Anomaly detection: Global v/s Local structures

Tirth Bhayani
Sanchita Kalra

Global Variable v/s Local Variables

Local structures deal with specific, localised interactions and patterns within a limited context, while global structures involve trends and relationship that span the entire system or dataset

Both global and local structures are important for gaining comprehensive insights into a complex dataset

Principal Component Analysis (PCA)

PCA is a dimensionality reduction technique that aims to transform high dimensional data into a lower dimensional representation while preserving as much of the original variance possible.

Algorithmic description of PCA:

- Data preprocessing: Given a dataset with n data points, each having m features, represent it as a matrix X of size n×m. Center the data by subtracting the mean of each feature from the corresponding feature values, I.e. making mean zero.
- Co-variance matrix: Covariance is used to identify any relationship between the variables if it exists. It shows the “spread” of points around the mean.

Mathematical formula for it:

$$\text{cov}(X, Y) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{n - 1}$$

Hence our covariance matrix will look like:

$$A = \begin{bmatrix} \text{cov}(y_1, y_1) & \text{cov}(y_1, y_2) & \cdots & \text{cov}(y_1, y_N) \\ \text{cov}(y_2, y_1) & \text{cov}(y_2, y_2) & \cdots & \text{cov}(y_2, y_N) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}(y_N, y_1) & \text{cov}(y_N, y_2) & \cdots & \text{cov}(y_N, y_N) \end{bmatrix}$$

Where $\text{cov}(y_i, y_j)$ represent covariance of i-th column and j-th column.

- Next, we use Eigen vectors and values to find principal components of a dataset. Principal components show direction of data that specify maximum amount of variance is line that capture most information present in the data.

To put it simply, principal axis are new axis along which we will study the data.

$A\mathbf{x} = \lambda\mathbf{x}$

- We will find eigenvalues and sort them in decreasing order.
- Calculate variance percentage for each column which will be:

$$\text{(corresponding eigenvalue)} \times \frac{100}{\left(\sum \text{ of all eigenvalues}\right)}$$

- Cumulative variance is also calculated: $\sum_{i=1}^{x} \text{Variance Percentage}_i$

If number of principal axis is given, we will directly use it, otherwise we will select a x such that cumulative variance is more than or equal to 85%.

- Projecting data into principal components

- Select number of principal components as above (let k)

- $X_{\text{reduced}} = X \cdot W_k$

Where W_k is a matrix containing respective k Eigen vectors

This will be our reduced dataset

When does global structure fails?

We propose to test how similar our co-relation matrix is with identity matrix. If it resembles it, global structure is bound to fail, as it signifies there will be little to no co-relation among different features

Null Hypothesis (H_0): Variance of k variables are equal (Thus no-covariance)

Alternate Hypothesis (H_1): At least one of k variable have unequal variance.

After trying out different functions and also Likelihood ratio test, we found out a test statistic which resembles chi-distribution with k-1 degree of freedom, when null hypothesis follows.

$$\chi^2 = \frac{(n-1)\ln(S^2) - \sum_{i=1}^{k}(n_i - 1)\ln(S_i^2)}{1 + \frac{1}{3(k-1)}\left(\sum_{i=1}^{k}\frac{1}{n_i - 1} - \frac{1}{n-k}\right)}$$

n : Total number of observations

k : Number of groups

n_i : Number of observations in the i-th group

X_{ij} : Observation in the i-th group, j-th sample

Compare the calculated test statistic to the critical value from the chi-squared distribution table*. If the test statistic is greater than the critical value, you reject the null hypothesis.

However, this failed when data was not normal. We took an inherent assumption of normality of data. So for non-normal data we further extended our test.

The only difference being a different test statistic:

$$W = \frac{(N-k)}{(k-1)} \frac{\sum_{i=1}^{k} n_i(\bar{Z}_i - \bar{Z})^2}{\sum_{i=1}^{k} \sum_{j=1}^{n_i} (Z_{ij} - \bar{Z}_i)^2}$$

N is the total number of observations.

k is the number of groups.

n_i is the number of observations in the i-th group.

Z_{ij} is the j-th observation in the i-th group.

$\bar{Z}_i$ is the mean of the i-th group.

$\bar{Z}$ is the overall mean.

Compare the calculated test statistic to the critical value from the F-distribution table**. If the test statistic is greater than the critical value, you reject the null hypothesis, suggesting that at least one group has a different variance. If it's smaller, you fail to reject the null hypothesis, indicating that there's not enough evidence to suggest unequal variances.

Local Outlier Factor(LOF)

LOF is a technique used for anomaly detection within a dataset. It works on outlier detection.

In this method, we calculate local density of a point with respect to its neighbours. Points with substantially lower density that the neighbours are considered outliers.

LOF Algorithm

1) Local Reachability Density (LRD) :

First, for each data point in dataset, LRD is calculated . The metric represents how densely packed the data points are around the point in question:

$$\text{Average RD}_A = \frac{1}{k}\sum_{i=1}^{k}\left(\text{distance}\left(A, \text{kth neighbor}_i\right)\right)$$

LRDA = 1/RDA

LRD is just the reciprocal of RD(Reachability Distance).

2) LocalOutlierFactor(LOF)

LOF is like a measure of outlierness of data points with respect to local neighbourhood . It compares LRD of the points to the neighbours LRD. A point with a significantly lower LRD compared to its neighbourhood, will have higher LOF , indicating it to be an outlier.

FORMULA:

$$\text{LOFA}_A = \frac{1}{k}\sum_{i=1}^{k}\frac{\text{LRD(kth neighbor}_i)}{\text{LRD}(A)}$$

3) Identification of Outliers:

Using the value of LOF , we have judge if a point is an outlier or a possible anomaly . Higher LOF value indicates point to be located in a sparser region of data space compared to neighbour suggesting that it is an outlier.

USES:

LOF algorithm is used in various domains like:

o Fraud detection

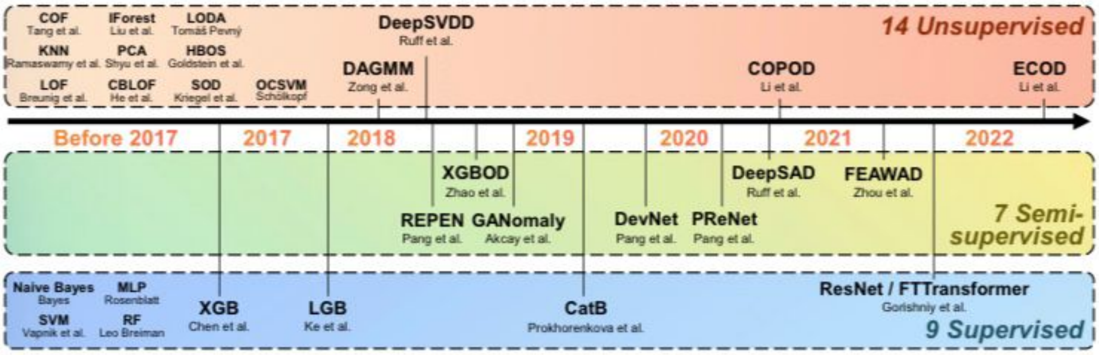
o Network security

o Industrial system

o Cyber data protection Preservation of local structures

Capturing and preserving local structure, such as neighbourhood relationships or local clusters of PCA / LDA especially for tasks like anomaly detection.

Improvisation of Local methods:



These are existing models currently to deal with local structures. Our proposal is to classify datasets in categories and then apply more than 1 models for specific categories of dataset to get improved results.

We decided to take one model from each timeframe to work on.

Before 2017: PCA and LOF

2017: OCSVM

2018: LGB

2019: XGBOD

2020: CatB

2021: DeepSAD

2022: ECOD

As we see the latest model ECOD (Ensemble of collaborative outlier detection) is a similar approach. We have tried to extend it here. We tried to apply it on different datasets*.

Summary of Technical Results:

Algorithm Robustness Under Noisy and Corrupted Data:

1. Duplicated Anomalies:

- **Unsupervised Methods:** Median ΔAUCROC of -16.43% observed when anomalies are duplicated six times.
- **Semi-Supervised Methods:** Showed significantly less degradation with a median ΔAUCROC of -0.05%.
- **Supervised Methods**: Performed even better with a median ΔAUCROC of 0.13% under the same conditions.

Insight: The imbalance assumption of anomalies in unsupervised methods becomes violated with increased duplication, contributing to their degradation.

2. Irrelevant Features:

- **Supervised Methods:**

A. Demonstrated resilience due to the feature selection process guided by data labels.

B. Even with 50% of input features corrupted by uniform noise, the worst-performing supervised algorithm showed ≤ 5% degradation.

C. In contrast, un- and semi-supervised methods faced up to 10% degradation.

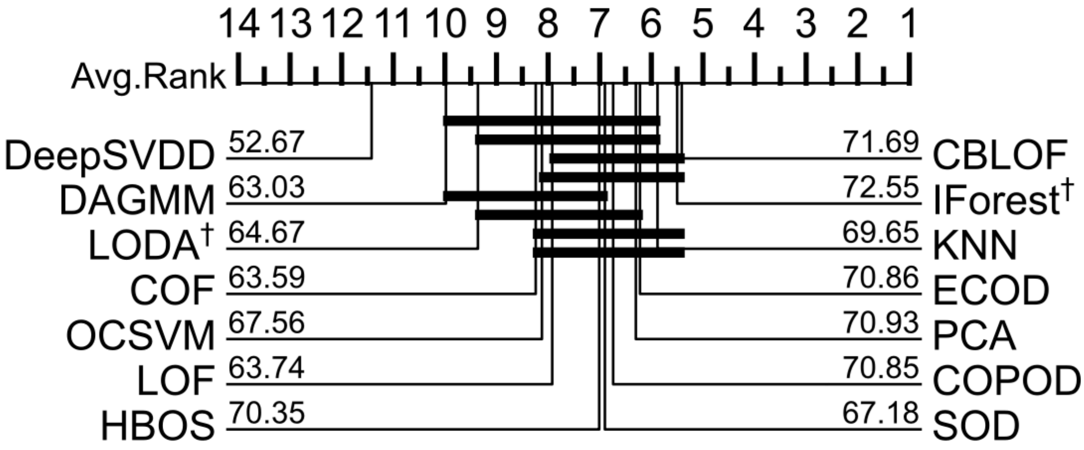
Insight: Supervised methods effectively filter out irrelevant features, showcasing the importance of label-guided training.

3. Annotation Errors:

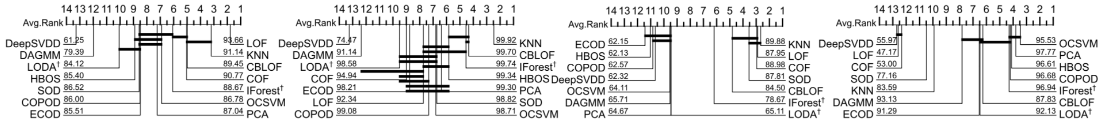
Both semi- and fully-supervised methods exhibited great resilience to minor annotation errors (5%) with median ΔAUCROC of -1.52% and -1.91%, respectively.

Severe annotation errors significantly degraded detection performance.

Ranking of different models(Overall):



Ranking under different anomalies:



(a) Local anomalies (b) Global anomalies (c) Dependency anomalies (d) Clustered anomalies

1. Unsupervised Methods: The results indicate that no unsupervised algorithm statistically outperforms the rest. This suggests that the choice of unsupervised anomaly detection algorithm does not have a clear winner, and different algorithms perform differently across datasets.

2. Semi-supervised vs. Fully-supervised: Semi-supervised methods outperform fully-supervised methods when the amount of labeled data is limited (γl ≤ 5%). In these cases, the detection performance of semi-supervised methods is generally better than that of fully-supervised algorithms.

3. DL-based Unsupervised Methods: Surprisingly, some deep learning-based unsupervised methods like DeepSVDD and DAGMM perform worse than shallower methods. The authors attribute this to the difficulty of training deep models without the guidance of label information and the challenges associated with tuning more hyper parameters.
4. Ensemble Methods: Newer network architectures like Transformer and ensemble methods (e.g., XGBoost and CatBoost) show competitive performance in anomaly detection. These methods, although not specifically designed for anomaly detection, provide satisfying detection performance, especially when the labeled anomaly ratio (γl) is low.

Algorithm Performance under Different Types of Anomalies:

Performance on Different Types of Anomalies: The results show that the performance of unsupervised algorithms depends on the alignment of their assumptions with the type of anomaly present in the data. For example, LOF performs well for local anomalies, while KNN is better for global anomalies.

Label-Informed Methods: Surprisingly, label-informed methods, both semi-supervised and fully-supervised, do not consistently outperform the best unsupervised methods, especially when there are incomplete labels. This suggests that incomplete label information can bias the learning process and lead to inferior performance for label-informed methods.

Prior Knowledge of Anomaly Types: The results emphasize the importance of understanding and leveraging knowledge about the type of anomalies in achieving high detection performance, even without labeled data. The authors suggest the need for designing anomaly-type-aware detection algorithms.

Conclusion:

- Understanding algorithm behaviour under diverse scenarios is crucial for practical application.
- Incorporating label information is beneficial, but careful consideration of the type and quality of labels is essential.
- Future directions include the improvement of robust unsupervised anomaly detection algorithms and the development of anomaly-type-aware detection techniques.

This study not only contributes to the understanding of anomaly detection algorithm behaviour but also provides valuable insights for refining existing models and paving the way for future advancements in this field.

*Datasets:

Primary Dataset: UNSW-NB15 Dataset
Link: https://research.unsw.edu.au/projects/unsw-nb15-dataset
Description: A dataset designed for network intrusion detection, providing labeled data for various types of network attacks. It includes a variety of features related to network traffic, making it suitable for evaluating intrusion detection systems.

Others:
KDD Cup 1999 Data (Intrusion Detection):
Link: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
Description: A widely used dataset for intrusion detection in network security.

Credit Card Fraud Detection:
Link: https://www.kaggle.com/mlg-ulb/creditcardfraud
Description: Anonymized credit card transactions labeled as fraudulent or genuine.

NASA Prognostics Data:
Link: https://www.nasa.gov/content/prognostics-center-of-excellence-data-set-shifts-to-public
Description: Time-series data from aircraft engines, used for predicting failures.

Thyroid Disease Dataset:
Link: https://archive.ics.uci.edu/ml/datasets/Thyroid+Disease
Description: Medical dataset for detecting thyroid diseases based on patient data.

Forest Cover Type Dataset:
Link: https://archive.ics.uci.edu/ml/datasets/Covertype
Description: Dataset for predicting forest cover types based on cartographic variables.

Numenta Anomaly Benchmark (NAB):
Link: https://www.kaggle.com/boltzmannbrain/nab
Description: Benchmark dataset with real-world anomalies, designed for evaluating anomaly detection algorithms.

IoT Sensor Data for Predictive Maintenance:
Link: https://www.kaggle.com/billstuart/predictive-maintenance
Description: Sensor data from IoT devices for predicting maintenance needs.

Shuttle Landing Control Dataset:
Link: https://archive.ics.uci.edu/ml/datasets/Statlog+(Shuttle)
Description: Dataset for classifying anomalies in shuttle landing control.

Covertype (Multiclass Version):
Link: https://archive.ics.uci.edu/ml/datasets/Covertype
Description: Multiclass version of the forest cover type dataset.

EEG Eye State Dataset:
Link: https://archive.ics.uci.edu/ml/datasets/EEG+Eye+State
Description: EEG data for predicting eye state (open/closed), useful for anomaly detection.