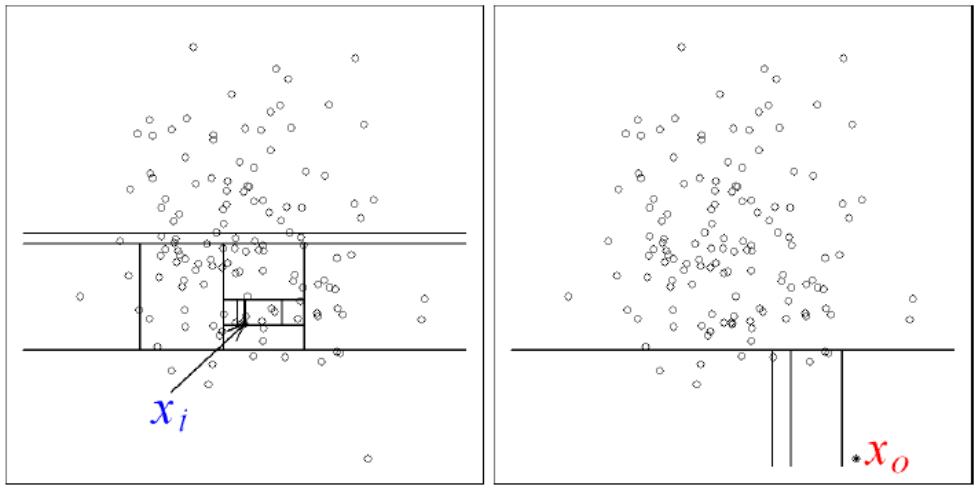
Anomaly Detection:
Isolation Forest and Its Variations

Pilsung Kang
School of Industrial Management Engineering
Korea University

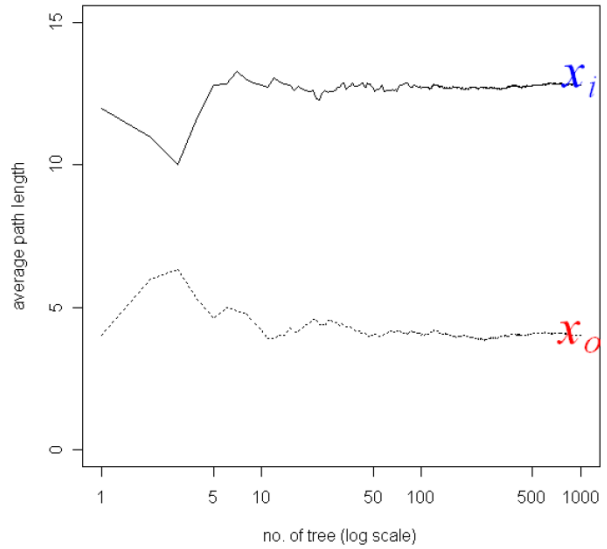
Isolation Forest

Liu et al. (2008, 2012)

- Motivation: Few and Different
  - ✓ The minority consists of fewer instances
  - ✓ They have attribute-values, which are very different from those of normal instances
- A tree structure can be constructed effectively to isolate every single instances
  - ✓ Novel instances are isolated closer to the root of the tree
  - ✓ Normal instances are isolated at the deeper end of the tree

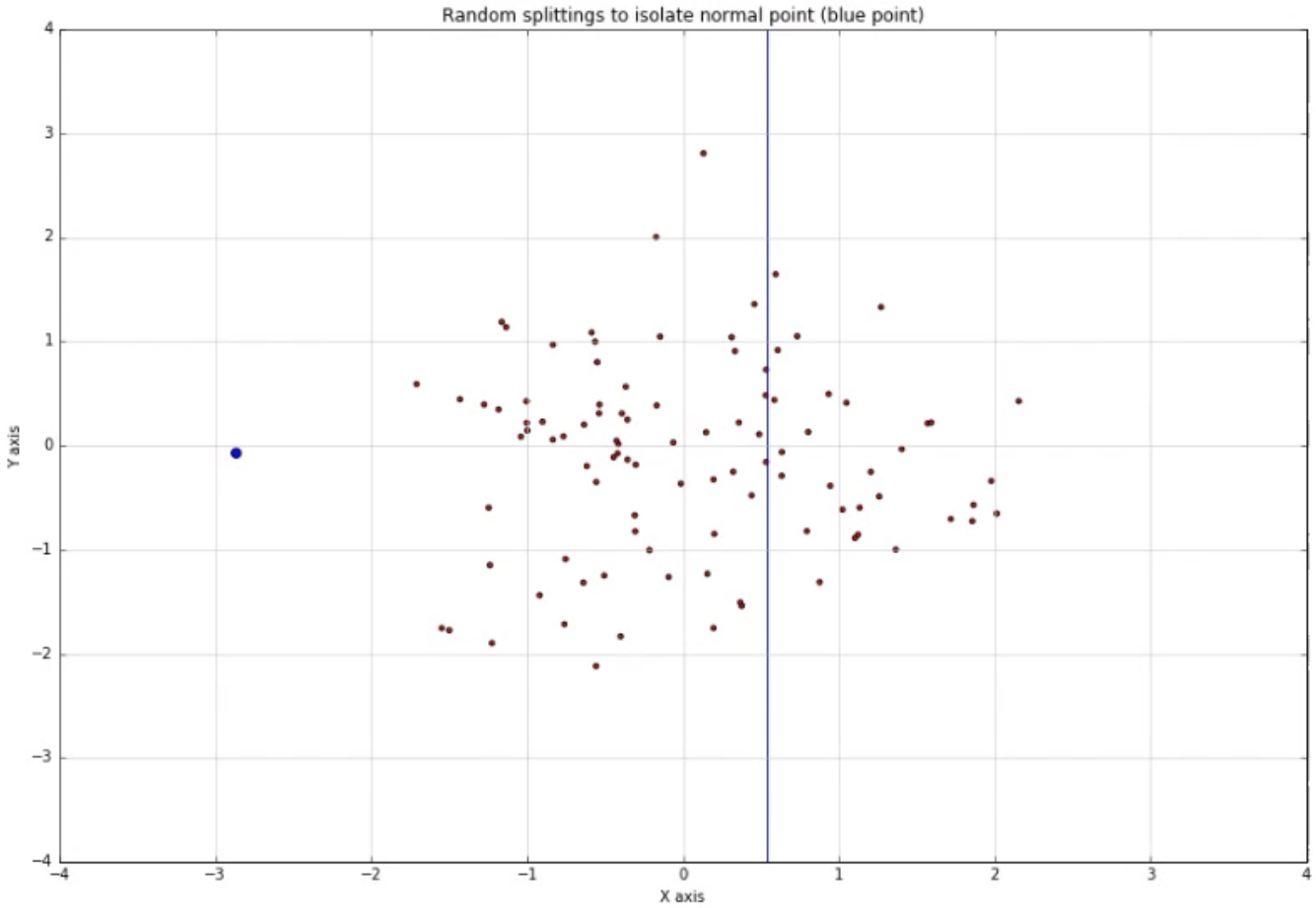


(a) Isolating x_i (b) Isolating x_o



Isolation Forest

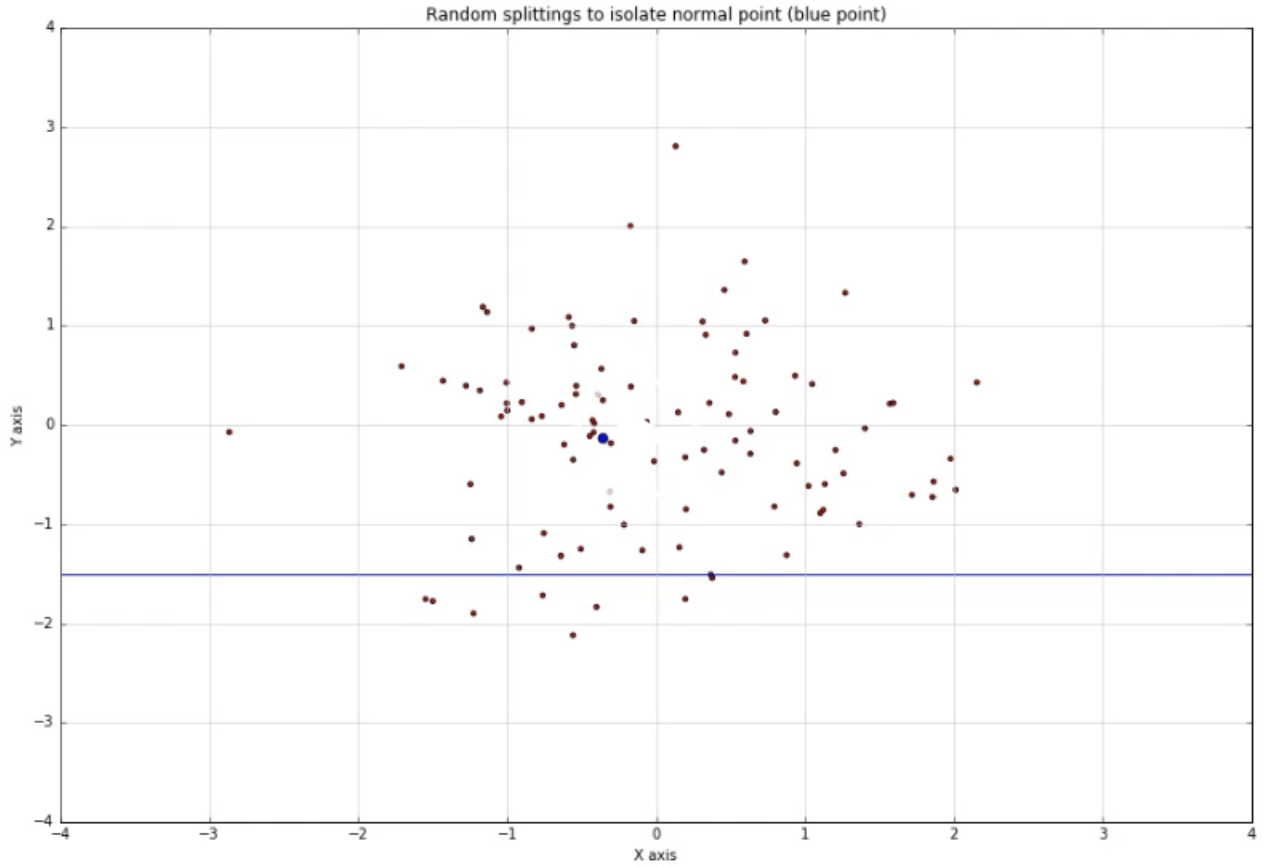
- Isolating a normal instance



https://blog.easysol.net/using-isolation-forests-anamoly-detection/

Isolation Forest

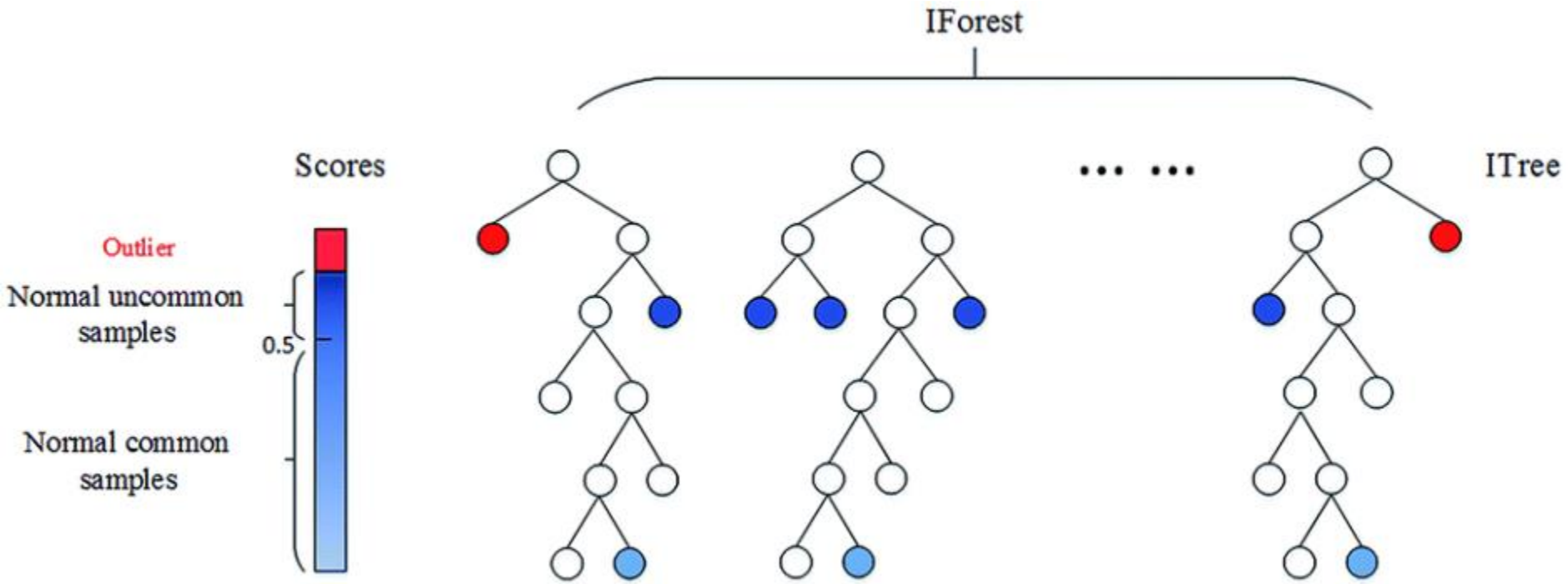
- Isolating an abnormal instance



https://blog.easysol.net/using-isolation-forests-anamoly-detection/

Isolation Forest

- The isolation characteristics of tree forms the basis of the method to detect novel instances
  - ✓ The average path to the terminal node can be used as a novelty score of an instance



Isolation Forest

- Definition: Isolation Tree (iTree)
  - ✓ Given a sample of data X of n instances, the dataset X is recursively divided by randomly selected attribute q with a split value p, until either
    - The tree reaches a height limit
    - |X| = 1
    - All instances in X have the same value
- Definition: Path Length
  - ✓ The path length h(x) of an instance x is measured by the number of edges x traverses an iTree from the root node to the terminal node in which the instance x is located
  - ✓ h(x) is normalized by the average path length of h(x) given n
    - $c(n) = 2H(n-1)-(2(n-1)/n)$ ($H(i) = \ln(i) + 0.5772156649$ (Euler's constant))

Isolation Forest

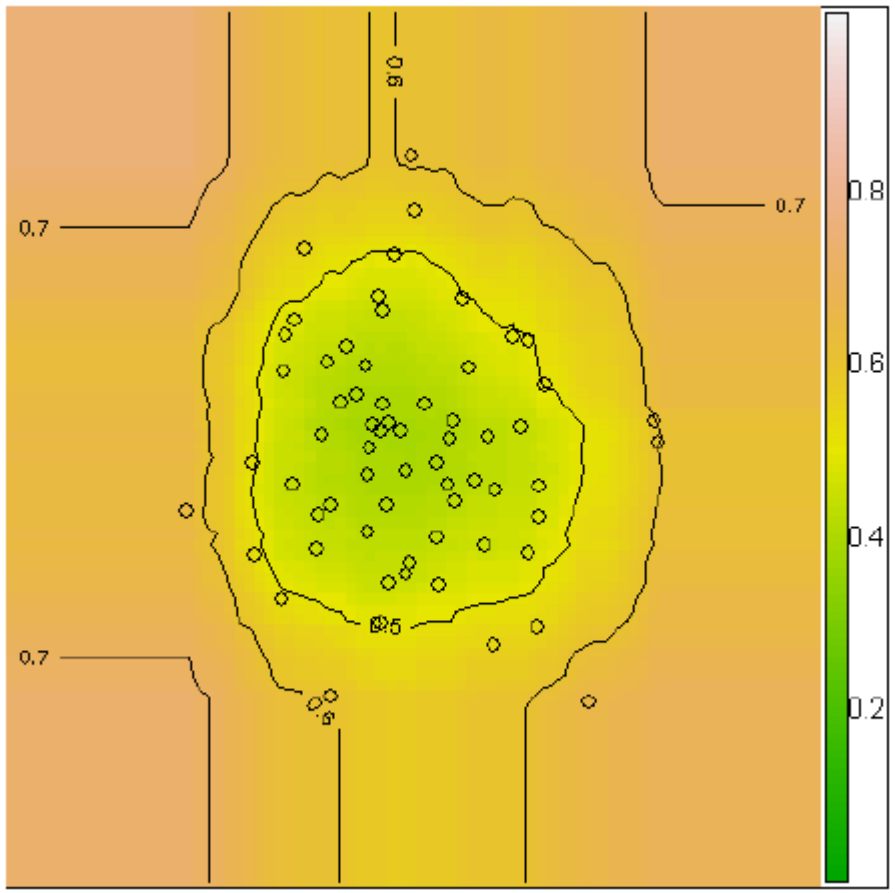
- Definition: Novelty score
  - ✓ The path length h(x) of an instance x is measured by the number of edges x traverses an iTree from the root node to the terminal node in which the instance x is located
  - ✓ h(x) is normalized by the average path length of h(x) given n
    - c(n) = 2H(n-1)-(2(n-1)/n) (H(i) = ln(i) + 0.5772156649 (Euler's constant))
  - ✓ The novelty score s of an instance x is defined by

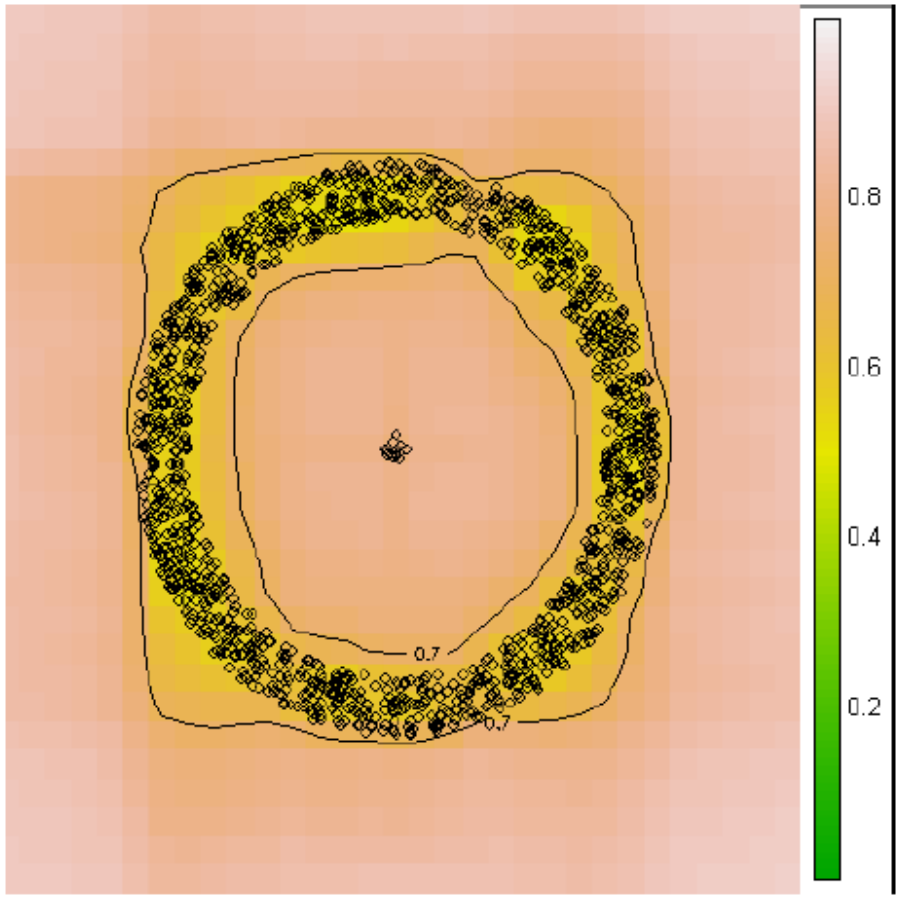
$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}}$$

- When $E(h(x)) \to c(n), \quad s \to 0.5$
- When $E(h(x)) \to 0, \quad s \to 1$
- When $E(h(x)) \to n - 1, \quad s \to 0$

Isolation Forest

- Novelty score contour





Isolation Forest

- Training Isolation Forest
  - ✓ Randomly sample datasets
  - ✓ Construct iTree
  - ✓ Compute the path length

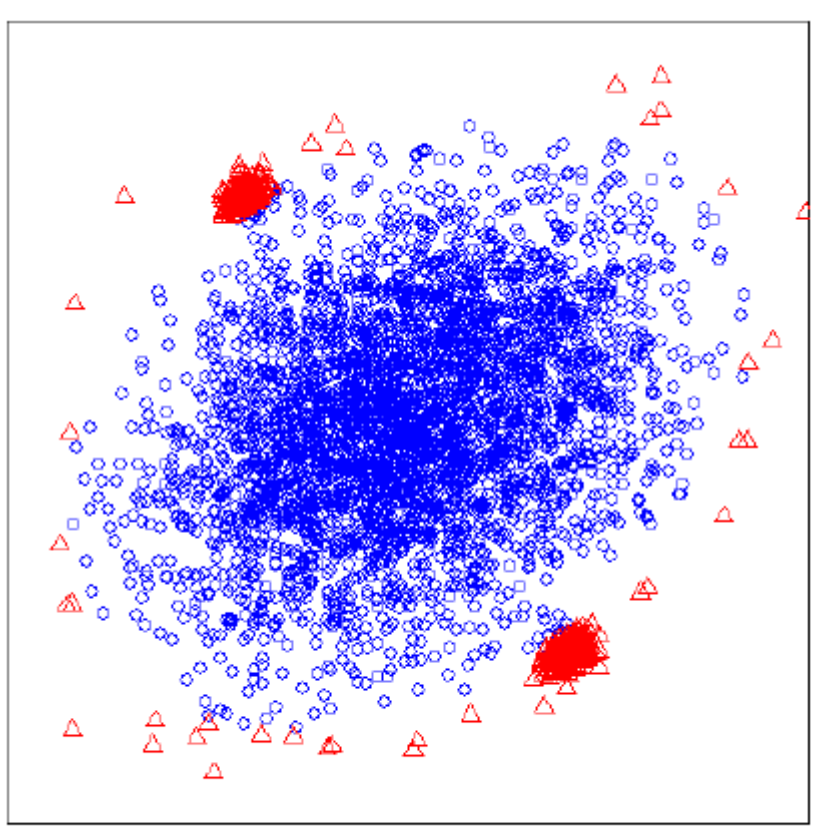
Algorithm 1 : $iForest(X, t, \psi)$

Inputs: X - input data, t - number of trees, ψ - subsampling size
Output: a set of t $iTrees$

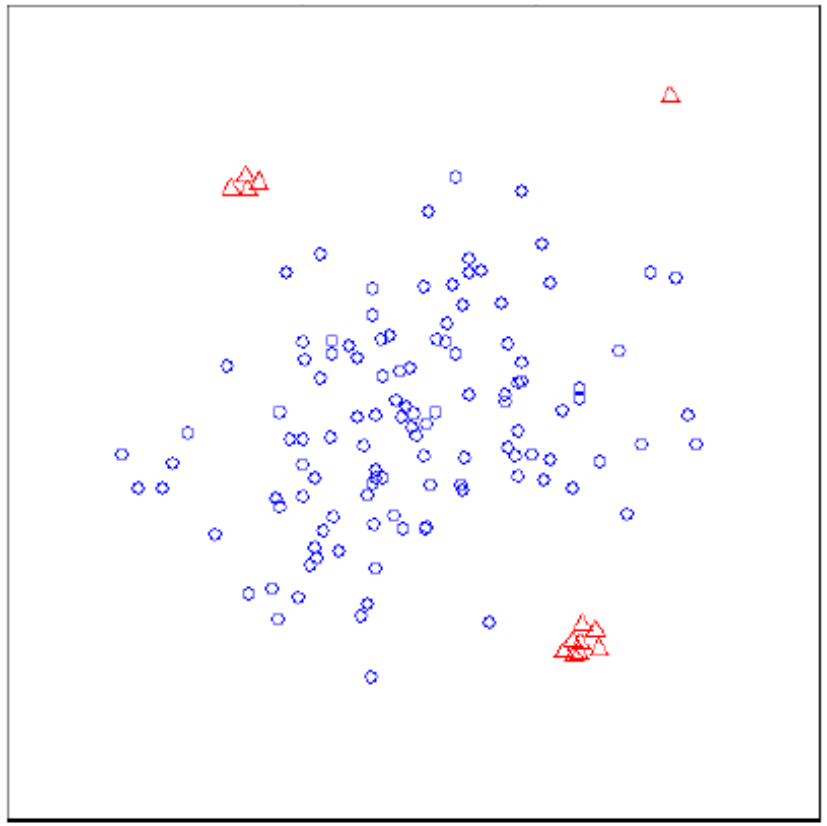
1: **Initialize** $Forest$
2: **for** $i = 1$ to t **do**
3: $\quad X' \leftarrow sample(X, \psi)$
4: $\quad Forest \leftarrow Forest \cup iTree(X')$
5: **end for**
6: **return** $Forest$

Isolation Forest

- Training Isolation Forest
  - ✓ Randomly sample datasets: 256 is generally enough



(a) Original sample
(4096 instances)



(b) Sub-sample
(128 instances)

Isolation Forest

- Training Isolation Forest
  - ✓ Construct iTree

Algorithm 2 : $iTree(X')$

Inputs: X' - input data
Output: an iTree

if X' cannot be divided **then**
  return $exNode\{Size \leftarrow |X'|\}$
else
  let Q be a list of attributes in X'
  randomly select an attribute $q \in Q$
  randomly select a split point p between the max and min values of attribute q in X'
  $X_l \leftarrow filter(X', q < p)$
  $X_r \leftarrow filter(X', q \geq p)$
  return $inNode\{Left \leftarrow iTree(X_l),$
    $Right \leftarrow iTree(X_r),$
    $SplitAtt \leftarrow q,$
    $SplitValue \leftarrow p\}$
end if

Isolation Forest

- Training Isolation Forest
  - ✓ Compute the path length

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

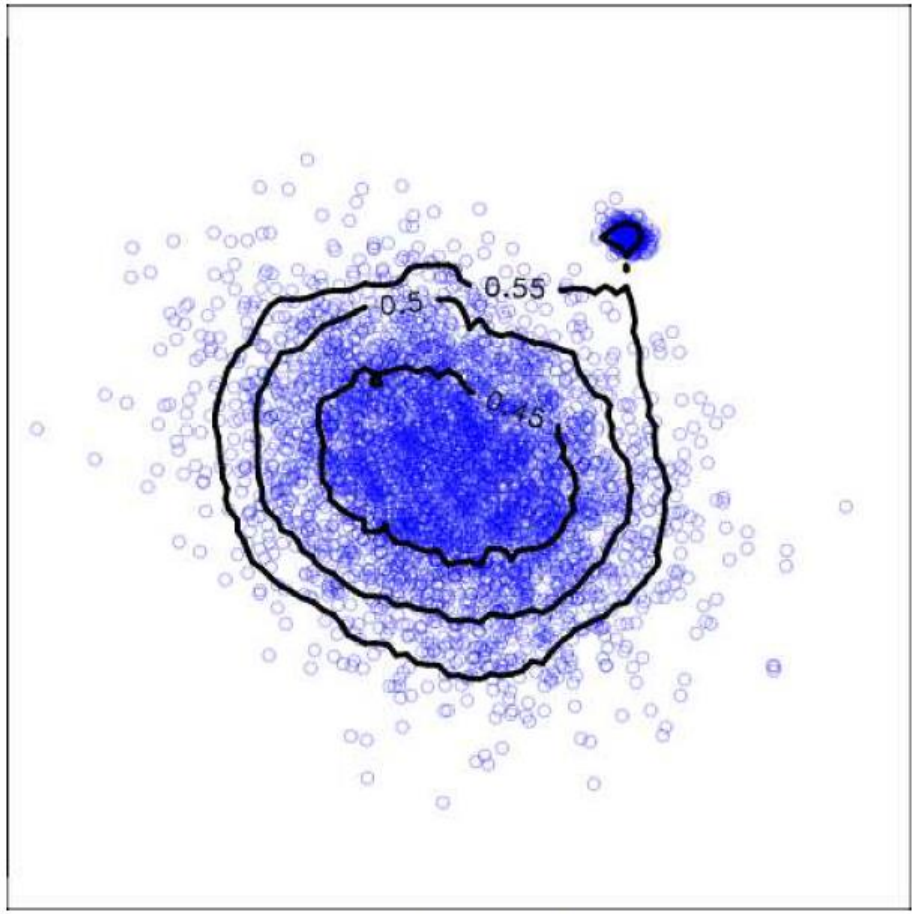
Inputs : x - an instance, T - an iTree, $hlim$ - height limit, e - current path length; to be initialized to zero when first called

Output: path length of x

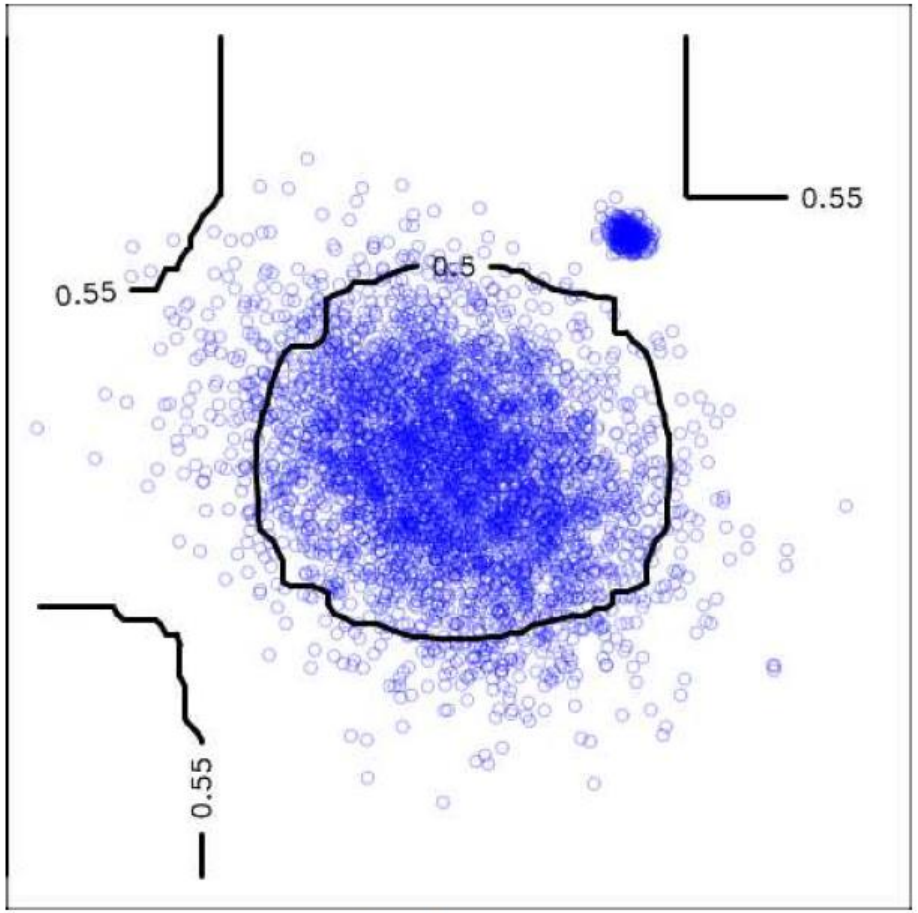
1: **if** T is an external node or $e \geq hlim$ **then**
2: return $e + c(T.size)$ {$c(.)$ is defined in Equation 1}
3: **end if**
4: $a \leftarrow T.splitAtt$
5: **if** $x_a < T.splitValue$ **then**
6: return $PathLength(x, T.left, hlim, e + 1)$
7: **else** $\{x_a \geq T.splitValue\}$
8: return $PathLength(x,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

Isolation Forest

- Effect of the height limit



(a) $hlim = 6$,



(b) $hlim = 1$,

Isolation Forest

- Empirical evaluation
  - ✓ Datasets

	n	d	anomaly class
Http (KDDCUP99)	567497	3	attack (0.4%)
ForestCover	286048	10	class 4 (0.9%) vs. class 2
Mulcross	262144	4	2 clusters (10%)
Smtp (KDDCUP99)	95156	3	attack (0.03%)
Shuttle	49097	9	classes 2,3,5,6,7 (7%)
Mammography	11183	6	class 1 (2%)
Annthyroid	6832	6	classes 1, 2 (7%)
Satellite	6435	36	3 smallest classes (32%)
Pima	768	8	pos (35%)
Breastw	683	9	malignant (35%)
Arrhythmia	452	274	classes 03,04,05,07, 08,09,14,15 (15%)
Ionosphere	351	32	bad (36%)
hbk	75	4	14 points (19%)
wood	20	6	6 instances (30%)

Isolation Forest

- Empirical evaluation
  - ✓ Performance (in terms of AUROC)

	AUC				
	*i*Forest	ORCA	SVM	LOF	RF
Http (KDDCUP99)	**1.00**	0.36	0.90	*	**
ForestCover	0.87	0.83	**0.90**	0.57	**
Mulcross	**0.96**	0.33	0.59	0.59	**
Smtp (KDDCUP99)	**0.89**	0.80	0.78	0.32	**
Shuttle	**1.00**	0.60	0.79	0.55	**
Mammography	**0.84**	0.77	0.65	0.67	**
Annthyroid	**0.84**	0.68	0.63	0.72	**
Satellite	**0.73**	0.65	0.61	0.52	**
Pima	0.67	**0.71**	0.55	0.49	0.65
Breastw	**0.98**	**0.98**	0.66	0.37	0.97
Arrhythmia	**0.81**	0.78	0.71	0.73	0.60
Ionosphere	0.83	**0.92**	0.71	0.89	0.85

(a) AUC performance

Isolation Forest

- Empirical evaluation
  - ✓ Performance (in terms of computational complexity)

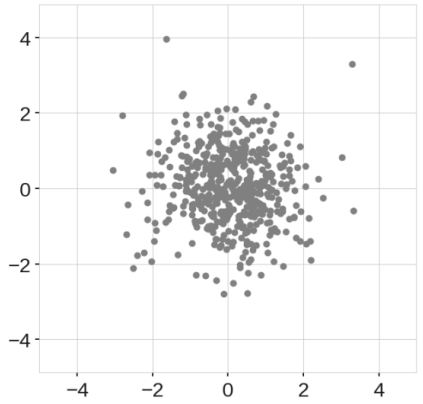
	Time (seconds)						
	*i*Forest			ORCA	SVM	LOF	RF
	Train	Eval.	Total				
Http	0.25	15.33	**15.58**	9487.47	35872.09	*	**
ForestCover	0.76	15.57	**16.33**	6995.17	9737.81	224380.19	**
Mulcross	0.26	12.26	**12.52**	2512.20	7342.54	156044.13	**
Smtp	0.14	2.58	**2.72**	267.45	986.84	24280.65	**
Shuttle	0.30	2.83	**3.13**	156.66	332.09	7489.74	**
Mammography	0.16	0.50	**0.66**	4.49	10.8	14647.00	**
Annthyroid	0.15	0.36	**0.51**	2.32	4.18	72.02	**
Satellite	0.46	1.17	**1.63**	8.51	8.97	217.39	**
Pima	0.17	0.11	0.28	**0.06**	**0.06**	1.14	4.98
Breastw	0.17	0.11	0.28	**0.04**	0.07	1.77	3.10
Arrhythmia	2.12	0.86	2.98	**0.49**	0.15	6.35	2.32
Ionosphere	0.33	0.15	0.48	**0.04**	0.04	0.64	0.83

(b) Actual processing time

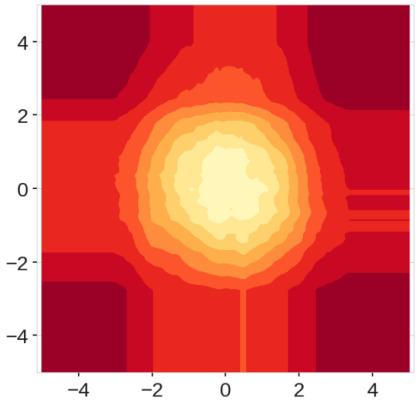
Extended Isolation Forests

Hariri et al. (2018)

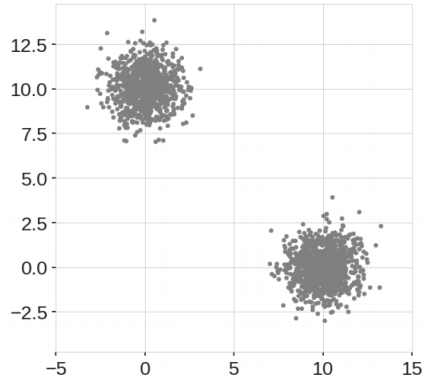
- Motivation



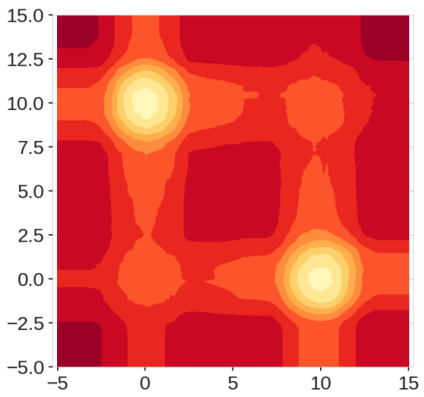
(a) Normally Distributed Data



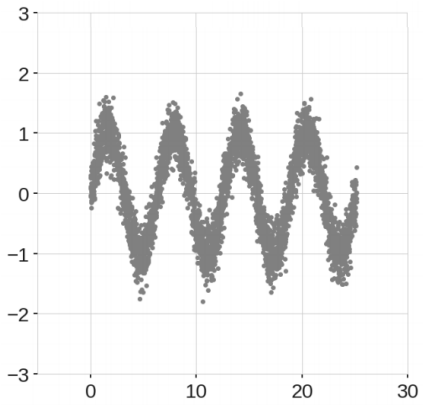
(b) Anomaly Score Map



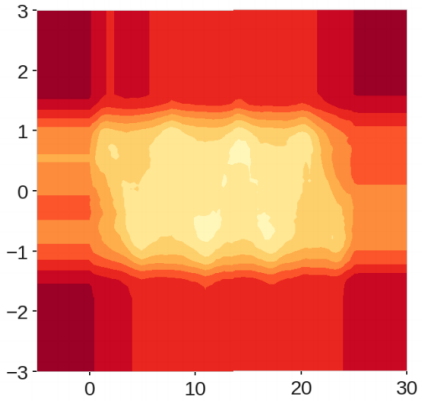
(a) Two normally distributed clusters



(b) Anomaly Score Map



(a) Sinusoidal data points with Gaussian noise.



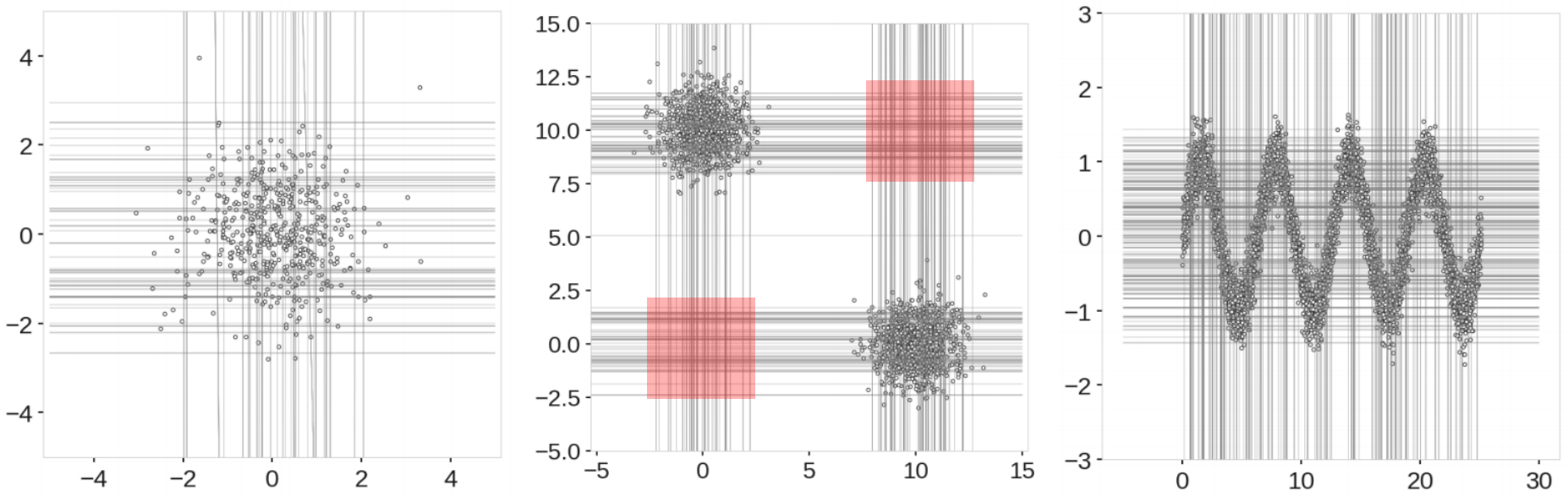
(b) Anomaly Score Map

Standard Isolation Forest
cannot work well for this dataset

Extended Isolation Forests

- Contribution

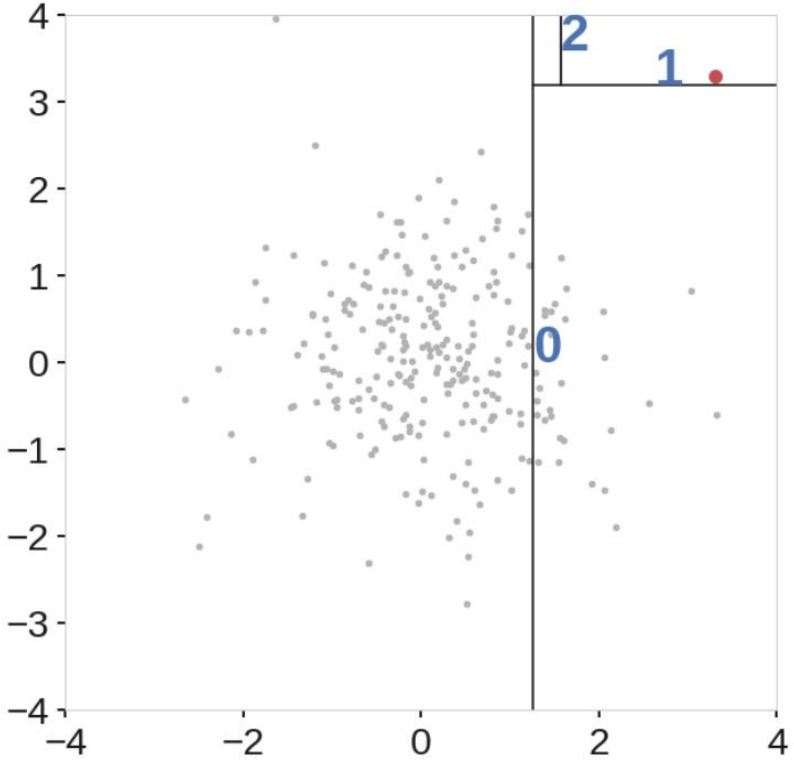
But as we have seen, the branch cuts are always either horizontal or vertical, and this introduces a bias and artifacts in the anomaly score map. There is no fundamental reason in the algorithm that requires this restriction, and so at each branching point, we can select a branch cut that has a random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



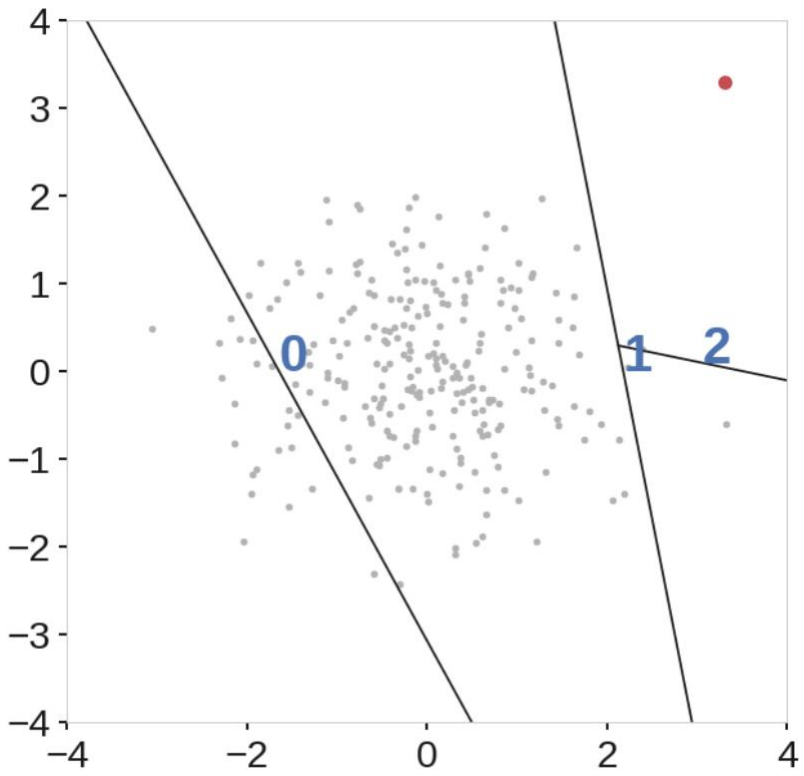
Extended Isolation Forests

- Illustrative example

Standard Isolation Forest



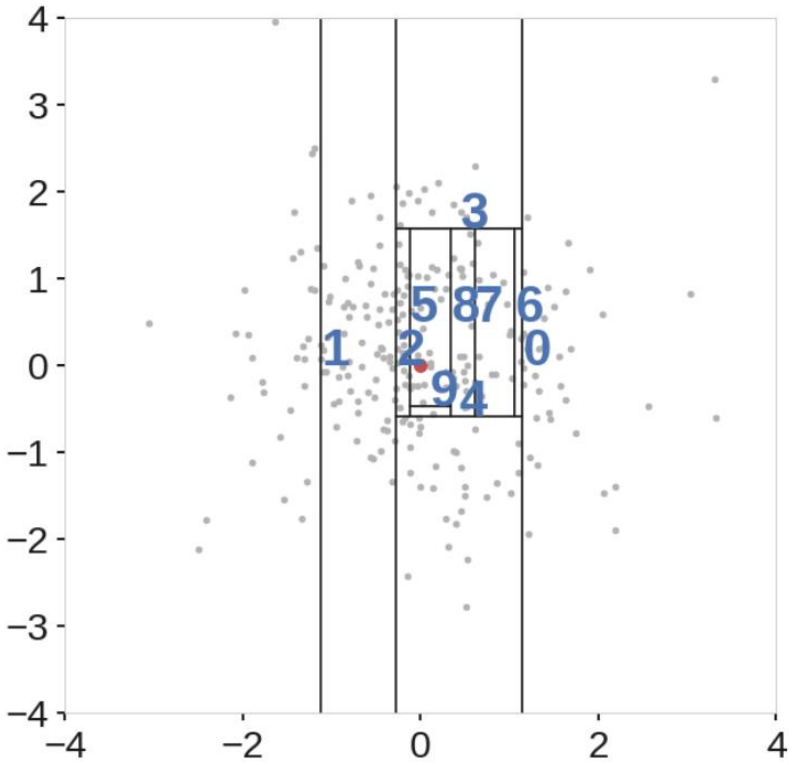
Extended Isolation Forest



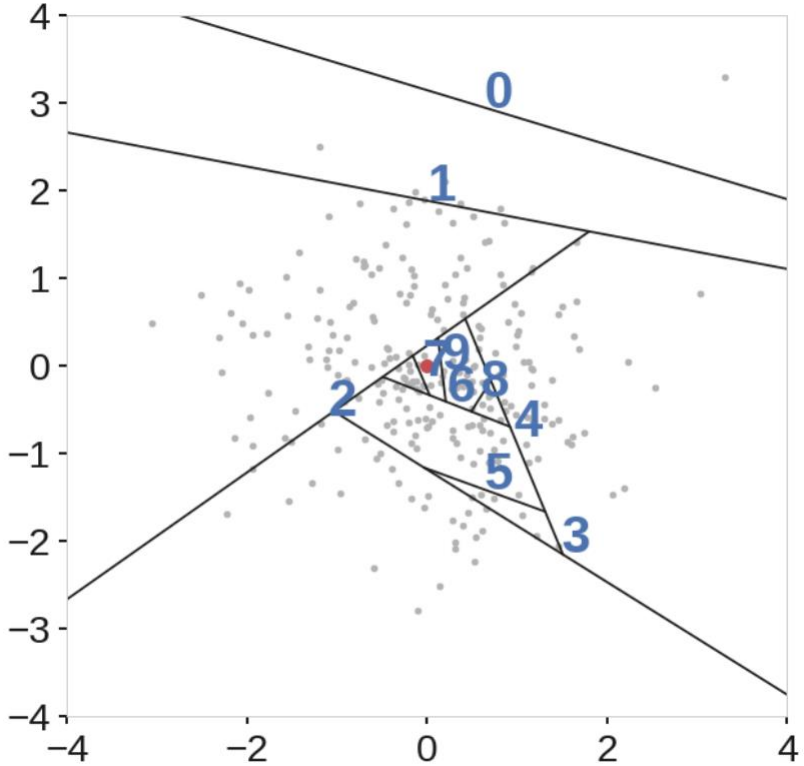
Extended Isolation Forests

- Illustrative example

Standard Isolation Forest



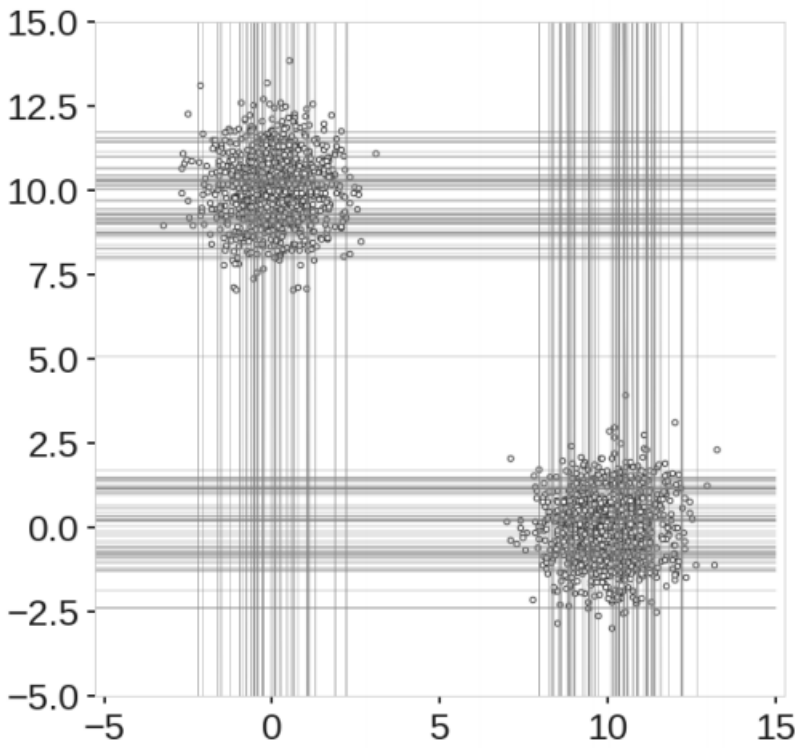
Extended Isolation Forest



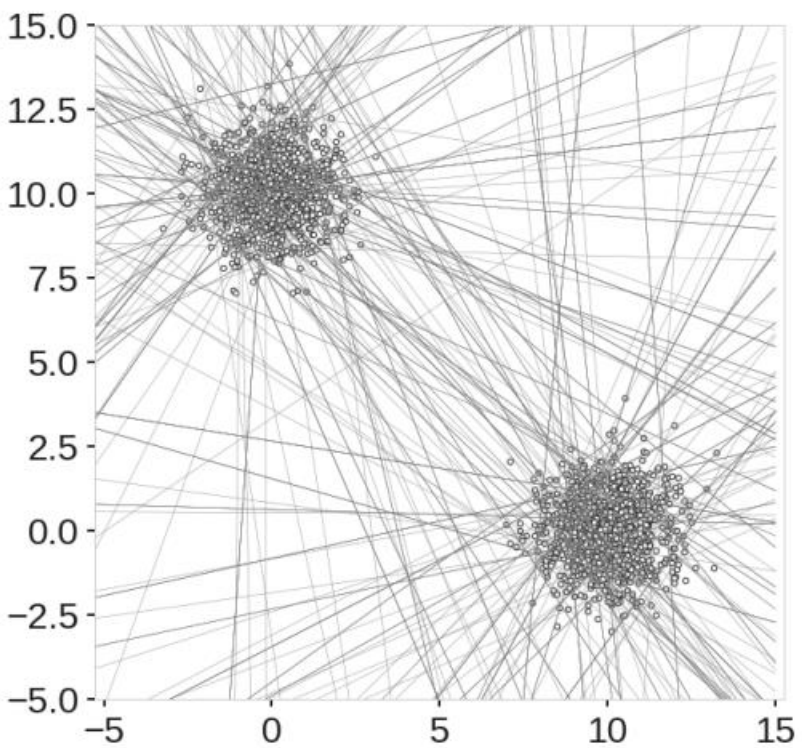
Extended Isolation Forests

- How are the biases reduced?

Standard Isolation Forest



Extended Isolation Forest



Extended Isolation Forests

- 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 extension level

7: $X_l \leftarrow filter(X, (X - p) \cdot n \leq 0)$

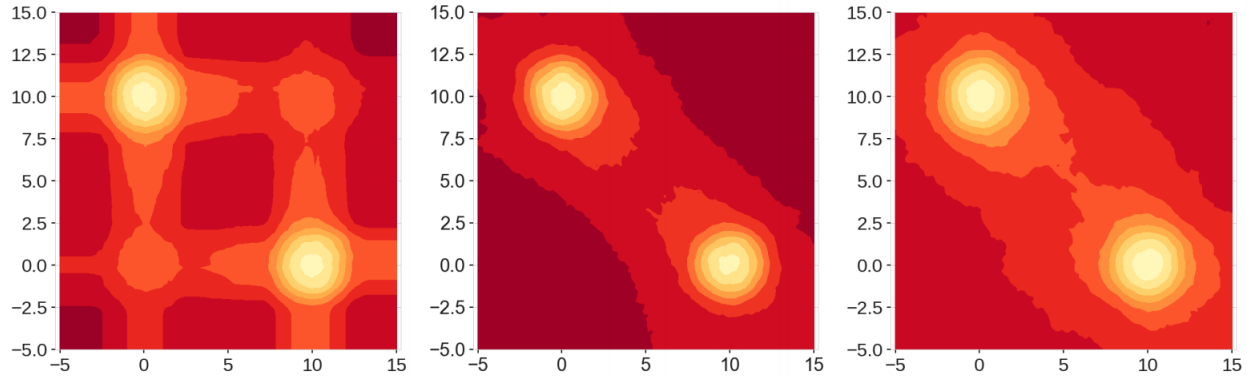
8: $X_r \leftarrow filter(X, (X - p) \cdot n > 0)$

9: **return** inNode$\{ Left \leftarrow iTree(X_l, e + 1, l),$
$Right \leftarrow iTree(X_r, e + 1, l),$
$Normal \leftarrow n,$
$Intercept \leftarrow p\}$

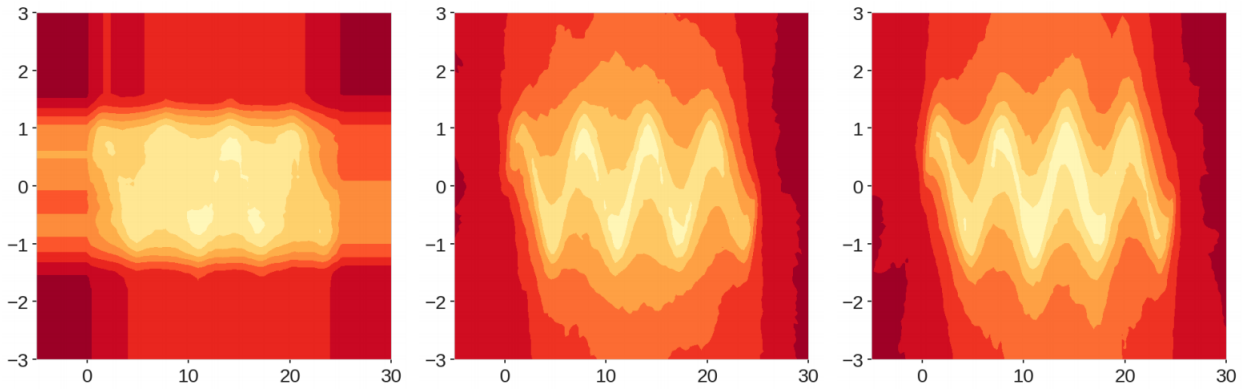
10: **end if**

Extended Isolation Forests

- Anomaly score distribution



(a) Standard IF (b) Rotated IF (c) Extended IF



(a) Generic IF (b) Rotated IF (c) Extended IF


Q & A

References

Research Papers

- Breunig, M.M., Kriegel, H.-P., Ng, R.T., and Sander, J. (2000). LOF: Identifying density-based local outliers. Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data: 93-104.
- Chalapathy, R., Menon, A. K., & Chawla, S. (2018). Anomaly detection using one-class neural networks. arXiv preprint arXiv:1802.06360.
- Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. ACM computing surveys 41(3): 15.
- Harmeling, S. Dornhege, G., Tax, D. Meinecke, F., and Muller, K-R. (2006). From outliers to prototype: Ordering data. Neurocomputing 69(13-15): 1608-1618.
- Hariri, S., Kind, M. C., & Brunner, R. J. (2018). Extended Isolation Forest. arXiv preprint arXiv:1811.02141.
- Kang, P. and Cho, S. (2009). A hybrid novelty score and its use in keystroke dynamics-based user authentication. Pattern Recognition 42(11): 3115-3127.
- Liu, F.T., Ting, K. M., & Zhou, Z. H. (2008, December). Isolation forest. In Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on (pp. 413-422). IEEE.
- Liu, F.T., Ting, K. M., & Zhou, Z. H. (2012). Isolation-based anomaly detection. ACM Transactions on Knowledge Discovery from Data (TKDD), 6(1), 3.
- Oza, P., & Patel, V. M. (2018). One-class convolutional neural network. IEEE Signal Processing Letters, 26(2), 277-281.
- Perera, P., Nallapati, R., & Xiang, B. (2019). Ocgan: One-class novelty detection using gans with constrained latent representations. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2898-2906).

References

Research Papers

- Pimentel, M.A.F., Clifton, D.A., Clifton, L., and Tarassenko, L. (2014). A review of novelty detection, Signal Processing 99: 215-249.
- Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U., & Langs, G. (2017, June). Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In International Conference on Information Processing in Medical Imaging (pp. 146-157). Springer, Cham.
- Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., and Williamson, R. C.(2001). Estimating the support of a high-dimensional distribution. Neural computation 13(7): 1443-1471.
- Tax, D.M. (2001). One-class classification, Ph.D. Thesis, Delft University of Technology, Netherlands.
- Tax, D.M. and Duin, R.P. (2004). Support vector data description. Machine learning 54(1): 45-66.

Other materials

- Pages 28-33 & 36: http://research.cs.tamu.edu/prism/lectures/pr/pr_l7.pdf
- Figures in Auto-encoder section: https://dl.dropboxusercontent.com/u/19557502/6_01_definition.pdf
- Gramfort, A. (2016). Anomaly/Novelty detection with scikit-learn: https://www.slideshare.net/agramfort/anomalynovelty-detection-with-scikitlearn