On Evaluating Machine Learning Models for Anomaly Detection

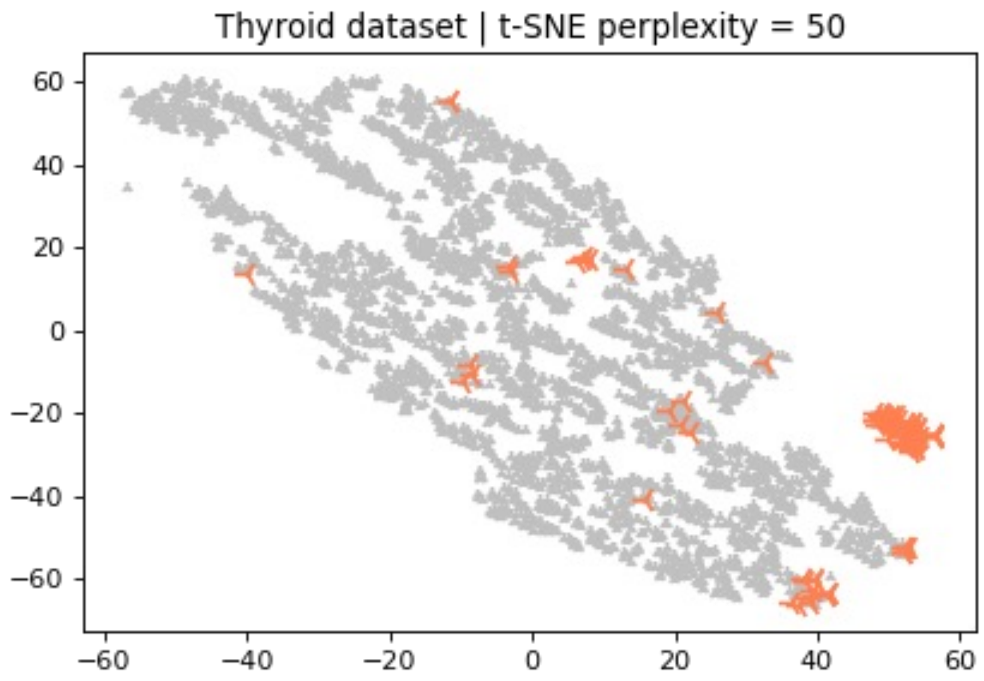
Presented by :
D'Jeff Kanda Nkashama
https://djeffkanda.github.io/

Université de Sherbrooke

What is anomaly detection?

- Anomaly detection is identifying observations that deviate from what is deemed normal observations.

- Depending on the situation, such an observation is considered unusual, irregular, atypical, inconsistent, unexpected, rare, erroneous, faulty, fraudulent, malicious,



Anomaly detection algorithms

- Families of anomaly detection algorithms:
  - Distance-based methods (LOF)
  - Methods learning decision boundaries (SVM)
  - Probabilistic methods (GMM)
  - Reconstruction-based methods (Autoencoders)

- Algorithms output a score, then a threshold is used on this score to determine whether the sample is an anomaly

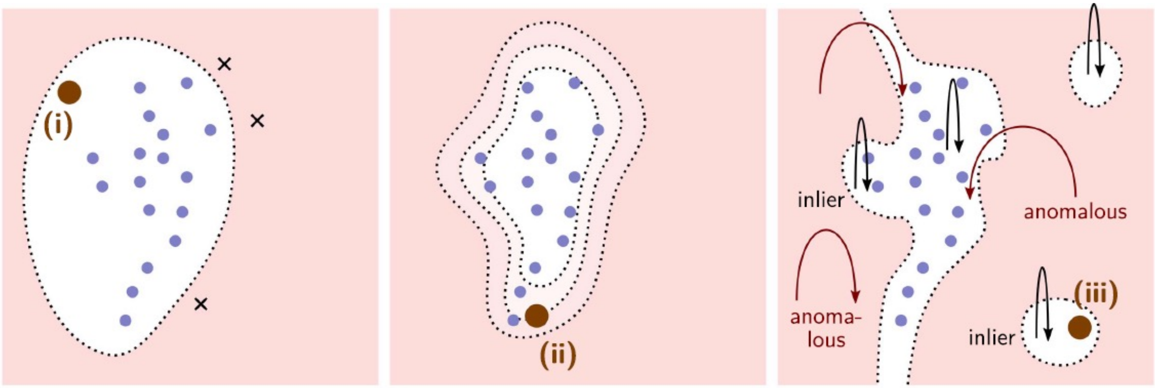


Figure 5 of *A Unifying Review of Deep and Shallow Anomaly Detection*, Ruff et al. 2021.

Problems with current evaluation protocols

- The evaluation protocols used in the literature are inconsistent
- Makes comparisons of reported results difficult to interpret from paper to paper

Method	KDDCUP			Thyroid		
	Precision	Recall	F_1	Precision	Recall	F_1
OC-SVM	0.7457	0.8523	0.7954	0.3639	0.4239	0.3887
DSEBM-r	0.1972	0.2001	0.1987	0.0404	0.0403	0.0403
DSEBM-e	0.7369	0.7477	0.7423	0.1319	0.1319	0.1319
DCN	0.7696	0.7829	0.7762	0.3319	0.3196	0.3251
GMM-EN	0.1932	0.1967	0.1949	0.0213	0.0227	0.0220
PAE	0.7276	0.7397	0.7336	0.1894	0.2062	0.1971
E2E-AE	0.0024	0.0025	0.0024	0.1064	0.1316	0.1176
PAE-GMM-EM	0.7183	0.7311	0.7246	0.4745	0.4538	0.4635
PAE-GMM	0.7251	0.7384	0.7317	0.4532	**0.4881**	0.4688
DAGMM-p	0.7579	0.7710	0.7644	0.4723	0.4725	0.4713
DAGMM-NVI	0.9290	**0.9447**	0.9368	0.4383	0.4587	0.4470
DAGMM	**0.9297**	0.9442	**0.9369**	**0.4766**	0.4834	**0.4782**

Table 2 of the DAGMM paper (Zong et al., 2018).

Method	KDD99			Thyroid		
	Presion	**Recall**	F_1	**Presion**	**Recall**	F_1
PCA	0.8312	0.6266	0.7093	0.9258	0.7322	0.8089
Kernel PCA	**0.8627**	0.6319	0.7352	0.9537	0.7493	0.8402
KDE	0.8119	0.6133	0.6975	0.9275	0.7129	0.7881
RKDE	0.8596	0.6328	0.7322	0.9437	0.7538	0.8429
OC-SVM	0.8050	**0.6512**	0.7113	**0.9602**	0.7424	**0.8481**
AEOD	0.7624	0.6218	0.6885	0.9157	0.6927	0.7873
DSEBM-r	0.8521	0.6472	0.7328	0.9527	0.7479	0.8386
DSEBM-e	0.8619	0.6446	**0.7399**	0.9558	**0.7642**	0.8375

Table 2 of the DSEBM paper (Zhai et al., 2016).

Université de Sherbrooke

Inconsistencies in data split and threshold choice

Because we train on normal data only, we are left with anomalies after splitting the dataset in a training and test set. We spotted three strategies:

- Discarding the anomalies found in the training set. DSEBM (Zhai et al., 2016)
- Injecting all the anomalies in the test set. DAGMM (Zong et al., 2018), ALAD (Zenati et al., 2018)
- Making the test set balanced (by putting as many normal samples as anomalous samples). DROCC (Goyal et al., 2020)

Two strategies to set the threshold:

- Fixing the threshold based on the anomaly ratio DAGMM, ALAD
- Looking for the optimal threshold NeuTraL AD (Qiu et al., 2021)

Inconsistencies in reported metrics

Which metrics do we use and how do we set the threshold for classification?

- Accuracy, Precision, Recall, F1-score, Area Under the Receiver-Operating Curve (AUROC), Area Under the Precision-Recall curve (AUPR)

What is the class of interest?

- Changes the anomaly ratio by reversing the balance of the dataset

Method \ Metric	Accuracy	Recall	Precision	F1-Score	AUPR	AUROC
ALAD	x	~	~	~	x	~
DAGMM	x	O	O	O	x	x
DeepSVDD	x	x	x	x	x	O
DROCC	x	x	x	~	x	~
DSEBM	x	O	O	O	x	x
DUAD	x	x	x	x	~	O
GOAD	x	x	x	~	x	~
LOF	x	x	x	x	x	x
MemAE	x	~	~	~	x	~
OC-SVM	x	x	x	x	x	x
RecForest	x	x	O	x	x	O
SOM-DAGMM	O	O	O	O	x	x
NeuTral-AD	x	~	~	~	x	~

Reported metrics in various papers.

Proposed evaluation protocol

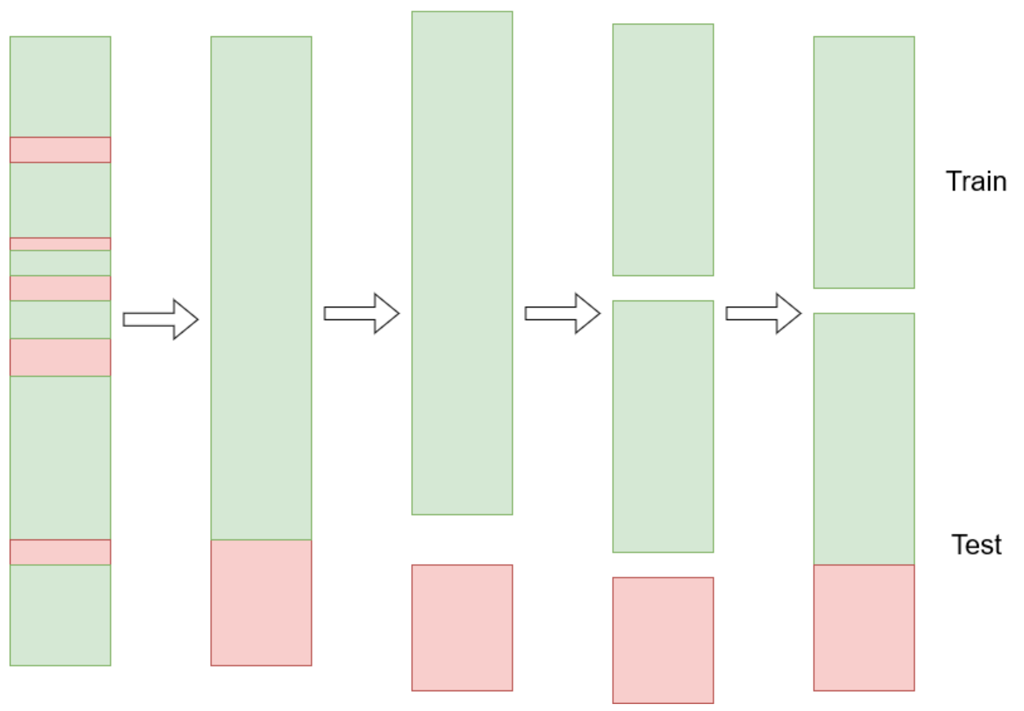
The evaluation protocol we propose fixes the following issues:

- How to choose the class of interest
- How to split the dataset
- Which metrics to report
- How to set the threshold

A Revealing Large-Scale Evaluation of Unsupervised Anomaly Detection Algorithms
Maxime Alvarez*, Jean-Charles Verdier*, D'Jeff K Nkashama*, and 3 more authors
In Workshop on Setting up ML Evaluation Standards to Accelerate Progress, International conference on learning representations 2022

Class of interest & Data split

- Use the minority class as the class of interest
- Split the normal data 50/50 and put all the anomalies in the test set



Threshold & Reported metrics

- Choose the optimal threshold
- Report F1-score, Precision, Recall, and AUPR
  - Precision and Recall, together, allow us to consider the imbalance of the dataset
  - AUPR allows us to evaluate the performance independently of the threshold
- AUROC considered too optimistic for unbalanced datasets

	KDD10	
	AUROC	AUPR
DAE	0.982(0.000)	0.947(0.001)
DAGMM	0.991(0.003)	**0.973(0.006)**
SOM-DAGMM	0.989(0.002)	0.958(0.013)
DUAD	0.983(0.010)	0.932(0.035)
MemAE	0.982(0.002)	0.947(0.006)
DeepSVDD	**0.994(0.002)**	0.971(0.010)
DROCC	0.975(0.000)	0.932(0.000)
DSEBM-e	0.986(0.001)	0.939(0.005)
DSEBM-r	0.990(0.000)	0.956(0.002)
ALAD	0.990(0.002)	0.953(0.011)
NeuTraLAD	0.988(0.001)	0.970(0.001)
OC-SVM	0.988(0.000)	0.949(0.000)
LOF	0.911(0.000)	0.899(0.000)

Anomaly detection datasets

- **Tabular data**, time series, images
- Anomaly detection datasets are often **imbalanced**
- We train unsupervised algorithms on normal data only
  - We may want to train on normal data contaminated with anomalies to test the robustness of the algorithm

Dataset	Number of samples (N)	Number of features (D)	Anomaly ratio (ρ)
Arrhythmia	452	274	0.1460
CSE-CIC-IDS2018	16 232 944	83	0.1693
KDD 10%	494 021	42	0.1969
NSL-KDD	148 517	42	0.4811
Thyroid	3772	6	0.0246

Table 1. General information on the datasets.

Experiments

- 12+ unsupervised anomaly detection algorithms
- 5+ tabular datasets from cybersecurity and medical domains
- All evaluated following the proposed evaluation protocol
- Used the hyperparameters from the original paper when available
- Goal: To give a more accurate picture of the relative performances of these algorithms!

	KDDCUP 10		
	Precision	Recall	F_1
DAE	0.932(0.013)	0.932(0.026)	0.932(0.020)
DAGMM	0.936(0.009)	0.984(0.019)	0.959(0.014)
SOM-DAGMM	0.957(0.007)	0.998(0.002)	0.977(0.003)
DUAD	0.940(0.007)	0.991(0.014)	0.965(0.010)
MemAE	0.930(0.012)	0.971(0.022)	0.950(0.017)
DeepSVDD	0.908(0.02)	0.876(0.02)	0.891(0.02)
DROCC	0.840(0.000)	0.996(0.000)	0.911(0.000)
DSEBM-e	0.957(0.001)	0.976(0.001)	0.966(0.001)
DSEBM-r	**0.966(0.001)**	0.994(0.001)	**0.980(0.001)**
ALAD	0.951(0.005)	0.966(0.010)	0.959(0.007)
NeuTraLAD	0.931(0.003)	**0.997(0.001)**	0.964(0.002)
OC-SVM	0.942(0.000)	0.994(0.000)	0.967(0.000)
LOF	0.930(0.000)	0.972(0.000)	0.951(0.000)

	KDD10	
	AUROC	AUPR
DAE	0.982(0.000)	0.947(0.001)
DAGMM	0.991(0.003)	**0.973(0.006)**
SOM-DAGMM	0.989(0.002)	0.958(0.013)
DUAD	0.983(0.010)	0.932(0.035)
MemAE	0.982(0.002)	0.947(0.006)
DeepSVDD	**0.994(0.002)**	0.971(0.010)
DROCC	0.975(0.000)	0.932(0.000)
DSEBM-e	0.986(0.001)	0.939(0.005)
DSEBM-r	0.990(0.000)	0.956(0.002)
ALAD	0.990(0.002)	0.953(0.011)
NeuTraLAD	0.988(0.001)	0.970(0.001)
OC-SVM	0.988(0.000)	0.949(0.000)
LOF	0.911(0.000)	0.899(0.000)

Lessons Learned

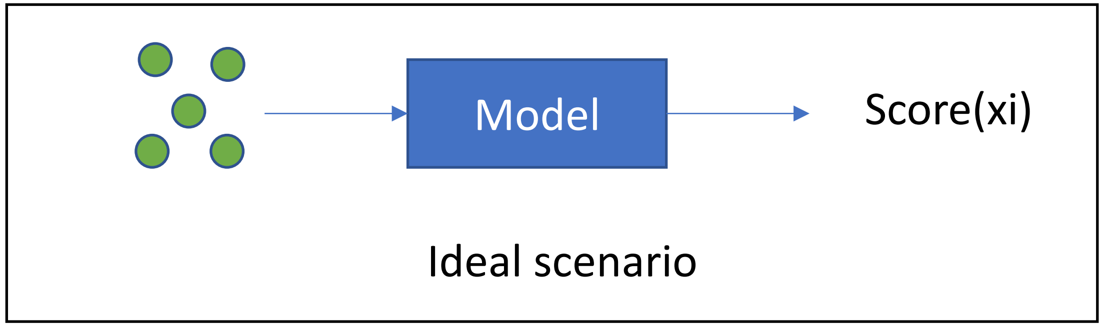
- The relative performance of algorithms in the literature is not the same as the one we get when we use our consistent evaluation protocol

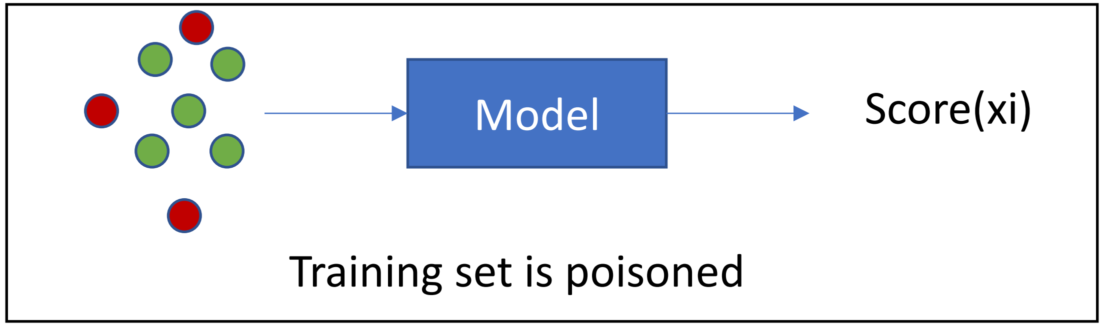
- Our vanilla auto-encoder DAE outperforms more sophisticated reconstruction-based methods like DAGMM and MemAE on CIC-IDS2018
- Baseline methods with optimized hyper-parameters achieve more competitive F1-scores than reported in the literature so far
- NeuTraLAD, the transformation based approach, offers consistently above-average performance across all datasets

- Taking the majority class as the class of interest gives overly optimistic results
- AUPR is more informative than AUROC on unbalanced datasets

Models' Robustness

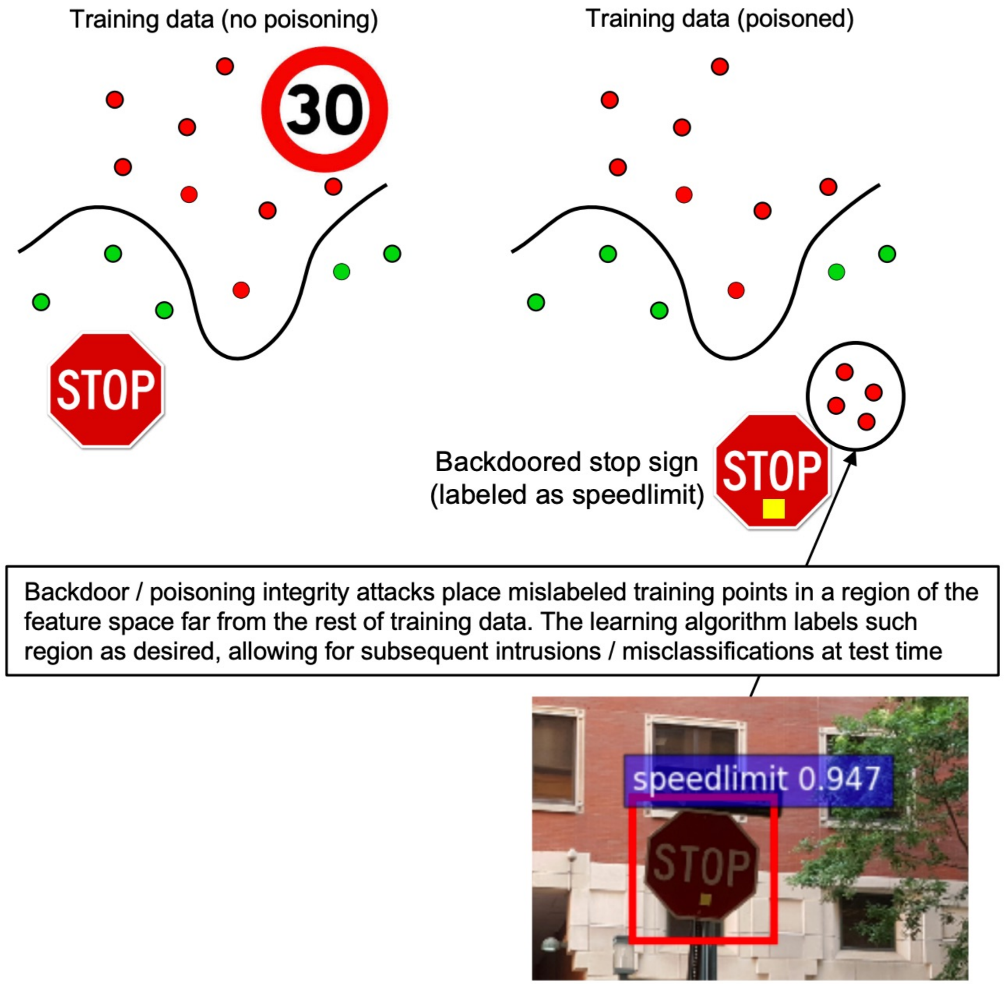
- AD models assume data is clean
- Problem : Data can be contaminated in real-world





Université de Sherbrooke

Models' Robustness

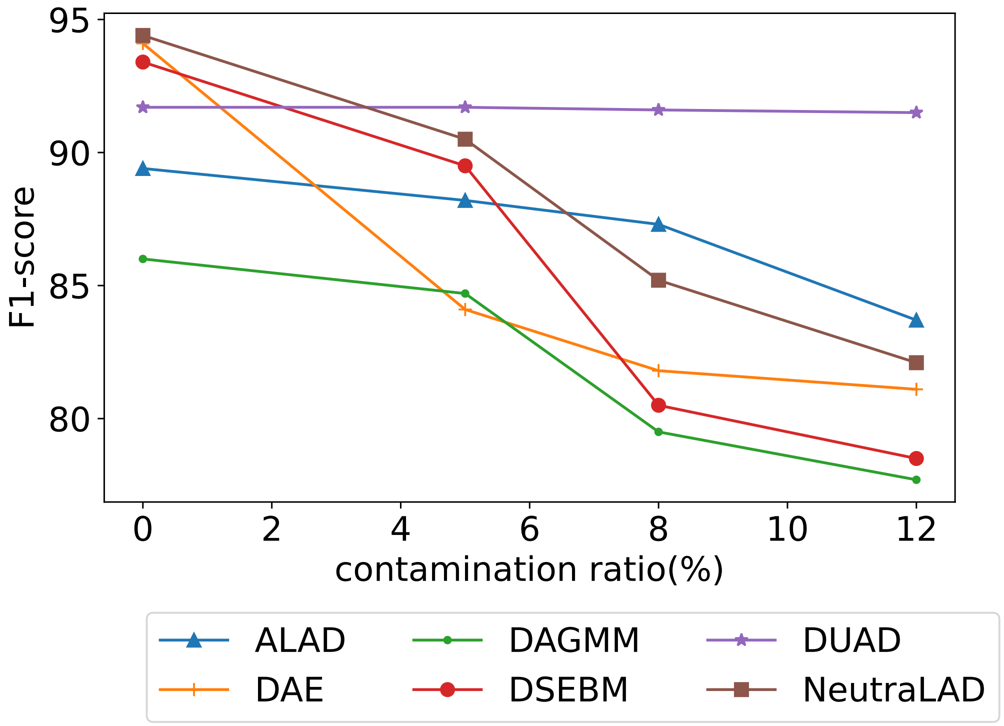


Attacks against ML Models

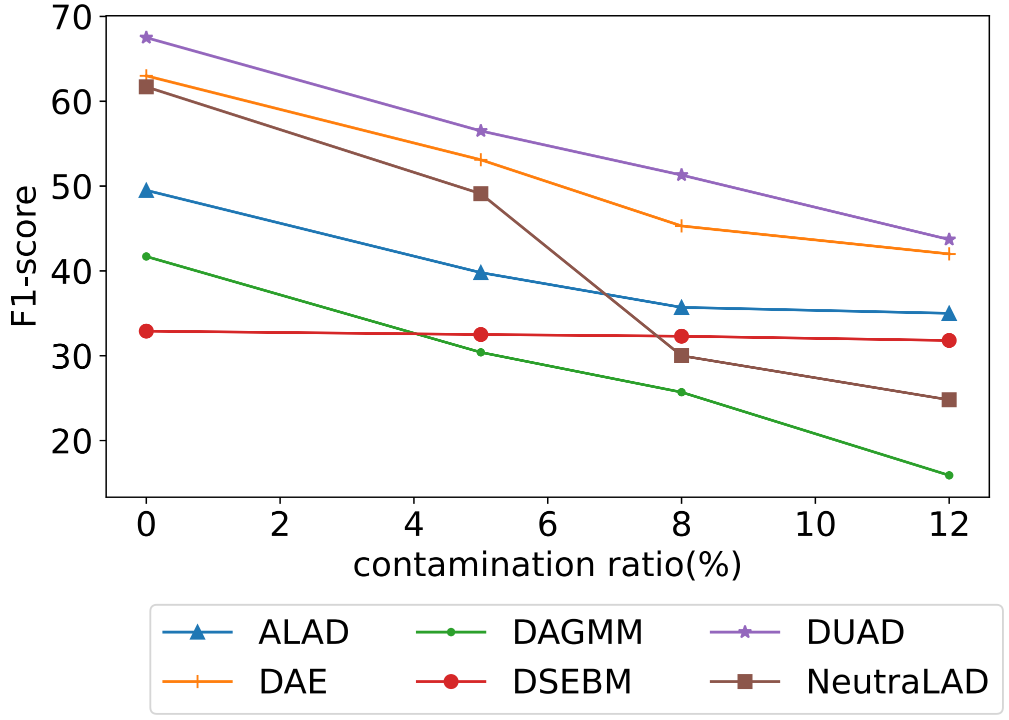
Attacker's Goal	Misclassifications that do not compromise normal system operation	Misclassifications that compromise normal system operation	Querying strategies that reveal confidential information on the learning model or its users
Attacker's Capability	**Integrity**	**Availability**	**Privacy / Confidentiality**
Test data	Evasion (a.k.a. adversarial examples)	-	Model extraction / stealing and model inversion (a.k.a. hill-climbing attacks)
Training data	Poisoning (to allow subsequent intrusions) – e.g., backdoors or neural network trojans	Poisoning (to maximize classification error)	-

https://www.sciencedirect.com/science/article/pii/S0031320318302565?casa_token=HaMtrIpmJYsAAAAA:39tZI6rZ29n2nKQ0t-SLT4ByTTYhZOCS2oB354wOzM6zsqGo2ss9oCjCk19PN7De8ZHHWuQNbkc

Robustness evaluation



NSL-KDD



CIC-IDS

On Evaluating the Robustness of Deep Unsupervised Learning Methods for Network Intrusion Detection
D'Jeff K Nkashama, Soltani Arian, Jean-Charles Verdier, and 3 more authors
In Workshop on Machine Learning for Cybersecurity, International Conference on Machine Learning 2022

Paper : https://arxiv.org/pdf/2207.03576.pdf
Code : https://github.com/intrudetection/robevalanodetect

Université de Sherbrooke

Conclusion

- A consistent evaluation protocol as a basis to compare unsupervised anomaly detection algorithms
- Updated and more precise picture of the relative performance of twelve methods on five widely used tabular datasets

	KDDCUP 10		
	Precision	Recall	F_1
DAE	0.932(0.013)	0.932(0.026)	0.932(0.020)
DAGMM	0.936(0.009)	0.984(0.019)	0.959(0.014)
SOM-DAGMM	0.957(0.007)	0.998(0.002)	0.977(0.003)
DUAD	0.940(0.007)	0.991(0.014)	0.965(0.010)
MemAE	0.930(0.012)	0.971(0.022)	0.950(0.017)
DeepSVDD	0.908(0.02)	0.876(0.02)	0.891(0.02)
DROCC	0.840(0.000)	0.996(0.000)	0.911(0.000)
DSEBM-e	0.957(0.001)	0.976(0.001)	0.966(0.001)
DSEBM-r	**0.966(0.001)**	0.994(0.001)	**0.980(0.001)**
ALAD	0.951(0.005)	0.966(0.010)	0.959(0.007)
NeuTraLAD	0.931(0.003)	**0.997(0.001)**	0.964(0.002)
OC-SVM	0.942(0.000)	0.994(0.000)	0.967(0.000)
LOF	0.930(0.000)	0.972(0.000)	0.951(0.000)

Your new algorithm → ?

Université de Sherbrooke