Anomaly Detection in Sensing Data Based on RRCF

- 2020 KSIAM Annual Meeting
- Minjung Gim at NIMS
- Join work with Jin-Hwan Cho, Dong Heon Choe

Innovation Center for
Industrial Mathematics

About me

- Ph.D. in Mathematics(Probability)
- Senior research scientist at NIMS
- Research interests
  – Anomaly detection methods
  – Mathematical data science
- Research Projects
  – 한국연구재단 '4차 산업혁명과 수학, 전략과제'
  – 중소기업기술정보진흥원 '창업성장기술개발사업 혁신형 창업과제'
  (주)타키온테크 수탁 과제

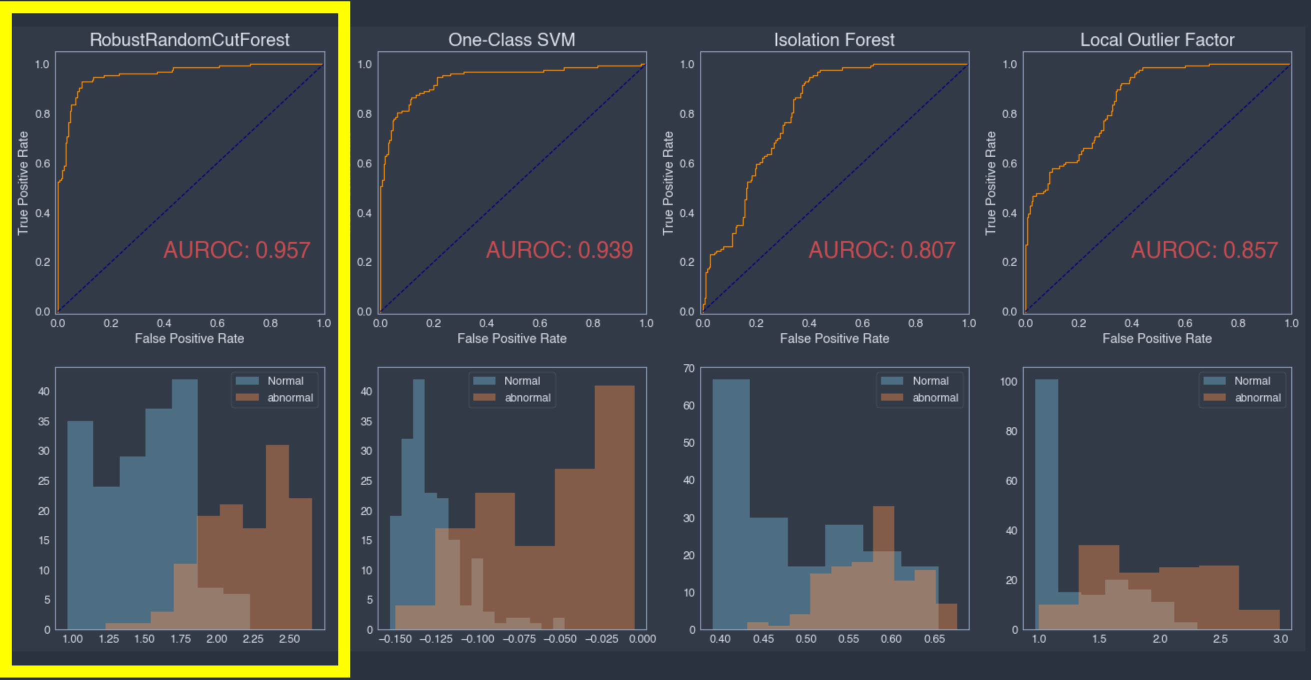
- Tachyon Tech(Inc.)
  – Startup company own technology related to smart factory
  – Solution for detecting defects and abnormal machine status by analyzing manufacturing process data

- (rough) Explanation
  – One class classification in sensing data
  – Mathematical improvement of anomaly detection method

Anomaly Detection

- Task of discerning unusual samples in data
- Identifying unexpected observation or event in data
- Variants of anomaly detection problem
  – Binary classification
  – Highly imbalanced binary classification
  – Outlier detection
  – One class classification(novelty detection)

- (detailed) Explanation
  - Real time anomaly detection solution
  - RRCF is effective method for OCC in sensing data(2019 May)
  - High accuracy and AUROC
  - (Weakness) too slow and too big model size



- Robust Random Cut Forest Based Anomaly Detection On Streams, Proceedings of the 33rd ICML, New York (2016)
  – S. Guha, N. Mishra, G. Roy, O. Schrijvers
- Anomaly detection algorithm for dynamic data streams
- A variant of "Isolation Forest" (2008)
- Built in Amazon SageMaker
- Random tree based and bagging ensemble method

- Generate binary tree called RRCTree
  – RRCF is a collection of RRCTrees

- Calculate Collusive displacement of $\mathbb{x}$, anomaly score of $\mathbb{x}$

- Techniques for Streaming data
  – Insertion and Deletion
  – RRCTree Prob. space

- Definition.
  Robust random cut tree(RRCTree) on point set S is generated as follows:

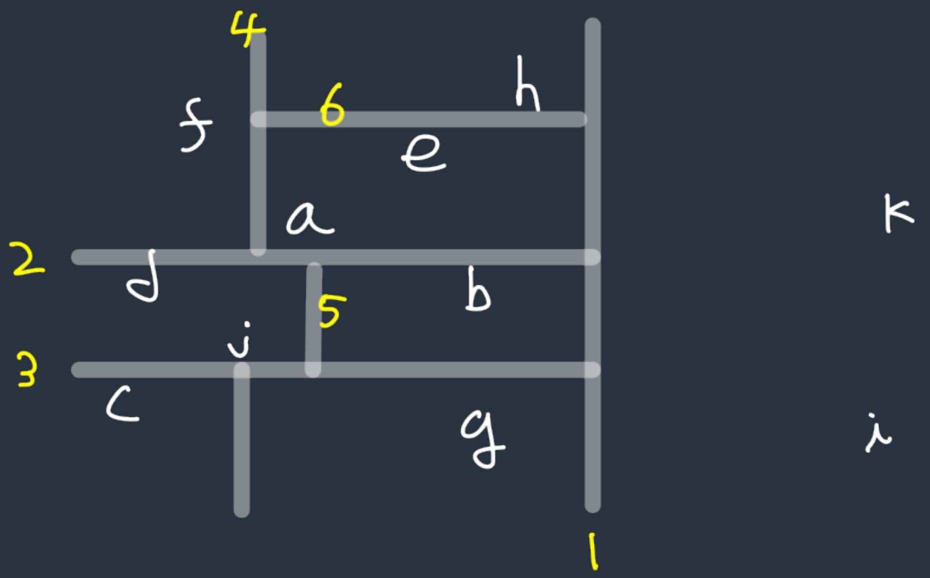
  1. Choose a random dimension proportional to $\frac{\ell_i}{\sum_j \ell_j}$ where
  $\ell_i = max_{\mathbb{x}\in S} x_i - min_{\mathbb{x}\in S} x_i$
  2. Choose $X_i \sim Uniform[min_{\mathbb{x}\in S} x_i, max_{\mathbb{x}\in S} x_i]$
  3. Let $S_1 = \{\mathbb{x} \in S, x_i \leq X_i\}$ and $S_2 = S \setminus S_1$ and recurse on S_1 and S_2

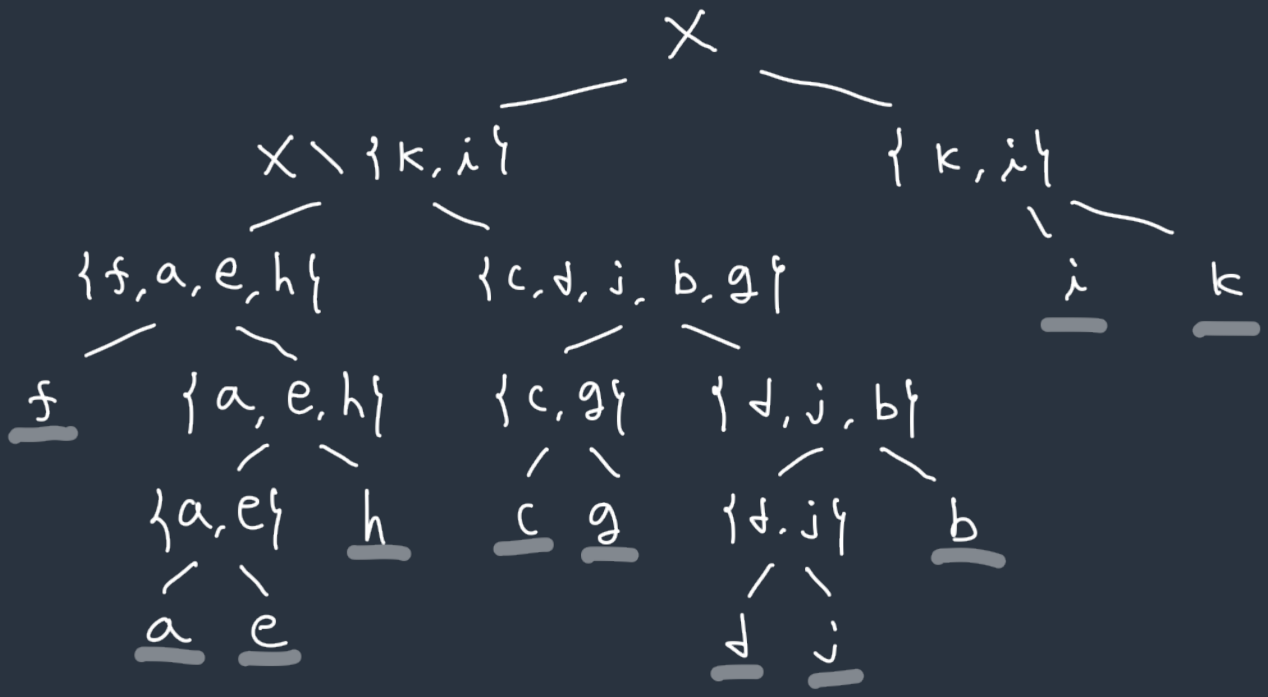
- Remark: Reduce the impact of unrelated features

ICIM Innovation Center for Industrial Mathematics

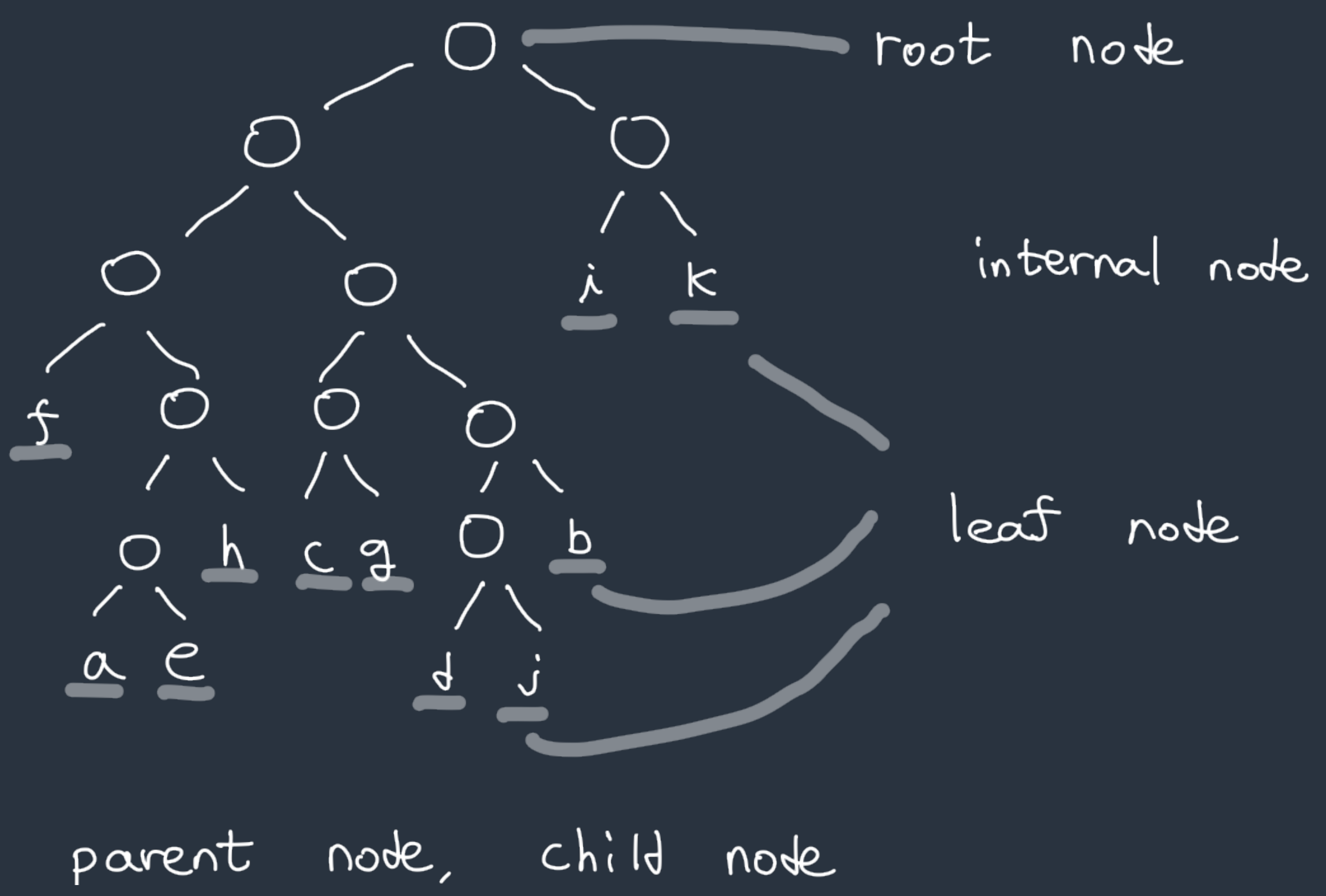
- Example

$X = \{a, b, c, \ldots, k\}$



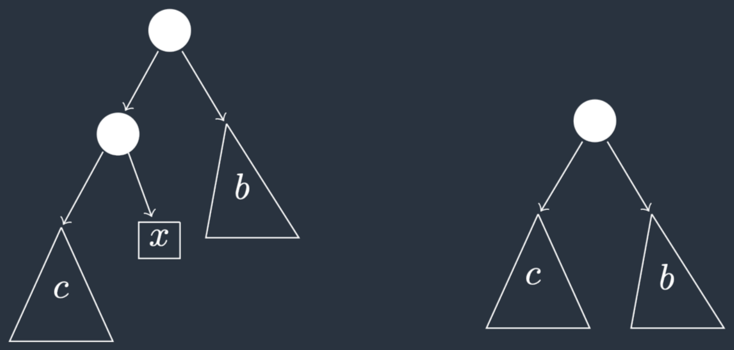


- Example


root node
internal node
leaf node
i
k
f
h
c
g
b
a
e
d
j
parent node, child node

- $DISP(\mathbb{x})$: the change of sum of depths when $\mathbb{x}$ is deleted
  – the number of sibling or their descendants
  – the change in the model complexity of all other points

- $CoDISP(\mathbb{x})$: maximum among $DISP$ of $\mathbb{x}$ and of its neighbors
  – to remove masking effects(colluder)

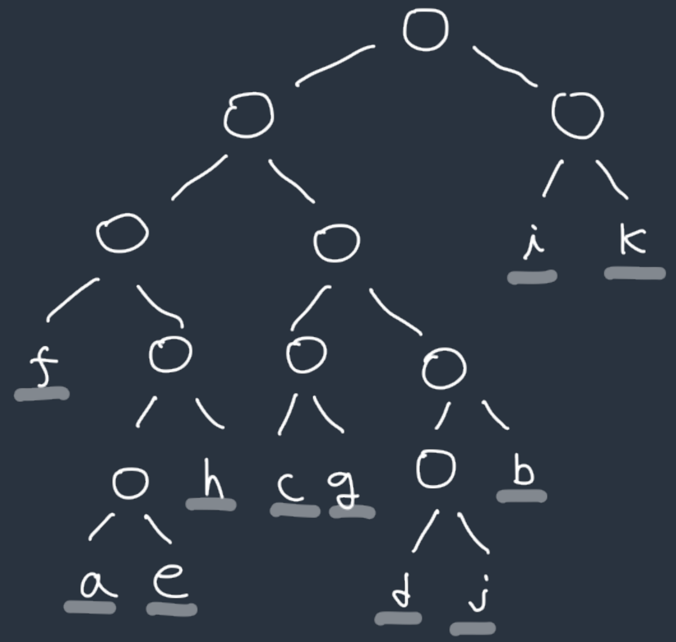


(a) Before: T (b) After: T'

Figure 1. Decremental maintenance of trees.

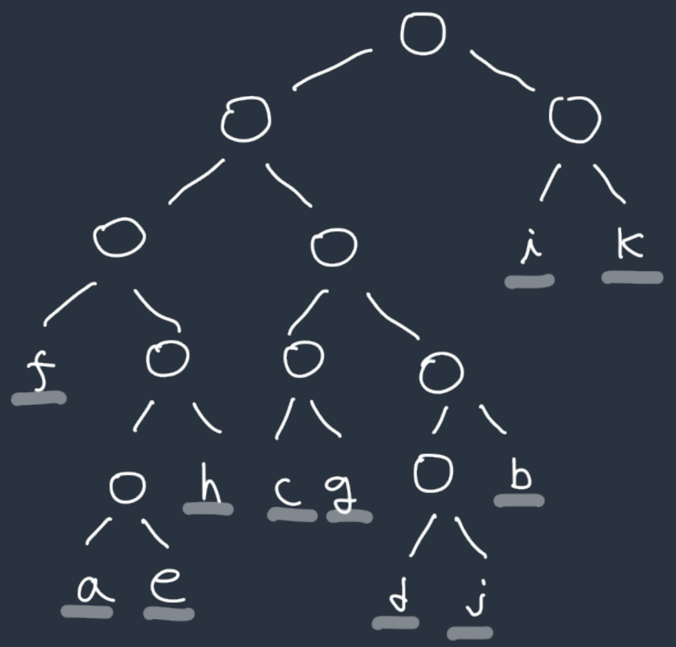
- RRCTree Prob. Space $RRCF(S)$ for a set S, some tree T

- Deletion of a point p in T
  – remove p and its parent node, then $T'(S \setminus p)$ is uniquely determined



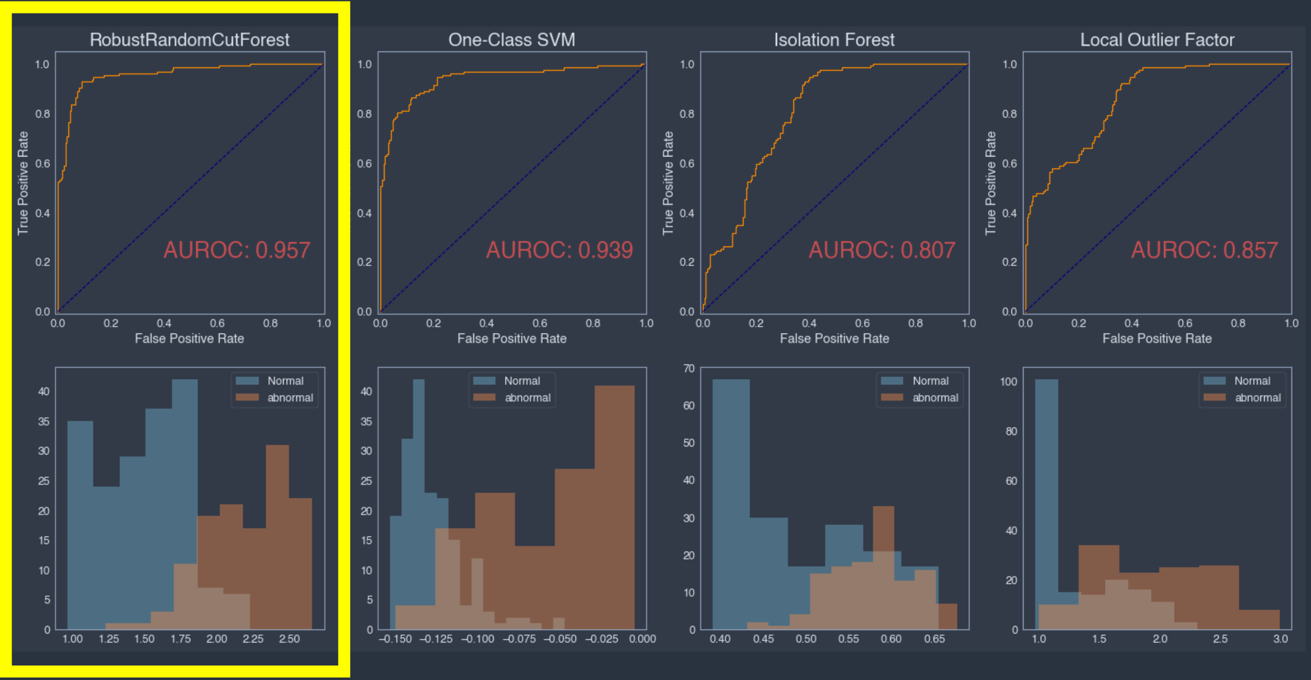
- RRCTree Prob. Space $RRCF(S)$ for a set S, some tree T

- Insertion of a point p, not in T
  - insert p into a tree T and produce tree $T'(S \cup \{p\})$
  - with Insert Point Algorithm
  - tree $T'(S \cup \{p\})$ is not uniquely determined



Industrial Problem

- (detailed) Explanation
  - Real time anomaly detection solution
  - RRCF is effective method for OCC in sensing data(2019 May)
  - High accuracy and AUROC
  - (Weakness) too slow and too big model size



Proposed Technique

- Deletion and Insertion for OCC(2019 May)

- Optimize calculation of $CoDISP$ of a new observation
  – novelty detection setting
  – no insertion and deletion

- Deterministic $CoDISP$(called expected $CoDISP$)
  – $CoDISP$ of RRCF model is not deterministic because of Insert Point Algorithm

- Feature sampling(exploit and explore)
  – randomly sample from all features before generate a tree

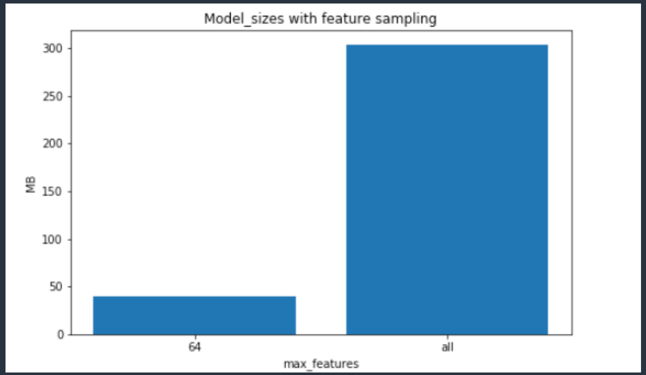
Result

- Test setting
  - Intel® Core™ i7-8750H CPU @ 2.20GHz 6 Cores
  - speed improvement

n_tree	128	256	512
Original RRCF	124.3	248.0	502.5
Our RRCF	16.7	33.4	66.1

(second)

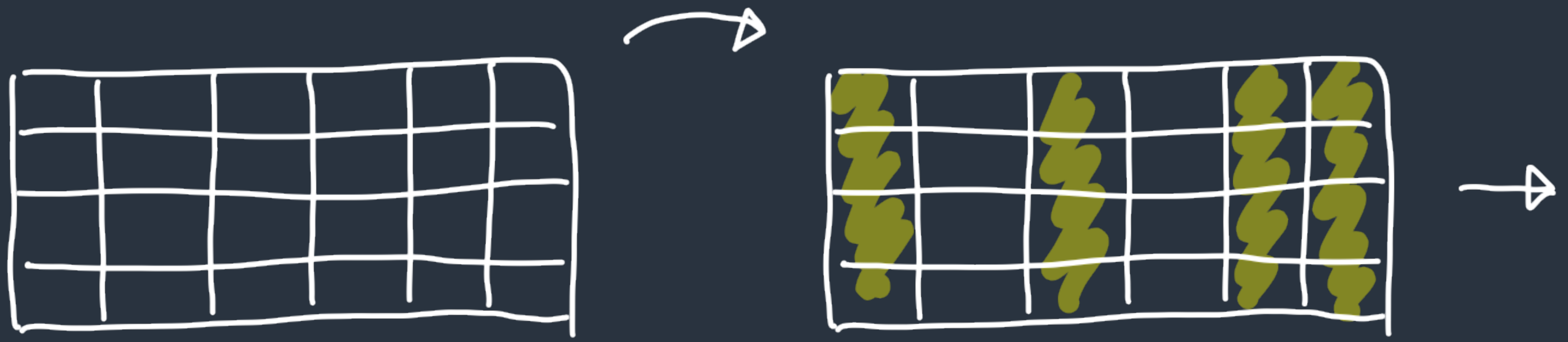
  - size improvement



Feature sampling technique

- Feature sampling(explore all features)
  – randomly sample from all features before generate a tree
  – efficiently detect anomaly in small scale features

feature sampling(randomly)

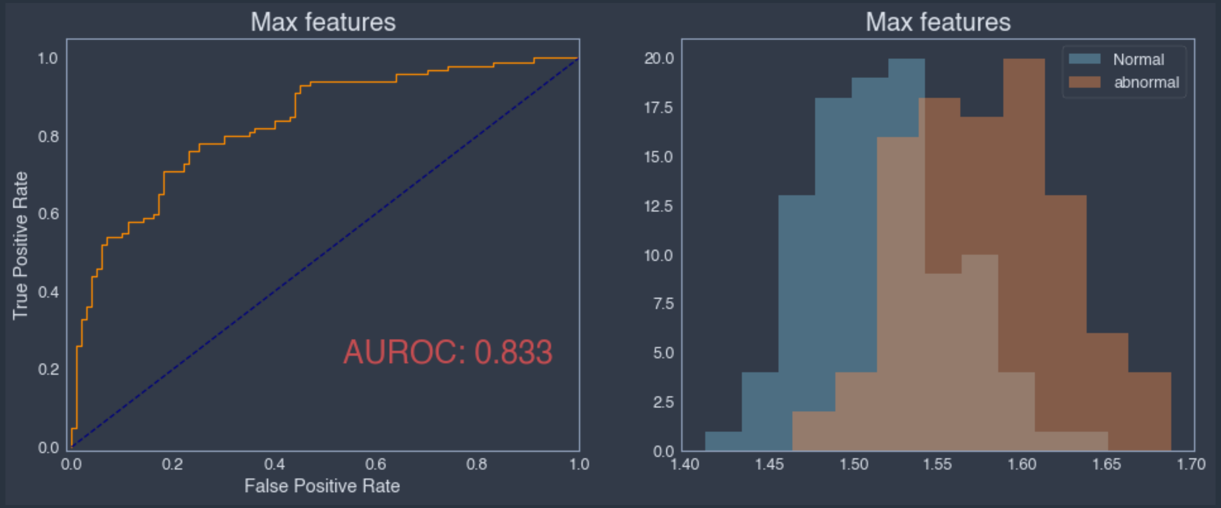


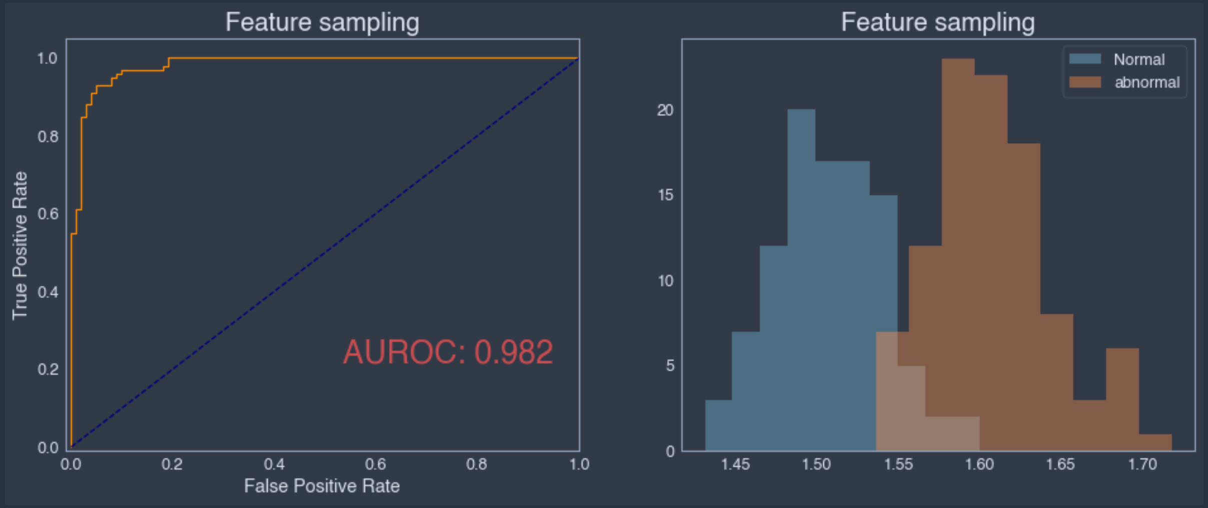
tree generate

- Test
  - train: $(1000, 100)$ distributed in $(U(0,10)\times 95, U(0,1)\times 5)$
  - test: $(200, 100)$ *inliers* and *outliers* $(U(0,10)\times 95, U(1,2)\times 5)$

Feature sampling technique

- Test result(all features vs feature sampling)





- Use all features
  $AUROC: 0.833$

- Sample 4 features
  $AUROC: 0.982$

Future work

- Develop new scoring function(stable and explainable)
- Find new approach using random cut tree
- Analyze and verify random tree based anomaly detection

Thank you

- $DISP(\mathbb{x}, Z)$: the increase in the model complexity of all other points, i.e., for a set Z, to capture the externality introduced by $\mathbb{x}$, define, where $T' = T(Z - \{\mathbb{x}\})$,

$$DISP(\mathbb{x}, Z) = \sum_{T, \mathbb{y} \in Z - \{\mathbb{x}\}} \mathbb{P}[T] \left(f(y, Z, T) - f(y, Z - \{\mathbb{x}\}, T') \right)$$

Innovation Center for Industrial Mathematics

- Insert Point Algorithm

Algorithm 2 Algorithm InsertPoint.

1: We have a set of points S' and a tree $T(S')$. We want to insert p and produce tree $T'(S' \cup \{p\}$.
2: If $S' = \emptyset$ then we return a node containing the single node p.
3: Otherwise S' has a bounding box $B(S') = [x_1^\ell, x_1^h] \times [x_2^\ell, x_2^h] \times \cdots [x_d^\ell, x_d^h]$. Let $x_i^\ell \leq x_i^h$ for all i.
4: For all i let $\hat{x}_i^\ell = \min\{p_i, x_i^\ell\}$ and $\hat{x}_i^h = \max\{x_i^h, p_i\}$.
5: Choose a random number $r \in [0, \sum_i (\hat{x}_i^h - \hat{x}_i^\ell)]$.
6: This r corresponds to a specific choice of a cut in the construction of $RRCF(S' \cup \{p\})$. For instance we can compute $\arg\min\{j | \sum_{i=1}^{j} (\hat{x}_i^h - \hat{x}_i^\ell) \geq r\}$ and the cut corresponds to choosing $\hat{x}_j^\ell + \sum_{i=1}^{j} (\hat{x}_i^h - \hat{x}_i^\ell) - r$ in dimension j.
7: If this cut separates S' and p (i.e., is not in the interval $[x_j^\ell, x_j^h]$) then and we can use this as the first cut for $T'(S' \cup \{p\})$. We create a node – one side of the cut is p and the other side of the node is the tree $T(S')$.
8: If this cut does not separate S' and p then we throw away the cut! We choose the exact same dimension as $T(S')$ in $T'(S' \cup \{p\})$ and the exact same value of the cut chosen by $T(S')$ and perform the split. The point p goes to one of the sides, say with subset S''. We repeat this procedure with a smaller bounding box $B(S'')$ of S''. For the other side we use the same subtree as in $T(S')$.
9: In either case we update the bounding box of T'.