Isolation Forest

Handong Global University
Bigdata Design 24F
21900707 조영우

Cases

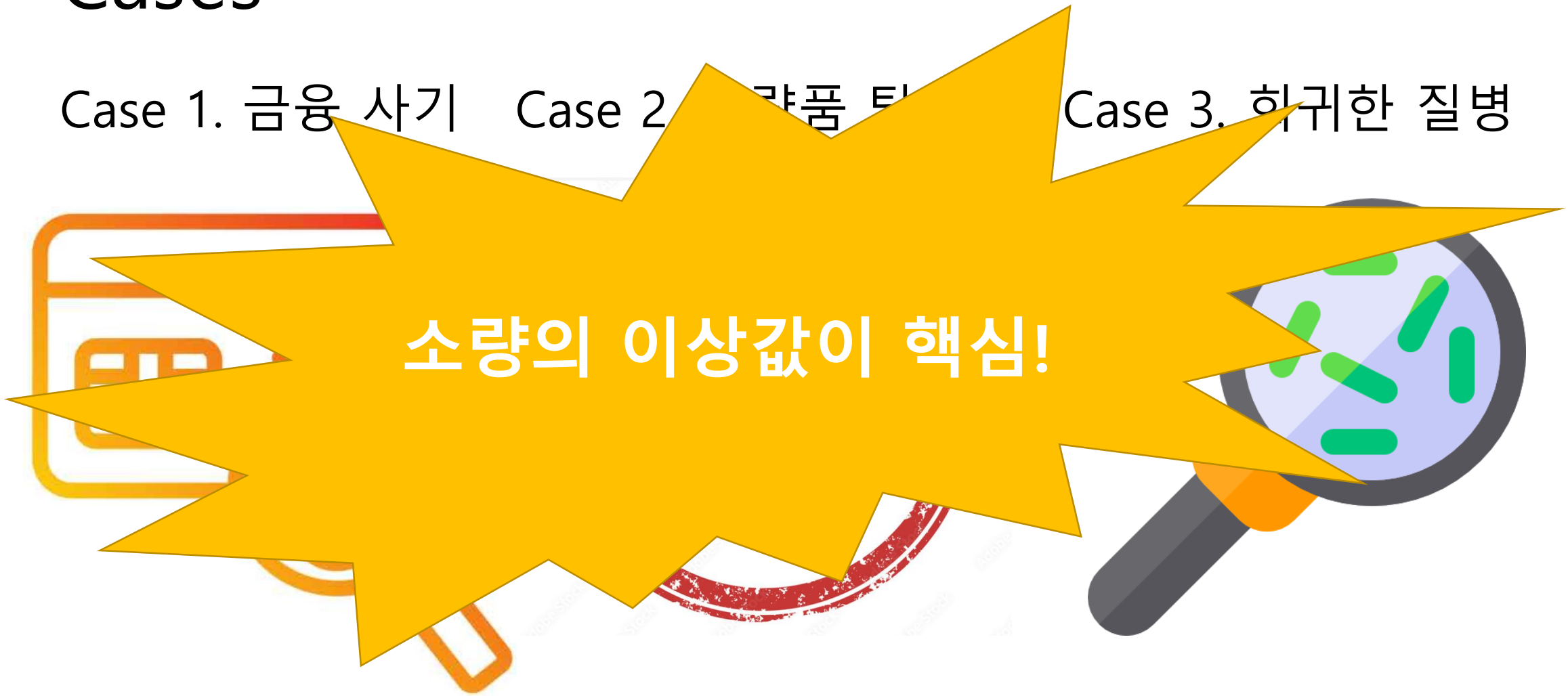
Case 1. 금융 사기

Case 2. 불량품 탐지

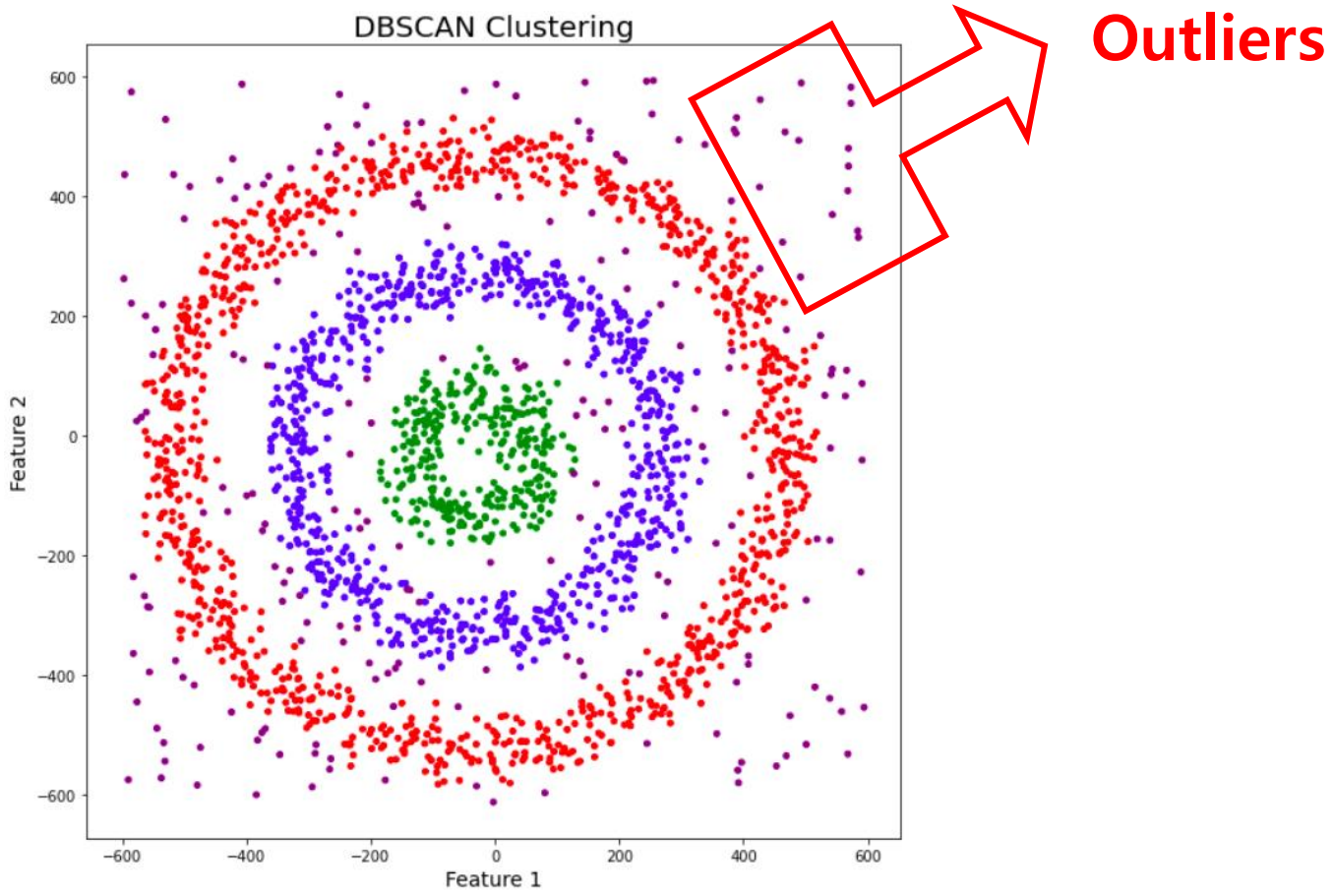
Case 3. 희귀한 질병

Cases

Case 1. 금융 사기　Case 2.　　Case 3. 희귀한 질병

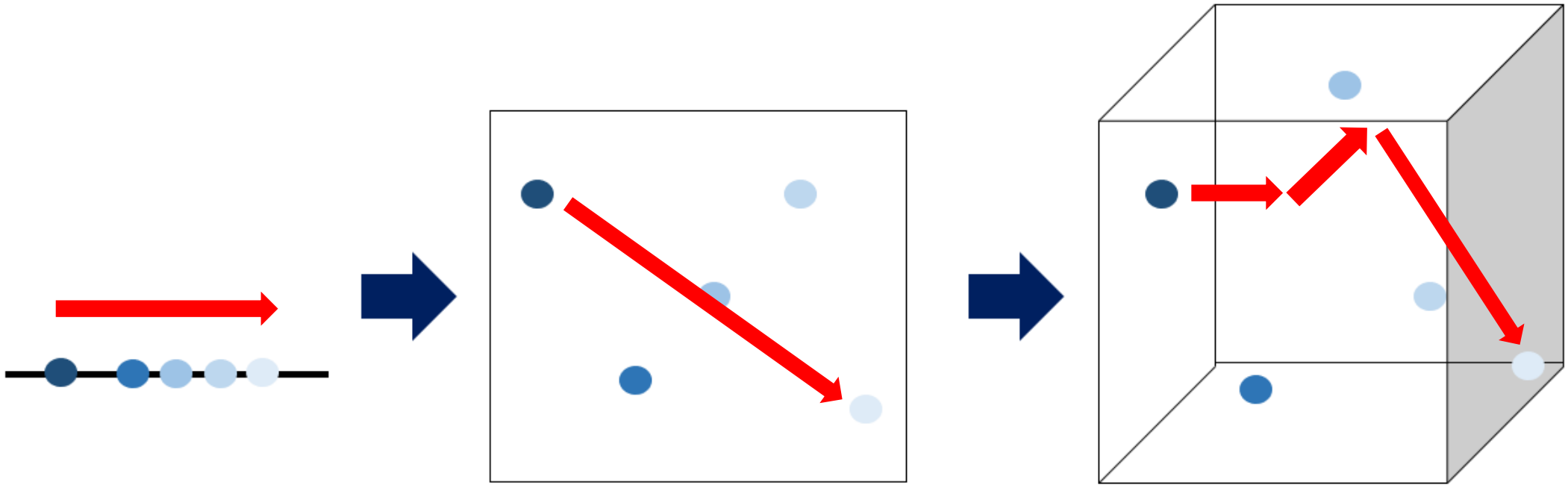


DBSCAN Algorithm



https://miro.medium.com/v2/resize:fit:875/0*PAjsvIpK5dmNXfM0.png

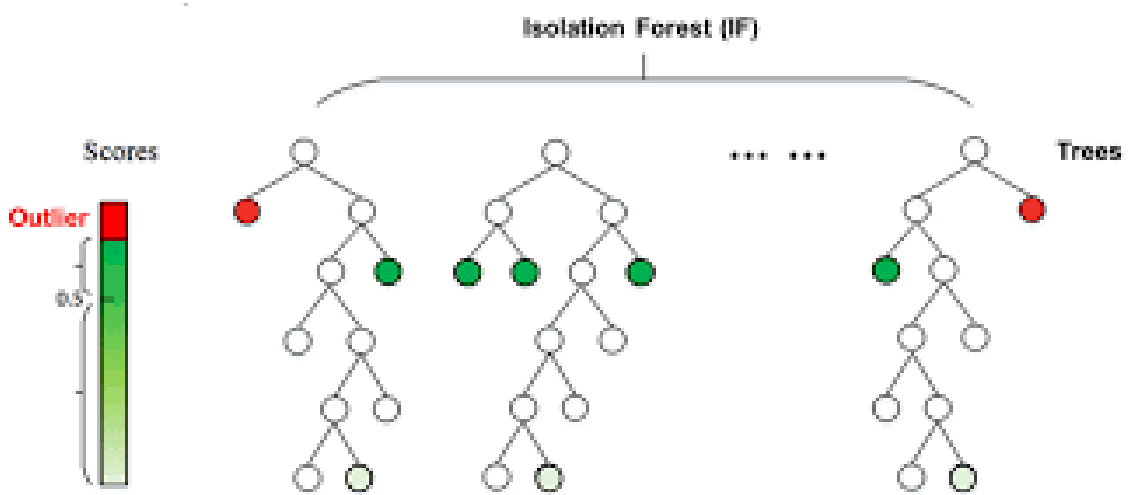
Problem of DBSCAN, Dimensional Curse



https://blog.kakaocdn.net/dn/6g8f5/btqv0yFthYE/NqRQzG0fTTuhYxeCJe865k/img.png

Division, instead of Distance

1. Based on decision tree
2. Add random forest concept
3. Assume that outliers may be detected with a little division



Process

- **Step 1. Select Random data points for training**
  - Prevent masking(When outliers are clustered together and are mistaken for normal)
- **Step 2. Partition the data until each data point become unique**
  - Using random features (difference from decision tree)
- **Step 3. Calculate the number of divisions of each data**
  - It will be used in calculating **anomaly score**

Process

- **Step 4. Create multiple trees randomly**
  - Concept of random forest

- **Step 5. Calculate the anomaly score**

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}} \qquad s(x,n) = \begin{cases} 1 & \text{if } E(h(x)) = 0 \\ 0.5 & \text{if } E(h(x)) = c(n) \\ 0 & \text{if } E(h(x)) = n-1 \end{cases}$$

  - E(h(x)) = (average number of divisions for each data x)
  - C(n) = (average number of division for every data)
  - Example1) E(h(x)) increasing gradually -> $2^{-\infty}$ ≒ 0
  - Example2) E(h(x)) decreasing gradually -> 2^{-0} ≒ 1

Toy Example

- Iris data(150) + Outlier data(10)
  - Add column (Iris = 1, Outlier = -1)

```
IsolationForest(n_estimators=100, max_samples=256, contamination=0.0625, random_state=19)
```

  - n_estimators: number of decision trees
  - max_samples: number of data to use
  - contamination: Detecting outlier rate

Fig1

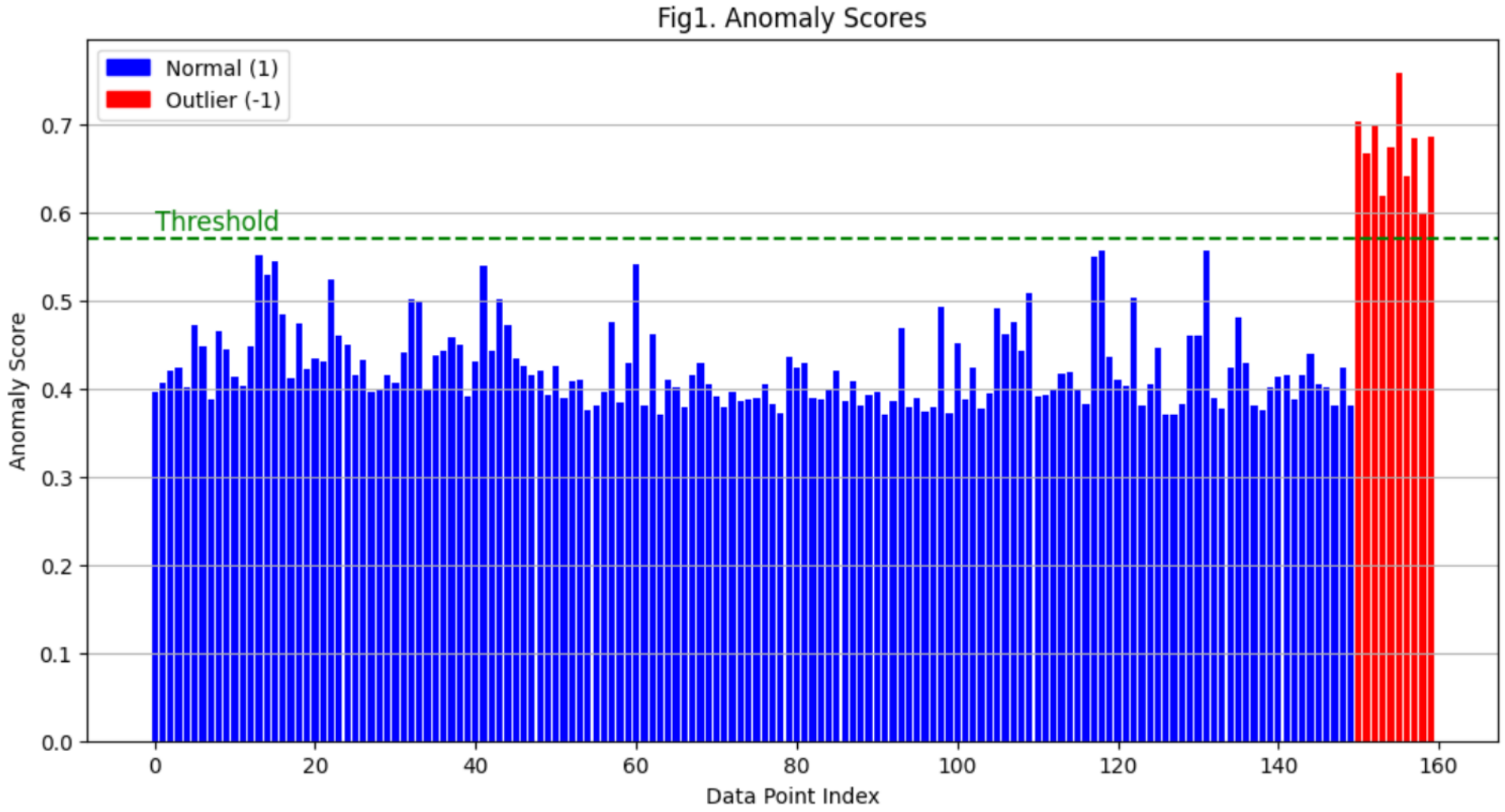


Fig1. Anomaly Scores
Normal (1)
Outlier (-1)
Threshold
Anomaly Score
Data Point Index
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0
20
40
60
80
100
120
140
160

Fig2

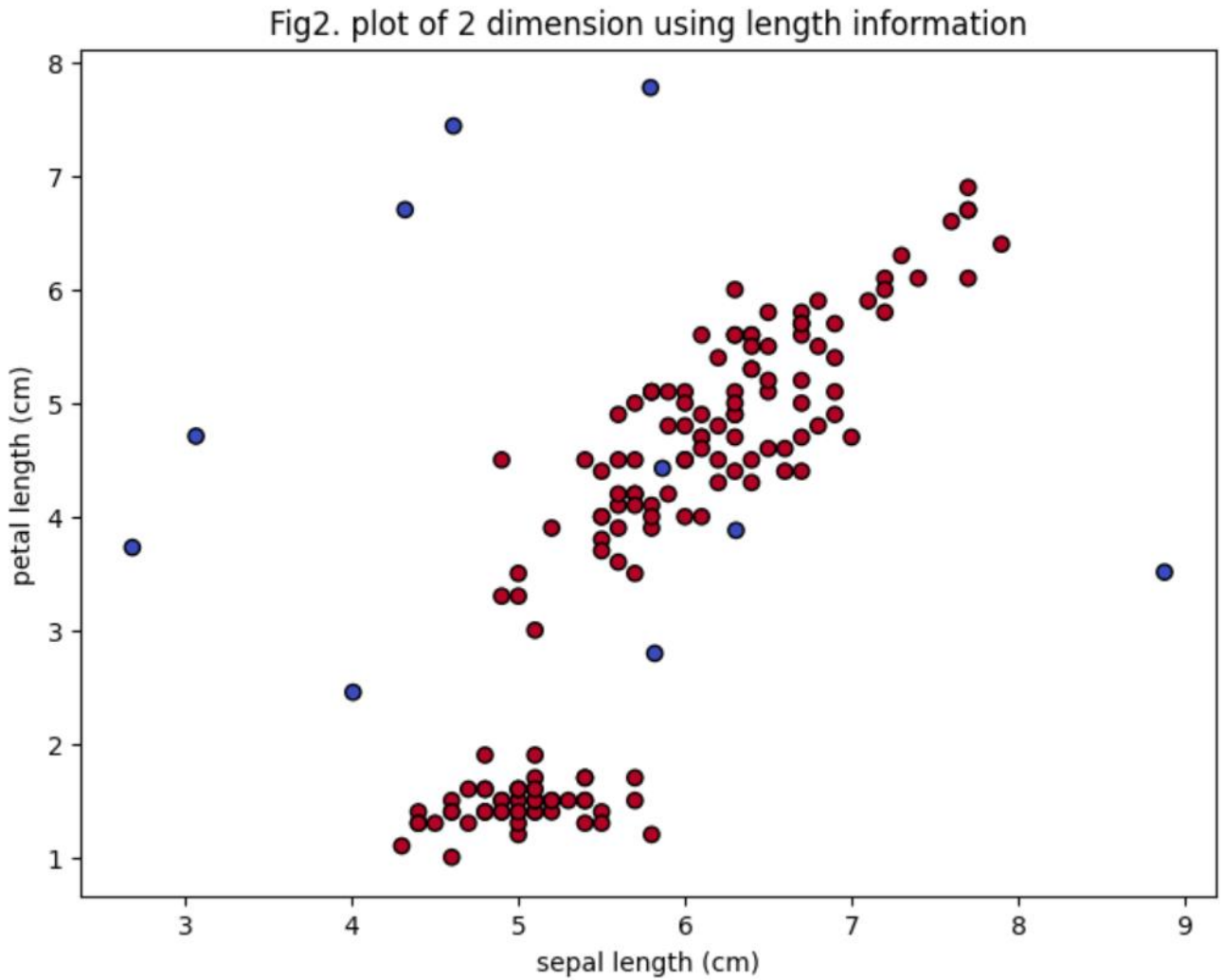


Fig2. plot of 2 dimension using length information
petal length (cm)
sepal length (cm)
1
2
3
4
5
6
7
8
3
4
5
6
7
8
9

Reality Example

- Creditcard data(Normal=170,614 / Outlier=136 / %=0.0797)

```
IsolationForest(n_estimators=200, contamination="auto",
```

```
Time taken to train the model: 0.50seconds

<Evaluation of Isolation Forest>
[[82542  2765]
 [   23   113]]

Accuracy: 0.9674
Precision: 0.0393
Recall: 0.8309
```

```
dbscan = DBSCAN(eps=2, min_samples=3)
```

```
Time taken to train the model: 91.98seconds

<Evaluation of DBSCAN>
[[41052 44255]
 [    8   128]]

Accuracy: 0.4820
Precision: 0.0029
Recall: 0.9412
```

comparison

- Isolation Forest
  - Good in both time and accuracy
  - FP(2,765) > FN(23) -> Find as many fraud cases as possible

- DBSCAN
  - Accuracy 50% means randomly choose
  - Improvement: decrease the number of outlier detection
    -> eps=2 is too short to make large group (curse of dimension)
    -> time, and memory are consumed more

More information..

- E-mail: verysp129@handing.ac.kr
- Github: https://github.com/jayjo9/bigdata_design