Sound Anomaly Detection

Alina Golkina

MSAI MIPT, Evocargo

2023

girafe
ai

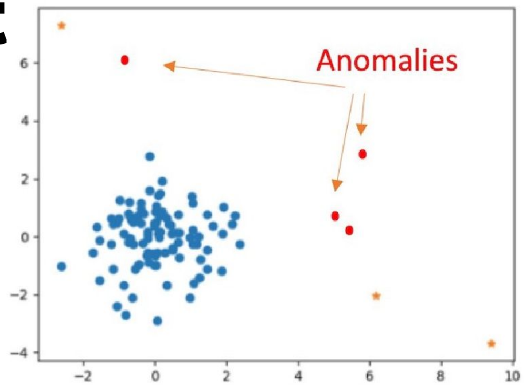
Outline

1. Problem Statement
    a. Application Domains
    b. Types of anomalies
2. Datasets
3. Sound feature extraction
4. Models
    a. Classical Key Algorithms
    b. Deep Anomaly Detection
5. Current results

Problem Statement

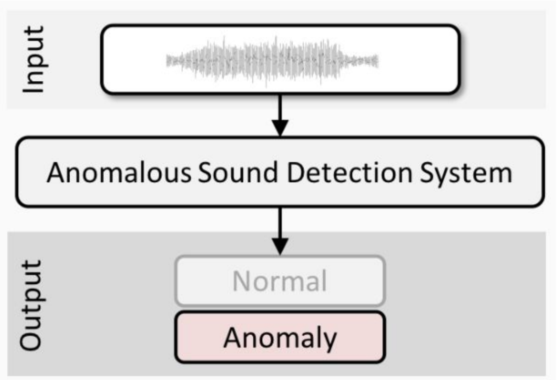
girafe
ai

General problem statement



Anomaly detection is finding patterns in data that do not conform to expected behavior

Sound anomaly detection in electrical engines



detect abnormal engine operation. Anything that deviates from the main mode is an anomaly, if it is necessary to detect it, it is possible to categorize these deviations.

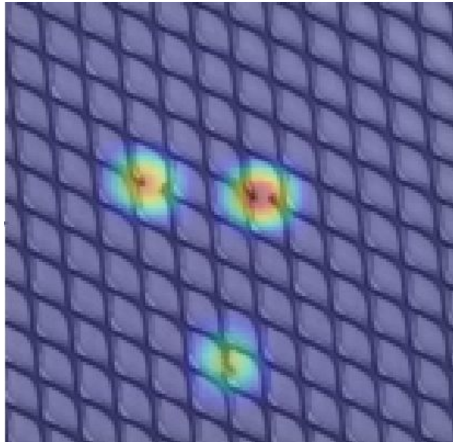
girafe ai

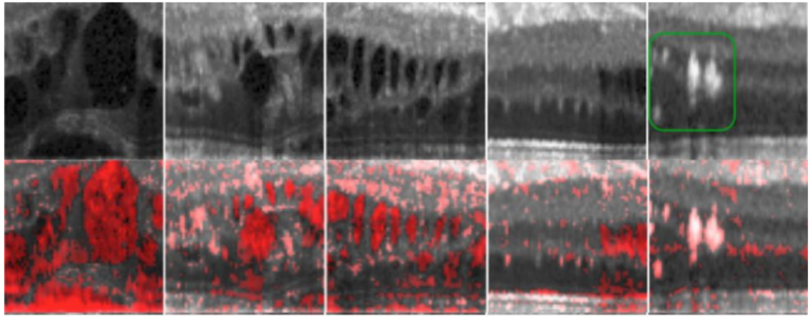
Evocargo electric engines

Self driving cars powered by own electric and hybrid (hydrogen electric) engines, so we need to monitor functioning state automatically 24/7

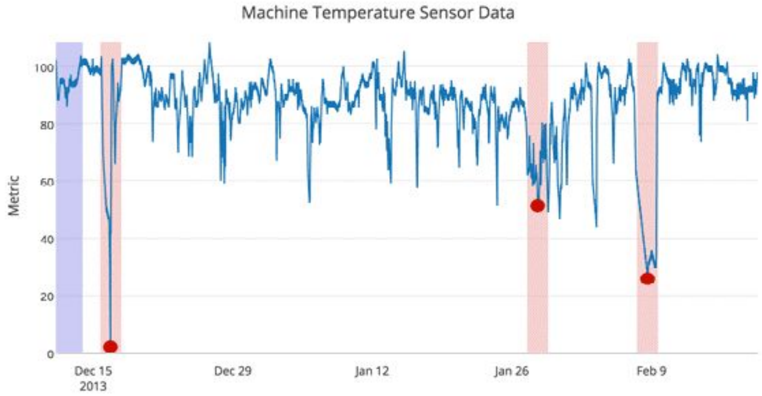
Application Domains

- Intrusion Detection
- Fraud Detection
- Industrial Damage Detection
- Medical and Health Anomaly Detection
- etc...





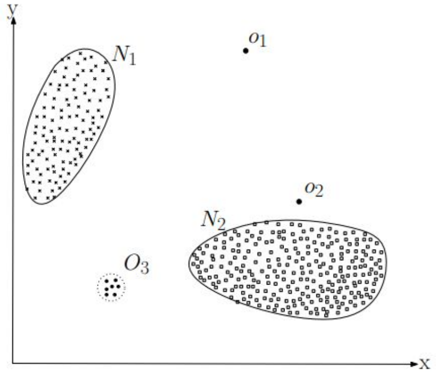
Detecting Retinal Damage



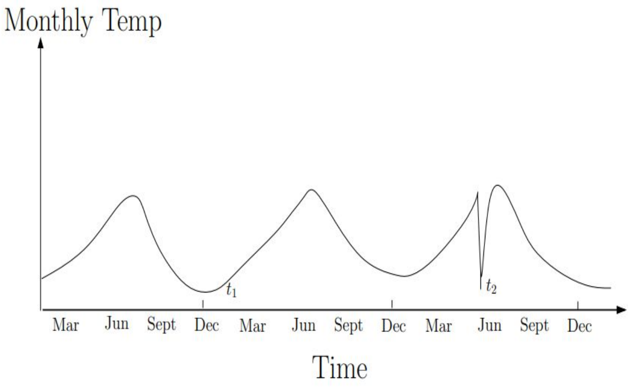
girafe ai

Types of Anomalies

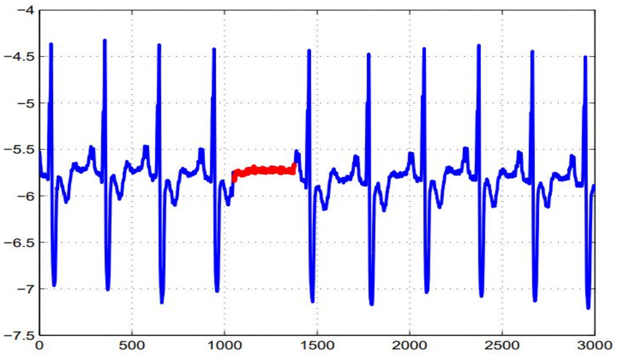
Point anomalies



Contextual anomalies

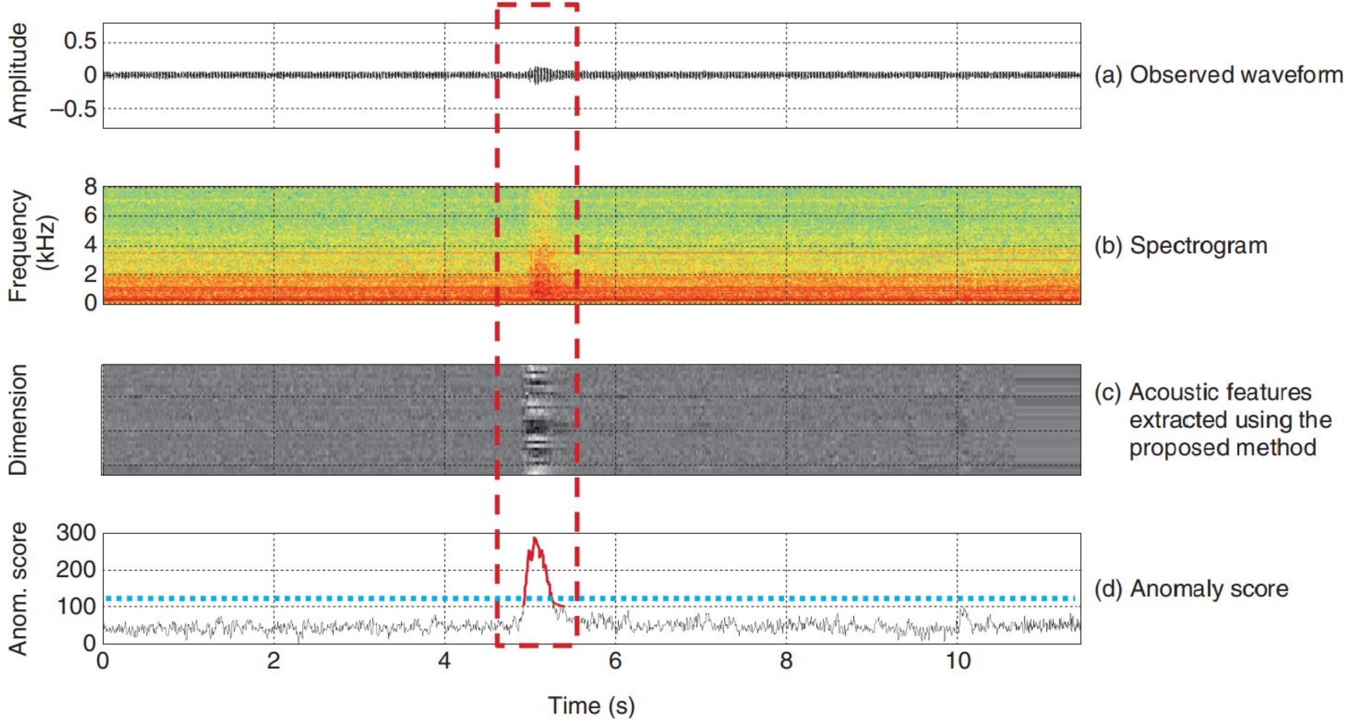


Collective anomalies



In this problem we reduce anomalies to point type

Sound Anomaly Detection

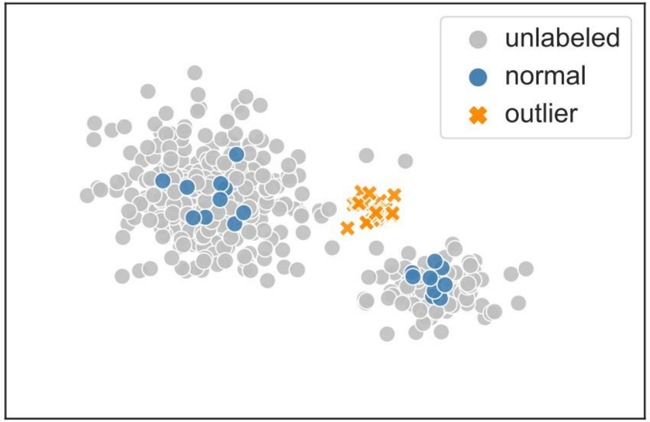


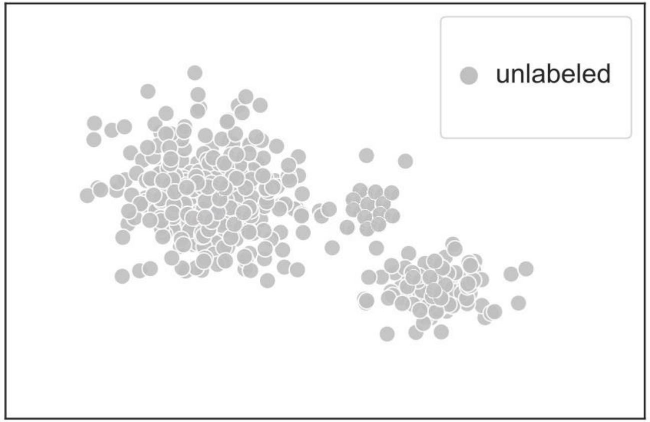
H. Uematsu, Y. Koizumi, S. Saito, A. Nakagawa, N. Harada “Anomaly Detection Technique in Sound to Detect Faulty Equipment”
https://www.ntt-review.jp/archive/ntttechnical.php?contents=ntr201708fa5.html

girafe ai

Semi-supervised and Unsupervised Anomaly Detection

Semi-supervised anomaly detection - the training data consists of mostly normal unlabeled data (gray) as well as a few labeled normal samples (blue) and labeled anomalies (orange).





Deep semi-supervised anomaly detection, L Ruff, RA Vandermeulen, N Görnitz, A Binder, ICLR 2020
https://arxiv.org/pdf/1906.02694.pdf

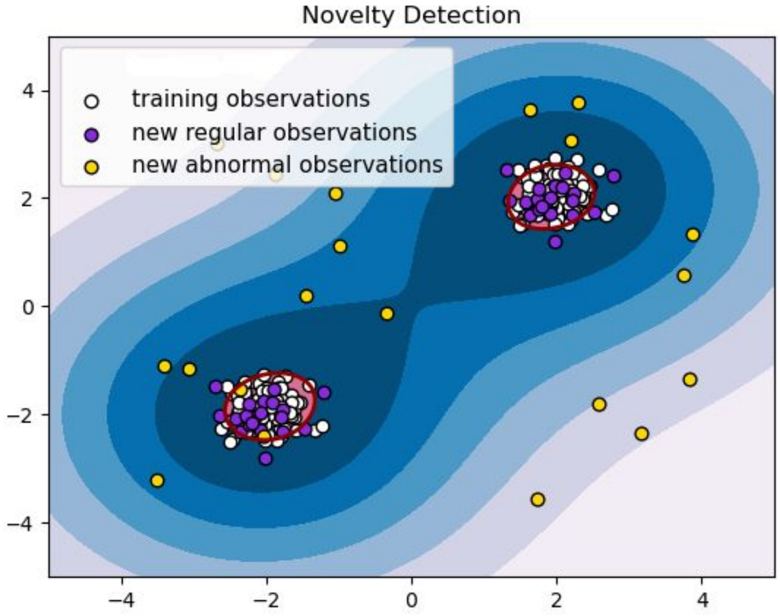
Anomaly detection vs Novelty detection

Anomaly detection (Outlier detection):

dataset may already have outliers , goal - to identify such outliers.

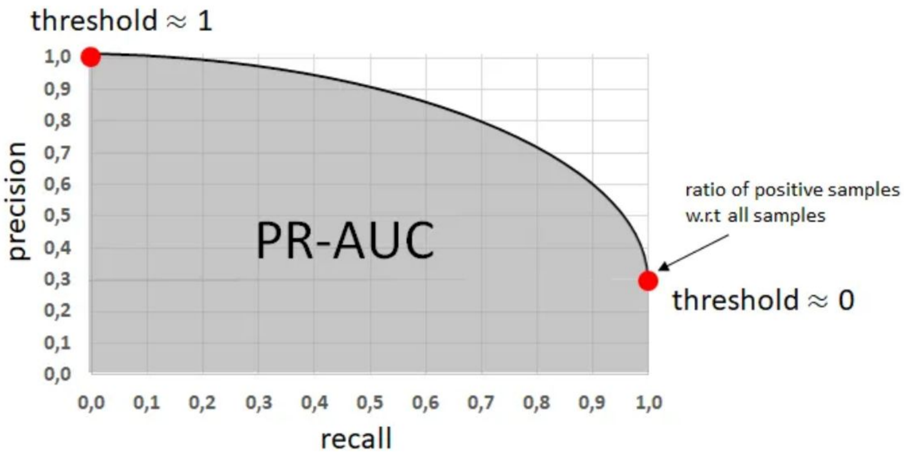
Novelty detection:

dataset contains only normal observations, goal - to check if new observations are outliers.



Metrics

- $Recall = TPR\ (True\ Positive\ Rate) = \frac{TP}{(TP+FN)}$
- $Precision = \frac{TP}{(TP+FP)}$
- $MAR\ (Missed\ Alarm\ Rate) = 1 - TPR$
- $FAR\ (False\ Alarm\ Rate) = \frac{FP}{(TN+FP)}$
- $F1 = 2\frac{Recall \cdot Precision}{Recall + Precision}$
- $ROC\text{–}AUC\ (Area\ Under\ ROC\ Curve)$
- $PR\text{–}AUC\ (Average\ Precision,\ Area\ Under\ The\ Precision\text{–}Recall\ curve)$

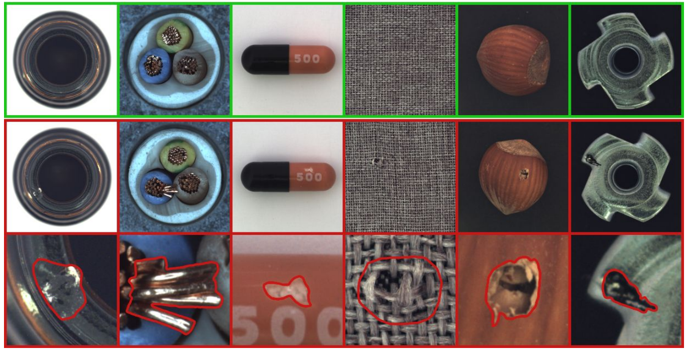


Datasets

girafe
ai

Datasets for General Anomaly Detection

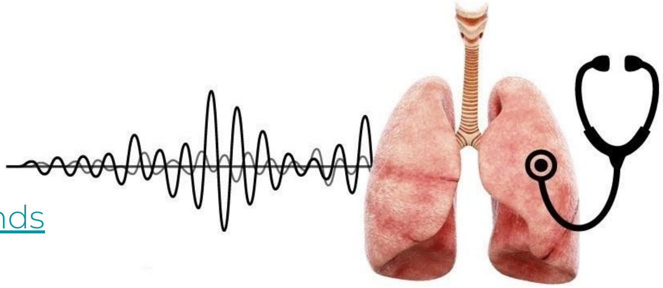
- ODDS (Outlier Detection Datasets) (http://odds.cs.stonybrook.edu):
  - Multi-dimensional point datasets
  - Time series graph datasets for event detection
  - Time series point datasets (Multivariate/Univariate)
  - Adversarial/Attack scenario and security datasets
  - Crowded scene video data for anomaly detection
- ADBench: Anomaly Detection Benchmark, https://github.com/Minqi824/ADBench
  - Classical tabular benchmark datasets
  - Adapted CV and NLP Datasets for Tabular AD
- Numenta Anomaly Benchmark (NAB), https://github.com/numenta/NAB/tree/master/data
- MVTecAD (MVTEC Anomaly Detection Dataset), https://www.mvtec.com/company/research/datasets/mvtec-ad/

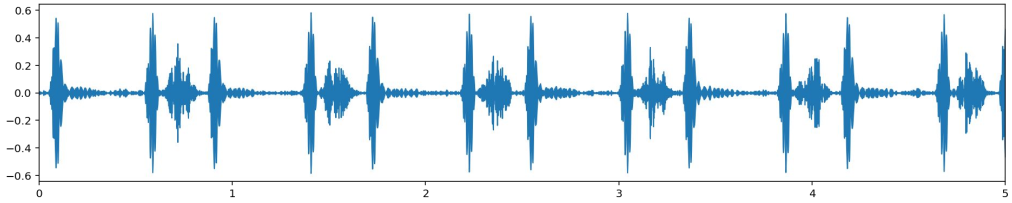


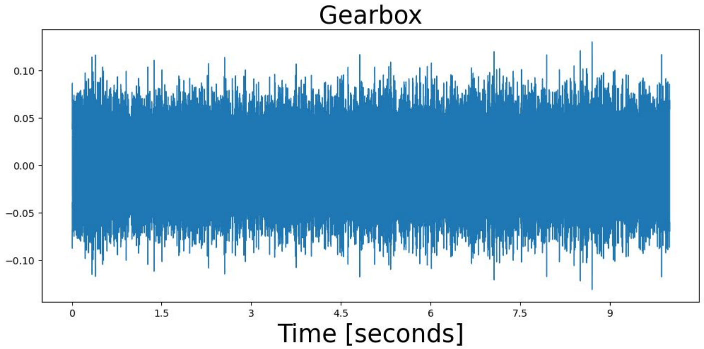
girafe ai

Datasets for Sound Anomaly Detection

- A Dataset of Lung Sounds Detect pulmonary diseases
  https://www.kaggle.com/datasets/arashnic/lung-dataset
- Heartbeat Sounds: Classifying heartbeat anomalies
  https://www.kaggle.com/datasets/kinguistics/heartbeat-sounds
- MIMII, DCASE 2019 Challenge https://dcase.community/
- MIMII DUE, DCASE 2022 Challenge
- ToyAdmos, DCASE 2019 Challenge
- ToyAdmos2, DCASE 2022 Challenge







girafe ai

https://dcase.community/challenge2022/task-unsupervised-anomalous-sound-detection-for-machine-condition-monitoring

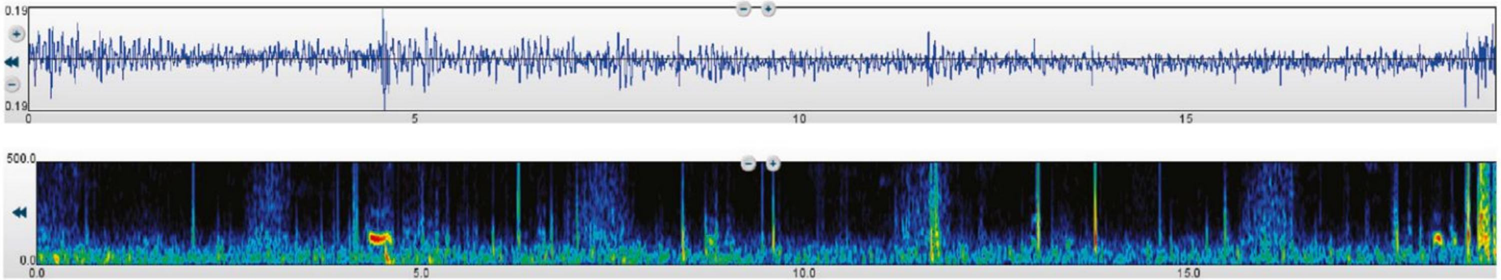
A Dataset of Lung Sounds

A dataset of lung sounds recorded from the chest wall using an electronic stethoscope (2021)

respiratory sounds:

- 112 subjects (35 healthy and 77 unhealthy)
- The subjected aged from 21 to 90, mean ±SD of 50.5 ±19.4,
- 43 females and 69 males.

Health Condition	No. of Subjects
Normal	35
Asthma	32
Pneumonia	5
COPD	9
BRON	3
Heart failure	21
Lung fibrosis	5
Pleural effusion	2



girafe ai

https://data.mendeley.com/datasets/jwyy9np4gv/3
https://www.sciencedirect.com/science/article/abs/pii/S0208521620301297?via%3Dihub

Dataset of Heartbeat Sounds

Classifying heartbeat anomalies from stethoscope.

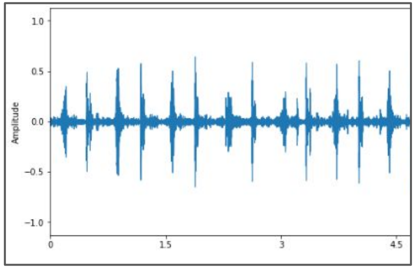
Dataset A (iStethoscope Pro iPhone app):

- Normal
- Murmur
- Extra Heart Sound
- Artifact

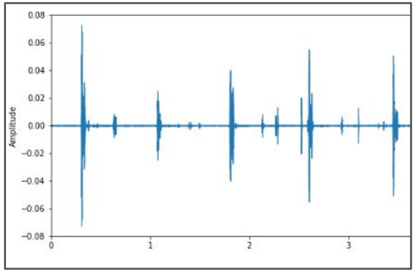
Dataset B (clinic trial in hospitals using the digital stethoscope DigiScope):

- Normal
- Murmur
- Extrasystole

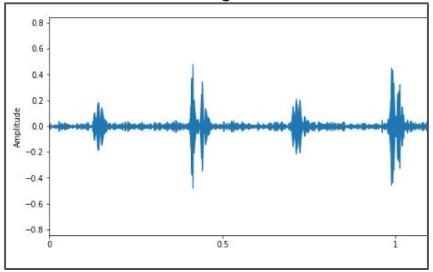
normal



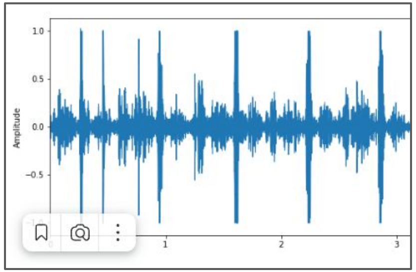
extra heart sounds



extrasystole



murmur



girafe ai

http://www.peterjbentley.com/heartchallenge/ ,
https://www.kaggle.com/datasets/kinguistics/heartbeat-sounds

MIMII, ToyAdmos datasets

DCASE Challenge:

IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events

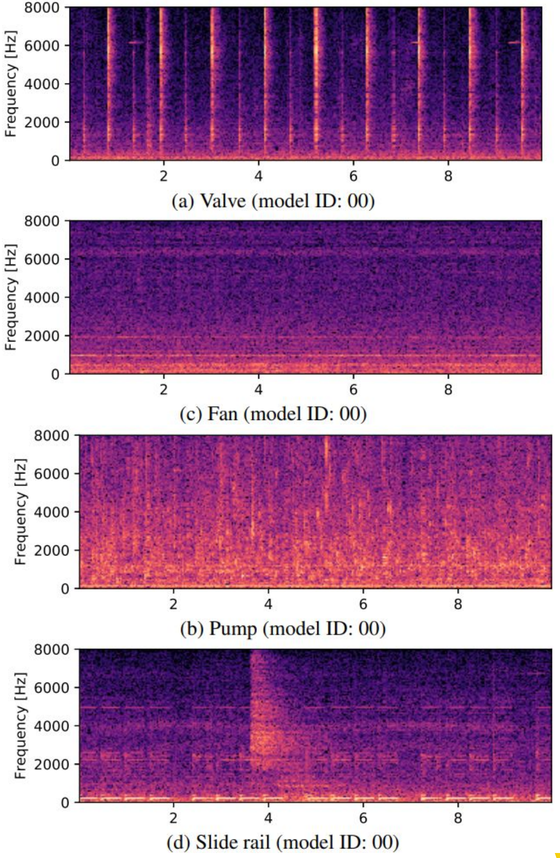
Task:
Unsupervised Anomalous Sound Detection for Machine Condition Monitoring.

MIMII (DCASE 2019)

Sound dataset for malfunctioning industrial machine investigation and inspection.

- Four different types of industrial machines in real factory environments.
- Each type contains 10-20 hours normal sound and ~2 hours anomalous sounds.

Machine type	Operations	Examples of anomalous conditions
Valve	Open / close repeat with different timing	More than two kinds of contamination
Pumpc	Suction from / discharge to a water pool	Leakage, contamination, clogging, etc.
Fan	Normal operation	Unbalanced, voltage change, clogging, etc.
Slide rail	Slide repeat at different speeds	Rail damage, loose belt, no grease, etc.

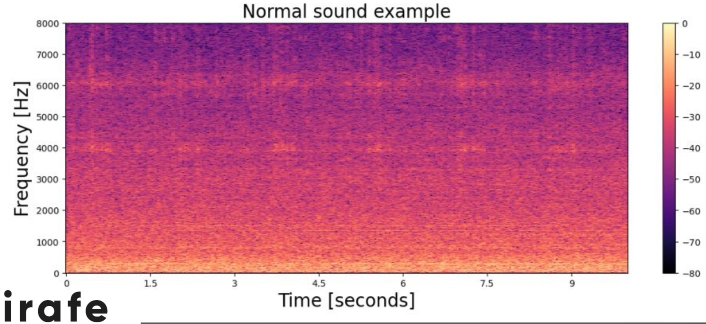


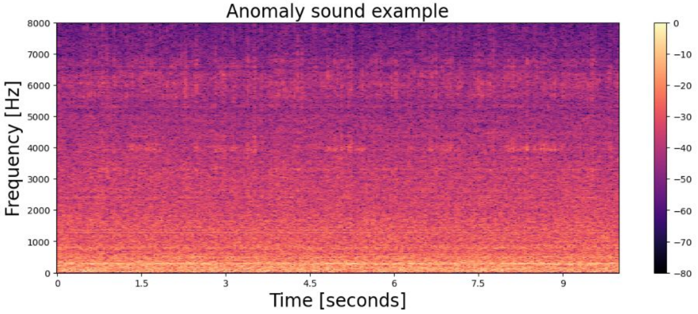
(a) Valve (model ID: 00)
(c) Fan (model ID: 00)
(b) Pump (model ID: 00)
(d) Slide rail (model ID: 00)

MIMI DUE (DCASE challenge 2021)

Sound dataset with domain shifts due to changes in operational and environmental conditions.

Machine type	Operations	Examples of anomalous conditions
Valve	Open / close repeat with different timing	More than two kinds of contamination
Pump	Suction from / discharge to a water pool	Contamination, clogging, leakage, dry run, etc
Fan	Normal operation	Wing damage, unbalanced, clogging, and over voltage
Slide rail	Slide repeat at different speeds	Rail damage, loose belt, no grease, etc.
Gearbox	Normal operation	Gear damage, overload, over voltage, etc

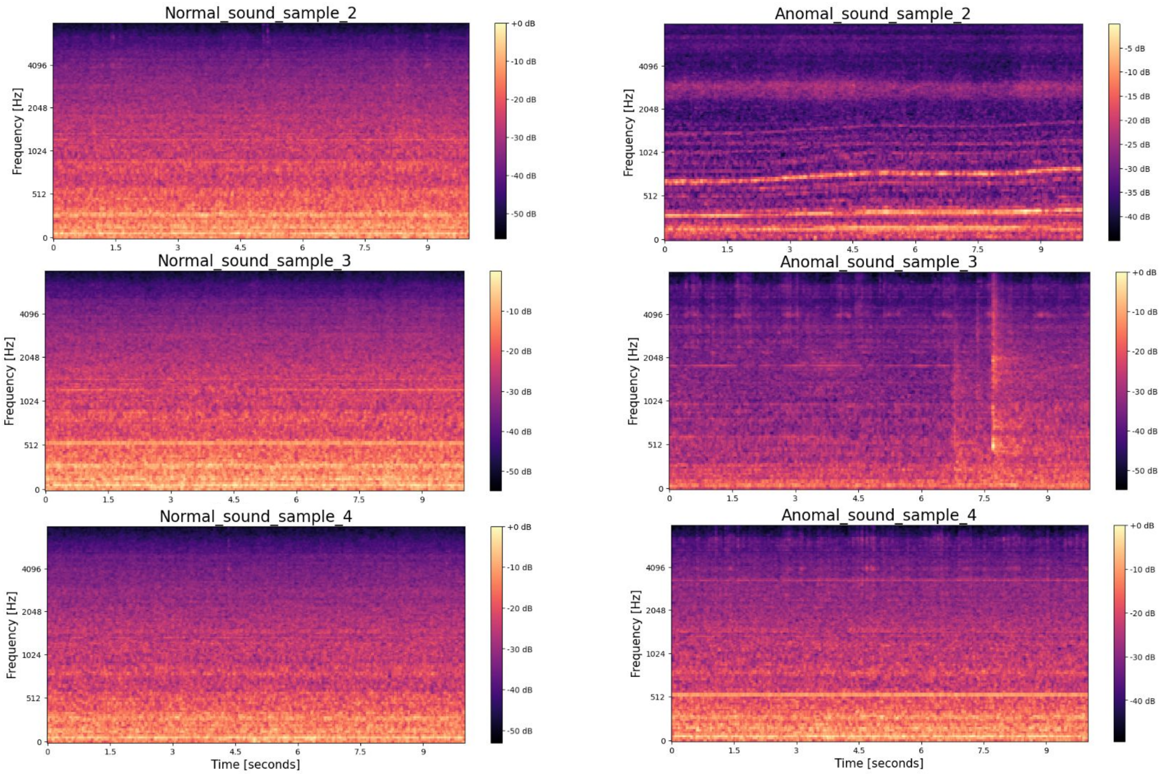




girafe ai

[MIMII DUE: SOUND DATASET FOR MALFUNCTIONING INDUSTRIAL MACHINE INVESTIGATION AND INSPECTION WITH DOMAIN SHIFTS DUE TO CHANGES IN OPERATIONAL AND ENVIRONMENTAL CONDITIONS]

MIMI DUE Normal / Anomaly Sounds



ToyAdmos (DCASE 2019)

- 540 hours of normal machine operating sounds
- over 12 k samples of anomalous sounds

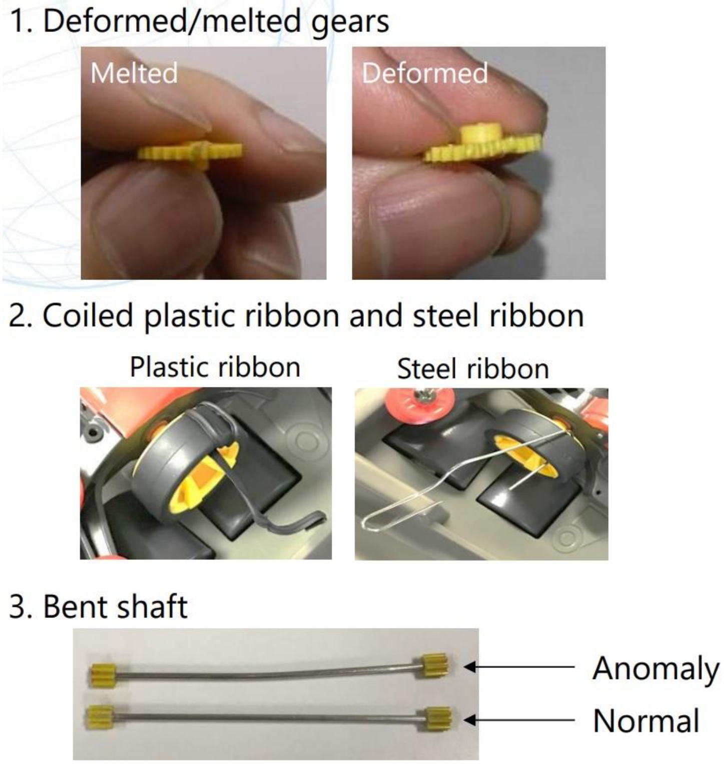
Anomaly detection tasks:

- product inspection (toy car),
- fault diagnosis for fixed machine (toy conveyor),
- fault diagnosis for moving machine (toy train).

Anomalous sound:

- deliberately damaging its components
- adding extraneous objects

Toy car anomaly conditions



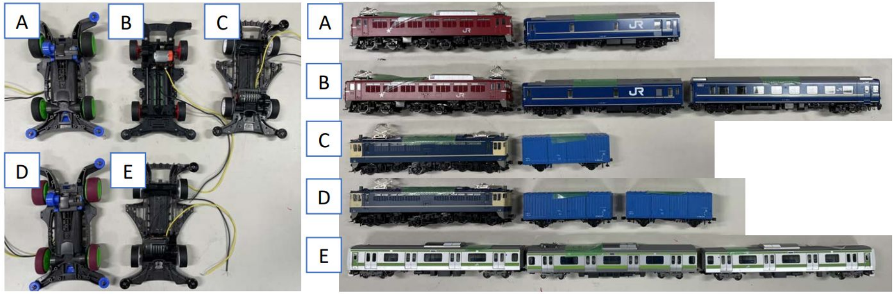
github: https://github.com/YumaKoizumi/ToyADMOS-dataset
dataset: https://zenodo.org/record/3351307#.Y-_0wFDP1D9

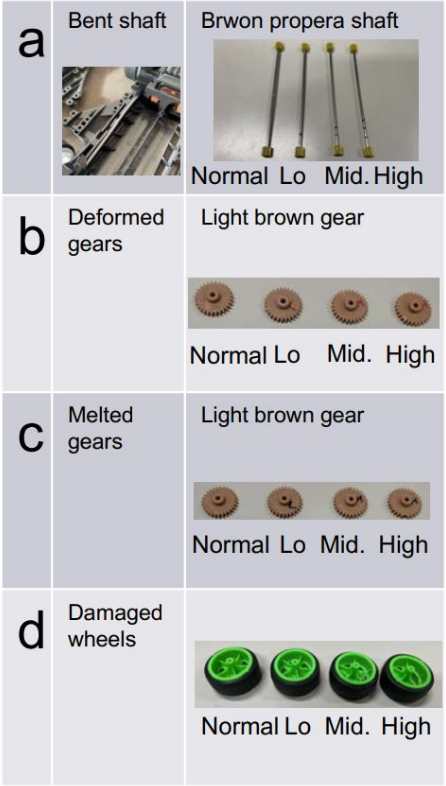
ToyAdmos2 (DCASE Workshop 2021)

- over 27 k samples of normal machine-operating sounds
- over 8 k samples of anomalous sounds

Dataset is designed for evaluating systems under domain-shift conditions. Domain shifts conditions:

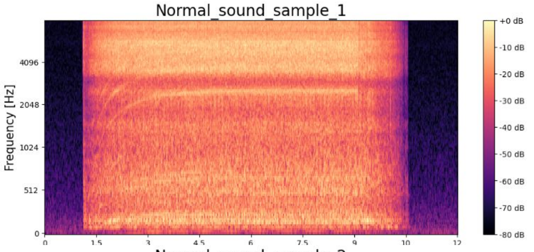
- using different machine models and parts configurations,
- different operating speeds,
- microphone arrangements, etc.

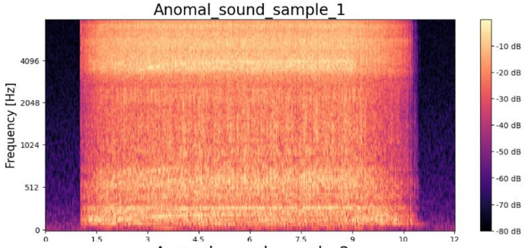


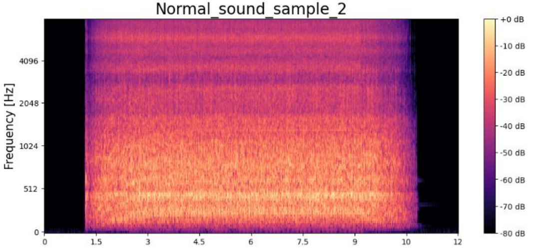


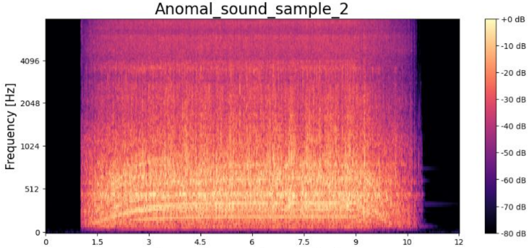
https://paperswithcode.com/paper/toyadmos2-another-dataset-of-miniature

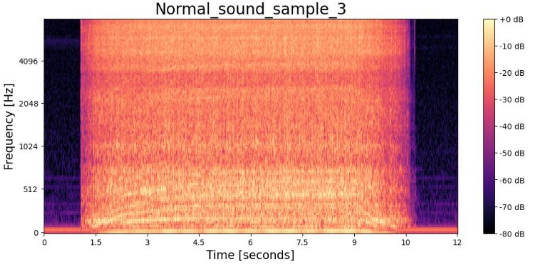
ToyAdmos2 Normal / Anomaly Sounds

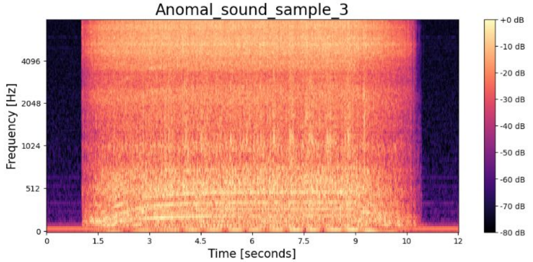












girafe ai

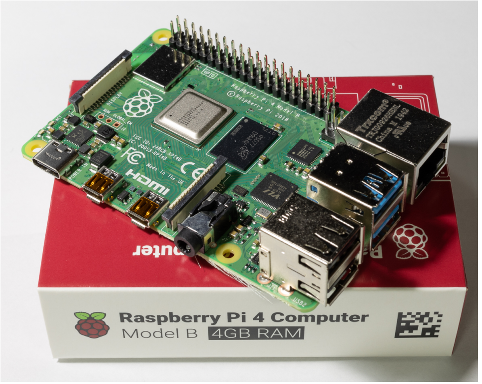
Evocargo electrical engines

The main challenge:

- to detect unknown anomalous sounds
- only normal sound samples as training data.

Fault diagnosis steps:

- engine sounds recording on Piezo contact mic REV2 using Raspberry pi4.
- collecting an audio dataset with 10 sec audio samples



Audio representation

- Time domain features
- Spectrogram
- Mel Spectrogram
- MFCC coefficients

girafe
ai

Time Domain Waveform

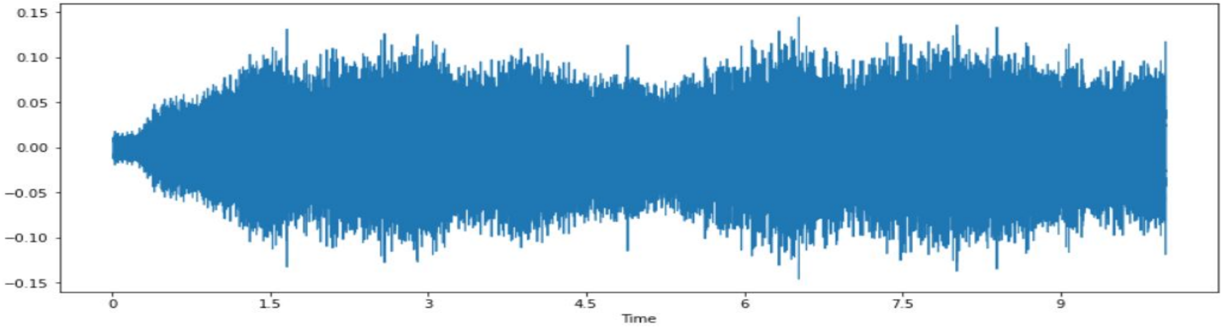
Original signal, time series array of amplitudes according sample rate

Advantages:

- Completely describes the waveform.
- Directly generates the output waveform.

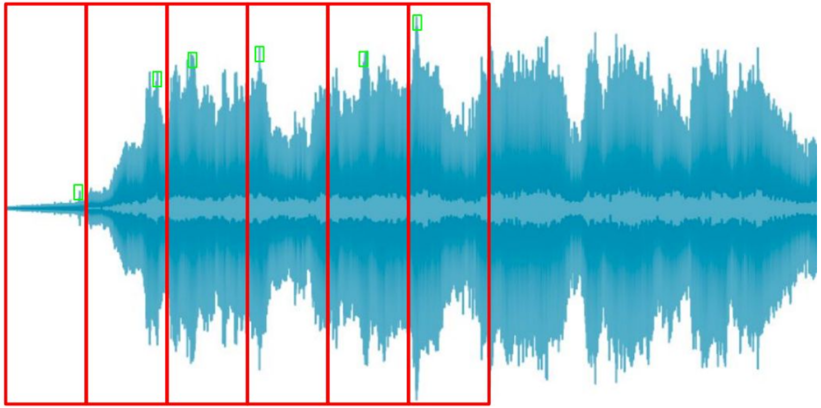
Disadvantages:

- Computationally expensive.
- Unstructured representation that does not reflect sound perception.

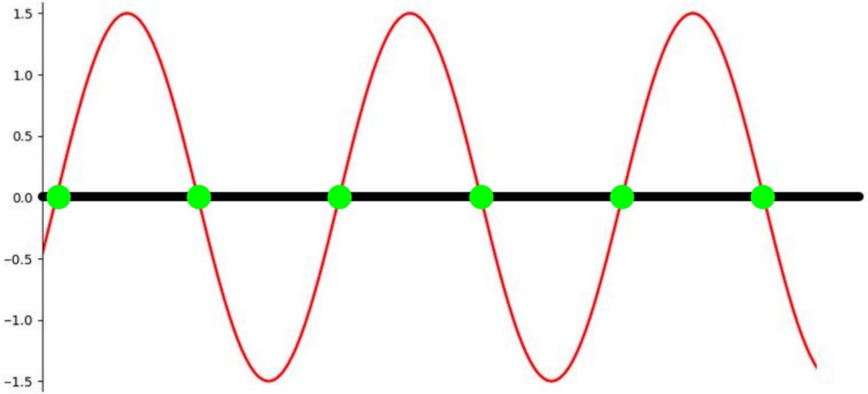


Time Domain Features

Amplitude envelope - Max amplitude value of all samples in a frame



Zero crossing rate - Number of times a signal crosses the horizontal axis



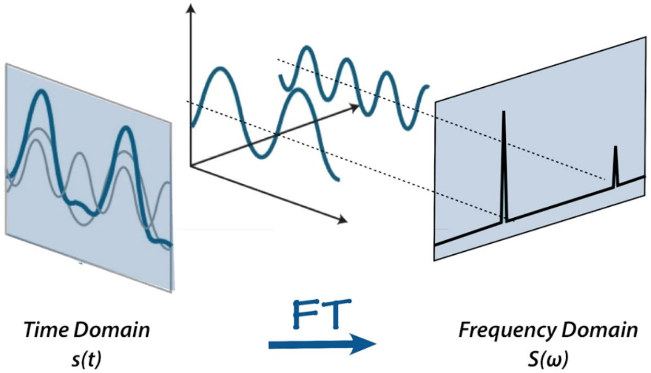
https://github.com/musikalkemist/AudioSignalProcessingForML

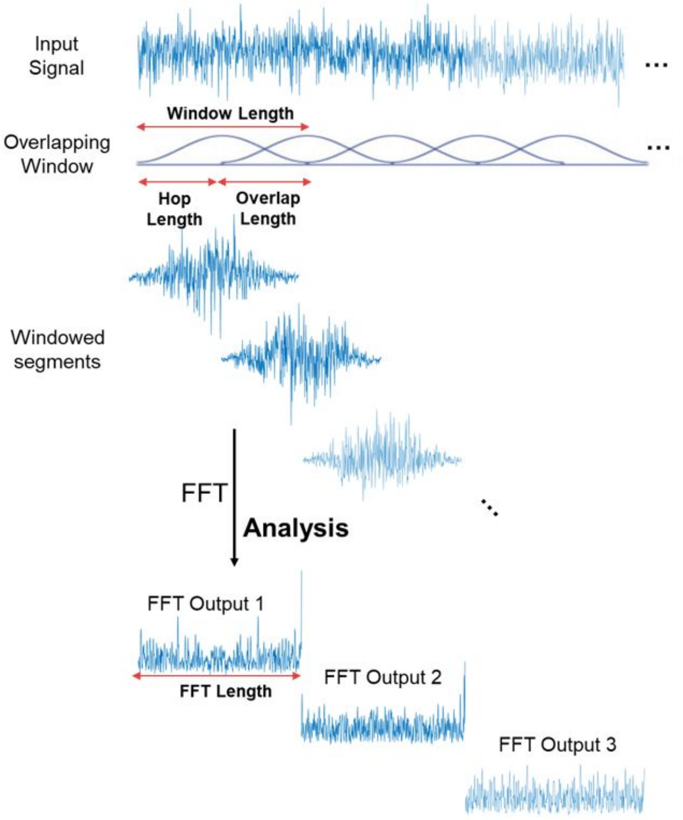
Spectrogram

Visual representation of frequencies of a given signal with time.

Creating spectrogram:

- break the audio signal into smaller frames(windows)
- calculate FFT for each window.
- getting frequencies for each window and window number will represent the time.





girafe ai

https://medium.com/analytics-vidhya/understanding-the-mel-spectrogram-fca2afa2ce53

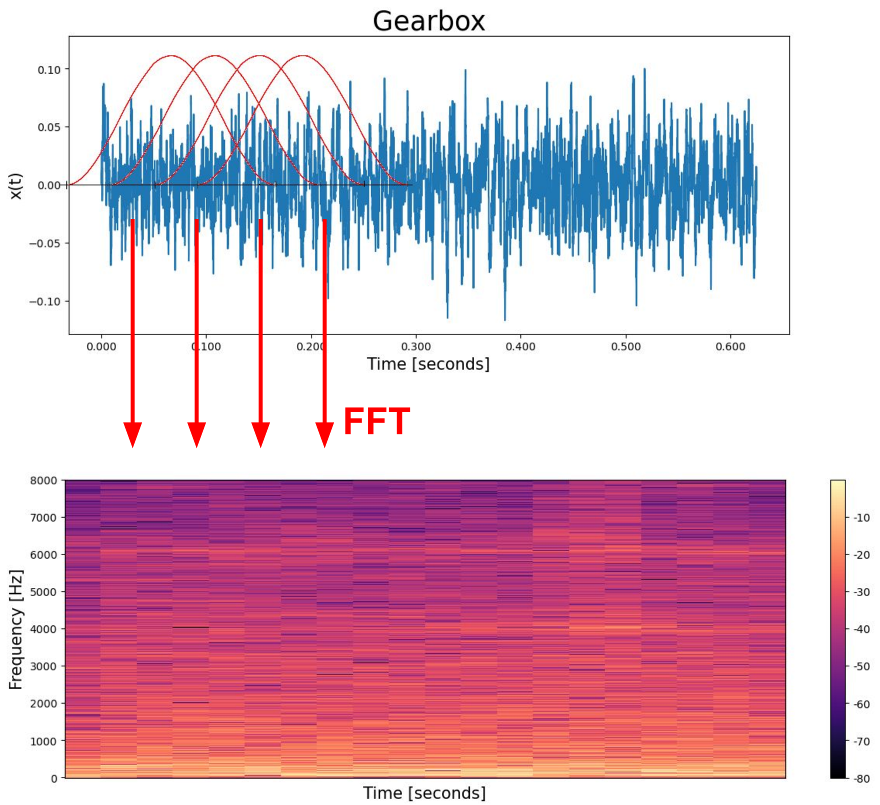
Spectrogram

Spectrogram representation plot:

- one axis represents the time
- the second axis represents frequencies
- the colors represent magnitude (amplitude) of the observed frequency at a particular time.

Advantages:

- Interpretable representations that are related to sound perception.



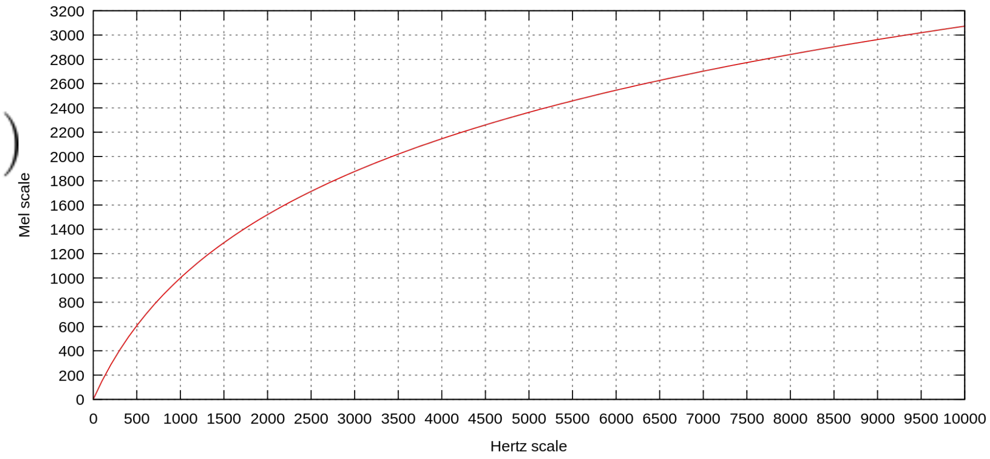
Mel-Frequency Spectrogram

A mel spectrogram is a spectrogram where the frequencies are converted to the mel scale (is a perceptual scale of pitches).

Mel Scale - is a logarithmic transformation of a signal’s frequency. Sounds of equal distance on the Mel Scale are perceived to be of equal distance to humans.

$$m = 2595 \log_{10}(1 + \frac{f}{700})$$

$$f = 700(10^{\frac{m}{2595}} - 1)$$

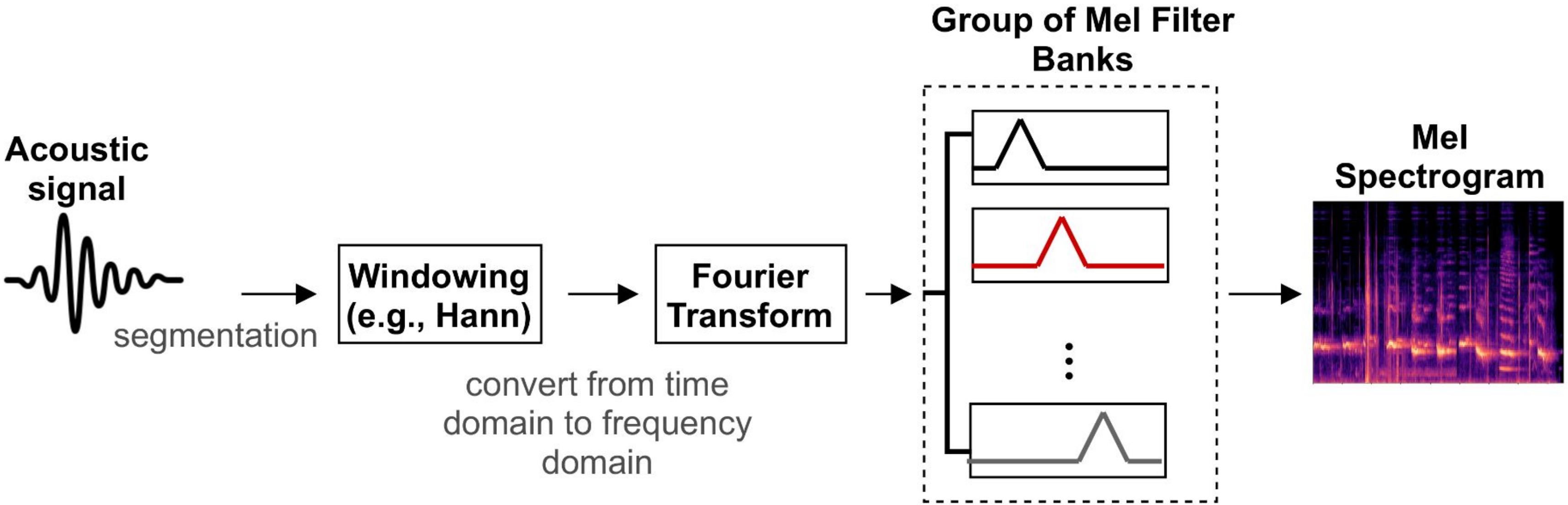


girafe ai

https://www.mdpi.com/1424-8220/21/9/3279

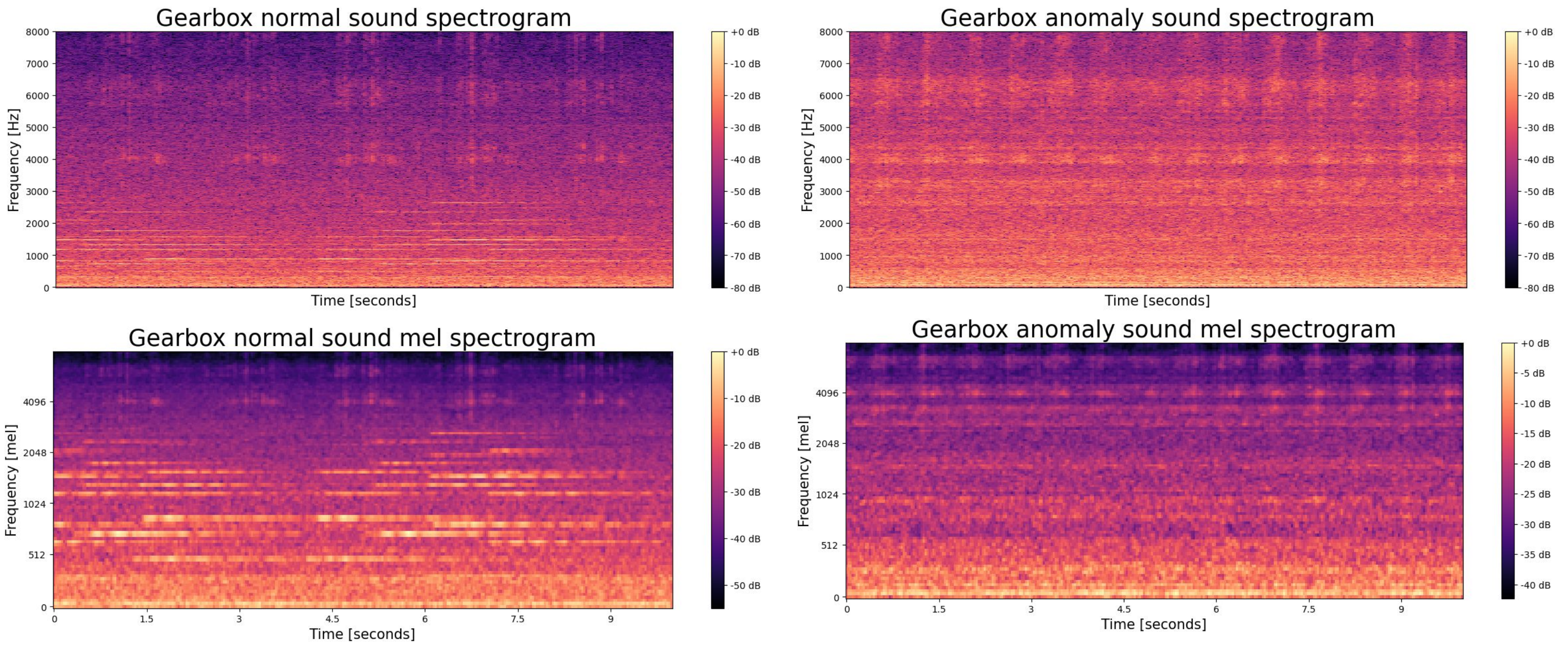
Mel Frequency Spectrogram

Process of extracting the Mel spectrogram from an acoustic signal:



https://www.mdpi.com/1424-8220/21/9/3279

Mel Frequency Spectrogram

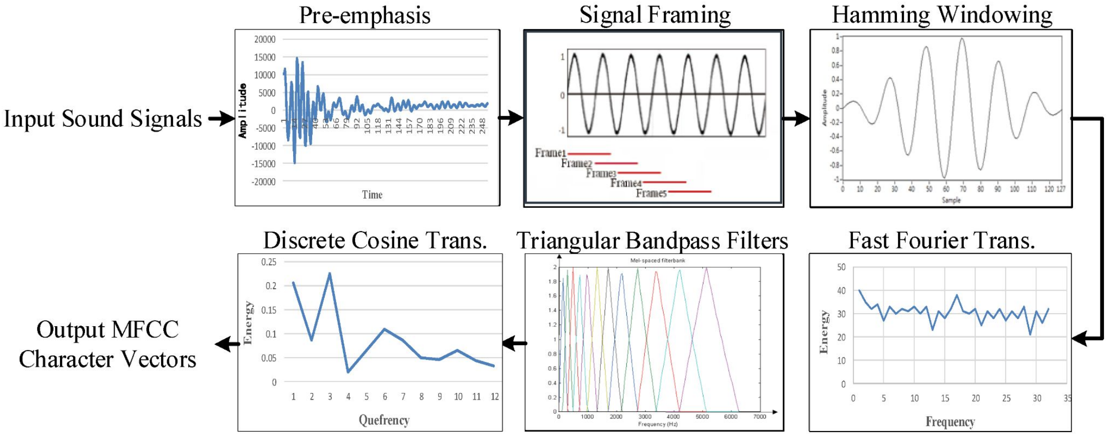


https://www.mdpi.com/1424-8220/21/9/3279

Mel Frequency Cepstral Coeffs (MFCC)

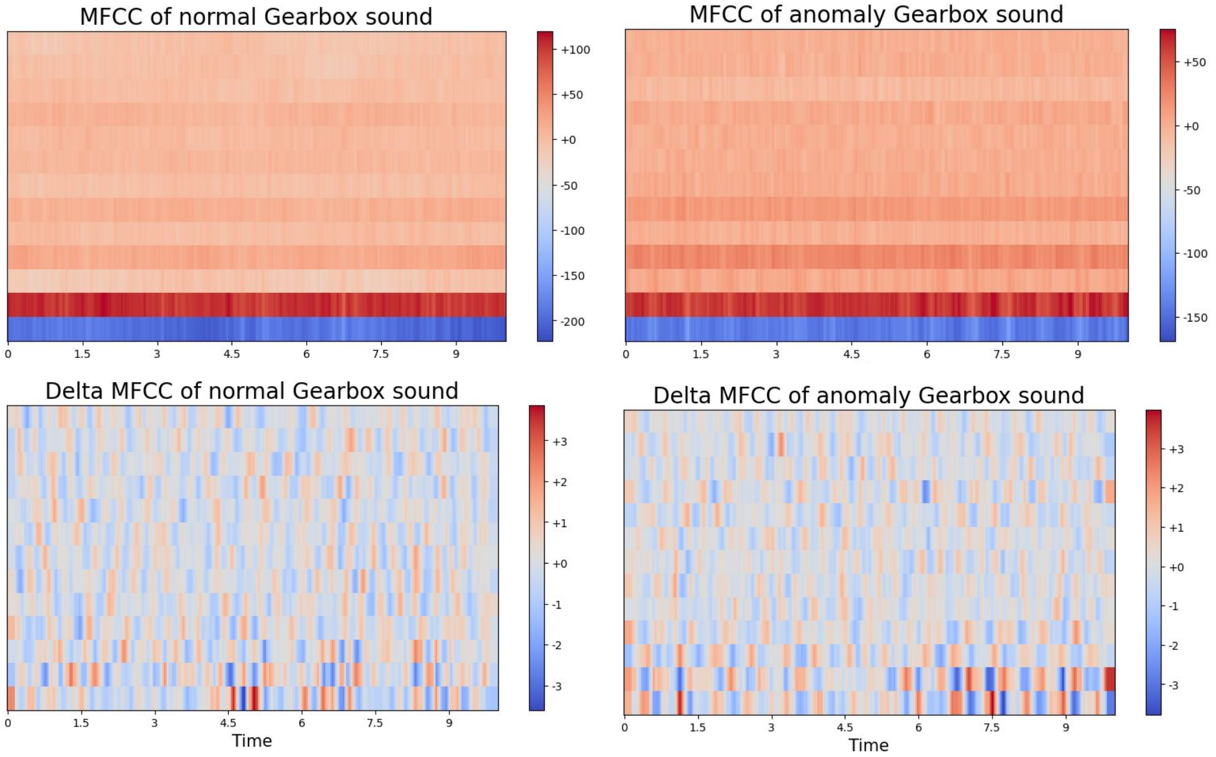
To get the MFCC we follow the following steps:

- Take the Fourier Transform of signal
- Map the power to the mel-scale using triangular overlapping windows
- The logs of the powers at each of the mel frequencies
- Take DCT — Discrete cosine transform of the mel log powers
- The MFCC are the amplitudes of the resulting spectrum



https://www.mdpi.com/2079-9292/8/9/924/htm

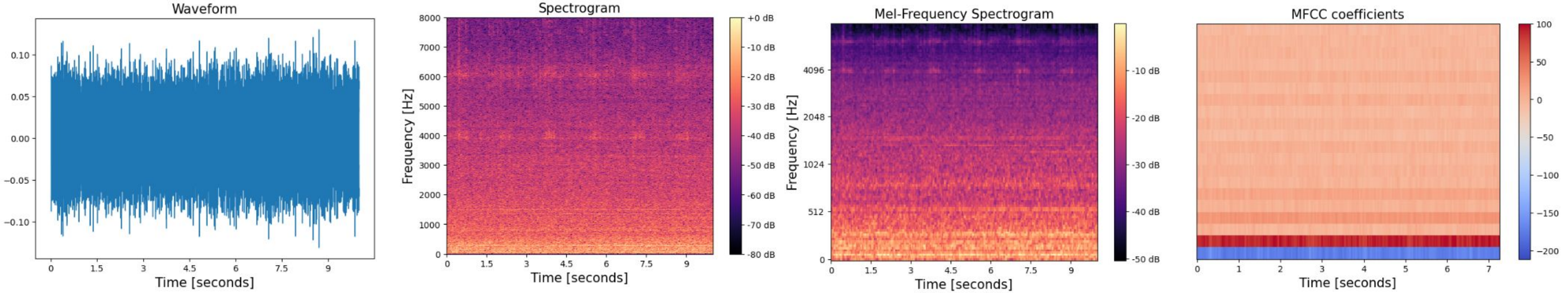
Mel Frequency Cepstral Coeffs (MFCC)



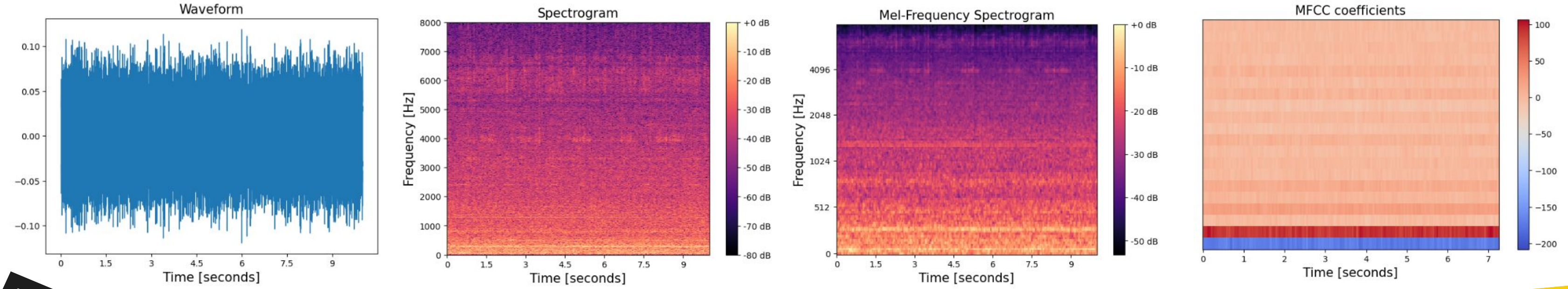
https://www.mdpi.com/2079-9292/8/9/924/htm

Gearbox Sounds Acoustic Features

Normal sound example (MIMI DUE, gearbox, section 1)



Anomaly sound example (MIMI DUE, gearbox, section 1)



Models

Classical Algorithms

1. K-Nearest Neighbor (KNN)
2. Local Outlier Factor (LOF)
3. Isolation Forest (iForest)
4. One Class SVM (OCSVM)

PyOD (Python Outlier Detection)

PyOD is the most comprehensive and scalable Python library for detecting outlying objects in multivariate data.

- includes more than 40 detection algorithms
- Since 2017 more than 10 million downloads.

PyOD module offers methods to aggregate the outcome:

- Average
- Maximum of Maximum (MOM)
- Average of Maximum (AOM)
- Maximum of Average (MOA)

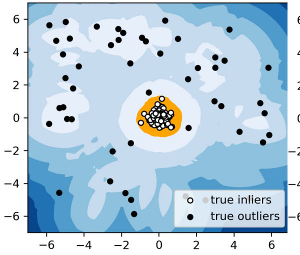
Zhao, Y., Nasrullah, Z. and Li, Z., 2019. PyOD: A Python Toolbox for Scalable Outlier Detection. Journal of machine learning research (JMLR), 20(96), pp.1-7. http://jmlr.org/papers/v20/19-011.html

Classical Anomaly Detection Algorithms

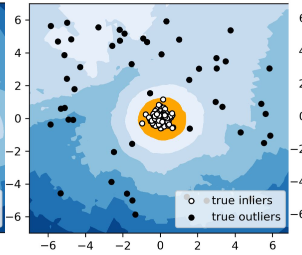
Unsupervised Anomaly Detection Algorithms

- **Distance based**
  - KNN
- **Density based**
  - LOF
  - COF
  - LOCI
- **Clustering based**
  - DBSCAN
  - CBLOF
- **Statistical**
  - GMM
  - KDE
  - HBOS
- **Classifier based**
  - One class SVM
- **Ensembles**
  - Isolation Forest

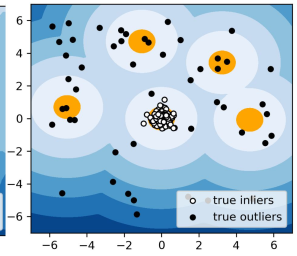
K-Nearest Neighbors (KNN)



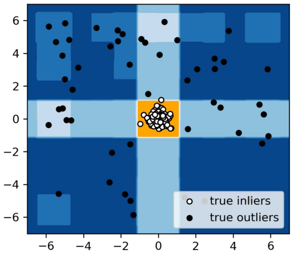
Local Outlier Factor (LOF)



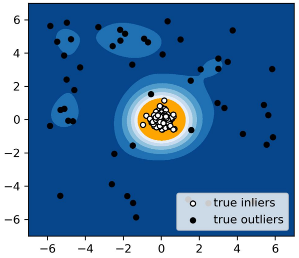
Cluster based Local Outlier Factor



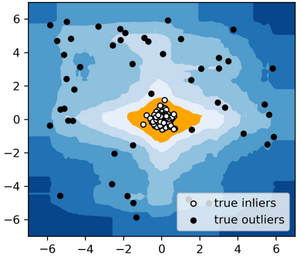
Histogram-based Outlier Detection (HBOS)



One-class SVM (OCSVM)



Isolation Forest (iForest)

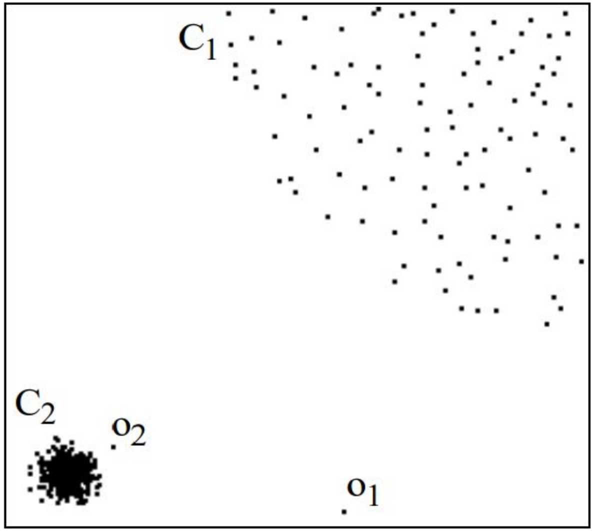


[Python Outlier Detection (PyOD)]
[Progress in Outlier Detection Techniques: A Survey]

K-Nearest Neighbor (KNN)

Computes the distance to the k nearest neighbors and uses the distance to define the outlier scores:

- For each data point, calculate the distance to other data points.
- Sort the data points from smallest to largest by the distance.
- Pick the first K entries.



Problems:

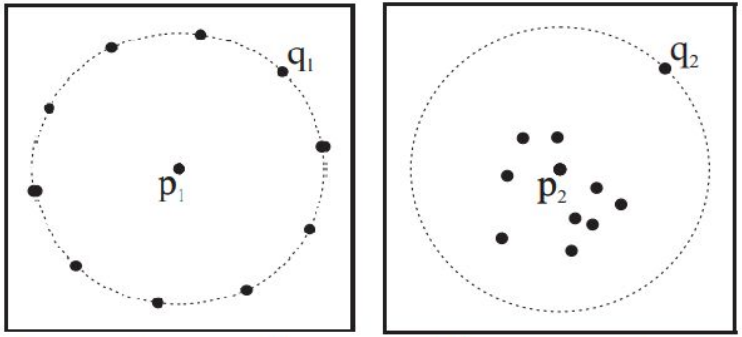
kNN is useful for finding global anomalies, but is less able to surface local outliers.

Fast Outlier Detection in High Dimensional Spaces, F Angiulli, C Pizzuti, PKDD 2002
https://link.springer.com/chapter/10.1007/3-540-45681-3_2

K-Nearest Neighbor (KNN)

There are three methods:

- Maximum: the distance to the k-th neighbor as the outlier score
- Average: the average of all k neighbors as the outlier score
- Median: the median of the distances to k neighbors as the outlier score



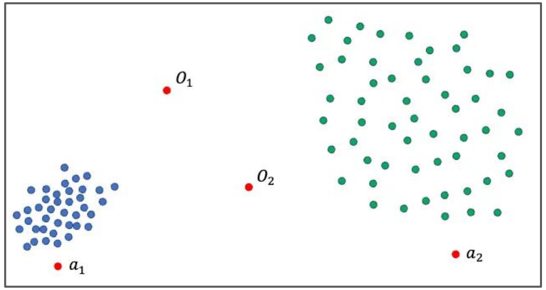
Two points with same D^k values (k=10)

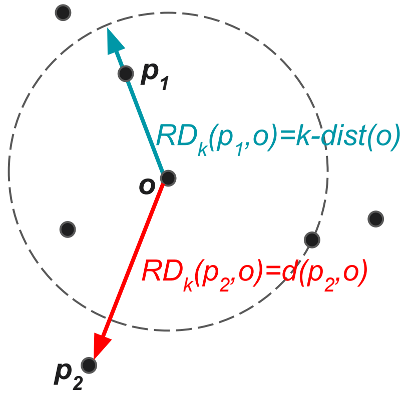
Local Outlier Factor (LOF)

LOF is the ratio between the density of neighborhood where an outlier locates, and the density of the data cluster near the outlier.

- **K-distance:** the distance between the point, and it's Kth nearest neighbor.
- **K-neighbors:** includes a set of points that lie in or on the circle of radius K-distance. K-neighbors can be more than or equal to the value of K.
- **Reachability-distance (RD):** (max - to reduce the statistical fluctuations of d(p,o) for all the points P close to point O)

$$RD_k(p, o) = max\{k{-}dist(o), d(p, o)\}$$





LOF: identifying density-based local outliers, MM Breunig, HP Kriegel, RT Ng, J Sander, 2000
ACM SIGMOD international conference on Management of data, 2000
https://dl.acm.org/doi/pdf/10.1145/342009.335388

Local Outlier Factor (LOF)

- **Local reachability density (LRD)** - the inverse of the average reachability distance of o from its neighbors.

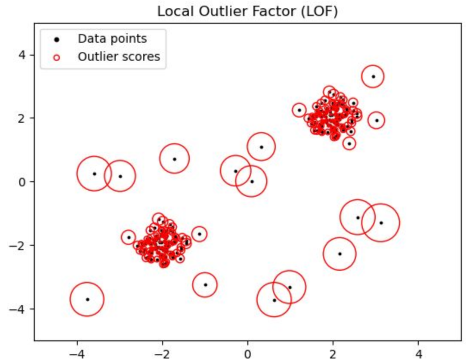
$$LRD_k(p) = \frac{1}{\sum_{o \in N_k(p)} \frac{RD(p,o)}{|N_k(p)|}}$$

- **Local Outlier Factor of K-neighbor: LOF(k)** - the ratio of the average LRD of the K neighbors of Point p to its LRD

Average LRD of neighbors

$$LOF_k(p) = \boxed{\frac{\sum_{o \in N_k(p)} LRD_k(o)}{|N_k(p)|}} \frac{1}{\boxed{LRD_k(p)}}$$

LDR of point p

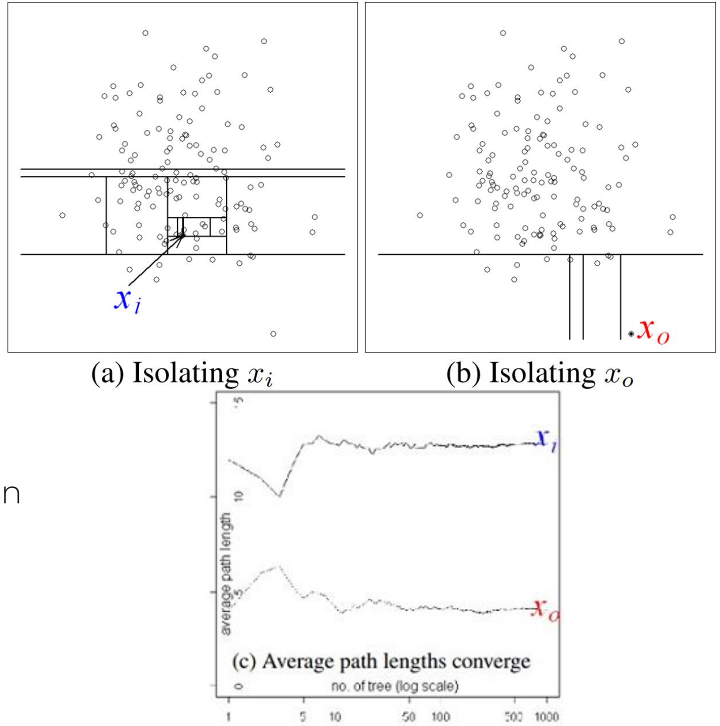


- LOF > 1 more likely to be anomalous
- LOF ≤ 1 less likely to be anomalous
- Large LOF values indicate more isolated points

https://medium.com/dataman-in-ai/handbook-of-anomaly-detection-with-python-outlier-detection-9-lof-8c1831359cc3

Isolation Forest (iForest)

- IForest identifies anomalies directly
- Applies a tree structure to isolate every observation
- Anomalies will be the data points first to be singled out
- Normal points tend to hide deep in the tree
- iForest - an ensemble of iTrees.
- Anomalies - instances with short average path lengths on the iTrees.



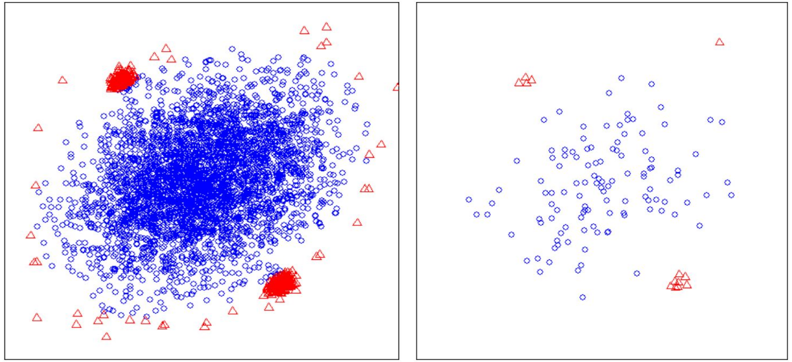
(a) Isolating x_i (b) Isolating x_o

(c) Average path lengths converge

Isolation Forest, Fei Tony Liu, Kai Ming Ting, Zhi-Hua Zhou
Data Mining, 2008. ICDM’08. Eighth IEEE International Conference
https://www.researchgate.net/publication/224384174_Isolation_Forest

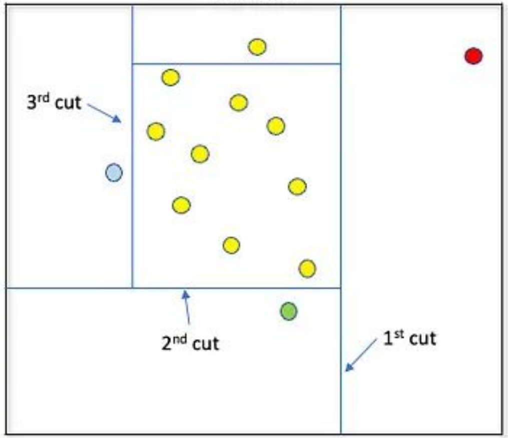
iForest Training Stage

- building trees using dataset subsamples - to reduce the effect of swamping and masking.



(a) Original sample (b) Sub-sample

- recursively split subsample by a randomly selected feature q and the value p, p∈ [q.min, q.max]



- anomalies closer to the root - it is not necessary to construct a large iTree: maximum depth of each tree = $\log_2(n)$.

girafe ai

iForest Evaluating Stage

Anomaly score:

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

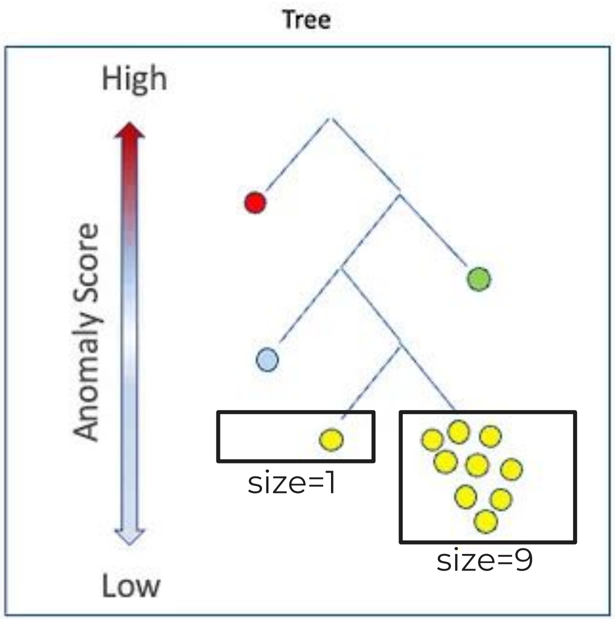
The average value of h(x) from N iTrees

The adjustment if size of x^{th} external node with x > 1

$$\boxed{c(n) = 2H(n-1) - 2\frac{n-1}{n}}$$

$$H(i) = ln(i) + \gamma$$

H - harmonic number
γ - Euler's constant

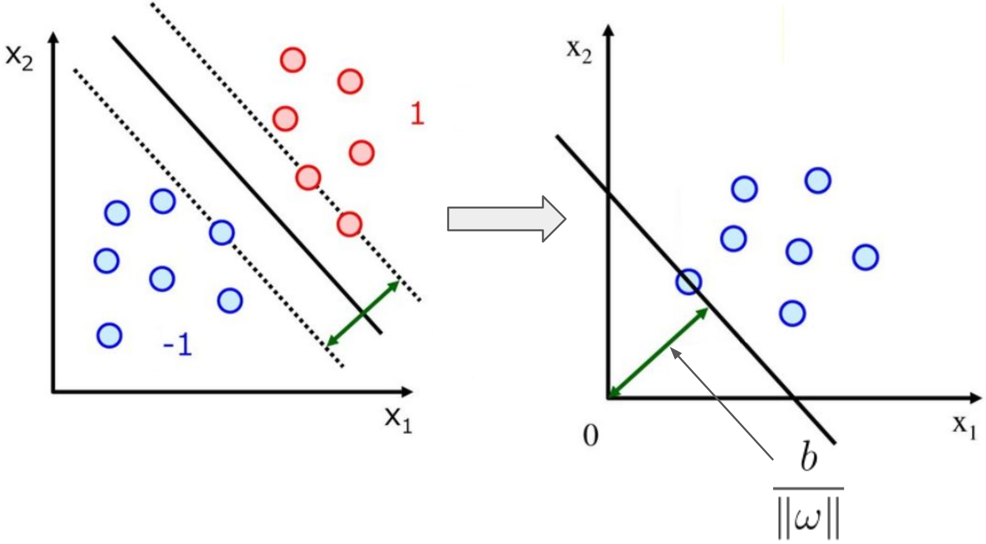


- $E(h(x)) \to 0$, $s \to 1$, x - anomaly
- $E(h(x)) \to (n-1)$, $s \to 0$, x - a normal instance
- $E(h(x)) \to c(n)$, $s \to 0.5$, if all $s \approx 0.5$ - entire sample does not have any anomalies

girafe ai

One Class SVM (OCSVM)

from SVM to One Class SVM



$$\begin{cases} \frac{1}{2}\|w\|^2 - b + \frac{1}{n\nu}\sum_{i=1}^{n}\xi_i \longrightarrow min \\ \langle \omega, x \rangle \geq b - \xi_i \\ \xi_i \geq 0 \end{cases}$$

ν - upper bound on the outliers fraction

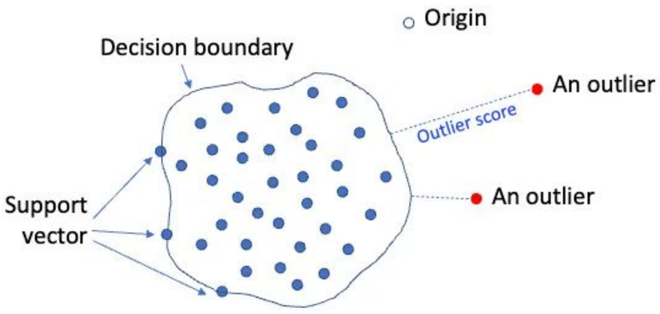
Handbook of Anomaly Detection: With Python Outlier Detection — (6) OCSVM | by Chris Kuo/Dr. Dataman | Dataman in AI | Medium

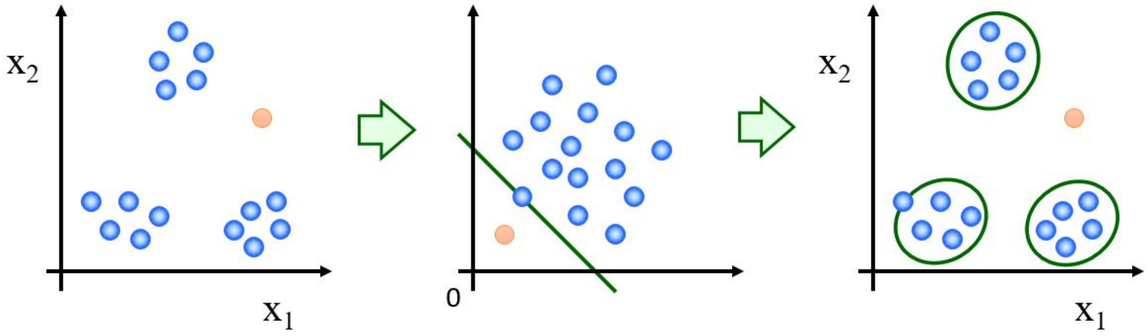
One Class SVM (OCSVM)

separates all the data points from the origin in a higher dimensional space and maximizes the distance from this hyperplane to the origin.

Two approaches:

- use a hyperplane for separation [1]
- use a sphere [2]





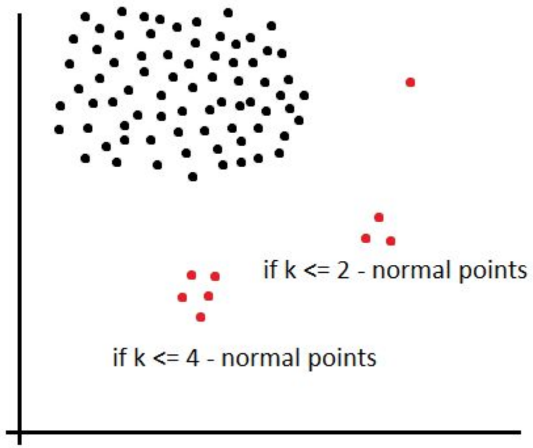
Ensemble of Models

Using ensemble of model with different hyperparameters for model stability.

- different k in KNN
- iTrees number (n_estimators) and sample size in iForest

Aggregation of the scores produced by multiple models.
The PyOD module offers four methods to aggregate the outcome:

- Average
- Maximum of Maximum (MOM)
- Average of Maximum (AOM)
- Maximum of Average (MOA)



We can aggregate the scores produced by different models:

- more models indicated the instance as an outlier, the higher the probability that this instance the real outlier

Deep Anomaly detection

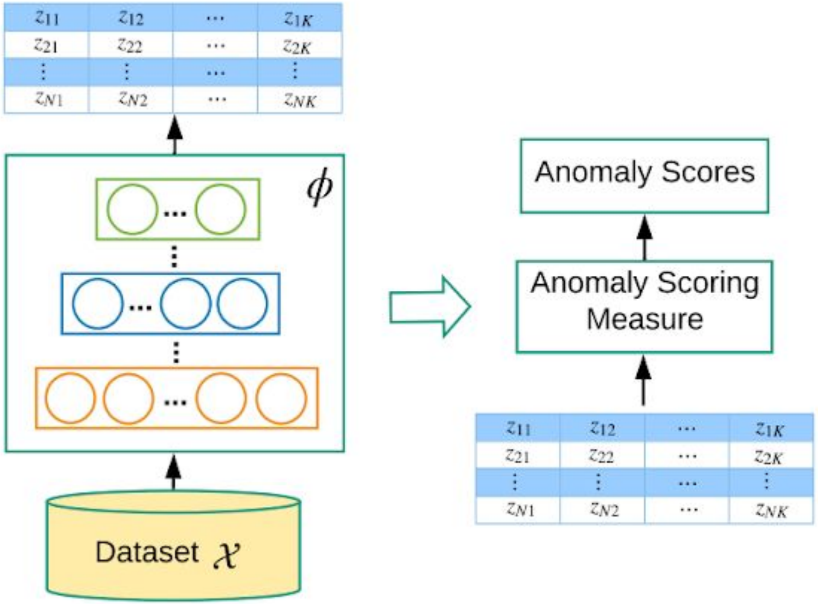
Consists of three conceptual paradigms:

1. Deep learning for feature extraction
2. Learning feature representation of normality
3. End-to-end Anomaly Score Learning

Deep learning for anomaly detection: A review
G Pang, C Shen, L Cao, AVD Hengel
ACM computing surveys (CSUR), 2021,
https://arxiv.org/pdf/2007.02500.pdf

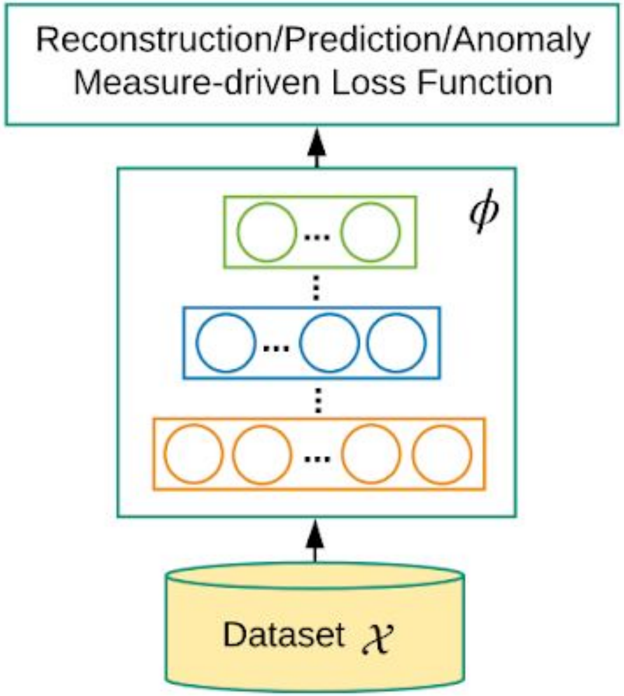
Deep Learning for Feature Extraction

- The feature extraction and the anomaly scoring are fully disjointed and independent tasks. Deep learning for dimensionality reduction only.
- it is possible to use pre-trained deep learning models (CNN for images, RNN for sequential data, for audio data (SoundNet, VGGish, OpenL3, VGGVox, PANNs)
- extracted features (low dimensional) as input to classic key algorithms for anomaly detection.



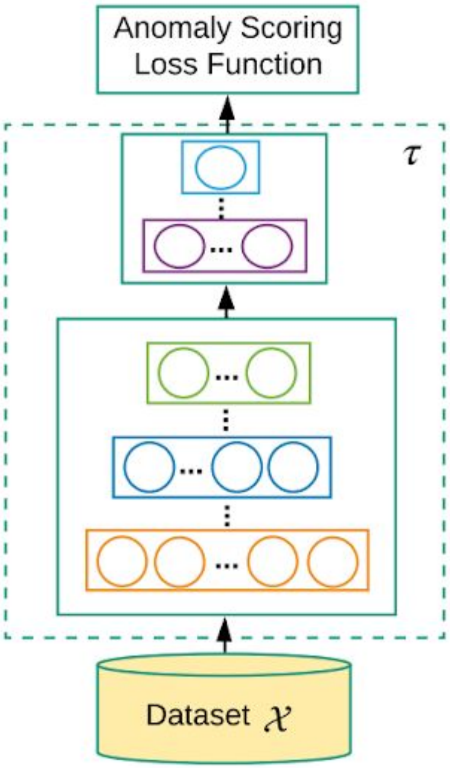
Learning Feature Representation of Normality

- Autoencoders (CNN, LSTM based AE, etc.)
- Generative Adversarial Networks [3]
- Self-supervised classification
- Anomaly Measure-dependent Feature Learning (Distance-based Measure, One-class Classification-based Measure, Clustering-based Measure) [4]



End-to-end Anomaly Score Learning

neural network directly learns the anomaly scores. The methods in this category simultaneously learn the feature representations and anomaly score. [5] [6] [7] [8]



DL Algorithms

1. Autoencoders
2. DevNet
3. STgramNet

Tools & Libs Deep Anomaly Detection

Anomalib: A Deep Learning Library for Anomaly Detection, 2022,
https://github.com/openvinotoolkit/anomalib

- ready-to-use implementations of anomaly detection algorithms
- a set of tools for development and implementation of custom models.
- has a strong focus on image-based anomaly detection (to identify anomalous images or anomalous pixel regions)

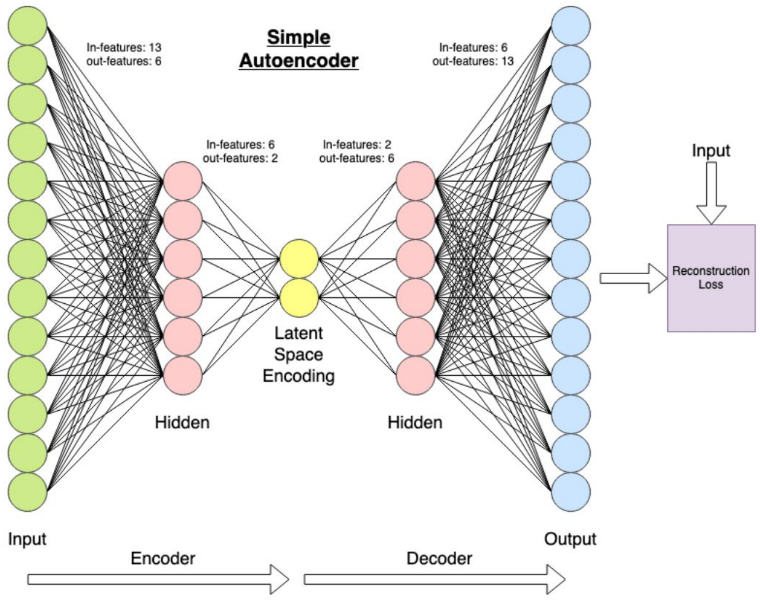
Python Deep Outlier/Anomaly Detection (DeepOD), 2022
https://github.com/xuhongzuo/DeepOD

- open-source python framework for deep learning-based anomaly detection on multivariate data.
- 11 deep outlier detection / anomaly detection algorithms (in unsupervised/weakly-supervised paradigm) based on PyTorch

Autoencoders for Anomaly Detection

Autoencoder-based approaches ([1], [2], [3]) use a bottleneck network architecture to learn a low-dimensional representation space.

- Reconstruction errors as anomaly scores
- Feature extraction for classic key algorithms (KNN, LOF, IForest, etc...)

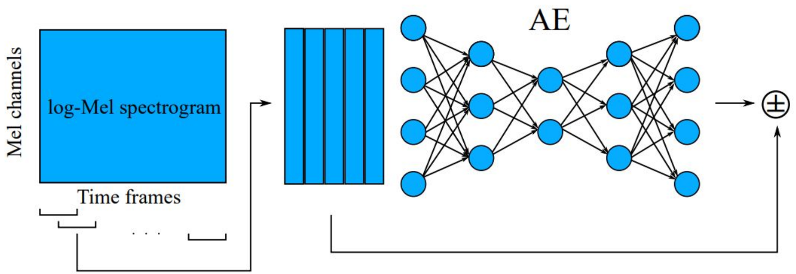


[1] S. Hawkins, H. He, G. Williams, R. Baxter, “Outlier detection using replicator neural networks,” in DaWaK. 2002
[2] J. Chen, S. Sathe, C. Aggarwal, D. Turaga, “Outlier detection with autoencoder ensembles,” in SDM. SIAM, 2017,
[3] C. Zhou and R. C. Paffenroth, “Anomaly detection with robust deep autoencoders,” in KDD. ACM, 2017

AE for Sound Anomaly Detection

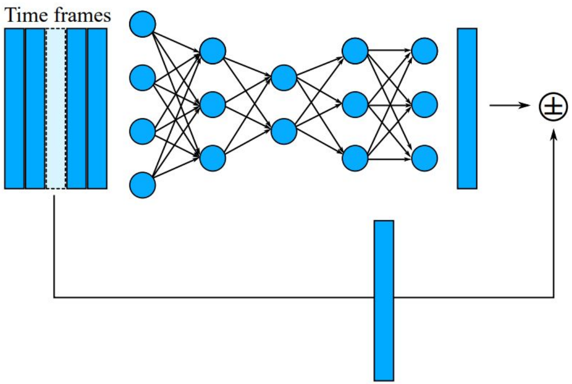
conventional approach :

multiple frames of a spectrogram are used as an input feature, and the same number of frames are generated as an output.



interpolation deep neural network (IDNN):

center frame is removed as an input,
and it predicts an interpolation of the removed frame as an output



girafe ai

[1] K Suefusa, T Nishida, H Purohit, "Anomalous Sound Detection Based on Interpolation Deep Neural Network", ICASSP 2020 - 2020 IEEE https://arxiv.org/pdf/2005.09234.pdf

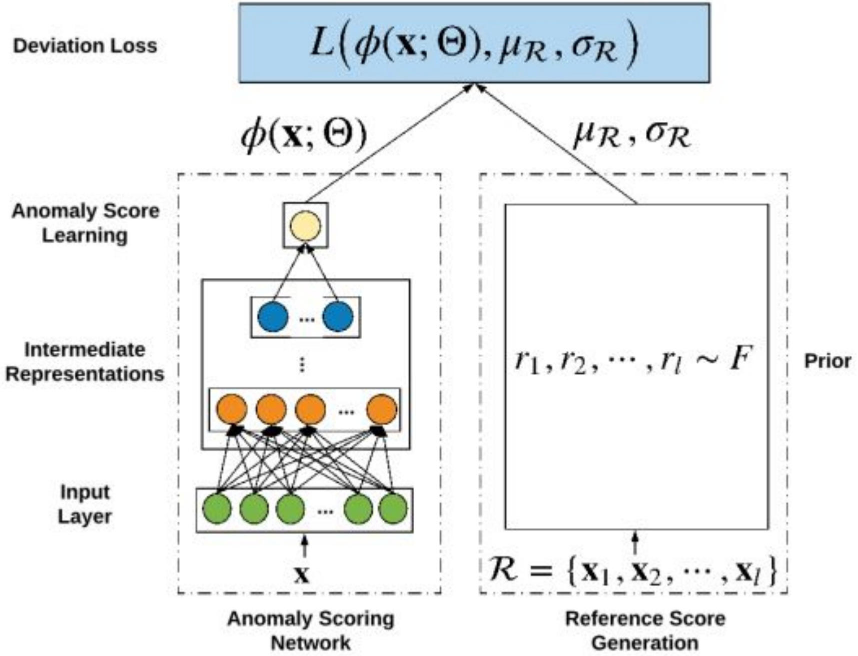
[2] B Bayram, TB Duman, G Ince, "Real time detection of acoustic anomalies in industrial processes using sequential autoencoders" 2021,

DevNet (2019)

Deep Anomaly Detection with Deviation Networks

Given a set of N + K training data objects:

- N - is unlabeled data
- K - is a very small set of labeled anomalies that provide some prior knowledge of anomalies
- K << N



Deep anomaly detection with deviation networks, G Pang, C Shen, A van den Hengel, 2019
KDD '19: International Conference on Knowledge Discovery & Data Mining
https://paperswithcode.com/paper/deep-anomaly-detection-with-deviation

girafe ai

DevNet Loss Function

Z-score-based Deviation Loss

$$dev(\mathbf{x}) = \frac{\phi(\mathbf{x};\Theta) - \mu_{\mathcal{R}}}{\sigma_{\mathcal{R}}},$$

$$L\left(\phi(\mathbf{x};\Theta), \mu_{\mathcal{R}}, \sigma_{\mathcal{R}}\right) = (1-y)|dev(\mathbf{x})| + y \max\left(0, a - dev(\mathbf{x})\right)$$

y = 1 – anomaly, y = 0 - normal object
a is equivalent to a Z-Score confidence interval parameter

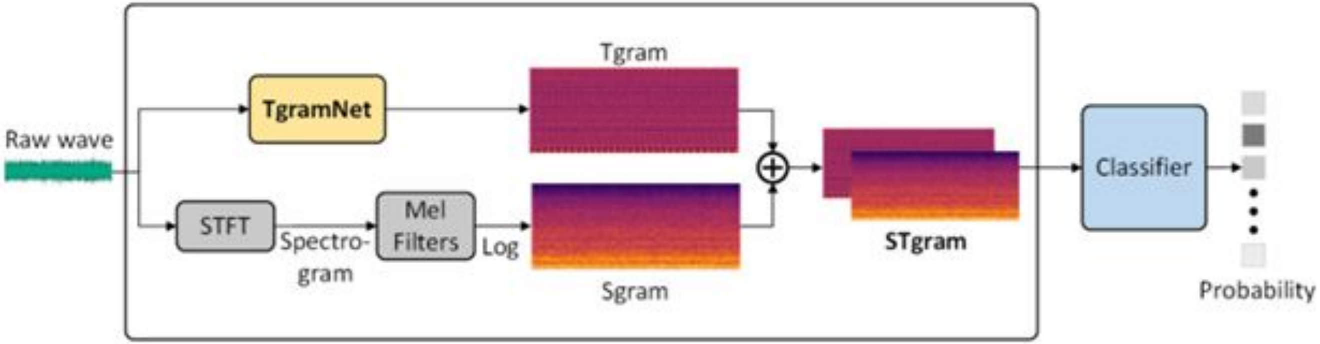
DevNet - semi-supervised algorithm - we need label some known anomalies

STgramNet (2022)

Anomalous Sound Detection Using spectral-temporal information fusion.

The spectral-temporal feature extraction:

- for the temporal feature - from the raw wave through a CNN-based network (TgramNet)
- for the frequency feature - the log-Mel spectrogram.
- concatenation operation for feature fusion and classifier for anomalous sound detection



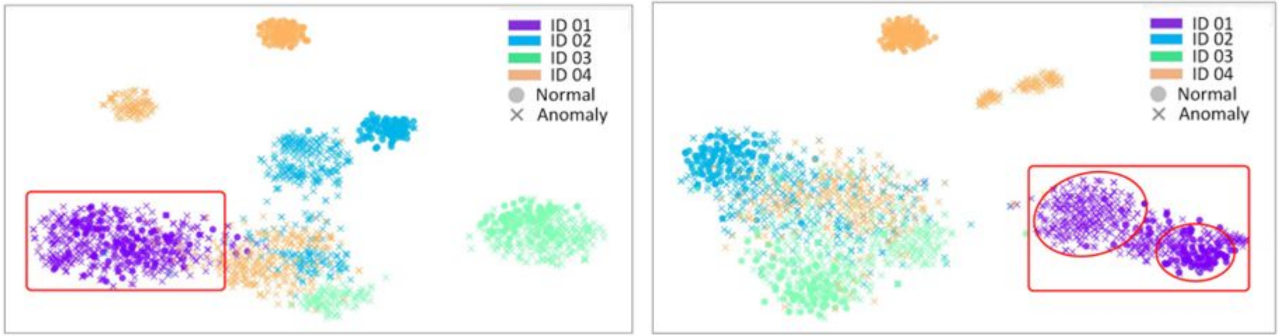
Anomalous Sound Detection Using Spectral-Temporal Information Fusion, Y Liu, J Guan, Q Zhu, W Wang
CASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing,
https://arxiv.org/pdf/2201.05510.pdf

STgramNet

t-SNE cluster visualization of the latent features of log-Mel spectrogram and Tgram shows that log-Mel spectrogram and Tgram are complementary:

- anomaly and normal features of machine “ID 01” are overlapping in terms of the log-Mel spectrogram
- more distinguishable by Tgram.
- the log-Mel spectrogram may filter out useful information about anomalies.

t-distributed stochastic neighbor embedding (t-SNE) cluster visualization



(a) log-Mel spectrogram (b) Tgram

Current Results

girafe
ai

Current Results on MIMII DUE Dataset

Feature Extraction	Classic Algorithms					
	iForest		LOF		KNN	
	F1-score	AP	F1-score	AP	F1-score	AP
Time-domain features + MFCC-agg	0,57	0,59	0,68	0,62	0,52	0,52
MFCC	0,59	0,58	0,63	0,56	0,69	0,61
Melspectrogram	0,71	0,62	0,44	0,52	0,46	0,55

- For Mel-Frequency Spectrogram - the best model iForest
- for MFCC and audio statistics - the best model LOF
- for MFCC coefficients - the best model KNN

girafe ai

Current Results on MIMII DUE Dataset

Feature Extraction	DL Algorithms			
	Deep SVD		RCA	
	F1-score	AP	F1-score	AP
Time-domain features + MFCC-agg	0,66	0,74	0,6	0,7
MFCC	0,64	0,71	0,51	0,57
Melspectrogram	0,65	0,77	0,57	0,59

DL Algorithms from DeepOD library:

- Deep SVDD: Deep one-class classification, L Ruff, R Vandermeulen, N Goernitz, 2018 http://proceedings.mlr.press/v80/ruff18a/ruff18a.pdf
- RCA: A deep collaborative autoencoder approach for anomaly detection, B Liu, D Wang, K Lin, PN Tan, J Zhou - IJCAI, 2021, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9036495/

Reference

[1] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. Neural Computation, 13(7):1443–1471, (2001). https://proceedings.neurips.cc/paper/1999/file/8725fb777f25776ffa9076e44fcfd776-Paper.pdf

[2] Tax, D.M., Duin, R.P. Support Vector Data Description. Machine Learning 54, 45–66 (2004). https://doi.org/10.1023/B:MACH.0000008084.60811.49)

[3] Houssam Zenati, Chuan Sheng Foo, Bruno Lecouat, Gaurav Manek, and Vijay Ramaseshan Chandrasekhar. 2018. Efficient gan-based anomaly detection, https://arxiv.org/pdf/1802.06222.pdf

[4] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. 2018. Learning Representations of Ultrahigh-dimensional Data for Random Distance-based Outlier Detection. In KDD. 2041–2050. https://arxiv.org/pdf/1806.04808.pdf

[5] Guansong Pang, Chunhua Shen, Huidong Jin, and Anton van den Hengel. 2019. Deep Weakly-supervised Anomaly Detection. arXiv preprint:1910.13601 (2019).

Reference

[5] Min-hwan Oh and Garud Iyengar. 2019. Sequential Anomaly Detection using Inverse Reinforcement Learning. In KDD. 1480–1490
https://arxiv.org/pdf/2004.10398.pdf

[6] Guansong Pang, Chunhua Shen, Huidong Jin, and Anton van den Hengel. 2019. Deep Weakly-supervised Anomaly Detection.
https://arxiv.org/pdf/1910.13601.pdf

[7] Ting Chen, Lu-An Tang, Yizhou Sun, Zhengzhang Chen, and Kai Zhang. 2016. Entity embedding-based anomaly detection for heterogeneous categorical events. In IJCAI. 1396–1403
https://arxiv.org/pdf/1608.07502.pdf

[8] Panpan Zheng, Shuhan Yuan, Xintao Wu, Jun Li, and Aidong Lu. 2019. One-class adversarial nets for fraud detection. In AAAI. 1286–1293.
https://ojs.aaai.org/index.php/AAAI/article/view/3924

Thank You for Attention

girafe
ai