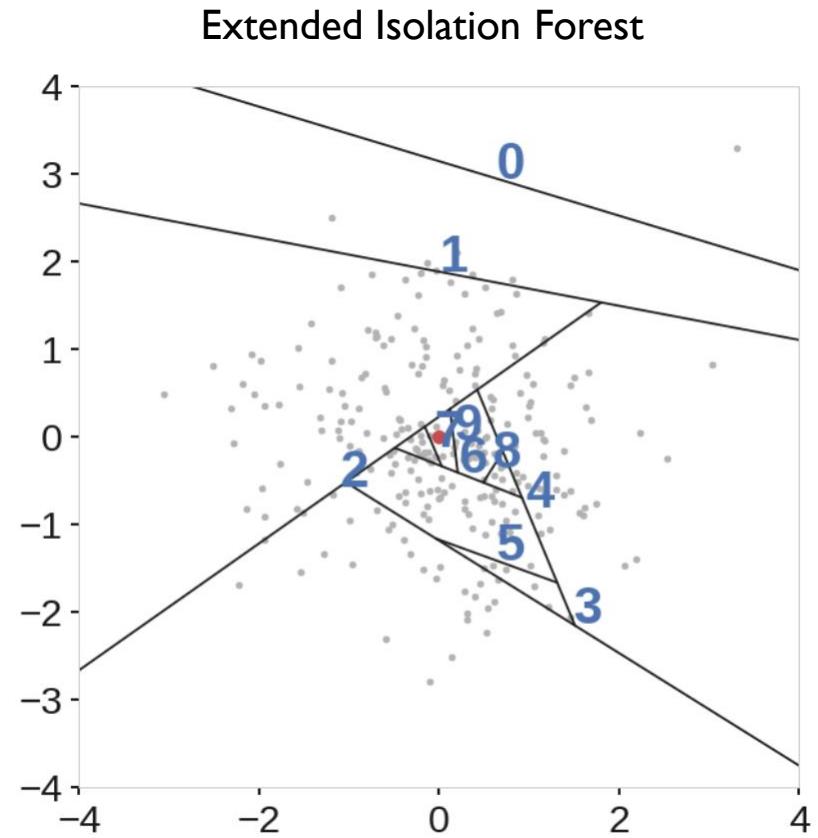
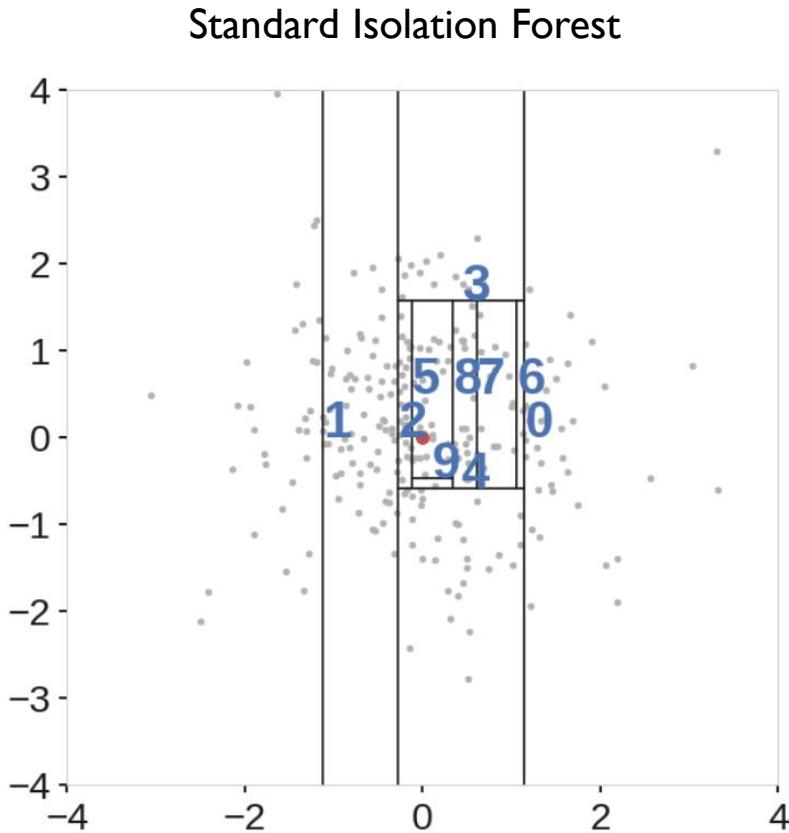


Extended Isolation Forest



모델 기반 이상치 탐지 기법: Extended IF

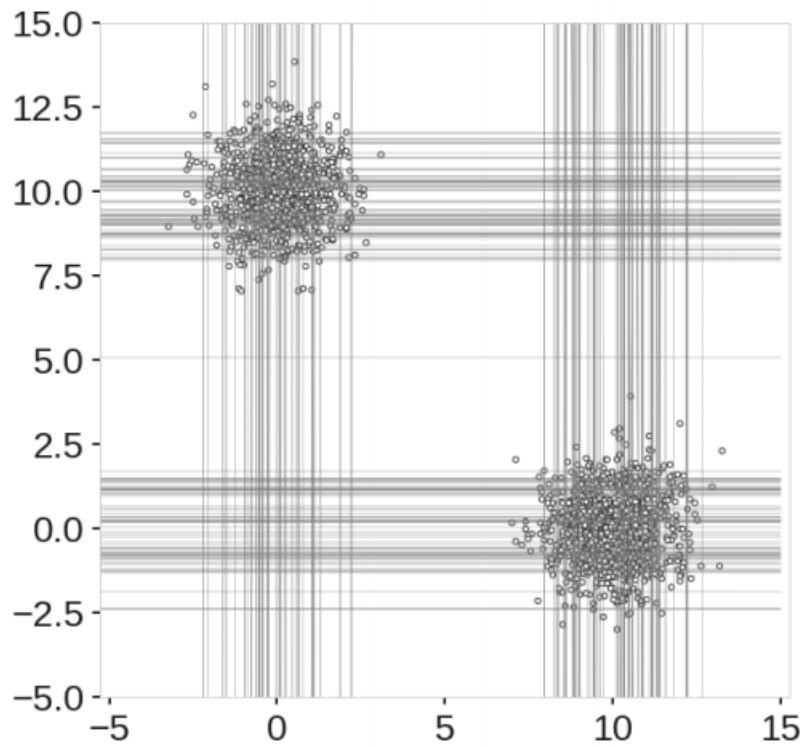
- Illustrative example



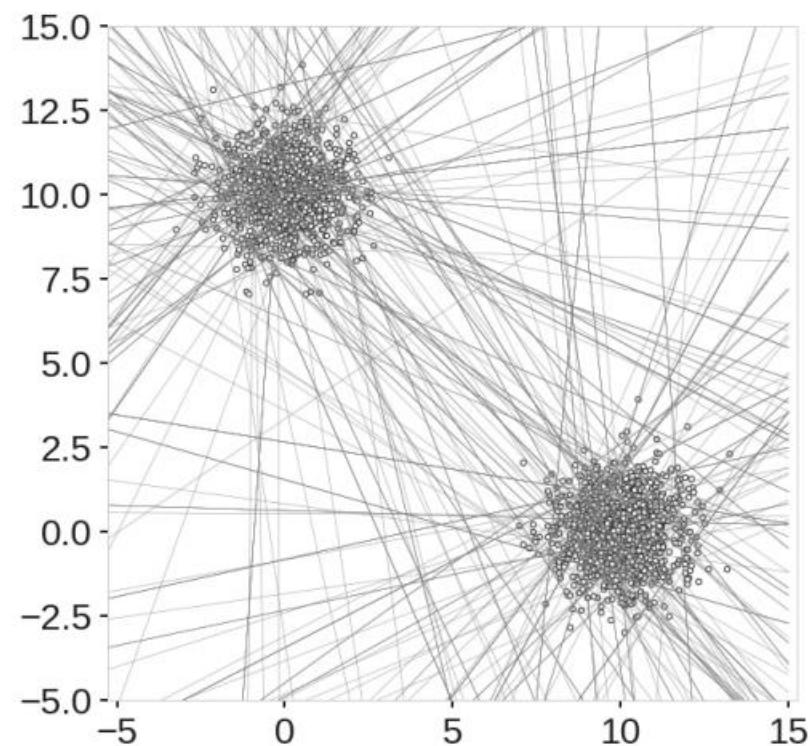
모델 기반 이상치 탐지 기법: Extended IF

- How are the biases reduced?

Standard Isolation Forest



Extended Isolation Forest



모델 기반 이상치 탐지 기법: Extended IF

- Algorithm

Algorithm 2 $iTree(X, e, l)$

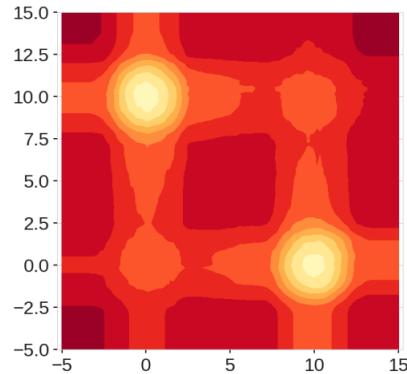
Input: X - input data, e - current tree height, l - height limit

Output: an iTree

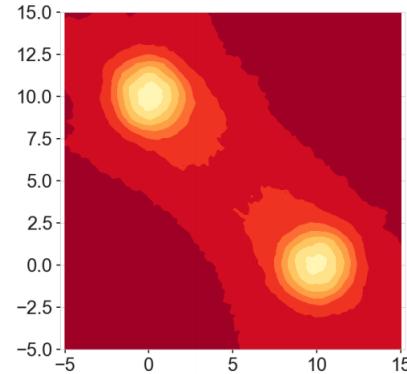
- 1: **if** $e \geq l$ or $|X| \leq 1$ **then**
 - 2: **return** $exNode\{Size \leftarrow |X|\}$
 - 3: **else**
 - 4: randomly select a normal vector $n \in \mathbb{R}^{|X|}$
 by drawing each coordinate of \vec{n} from a uniform
 distribution.
 - 5: randomly select an intercept point $p \in \mathbb{R}^{|X|}$ in
 the range of X
 - 6: set coordinates of n to zero according to exten-
 sion level
 - 7: $X_l \leftarrow filter(X, (X - p) \cdot n \leq 0)$
 - 8: $X_r \leftarrow filter(X, (X - p) \cdot n > 0)$
 - 9: **return** $inNode\{Left \leftarrow iTree(X_l, e + 1, l),$
 $Right \leftarrow iTree(X_r, e + 1, l),$
 $Normal \leftarrow n,$
 $Intercept \leftarrow p\}$
 - 10: **end if**
-

모델 기반 이상치 탐지 기법: Extended IF

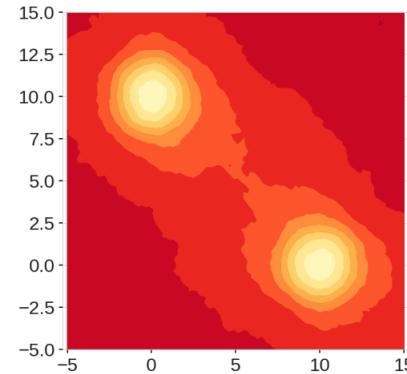
- Anomaly score distribution



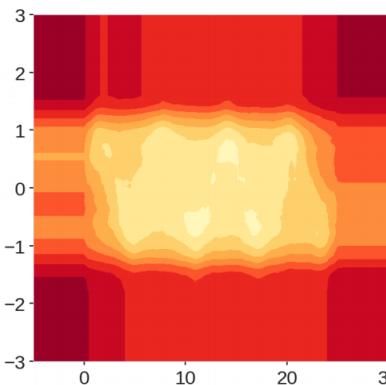
(a) Standard IF



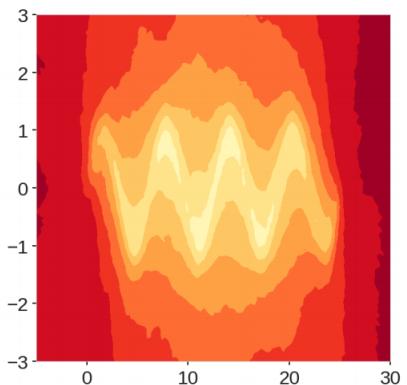
(b) Rotated IF



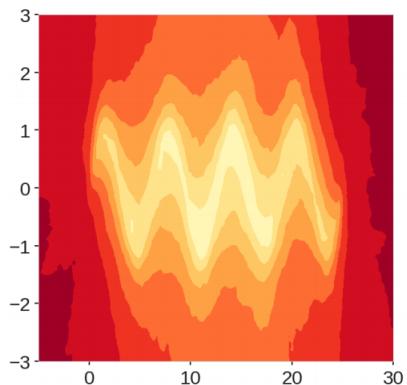
(c) Extended IF



(a) Generic IF



(b) Rotated IF



(c) Extended IF



ANY
questions?