Active learning for anomaly detection

Sofia Vergara Puccini

Table of contents

General Concepts

Active learning

- Interaction between learning algorithm and user to obtain true labels of data points.
- Learning algorithm queries instances to a user iteratively.
- Feedback of analyst is used to update the scoring function.

Advantages:

- Algorithm learns effectively the parameters with only a few examples
- Useful when labelling process is expensive

Goal: Maximize the amount of true positives (real anomalies) presented to the user from a limited budget of points.

Ensemble methods

- Combines predictions from two or more models.
- Examples: Random forest, AdaBoost, Gradient Boosting, IFOR

Advantages in anomaly detection:

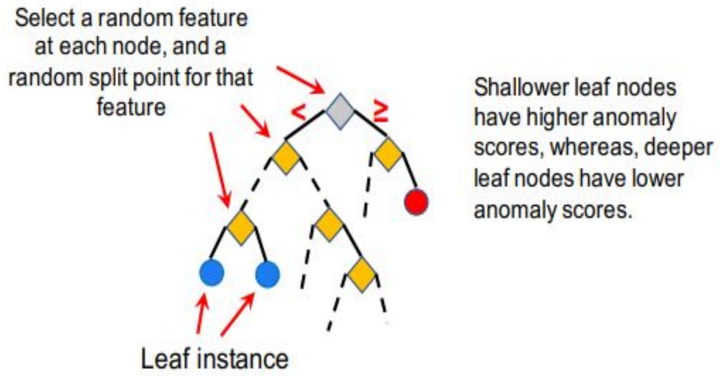
- Single detectors are highly susceptible to:
  - imbalanced data, e.g type of anomalies
  - Problem application.
- Multiple detectors make predictions more robust -> less false positives

Isolation Forest trees

- Isolates anomalies instead of profiling normal instances
- Binary tree structure
  - Repeated partitions of feature space
  - Random split point of attributes
- Anomalies isolated faster due to extreme attributes

Advantages:

- Has low memory requirement
- Can scale up to handle large data and many attributes



Source: S. Das, M. Islam, N. Kannapan, and J. Doppa. 2019. Active Anomaly detection via Ensembles: Insights, Algorithms and Interpretability. School of EECS, Washington State Univeristy (2019)

Isolation Forest trees

Remarks:

- Partitions are done recursively until instances reach a leaf node
- Path length: No. of partitions from root to leaf ***l***
- Only used in ensembles, so the average path length of an anomaly is shorter

Isolation Forest trees

- Ensemble ***E*** is composed by ***m*** detectors (leaf nodes)
- Score of each instance is the path length to the leaf
- Score of instance is normalized
- ***p*** weight or relevance of detector ***i***

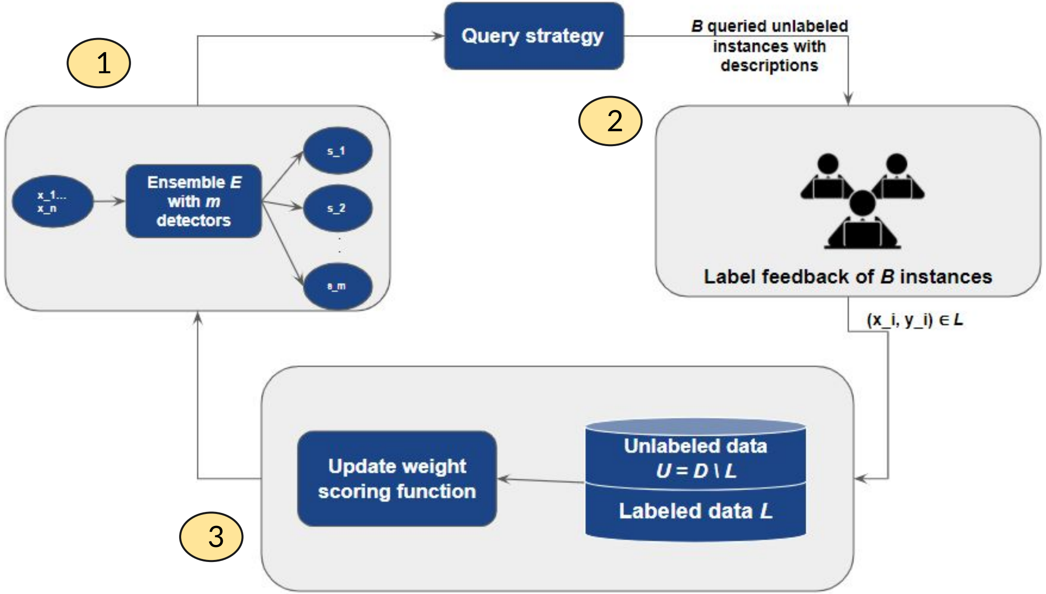
As a result:

$$\mathtt{Score}(\mathbf{x}) = \sum_{i=1}^{m} p_i(\mathbf{x}) \cdot s_i(\mathbf{x})$$

- Score vector is sparse
- Ideal set of weights produces anomalies to be in furthest positive region of scoring space

Framework for anomaly detection using active learning

Framework for AD using active learning



1) Create model for scoring instances as anomalies or nominals (e.g ensemble)
2) Selecting instances to be queried, e.g randomly, highest score
3) Update parameters of the model and repeat

Source: S. Das, M. Islam, N. Kannapan, and J. Doppa. 2019. Active Anomaly detection via Ensembles: Insights, Algorithms and Interpretability. School of EECS, Washington State Univeristy (2019)

Description of subspaces

Compact descriptions

- Provide a description to analyst about labeled instances
- Helps understanding how predictions are made
- Can be used to obtain anomalies from different subspaces (using Select-Diverse)

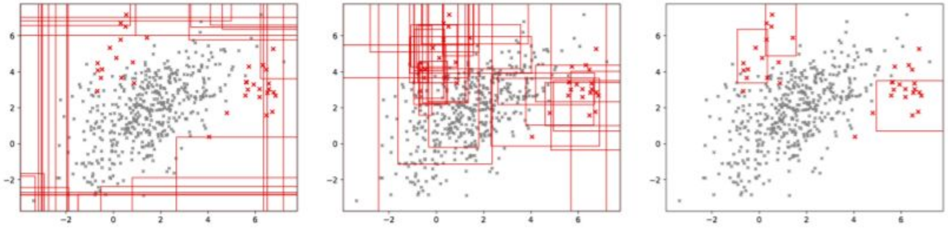
Goal:

- Find minimal region that includes all labeled instances

Compact descriptions

Steps:

1. Define set ***Z*** of instances to give a description, e.g true anomalies identified
2. Find subspaces ***S*** containing **Z**
3. Calculate the volume of the subspaces
4. Find smallest set of subspaces that contain all ***Z***
    a. Problem formulated as an integer linear program



(a) Baseline (b) Active Anomaly Detection (c) Compact Description

Source: S. Das, M. Islam, N. Kannapan, and J. Doppa. 2019. Active Anomaly detection via Ensembles: Insights, Algorithms and Interpretability. School of EECS, Washington State Univeristy (2019)

Complexity of subspaces

- Previous approach does not consider:
  - Precision of the subspace, i.e amount of nominals in subspace
  - Complexity of subspace, i.e predicate rules defining it.
- New approach: Penalizes subspace using complexity of rules and amount of nominals in **S**

Example of predicate rule:

“If credit score = ‘Low’ or (employed = False and savings < 100), then approve loan = False. “

Complexity of subspaces

Steps:

1. Select labeled and unlabeled instances (containing nominals) to provide simple description
2. Obtain subspaces ***S*** containing instances
3. Calculate volume, No. of nominals η and complexity ς of the subspaces
4. Find smallest subset of subspaces ***S****
5. Retain subspaces whose precision (based on η) is larger than threshold ***t***

$$\mathbf{S}^* = \arg\min_{\mathbf{x}\in\{0,1\}^k} \mathbf{x} \cdot (\mathbf{v} \circ (\mathbf{1}_k + \eta) + \varsigma)$$

ς = *2^(rule length(s)−1)*

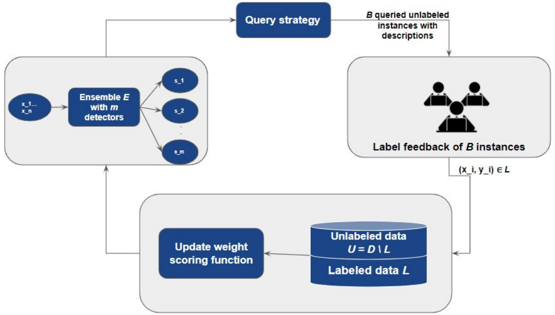
Algorithms for AD using Active Learning

BAL: Batch Active Learning

Remarks about the algorithm:

1. The algorithm starts by getting label ***y*** = {-1,+1} of selected instance from analyst
2. Store score ***z*** (from ensemble) in matrix ***H+*** or ***H-*** depending on label
3. Minimizes loss function based on labeled instances and calculates ensemble weights ***w***
4. **Result**: final set ***w***, ***H+*** and H-, after all ***B*** points are analyzed

$$\texttt{Score}(\mathbf{x}) = \sum_{i=1}^{m} p_i(\mathbf{x}) \cdot s_i(\mathbf{x})$$



BAL: Batch Active Learning

Remarks about loss function:

1. Composed by hinge loss and influence λ of the initial set of weights ***w_unif***
2. Penalizes model if scores are lower for true positives, and higher for nominals
3. Influence λ of initial weights decrease as more instances are labeled
4. ***q_T***(w(t−1)) current selected instance evaluated with the weights of the previous iteration

$$\ell(q, \mathbf{w}; (\mathbf{z}_i, y_i)) = \begin{cases} 0 & \mathbf{w} \cdot \mathbf{z}_i \geq q \text{ and } y_i = +1 \\ 0 & \mathbf{w} \cdot \mathbf{z}_i < q \text{ and } y_i = -1 \\ (q - \mathbf{w} \cdot \mathbf{z}_i) & \mathbf{w} \cdot \mathbf{z}_i < q \text{ and } y_i = +1 \\ (\mathbf{w} \cdot \mathbf{z}_i - q) & \mathbf{w} \cdot \mathbf{z}_i \geq q \text{ and } y_i = -1 \end{cases}$$

$$\lambda^{(t)} = \frac{0.5}{|\mathbf{H}_+| + |\mathbf{H}_-|}$$

$$\mathbf{w}_{unif} = [\frac{1}{\sqrt{m}}, \ldots, \frac{1}{\sqrt{m}}]^T$$

Contextual Anomaly detection

- Real world systems often produce anomalies that are catalogued as such depending on the situation.
- Global perspective can hide abnormal instances.

Example:

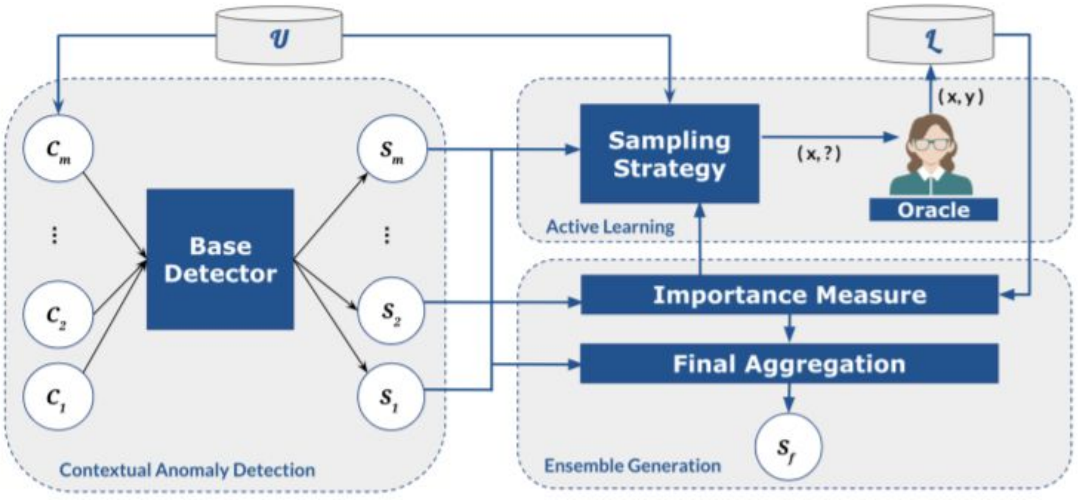
"High energy consumption is normal during winter but the same behaviour might be abnormal in summer"

- Environmental factor (attribute) contextualizes what an anomaly is.
- Distinction between Contextual and Behavioural attributes allows identification of anomalies
    - **All features = Behavioural + Contextual features**

Framework for WisCon

Remarks:

- *m* detectors are the contexts
- Scores defined to each detector



Source: E. Calikus, S. Nowaczyk, M. Bouguelia, and O Dikmen. 2021. Wisdom of the Contexts: Active Ensemble Learning for Contextual Anomaly Detection. (2021)

Wisdom of Contexts (WisCon): Ensemble

Steps:

1. Clusters for each instance **x** w.r.t each context
   a. Remark: Clustering algorithm depends on the data
2. Isolation forest trees for each cluster
3. Evaluate the deviation of the instance ***x_j*** to its cluster using the behavioural features
4. Create score vector for each context

Wisdom of Contexts (WisCon): Ensemble

Remarks:

- Contexts can be defined by all possible combinations of contextual features or PCA
- Contexts have different ranges, so scores are normalized
- Each instance is evaluated in all contexts

Wisdom of Contexts (WisCon): Active learning

Steps:

1. Provide instance to analyst to label
2. Store label in matrix ***L***
3. Provide a weight to labeled instance based on query strategy
    - If query strategy does not assume differences, $\theta = 1/t$ at iteration t

Goal:

Maximize the expected information gain of x based on the query strategy chosen

Wisdom of Contexts (WisCon): Update weights

Steps:

1. Calculate hard label ***p*** to instances ***x*** based on the score ***s*** of the context
2. If score of context > 0.9, **1** else **0**
   a. Each instance has ***m*** hard labels (***m*** contexts)
3. Compare the label of the analyst with the hard label
   a. If hard label = label analyst, then $l_{i,j}$ = **0** else **1**
4. Calculate detection error $e_{i,t}$ of the context at iteration ***t***
5. Calculate importance of the Context

$$\epsilon_{i,t} = \frac{\sum_{j=1}^{t} \theta_j l_{i,j}}{\sum_{j=1}^{t} \theta_j},$$

$$I_i = \frac{1}{2} ln(\frac{1-\epsilon_{i,t}}{\epsilon_{i,t}})$$

Wisdom of Contexts (WisCon): Weights update

Steps (continue):

6. Pruning of <u>context</u> with importance < 0 -> detection error of context > 0.5

7. With the remaining ***p*** contexts and their scores, recalculate scores of instances as:

$$s_j = \frac{\sum_{i=1}^{p} I_i \times s_{i,j}}{\sum_{i=1}^{p} I_i}$$

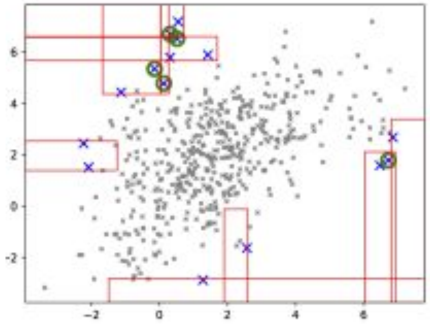
Query strategies

Query strategies

- Select critical instances, which help the model improve its accuracy
- Assumption: Analyst is only capable of labelling few instances.
- Some common techniques:
    - Most anomalous instances (highest ranked by the model)
    - Uncertainty sampling
- Select-Diverse and Low confidence anomalies

Select Diverse

- Search instances having minimum subspace overlap
- Most anomalous instances having minimum overlap given to analyst
- Similar approach like in Compact descriptions



S. Das, M. Islam, N. Kannapan, and J. Doppa. 2019. Active Anomaly detection via Ensembles: Insights, Algorithms and Interpretability. School of EECS, Washington State Univeristy (2019)

Select Diverse

Algorithm 1 `Select-Diverse` $(\mathbf{X}, b, n)$

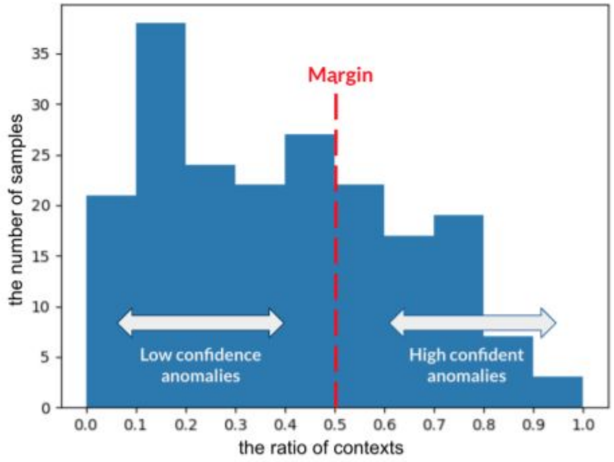
Input: Unlabeled dataset $\mathbf{X}$, # instances to select b, # candidate instances n $(n \geq b)$
Let $\mathcal{Z} = n$ top-ranked instances as candidates $\subseteq \mathbf{X}$ (blue points in Figure 8a)
Let $\mathbf{S}^* =$ subspaces with Equation 1 that contain $\mathcal{Z}$ (rectangles in Figures 8b and 8c)
Set $\mathbf{Q} = \emptyset$
while $|\mathbf{Q}| < b$ **do**
  Let $\mathbf{x} =$ instance with highest anomaly score $\in \mathcal{Z}$ s.t. $\mathbf{x}$ has minimal overlapping regions in $\mathbf{S}^*$ with instances in $\mathbf{Q}$
  Set $\mathbf{Q} = \mathbf{Q} \cup \{\mathbf{x}\}$ (green circles in Figure 8c)
  Set $\mathcal{Z} = \mathcal{Z} \setminus \{\mathbf{x}\}$
end while
return $\mathbf{Q}$

$$\mathbf{S}^* = \arg\min_{\mathbf{x} \in \{0,1\}^k} \mathbf{x} \cdot \mathbf{v}$$
$$\text{s.t.} \quad \mathbf{U} \cdot \mathbf{x} \geq \mathbf{1} \text{ (where } \mathbf{1} \text{ is a column vector of } p \text{ 1's)}$$

Low Confidence Anomalies

- Multiple contexts unveiling anomalies, but these are rare
- Many true positives are only scoring as anomalies in less than 20% of the contexts (low confidence anomalies)
- These rare contexts should have high importance

Goal: Select data points around the margin of the anomalies distribution.



E. Calikus, S. Nowaczyk, M. Bouguelia, and O Dikmen. 2021. Wisdom of the Contexts: Active Ensemble Learning for Contextual Anomaly Detection. (2021)

Low Confidence Anomalies

Steps:

1. Calculate margin of instances with the importance of the contexts
   - **Margin(x)**: How close is instance to margin of the distribution
2. Calculate sampling measure:
   - **Q_LCA** gives the instances with higher margin rates, higher probabilities of being selected
   - λ controls how influenced the sampling is towards margin rates, λ = 0 means random sampling
3. Margin of instance and importance of context are updated recursively

$$\text{margin}(x_j) = 100 \times \left(1 - \left|\frac{2\sum_{i=1}^{m} I_i \times p_{i,j}}{\sum_{i=1}^{m} I_i} - 1\right|\right) \qquad Q_{LCA} = \text{argmax} \frac{\exp(\lambda \times \text{margin}(x))}{u_x}$$

Low Confidence Anomalies

To avoid selecting confident anomalies and normal samples, i.e keeping them far from the margin, weights $\boldsymbol{\theta}$ are calculated for the labeled instances.

- **θj = margin(x)** if the true label is **1**, otherwise the weight is **0**
- Impact of normal data points is eliminated from the importance scores of the contexts
- Anomalies with higher margin rates -> strong impact on importance scores of contexts

Summary

- Active learning is useful in applications where labelling process is expensive
- Isolation forest focuses on isolating anomalies rather than profiling normal instances
- Compact descriptions allow analyst to understand predictions of the model
- BAL aims at giving high scores to anomalies and low to nominal instances
- WisCon scores instances as anomalies depending on contextual and behavioural attributes
- While Select-Diverse focuses on finding most anomalous instances without overlapping, LCA looks for anomalies not identified in most contexts.

References

1. E. Calikus, S. Nowaczyk, M. Bouguelia, and O Dikmen. 2021. Wisdom of the Contexts: Active Ensemble Learning for Contextual Anomaly Detection. (2021)
2. S. Das, M. Islam, N. Kannapan, and J. Doppa. 2019. Active Anomaly detection via Ensembles: Insights, Algorithms and Interpretability. School of EECS, Washington State Univeristy (2019)
3. F. Liu, K. Ting, and Z. Zhou. 2008. Isolation forest. Eighth IEEE International Conference on Data Mining (2008), 413–422. https: //doi.org/10.1109/ICDM.2008.17