Isolation Forest

Hanse Kim

Anomaly Detection

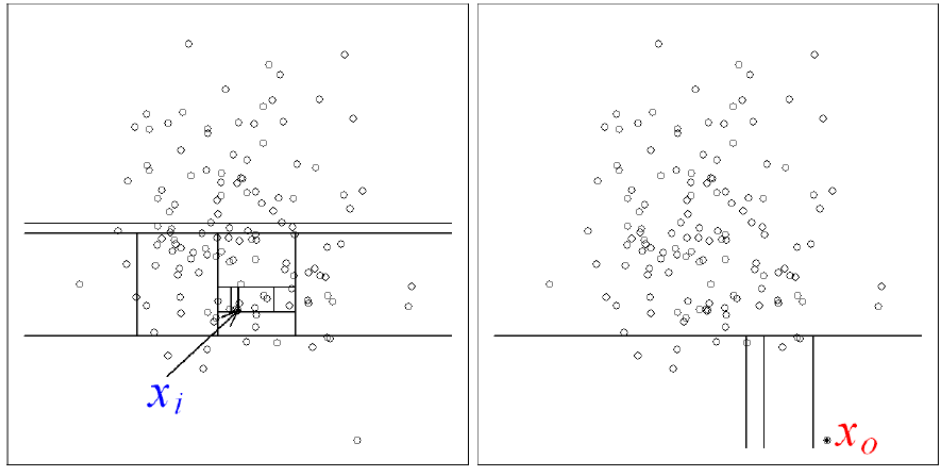
- Finding the ‘anomaly‘ from a group
- Well developed field with numerous examples;

Mostly distance-based, constructs profile of normal examples

Examples : k-NN, LOF

Isolation Forest

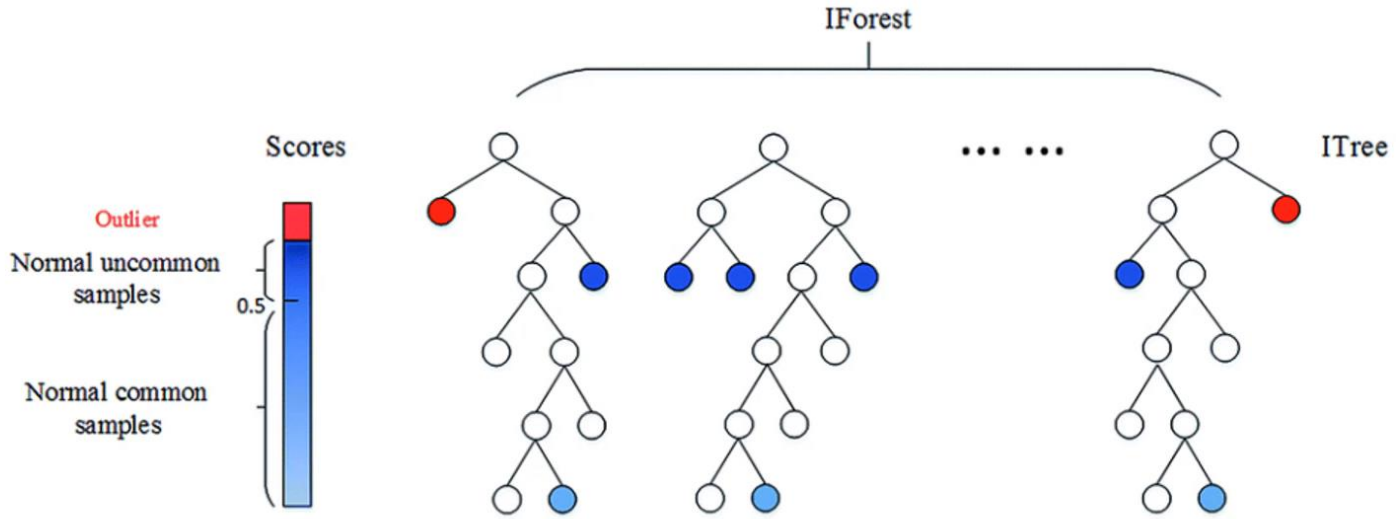
- Isolation forest focuses on explicitly isolating anomalies
- Utilising ‘few and different’ characteristic
- Leads to susceptibility to isolation; recursive decision trees to partition



(a) Isolating x_i (b) Isolating x_o

Isolation Forest

- Isolation forest focuses on explicitly isolating anomalies
- Utilising ‘few and different’ characteristic
- Leads to susceptibility to isolation; recursive decision trees to partition



Training Stage : Isolation Tree

- Binary decision tree
  - External node with no child; representing ‘group’ of data
  - Internal node with one test and two daughter nodes; representing ‘split’ of data
- ‘Split’ defined randomly
- Recursive partition until;
  - ‘Group’ of data to be split has one element
  - Height limit reached

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

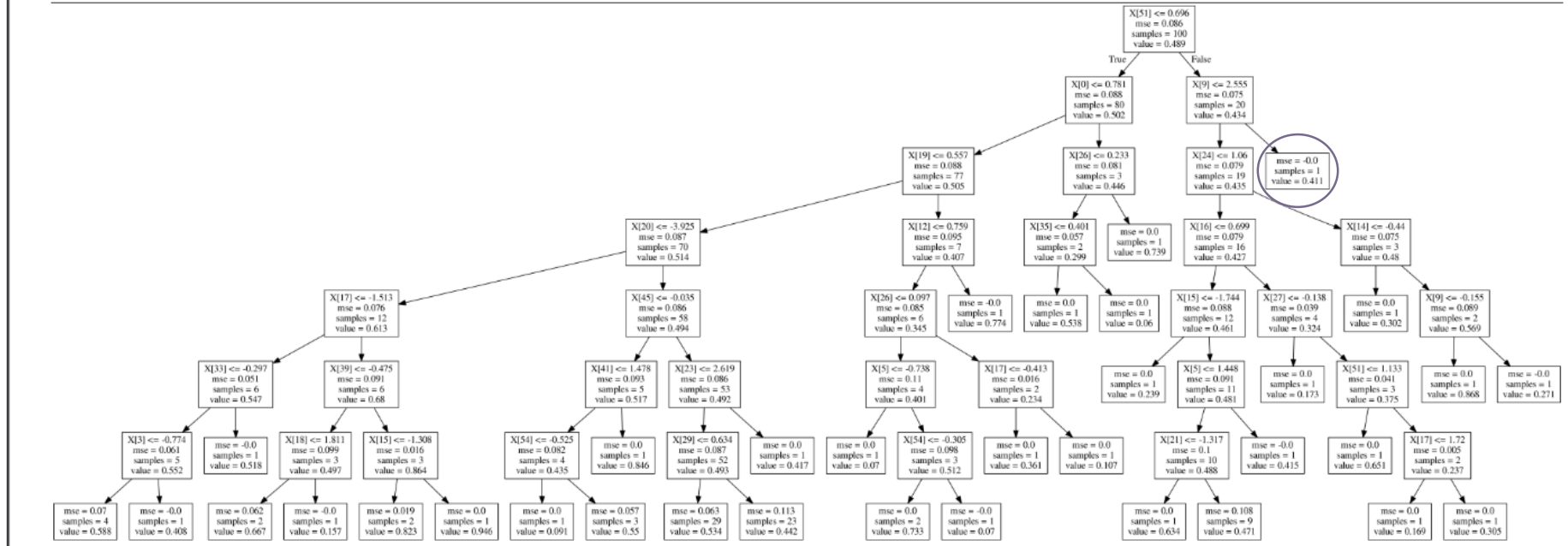
Inputs: X - input data, e - current tree height, l - height limit

Output: an iTree

1: **if** $e \geq l$ or $|X| \leq 1$ **then**
2: return *exNode*$\{Size \leftarrow |X|\}$
3: **else**
4: let Q be a list of attributes in X
5: randomly select an attribute $q \in Q$
6: randomly select a split point p from 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*$\{Left \leftarrow iTree(X_l, e+1, l)$,
10: $Right \leftarrow iTree(X_r, e+1, l)$,
11: $SplitAtt \leftarrow q$,
12: $SplitValue \leftarrow p\}$
13: **end if**

Training Stage : Isolation Tree

- Expect anomalies to have shorter path length



Training Stage : Isolation Forest

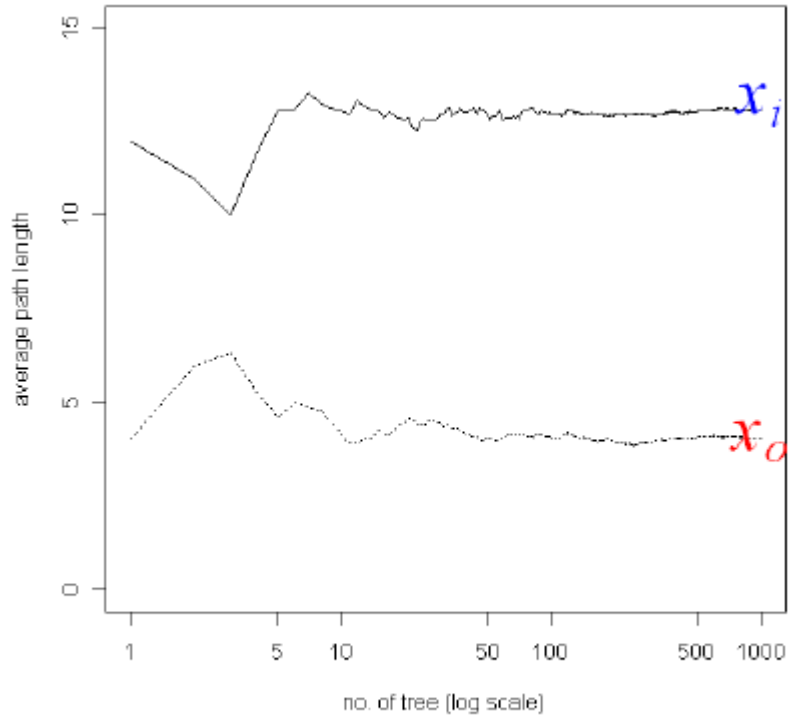
- Ensemble of isolation trees
- Multiple trees acting as ‘experts’ for different anomalies
- Around 50 trees gives converging path length

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

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

Output: a set of t *iTrees*

1: **Initialize** $Forest$
2: set height limit $l = ceiling(\log_2 \psi)$
3: **for** $i = 1$ to t **do**
4: $\quad X' \leftarrow sample(X, \psi)$
5: $\quad Forest \leftarrow Forest \cup iTree(X', 0, l)$
6: **end for**
7: **return** $Forest$



(c) Average path lengths converge

Evaluation Stage : Anomaly Score

- After initial phase of constructing isolation forest using dataset, anomaly score is calculated for each data point individually
- Calculate path length for each tree, expected value over entire forest
- Normalise with average path length of BST; ≈ 2log(n)

Algorithm 3 : $PathLength(x, T, e)$

Inputs : x - an instance, T - an iTree, e - current path length; to be initialized to zero when first called

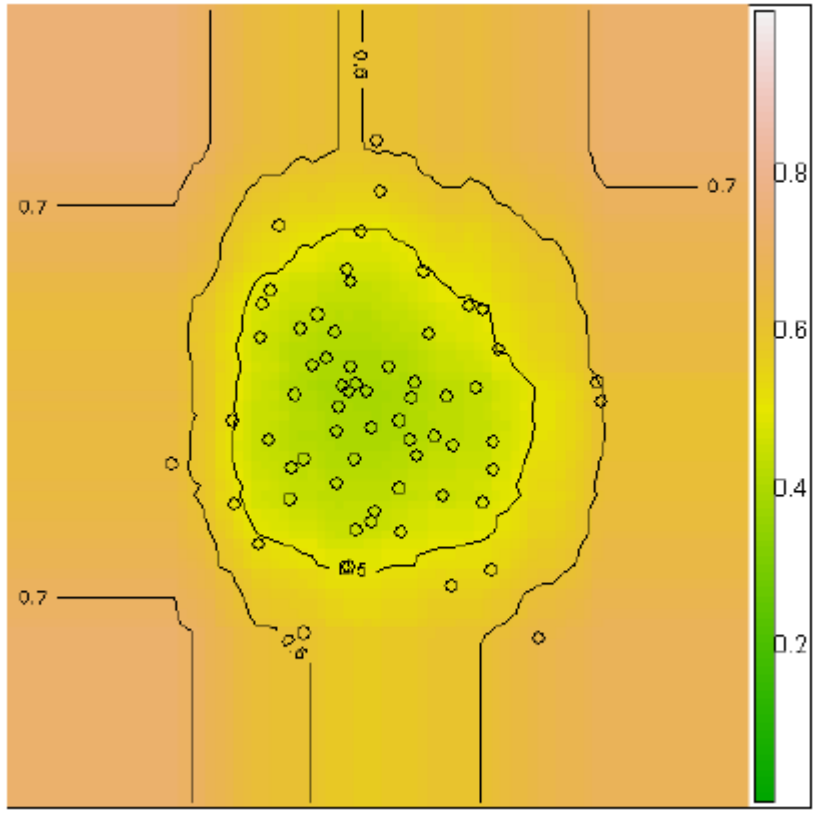
Output: path length of x

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

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

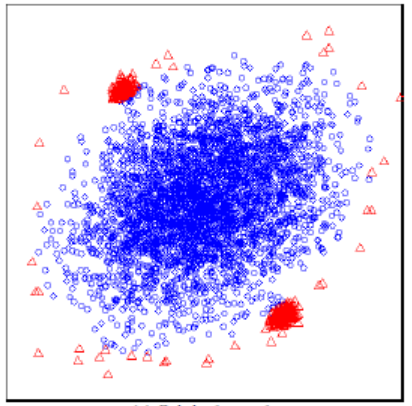
Evaluation Stage : Anomaly Score

- Obtain visual representation by passing through a lattice sample
- Higher score represents greater likelihood to be anomaly

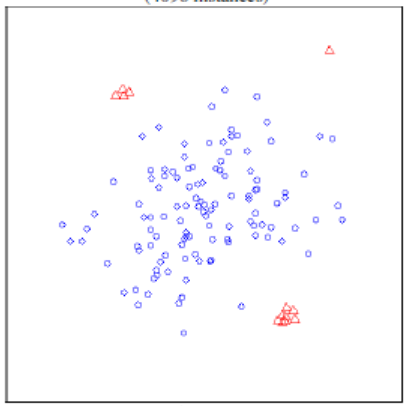


Characteristics of Isolation Forests

- Works best when sampling size is kept small
- Large sampling prevents isolation as normal instances can interfere
- Sub-sampling conducted
- Paper provides data on optimal sub-sample sizes; only a fraction of data required



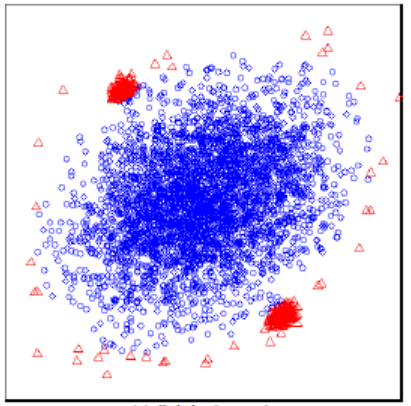
(a) Original sample
(4096 instances)



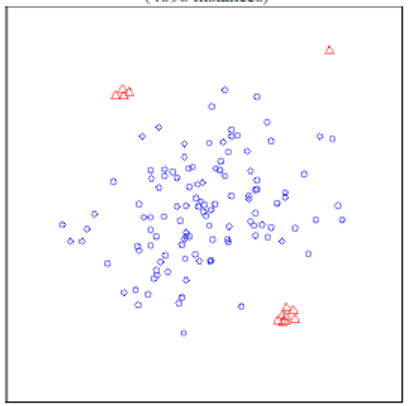
(b) Sub-sample
(128 instances)

Characteristics of Isolation Forests

- Especially prevents problems in anomaly detection
  - Swamping : normal data instances too close to anomalous data being wrongly identified
  - Masking : too many anomalies grouped together appearing as normal
- Lower data size helps isolation
- Each sub-sample includes different sets of anomalies, or none at all



(a) Original sample
(4096 instances)



(b) Sub-sample
(128 instances)

Further Analysis

- Time complexity of $O(t\psi \log \psi)$ for training stage, $O(nt\psi \log \psi)$ for evaluation stage; t number of trees and n number of data points
- Dataset with only normal instances : small reduction in AUC, restorable with larger sub-sampling sizes
- Comparisons to ORCA (a k-NN based method), LOF and Random Forests provided in paper

References

- Paper : Isolation Forest, Liu et al.
- Youtube : 03-7: Anomaly Detection - Isolation Forest (이상치탐지 - Isolation Forest)

https://www.youtube.com/watch?v=puVdwi5PjVA

- Isolation Forest (for anomaly detection)

https://dodonam.tistory.com/129

Thank you for listening