Non-Parametric Stochastic Autoencoder Model for Anomaly Detection

Raphael Alampay, Patricia Angela Abu
Ateneo Laboratory for Intelligent Visual Environment
Dept. of Information Systems and Computer Science
Ateneo de Manila University
Quezon City, Philippines

Abstract—**Anomaly detection is a widely studied field in computer science with applications ranging from intrusion detection, fraud detection, medical diagnosis and quality assurance in manufacturing. The underlying premise is that an anomaly is an observation that does not conform to what is considered to be normal. This study addresses two major problems in the field. First, anomalies are defined in a local context, that is, being able to give quantitative measures as to how anomalies are categorized within its own problem domain and cannot be generalized to other domains. Commonly, anomalies are measured according to statistical probabilities relative to the entire dataset with several assumptions such as type of distribution and volume. Second, the performance of a model is dependent on the problem itself. As a machine learning problem, each model has to have parameters optimized to achieve acceptable performance specifically thresholds that are either defined by domain experts of manually adjusted. This study attempts to address these problems by providing a contextual approach to measuring anomaly detection datasets themselves through a quantitative approach called *categorical measures* that provides constraints to the problem of anomaly detection and proposes a robust model based on autoencoder neural networks whose parameters are dynamically adjusted in order to avoid parameter tweaking on the inferencing stage. Empirically, the study has conducted a relatively exhaustive experiment against existing and state of the art anomaly detection models in a semi-supervised learning approach where the assumption is that only normal data is available to provide insight as to how well the model performs under certain quantifiable anomaly detection scenarios.**

Keywords—Neural networks; autoencoders; machine learning; anomaly detection; semi-supervised learning

I. Introduction

Anomaly detection is a widely studied field in computer science dating back to 1887 (Edgeworth) with applications ranging from intrusion detection, fraud detection, medical diagnosis, quality assurance and manufacturing. Anomalies / outliers / novelties are observations that exhibit characteristics that are not part of the usual pattern or expected behavior of what are considered as normal observations. These often result from an erroneous recording of information or fault in producing the information or in some cases, an intended act of disruption as with the case of intrusion in a computerized network system. The definition of what is normal however and consequently what constitutes to an anomaly, is largely dependent on the context of the domain being observed or practiced. For example, medical diagnosis might yield a relatively larger magnitude of deviation to consider something to be malignant rather than benign compared to quality assurance in manufacturing where a relatively smaller magnitude of deviation is observed to consider something to be acceptable or not. In this case, it is a simplistic definition of a large bias occuring or an extremely uneven ratio that defines anomalies. Often, this magnitude of deviation is defined by a domain expert in order to determine the impact of identifying anomalies. Otherwise, the measurement of deviation to define normal from anomalies is empirically defined through numerous experimentation to determine an acceptable value for descrimination which is also influenced by either policy or industry standards. From this, we can say that the study of outlier detection and its applications are generalized as a binary classification problem where observations are categorized as either normal or anomalous and that it is contextualized within the domain at hand. Its value in an operational or business perspective is that the identification of anomalies would always result in actionable items to either fix a system or process to improve the overall output of the application [1]. Mathematically, anomaly detection can be expressed in general using the following equation:

$$f(x) > t \tag{1}$$

where t is some threshold value, x is some unknown data point and $f(x)$ gives a score for the data point. If the score falls above (or below depending on context) of the threshold t then x is considered to be an outlier. The value of t is defined *a priori* usually by a domain expert or through experimentation.

II. Related Works

A. Anomaly Detection

The study of anomaly detection can be generalized as a binary classification problem with labels *normal* and *outliers*. As a classification problem, observations weather labelled normal or anomalous are characterized with a fixed set of attributes. Anomalies are observations whose attributes deviate from normal data based on some acceptable magnitude. In general, anomalies comprise of an extreme minority of the overall data having very low occurences. Other terms for anomalies are *outliers*, *novelties* or *abnormalities*. Regardless of normal or anomalous data, these observations can either be univariate or multivariate in nature.

B. Types of Anomalies

Depending on the problem at hand, an anomaly can be described in either of three major categories – point, contextual or collective anomalies.

1) Point Anomalies: Point anomalies are the simplest type of anomalies that are described as a single data point in n-dimensional space (regardless of it being univariate or multivariate data). Each data point weather anomalous or normal exist based on the values of its attributes. Point anomalies are also the easiest to visualize as they are simply points in the search space of the problem. For example, Fig. 1 illustrates data as point anomalies in the waveform dataset:

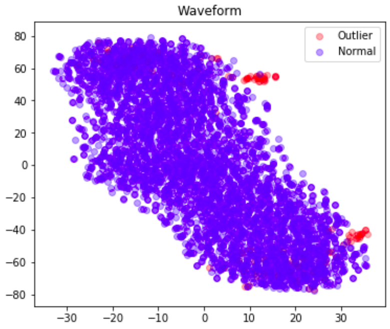


Fig. 1. Point Anomalies Example (TSNE Waveform)

Red points are considered anomalies if they deviate from majority of the normal data points (blue). Normal point data often cluster together as they exist more and exhibit similar measurements compared to anomalous observations. Thus this cluster of normal observations tend to form a cluster boundary. Points that lie beyond the boundary are considered to be outliers.

2) Contextual Anomalies: Also known as conditional anomalies, contextual anomalies are that which are bound to a specific context. A simple example of this would be the context of time. An outlier can be defined based on its measurement occuring at a specific point in time. When formulating an anomaly detection problem, it is integral to first define if the data can be simply described by its attributes (thus it is can be categorized as point anomalies) or if it can be contextualized by some dependent variable making it a contextual anomaly.

3) Collective Anomalies: Collective anomalies are defined to be a grouping of related data instances in relation to the entire data set. Individual observations may not be considered anomalies but if grouped together to form a higher level of observation, then we say that it is a collective anomaly. Defining collective anomalies are approached differently depending on the problem at hand. For example, in intrusion detection, a single observed usage of a protocol in a network may not be considered anomalous but if done in a certain sequence (a collection of protocol usage), such as network packets that use http, then ssh, then ftp protocols can be considered an attack vector (anomalies in this sense are defined as intrusions or attacks in the network).

C. Measuring Anomalies

As of this writing, there is no concrete definition of anomalies or anomaly detection datasets that distinguishes itself from any other binary classification problem. Furthermore, defining what anomalies are in a dataset are subject to the concrete problem at hand. Emott et. al however has proposed a set of measures to define how anomalies are measured in a local context. In their paper *Systematic Construction of Anomaly Detection Benchmarks from Real Data*, their study proposed four quantitative measures for defining anomalies relative to nominal data [2]. Of the four, three were implemented. Given a ”parent” or ”mother” dataset, it is possible to derive anomaly detection subsets based on difficulty constrained with a K parameter relating to the intended ratio of anomalies against a dataset. The three implemented measures for a given outlier data point are as follows. The fourth measure, *Feature Relevance* wasn't included in the paper and was open for interpretation and implementation. A portion of this study contributes to the implementation of this measure and the reasoning for the chosen approach. In depth discussion on the computation for the implemented scores are discussed in the methodology section.

D. Methods in Anomaly Detection

Defining a solution or training a model for anomaly detection can be categorized as either parametric or non-parametric. Regardless of approach, each category has its own share of advantages and disadvantages. Both categories are considered to be statistical approaches in anomaly detection.

1) Parametric Models: Anomaly detection models are considered to be parametric models if the problem assumes that the data being observed or generated follow a specific distribution. There are two key components for parametric model approaches. First would be the parameters θ for the assumed distribution. Second would be the probability density function $f(x, \theta)$, where x is an observation. The goal of parametric models is to derive or estimate the values of the parameters from the data itself where the parameters are largely dependent on the type of distribution assumed. The probability density function then outputs a score depicting how fit some observation x is given the estimated parameters of the distribution.

An example of a parametric model would be Gaussian based models. In this example, the structure of the data assumes to be gaussian in nature (exhibiting the normal distribution) that is, a symmetrical distribution where the mean is central and that data nearer to the mean occurs more frequently than data farther away from the mean exhibiting a bell shaped curve.

The parameters θ of the gaussian distribution are $\{\mu, \sigma^2\}$ which can be derived from training data x using equations 3, 4, 5 and 6. Given some unknown observation, we can check its probability score by fitting its attributes to the distribution's probability density function. The anomaly detection model can now be expressed using equation 2 wherein the unknown data $\hat{x}$ is considered anomalous if its probability value falls below some defined threshold t.

$$f(\hat{x}|\theta) < t \tag{2}$$

$$\forall x \in \{1...I\} x_{\mu_i} = \mu_i = \frac{\sum_{n=1}^{N} x_i}{N} \tag{3}$$

$$\forall x \in \{1...I\} x_{\sigma_i} = \sigma_i^2 = \frac{\sum_{n=1}^{N} (x_i - x_{\mu_i})^2}{N} \tag{4}$$

$$f(\hat{x}_i | \mu_i, \sigma_i^2) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp(\frac{-(\hat{x}_i - \mu_i)^2}{2\sigma_i^2}) \tag{5}$$

$$f(\hat{x}, \mu, \sigma^2) = \forall \hat{x} \in \{1...I\} \prod_{i=1}^{I} f(\hat{x_i} | \mu_i, \sigma_i^2) \tag{6}$$

As opposed to parametric models, non-parametric models allow the data to determine the underlying structure and boundaries for classification. No distribution is defined *a priori* unlike parametric models where it is largely based on an assumed distribution and its respective parameters.

2) Classical Outlier Detection Models: The following models are known to be used for outlier detection in literature and have served as standard benchmarks for newer models:

- Isolation Forest [3]
- Local Outlier Factor [4]
- Robust Covariance [5]

These models will be considered as part of performance measures against this study's proposed model.

3) One-Class Support Vector Machines: The One-Class SVM classification method is an extension of the original support vector machine classification algorithm as developed by Vapnik [6]. This approach however does not require data to be labeled as the algorithm returns a function that samples a small region from the probability distribution of the data that serves as the probability density of the training data. Because of this, One-Class SVM is categorized under the parametric class of anomaly detection algorithms. The function returns $+1$ for data points within the subregion and -1 elsewhere. The minimization function of One-Class SVM is slightly different from the original function and is characterized by equation 7.

$$\begin{aligned} \min_{w,\xi_i,\rho} \frac{1}{2}\|\mathrm{w}\|^2 + \frac{1}{Cn}\sum_{i=1}^{n}\xi_i - \rho \\ \text{subect to:} \\ (w \cdot \phi(x_i)) \geq \rho - \xi_i \text{ for all } i = 1, ..., n \\ \xi_i \geq 0 \text{ for all } i = 1, ..., n \end{aligned} \tag{7}$$

where ϕ is the kernel function to project data to a higher dimensional space, ξ_i are slack variables and C, as opposed to the original, decides the smoothness giving it a solution that a) sets an upper bound on the fraction of outliers and b) sets a lower bound on the number of training examples considered as support vectors. The solution creates a hyperplane characterized by w and ρ which has the maximum distance from the point of origin of the feature space and separates the data points from the origin. As such, this can be categorized as an unsupervised learning algorithm since it takes into account all data points regardless of label. The mathematical equations for this algorithm are explained more in detail in [7].

4) Autoencoders for Anomaly Detection: Autoencoders are neural network models whose output is the same as its input. It approximates how to reconstruct the input by first compressing the data into lower dimensional space to represent a more generalized version of the input in what is called the encoding process. The lower dimensional representation of data, otherwise known as the latent layer as seen in Fig. 2, is then forwarded to the output layer which has the same dimensionality as the input in what is called the decoding process. thus this neural network looks for correlations of features in a data set while taking advantage of non-linear properties of neural networks.

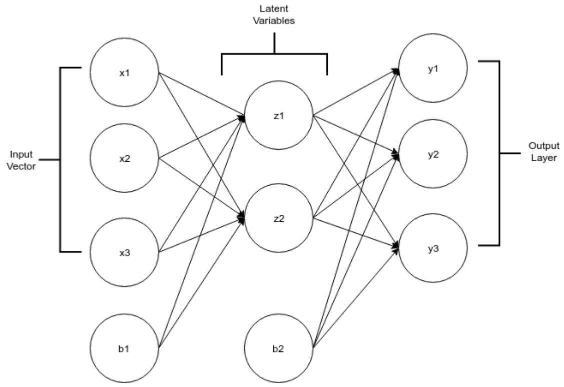


Fig. 2. Autoencoder Neural Network

Mathematically, the approximation of some input data x by the autoencoder model can be expressed in equation 8.

$$f(x) \approx x \tag{8}$$

During training, the error score of some input is the distance between the original input and resulting output. This is otherwise referred to as the reconstruction error. The distance or loss function commonly used is the mean of squared errors as computed in equations 9 and 10.

$$\mathrm{MSE}(x, y) = \mathrm{MSE}(x, f(x)) \tag{9}$$

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2 \tag{10}$$

The MSE value expresses the difficulty in reconstruction where the data fed to the network does not conform to what the model has learned from normal data. Thus if the value exceeds some threshold t, it can be concluded to be an outlier. This can be expressed in equation 11.

$$\mathrm{MSE}(x, f(x)) > t \tag{11}$$

Applications of autoencoders in anomaly detection has been used extensively in the field of performance assessment in computing systems such as [8]. It has also been applied to numerous outlier benchmarking datasets where outlier ratio for validation falls below 30%. This can be seen in [9] wherin Yoshiao et. al proposed a method for estimating reconstruction capabilities of the autoencoder by disregarding high reconstruction errors produced by the model during mini-batch training resulting in partially selecting sets of training results to update the model during back propagation. This however only attempts to address the efficiency of training the autoencoder models while still maintaining a high enough accuracy as compared to standard autoencoders, autoencoder ensembles as well as One-Class SVM.

5) One-Class Neural Networks: The One-Class Neural Network as popularized by Chalapathy et. al., is typical feed forward neural network with a single node as its output [10]. However, this method's novelty can be described in its objective function which takes inspiration from One-Class SVM as well as its utilization of autoencoders. The objective function can be described in equation 12.

$$\min_{w,V,r} \frac{1}{2}\|\mathrm{w}\|_2^2 + \frac{1}{2}\|V\|_F^2 + \frac{1}{v} \cdot \frac{1}{N} \sum_{n=1}^{N} \max(0, r - (w, g(VX_n))) - r \tag{12}$$

The key insight of the objective function is to replace the dot product from One-Class SVM's $(w, \phi(X_n))$ with the dot product $(w, g(VX_n))$ where V is the weight matrix from input to hidden layer that is optimized using an autoencoder. The change here allows transfer learning from an autoencoder model to be able to learn how the data points are reconstructed before applying it to a feed forward neural network. The values derived from the autencoder w and V are then used to optimize r which is theoretically the v-quantile of the array $(w, g(Vx_n))$. After getting the value of r, a score can be derived using equation 13:

$$S_n = \hat{y}_n - r \tag{13}$$

where $\hat{y}_n$ is the output of the feed forward network for data point n. If S_n is greater than or equal to 0 then x_n is considered normal. Else, the point is said to be anomalous.

E. Well Known Autoencoder based Anomaly Detection Models

1) Unsupervised Novelty Detection using Deep Autoencoders with Density based Clustering: Deep autoencoders with density based clustering (DADBC) is a system developed by Amarbayasgalan et. al. to solve the anomaly detection problem in a fully unsupervised fashion. The main steps of the method is a) dimensionality reduction and b) novelty identification through clustering [11]. An illustration of the system can be seen in Fig. 3.

F. Problems in Anomaly Detection

In this research, we try to address key problems in anomaly detection. These have been identified both in literature as well as observations in the nature of existing solutions weather it be parametric or non-parametric. The fundamental problems are identified as follows:

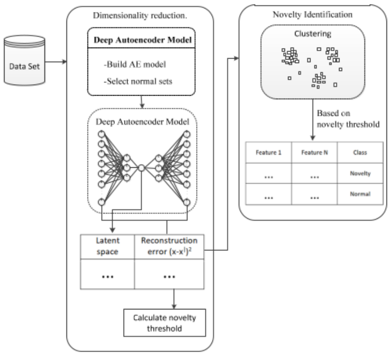


Fig. 3. Deep Autoencoders with Density based Clustering

1) If the model for anomaly detection contains parametric properties, there is heavy reliance on the assumed statistical distribution the data takes up.
2) Anomaly detection applications are largely contextual. In a non-parametric statistical setup, configuration of bins and clusters to profile data points have to be defined by a domain expert or empirically validated through experimentation. A wide variety of anomaly detection models require the definition of some threshold parameter to indicate the magnitude of variety between normal and anomalous data. This threshold is often defined by the practitioner or expert.
3) Anomaly detection datasets have a significantly low volume of data that are considered to be anomalous. In the preliminary results of this study, available standard anomaly detection datasets range from as little as 0.4% to at most 30% of data as anomalies making supervised learning techniques not viable. As such, this research only considers semi-supervised and unsupervised models for anomaly detection.

III. Methodology

This study's methodology is divided into two main parts. The first part contains the processes involved in defining quantitative measures of anomaly detection datasets which is largely based on Emott et. al's work on systematic construction of anomaly benchmarks. The main differences are the implementation of the feature relevance metric, removal of the constraint on selecting a user defined K relative frequency value and providing a global semantic variation score considering multidimensional nature of the data points. This modified approach in providing quantifiable features for anomaly detection datasets benchmarks is referred to as *categorical measures*. The second part of the methodology involves the construction of a neural network autoencoder based approach in solving the problem of anomaly detection. The novelties of the model lie in its non-parametric approach in defining a threshold value during inferencing allowing it to adapt to the patterns exhibitted by the data set itself and improving its performance with a stochastic component and a new loss function that prevents it from overfitting.

A. Categorical Measures

Given a data set, Emott et. al defined a set concrete measurable attributes to its outliers. An anomalous point can be measured according to its *a) point difficulty*, *b) semantic variation* and *c) feature relevance*. Although originally the study for these measures was intended to generate anomalies, our study modifies it in order to give quantitative features to an anomaly detection dataset D which can be simply expressed by equation 14:

$$D = \{f_1, f_2\} \tag{14}$$

f_1 corresponds to the ratio of anomalies present in the dataset relative to normal points whereas f_2 corresponds to the dataset's *semantic variation* score. Both these values are constrained to the following:

1) $f_1 \leq 0.05$ (5% contamination at most)
2) $f_2 \leq 1.5$ (score should be at most 1.5)

1) Contamination Ratio (f_1): The contamination ratio is allows a dataset to have the same characterization of anomaly detection situation as seen in literature to express its rarity of occurrence such as those in [9], [4], [5] and [2]. However the datasets used in these studies did not have consistent contamination ratio and only went to what was available in the dataset. In our experiments, given an initial dataset, we sample anomalies to force a ratio of 1%, 2%, 3%, 4% and 5% contamination in order to test the behavior of models under these circumstances. It is important to take note that there is a possibility that the sampled subset could maintain a ratio less than 5% but still have a semantic variation score higher than 1.5. In this case, we do not consider such a subset to be an anomaly detection dataset and resample until both the contamination ratio and semantic variation score is satisfied.

2) Semantic Variation Score (f_2): Semantic variation refers to the measure of how an anomalous data point is widely dispersed from the nominal group and fellow outliers. This means that the measure of dispersal should consider both labels of data points in terms of relative distance. Emott et. al chose a random seed point and computed $K-1$ data points that are closest to it using euclidean distance. This study's approach does not perform any random seeding since we compute a global score for all datapoints within a dataset that has already been subsampled to meet the first constraint of contamination ratio. This is also known as normalized clusterdness measure which can be expressed by equation 15:

$$\log(\frac{\sigma^2_{\text{normal}}}{\sigma^2_{\text{anomaly}}}) \tag{15}$$

where:

1) σ^2_{normal} is the variance of normal data
2) $\sigma^2_{\text{anomaly}}$ is the variance of anomaly data

To deal with multi-dimensional data, we compute for the variance σ^2 of anomaly (or normal) data points X by taking its covariance matrix using equation 16:

TABLE I. INVALID ANOMALY DETECTION DATASET

Dataset	Semantic Variation Score
Iris Versicolor Anomaly	1.63198
Iris Virginica Anomaly	1.57
Iris Setosa Anomaly	1.57212

$$\mathbf{Var}(X) = \mathbf{E}[(X - \mathbf{E}(X))(X - \mathbf{E}(X))^T] \tag{16}$$

We then take trace of the covariance matrix to give us the overall variance using equation 17:

$$\sigma^2 = \text{tr}(\mathbf{Var}(X)) \tag{17}$$

Although in the original paper, it was suggested that such a score did not provide any means of what threshold value constitutes to anomalies (also noting that it was used as a measure for ideal anomaly generation which is a separate study in itself). The data used by most literature suggests that datasets that exhibit an SV score of less than 1.5 tend to be the ones used for benchmarking anomaly detection models thus the constraint was applied together with the context of contamination ratio. An SV score greater than 0 suggests clustered anomalies as opposed to a score less than 0 which suggests more scattered data points. The intuition is that having more clustered anomalies has a higher chance of exhibiting a pattern since it's not simply an incorrect recording of data but also can be repeated with minimal variation within the class. This makes it difficult for density based methods to perform well as clustered points tend to be treated as normal instead of anomaly classes [2].

It is important to note that with these measures and constraints, anomaly detection datasets are not simply defined by the rarity of anomalies that occur but how clustered they are to reflect its difficulty in terms of detection. For example, Table I shows the popular Iris dataset used in [12] is known to be linearly separable but is also treated as an anomaly detection dataset by defining anomalies as part of the tail ends of an interquartile range. If this is the case, we can take a sample of such defined anomalies with a contamination ratio of 5% that exists within the tail ends of interquartile ranges of a class and treat the rest as normal but still have a high semantic variation score. Such conditions will not be trated as anomaly detection benchmarks.

B. Non-Parametric Stochastic Autoencoder Scoring

The proposed method discussed in this research primarily tries to solve for an acceptable t that will yield reliable results for anomaly detection as defined by the expression $f(x) > t$. The method is largely based on training an autoencoder model in order to address non-linear data and providing a new scoring mechanism that takes into account a non-parametric adaptive value assuming no distribution for the data as well as a new loss function that acts as an adaptive regularizer to prevent overfitting. To improve performance, a stochastic process is included and emperically proven to work better compared to using a vanilla autoencoder. Thus, this study names the model the *Non-parametric Stochastic Autoencoder Scoring* model.

1) Autoencoder Setup: The first step in the method is to train a standard autoencoder. The topology of the autoencoder will be a shallow one that is it will only consist of three layers, the input, the latent and the output layer which has the same dimensionality as the input. The number of latent variables is set to be approximately 3/4 of the original input size. The initial weights of the model are also set symmetrically that is given the initial weights W_{input} connecting the input to the latent layer, the initial weights W_{output} from the latent to the output layer will just be the transpose of W_{input}. Thus, $W_{output} = W_{input}^T$.

For the latent layer, the activation function used was the rectified linear unit (ReLU) given by the equation 18:

$$A_{latent}(x) = max(x, 0) \tag{18}$$

For the output layer, the activation function used was the sigmoid activation function 19:

$$A_{output}(x) = \frac{1}{1 + \exp(-x)} \tag{19}$$

2) Autoencoder's Mean Loss: The new loss function, *Autoencoder's Mean Loss*, to be used for training is a variation of mean squared errors. This loss function is composed of the sum of two terms. The first one is the standard mean squared errors and the second term is the sum of mean squared errors relative to each dimensionality's mean. This is unique to autencoders as the parameter μ_i can be derived initially from the data set and y_i is simply x_i unlike fully supervised learning methods where y has to be known. This property allows the loss function to adapt to

$$\epsilon(x) = \frac{1}{n}\sum_{i=1}^{n}(\hat{x_i} - y_i)^2 + \frac{1}{n}\sum_{i=1}^{n}(\hat{x_i} - \mu_i)^2 \tag{20}$$

A loss function is said to be valid if it is proven to be convex. This is considered to be a valid loss function as proven mathematically since the sum of two convex functions is also convex. Standard back propagation was used for training the autoencoder model.

3) Stochastic Latent Noise: A relevant part of the model is adding stochasticity to the latent layer of a trained autoencoder which is referred to as *Stochastic Latent Noise*. The intuition behind this is that if the model is trained only using positive / nominal data points, then determining a threshold to discriminate reconstructed points with high residual errors will still yield to a lot of false negatives. According to the original definition of outliers by Edgeworth, a possible reason for anomalies is the error of observation is the joint result of considerable, but finite, number of small sources of error. This concept is applied to the latent set Z from a trained autoencoder by determining its statistical properties μ_{z_i} and σ_{z_i} and sampling a normal distribution from it with parameter K corresponding to the size of data points found in the distribution. Another parameter d is given referring to the number of dimensions the sampling is applied to. Finally a parameter r is given to represent the ratio of random points taken at the tail end of each distribution to replace the value at latent index i. Using the trained autoencoder, the decoder part is then ran against these synthetic anomalies to get the reconstructed version at the original dimensional space. These values are then added to the histogram of residual errors to determine t as explained in the next section.

Another way of looking at *Stochastic Latent Noise* is that it's cyclical process of artificial reinforcement learning unique to autoencoders. Traditional machine learning algorithms rely on data to dictate the value of weights whereas reinforcement learning lets the model itself dictate the data. In a similar fashion, the autoencoder first learns of the approximation of the identity function from the data set and based on the weights forces some random aspect to its latent layer representing a compressed version of the data. This is then projected to reconstructed data or randomly synthesized instances that resemble the data as understood by the autoencoder model. An illustration of this can be seen in Fig. 4.

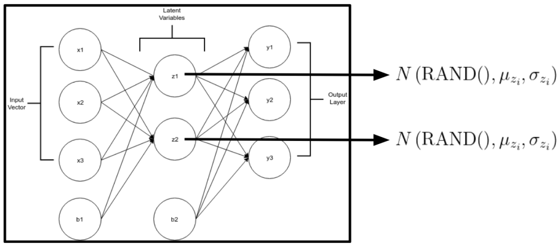

Latent Variables
Input Vector
x1
x2
x3
b1
z1
z2
b2
y1
y2
y3
Output Layer
$N(\text{RAND}(), \mu_{z_i}, \sigma_{z_i})$
$N(\text{RAND}(), \mu_{z_i}, \sigma_{z_i})$

Fig. 4. Stochastic Latent Noise

4) Determining the Threshold: The value of t is taken as the midpoint of the identified bin threshold for a histogram of residual errors as built by the reconstructed data points from the trained autoencoder including the added synthetic points from the previous section.

Histogram of Residual Errors: From the set of $\hat{X}$, the set of residual errors X_ϵ are used to build a histogram of q bins with starting range $\min(X_\epsilon)$ and ending in $\max(X_\epsilon)$. Each bin has its own local minimum and maximum values whose interval is given by equation 21.

$$\text{interval} = \frac{\max(X_\epsilon) - \min(X_\epsilon)}{p} \tag{21}$$

The value of p is automatically set using Freedman-Diaconis rule as seen in equation 22:

$$p = 2(\frac{\text{IQR}(X_\epsilon)}{\sqrt[3]{n}}) \tag{22}$$

An example result of the generated histogram of residual errors is seen in Fig. 5.

Determining t From Histogram of Residual Errors: To solve for t, given the histogram of residual errors, we first apply a head-tail break function (HTB) to return an array of possible break points. The following shows an implementation of the HTB algorithm:

Listing 1: HTB Python Implementation

```
def htb(data):
```

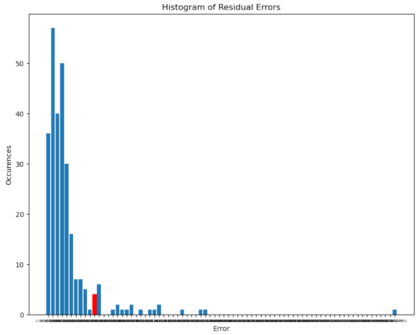


Fig. 5. Histogram of Residual Errors

```
outp = []  # array of break points

def htb_inner(data):
    dl = float(len(data))
   dm = sum(data) / dl
   head = [_ for _ in data if _ > dm]
   outp.append(dm)
   c_head = len(head) > 1
   c_thresh = len(head) / dl < 0.40
   while c_head and c_thresh:
       return htb_inner(head)

htb_inner(data)

return outp
```

For each break point, we get a candidate threshold by passing the bin configuration which is an array of values representing the bins (i.e. bin_0 and bin_1 defines the minimum and maximum value ranges of the first bin) and the occurrence counts in the form also of an array (i.e. in this case, the size of the occurrence counts). The final threshold is then naively selected to be the minimum from the array of candidate thresholds.

The implementation of fetching the threshold from the histogram is given by the following:

Listing 2: Fetch Candidate Threshold

```
def fetch_threshold(bs, counts, bp):
    index = 0
   m = 999999999
    t = -1

    for i in range(len(counts)):
       if abs(counts[i] - bp) <= m:
          m = abs(counts[i] - bp)
          index = i
          l = ((bs[i + 1] - bs[i]) / 2)
          r = bs[i]
          t = l + r
    return t
```

Now that t is determined, to classify an unknown data point x as either an outlier or an anomaly, x is passed to the trained autoencoder and its corresponding x_ϵ is taken. If the value is greater than t, then there is reason to believe that the model had a hard time reconstructing it and thus flagging it as an anomaly. A visual example of this can be seen in the illustration in Fig. 6.

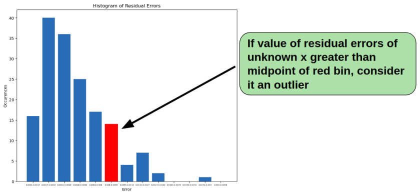
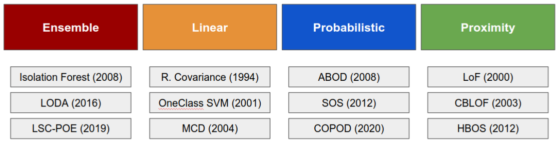


Fig. 6. Threshold Example

C. Methods for Benchmarking

To test the performance of the proposed method against existing, it was compared against 12 standard and state of the art methods for anomaly detection (summarized in Fig. 7 together with year of release) in a semi-supervised fashion where parameters of these methods were manually optimized to get the best possible results. These methods are grouped according to the nature of their methodologies as follows:



Ensemble	Linear	Probabilistic	Proximity
Isolation Forest (2008)	R. Covariance (1994)	ABOD (2008)	LoF (2000)
LODA (2016)	OneClass SVM (2001)	SOS (2012)	CBLOF (2003)
LSC-POE (2019)	MCD (2004)	COPOD (2020)	HBOS (2012)

Fig. 7. Summary of Methods

1) Ensemble:

1) Isolation Forest (ISO-F) [3]
2) Lightweight On-line Detector of Anomalies (LODA) [13]
3) Locally Selective Combination of Parallel Outlier Ensembles (LSCP) [14]

2) Linear:

1) Minimum Covariance Determinant (MCD) [5]
2) Robust Covariance (ROB-COV) [15]
3) OneClass SVM (OC-SVM) [6]

3) Probabilistic:

1) Angle-Based Outlier Detection (ABOD) [16]
2) Stochastic Outlier Selection (SOS) [17]
3) Copula-Based Outlier Detection (COPOD) [18]

4) Proximity:

1) Local Outlier Factor (LOF) [4]
2) Clustering-Based Local Outlier Factor (CBLOF) [19]
3) Histogram-Based Outlier Score (HBOS) [20]

D. Evaluating Performance

To evaluate the results of the proposed method against existing categories of methods mentioned in the previous section, for each sampled dataset (80 datasets in total), 10 simulations were conducted (total of 800 runs: 16 initial datasets partitioned to five different contamination configurations) with the MCC (Matthew's Correlation Coefficient) score extracted and applied to a two tailed t-test score with a significance level of 0.05. This allowed the study to determine with confidence if the proposed method is either:

1) Significantly better than other methods
2) Better but not significantly better than other methods
3) Poorer (at least one method is better than the proposed method) but not significantly poorer
4) Significantly poorer (at least one method is significantly better than the proposed method)

MCC (Matthew's Correlation Coefficient) was preferred over the commonly used F1-score due to the mathematical properties mentioned in [21] making it more ideal for anomaly detection with extremely biased data. For each initial dataset, 70% of normal data was randomly sampled and used for training with the remaining 30% used for evaluation. MCC is a measure of correctly predicting both majority of nominal and majority of anomalies. A value of -1 is reached for a perfect misclassification. A value of 1 is reached for a perfect classification. A value of 0 indicates a performance the same as a coin toss. This score can be computed by equation 23

$$\mathrm{MCC} = \frac{\mathrm{TP} \cdot \mathrm{TN} - \mathrm{FP} \cdot \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP}) \cdot (\mathrm{TP} + \mathrm{FN}) \cdot (\mathrm{TN} + \mathrm{FP}) \cdot (\mathrm{TN} + \mathrm{FN})}} \tag{23}$$

E. Datasets

Given the quantitative measures to define anomaly detection datasets, the following datasets were sampled from 16 datasets to meet the constraints mentioned in order to come up with a total of 80 datasets as shown in Tables II, III, IV, V, VI, VII, VIII, IX, X, XI, XII, XIII, XIV, XV, XVI and XVII. These tables show the values for each dataset's *categorical measures* in terms of contamination ratio and semantic variation score that characterizes it as an anomaly detection problem which also meets the constraints as mentioned previously.

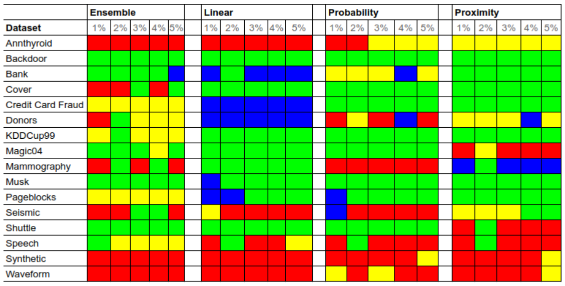
IV. Results and Analysis

A. Overview

A summary of the results is illustrated in Fig. 8 where:

- Green cells 🟢 mean that the proposed method did significantly better
- Blue cells 🔵 mean that the proposed method did better but not significantly better
- Yellow cells 🟡 mean that the proposed method did poorer but not significantly poorer
- Red cell 🔴 mean that the proposed method did significantly poorer

To better assess the results of the proposed model, each dataset was treated as a point in two dimensional space where each dimension corresponds to the a dataset's categorical measure. This allowed us to see if the proposed method performs relatively well in a certain range as bound by the values of categorical measures. Consequently, this allowed us to see at what range the method fails and where a certain category of outlier detection models would prove to be more useful. The next few sections would go through each anomaly detection category and how the proposed method performed comparatively.



	Ensemble					Linear					Probability					Proximity				
Dataset	1%	2%	3%	4%	5%	1%	2%	3%	4%	5%	1%	2%	3%	4%	5%	1%	2%	3%	4%	5%
Annthyroid	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
Backdoor	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Bank	🟩	🟩	🟩	🟩	🟦	🟦	🟩	🟦	🟦	🟦	🟨	🟨	🟨	🟦	🟨	🟩	🟩	🟩	🟩	🟩
Cover	🟥	🟥	🟩	🟥	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Credit Card Fraud	🟨	🟨	🟨	🟨	🟨	🟦	🟦	🟦	🟦	🟦	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Donors	🟥	🟩	🟨	🟨	🟨	🟦	🟦	🟦	🟦	🟦	🟥	🟨	🟥	🟦	🟥	🟨	🟨	🟨	🟦	🟨
KDDCup99	🟨	🟩	🟨	🟨	🟨	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Magic04	🟩	🟩	🟩	🟨	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟥	🟨	🟥	🟥	🟥
Mammography	🟥	🟩	🟥	🟩	🟥	🟩	🟩	🟩	🟩	🟩	🟥	🟥	🟥	🟥	🟥	🟦	🟩	🟦	🟦	🟦
Musk	🟩	🟩	🟩	🟩	🟩	🟦	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Pageblocks	🟨	🟨	🟨	🟨	🟨	🟦	🟦	🟩	🟩	🟩	🟦	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Seismic	🟥	🟥	🟩	🟩	🟥	🟨	🟥	🟥	🟥	🟥	🟦	🟥	🟥	🟥	🟥	🟨	🟨	🟨	🟩	🟩
Shuttle	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟥	🟥	🟥
Speech	🟩	🟨	🟨	🟨	🟨	🟥	🟩	🟥	🟥	🟨	🟥	🟩	🟥	🟥	🟥	🟥	🟩	🟥	🟥	🟥
Synthetic	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟨	🟥	🟥	🟥	🟥	🟨
Waveform	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟨	🟥	🟨	🟥	🟥	🟥	🟥	🟥	🟥	🟨

Fig. 8. Overview of Experimental Results

B. Performance vs Ensemble Methods

The projected categorical measures can be seen in Fig. 9 where majority of the datasets that the proposed method did significantly better against fall under the range of -0.75 at minimum and 0.60 at maximum in terms of semantic variation score regardless of outlier ratio.

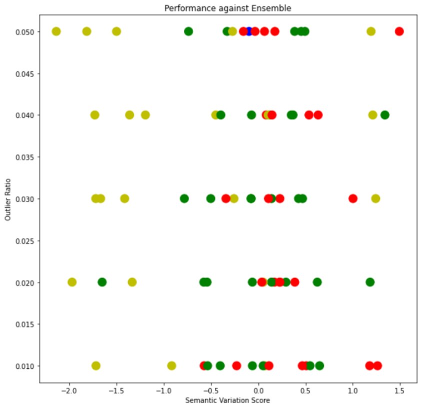


Fig. 9. Performance vs Ensemble

Ensemble methods require calibration of the sub-algorithms used and parameter tuning for each which makes it dependent on which exact algorithms are part of the ensemble. These methods tend to perform better than the proposed method semantic variation scores are negative contrary to the other method categories as seen in the next section. It is still noted though that the proposed method doesn't require such dependency on other algorithms making it less complex to calibrate.

C. Performance vs Linear Methods

The projected categorical measures can be seen in Fig. 10 where majority of the datasets that the proposed method did significantly better against has a semantic variation score of at most -0.50 regardless of outlier ratio. Although there are some instances beyond -0.50 that the proposed method can outperform linear methods, in most cases linear methods work significantly better.

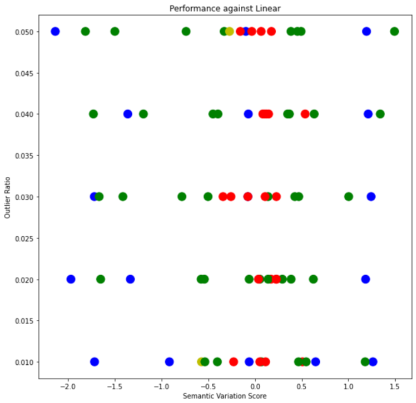


Fig. 10. Performance vs Linear

As with the results of proximity and probability based methods, linear methods fail to outscore the proposed methods in the negatively scored datasets in terms of semantic variation. This suggests that neural network based models can take advantage of non-linear properties that are not otherwise captured by more statistical properties of linear methods. In addition, it suggests that because of the negative values, the anomalies present in such datasets tend to be more scattered (less clustered) with more variation expressing a non-linear behavior.

D. Performance vs Probability Methods

The projected categorical measures can be seen in Fig. 11 where majority of the datasets that the proposed method did significantly better against has a semantic variation score of at most -0.50 regardless of outlier ratio. As with linear methods, compared to probability methods, the method may at times perform better if the semantic variation score is greater that -0.50 but in most cases it performs either poorer or significantly poorer in that domain.

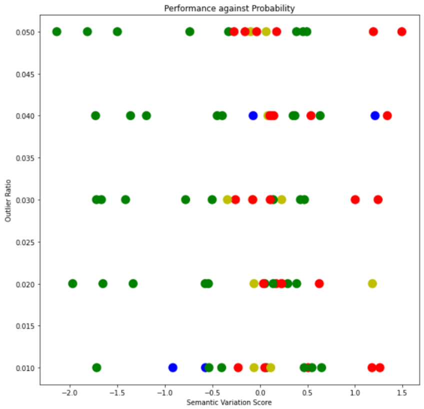


Fig. 11. Performance vs Probability

E. Performance vs Proximity Methods

The projected categorical measures can be seen in Fig. 12 and in similar comparison to linear and probability based methods, the proposed method performs significantly better than proximity based methods if the semantic variation score of the dataset to be evaluated is less than -0.50 regardless of outlier ratio.

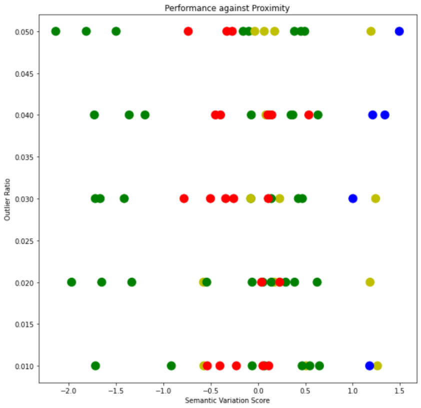


Fig. 12. Performance vs Proximity

As with the previous categories, the proposed method tend to perform better when the anomalies are less clustered. As with the nature of proximity based methods which is largely based on distance metrics such as nearest neighbors, a large variance of anomalies present will tend to fail as seen in the datasets that are negatively scored in terms of semantic variation.

F. General Discussion

As a general statement, it can be seen that in terms of the MCC performance against datasets constrained with the categorical measures, the proposed model can perform significantly well regardless of outlier ratio in most cases if the semantic variation score leans towards negative values, specifically -0.05. As mentioned by Emott et. al. [2], a dataset quantitatively characterized with a negative semantic variation score suggests that anomalies are more scattered in nature due to having a higher value in terms of variance that exceeds that of the variance of normal data points (taking the log of a value less than 1 as computed by semantic variation, will yield a negative value as a result of the variance of anomalies is higher than that of normal points).

Restricting the contamination ratio to at most 5% keeps it aligned with the concept of rarity as what most literature in anomaly detection states. It is important to note however that not all the initial datasets (that is, the dataset's original form without the sampled subsets for evaluation with applied categorical measures) necessarily comply with the constraints of categorical measures. Violating these constraints would not constitute to proper anomaly detection studies as it will have a positive semantic variation score (suggested threshold of the study is 1.5) that suggests occurrences of anomalies would tend to form certain patterns which does not appropriately reflect the description of anomalies in literature (incorrect data production and rare occurrence) and therefore could be a means of it being treated as either a biased two class classification problem wherein the minority class is not considered an anomaly.

In terms of application, if a scenario in the real world can determine anomalies that tend to cluster in such a way, then the proposed method is ideal for it in a sense that it will allow researchers / practitioners / stakeholders to take advantage of not needing to manually adjust inferencing parameters (reconstruction error threshold) upon usage. Being a neural network based model, it does seem to validate the approximation of non-linear functions that tend to address the unpredictable behavior of occurrences of anomalies. On the flip side however, anomalies that tend to be more clustered together thus exhibiting a higher semantic variation score (greater than -0.50 or a positive value for that matter), would still fall under the notion that the effectiveness of an anomaly detection algorithm will still be on a case to case basis.

Applying categorical measures allowed us to quantify the nature of an anomaly detection dataset and observe the behavior of various methods in terms of its MCC performance. With these measures, we can derive insights as to which possible range of values certain methods can work well against as opposed to the general characteristics of the anomaly detection problem where anomalies are simply said to rarely occur or fall within a certain deviation from what is normal. Such deviation can't be quantified as anomalies themselves are subject to the domain where data is captured or produced thereby not allowing a standard objective definition of it. With categorical measures, we can at the very least, quantify the structure of the dataset in relation to the existence of anomalies and its relation to normal data points.

V. Conclusion

The effectiveness of classification methods for anomaly detection are traditionally dependent on looking for the best parameters that fits a certain scenario or dataset at hand. This is primarily due to the fact that anomalies themselves cannot be quantified or given objective characteristics for all domains. Even in current literature, anomalies are restricted to definitions that vary from scenario to scenario in terms of rarity and what value to be used to express magnitude of deviation from what is normal. Given these, this study pushes the definition further by providing a quantitative definition not to the anomalies themselves but in context of datasets that are considered to be an anomaly detection problem so as to give insights as to how well anomaly detection methods perform. The study refers to these as *categorical measures*.

Anomaly detection datasets (specifically point anomalies and not context or time series type) characterized with *categorical measures* in this study was constrained to the following conditions:

1) The point anomaly instances comprise of at most 5% of the evaluated dataset to ensure its rarity of occurrence.
2) The dataset's semantic variation score does not exceed 1.5, since higher scores generally imply that there is clustering among the point anomaly instances, and this may indicate the presence of a non-anomalous process that generated the "anomaly" instances.

Both conditions have to be satisfied before a dataset can be considered as part of an anomaly detection study. Violating these constraints would not constitute to proper anomaly detection as the score suggests certain clusters forming depicting a pattern where methods that are density based are more likely to fail. Since the first constraint expresses rarity, it is still a ratio and one can still derive a subsample that meets the first criteria but fails in the second. Therefore, both have to be met to constitute to a valid anomaly detection problem instead of just leaning on the notion of rarity.

With the proposed method with a neural network autoencoder model as its base, it can be modified to be adaptive in terms of automatically setting the parameter in the inferencing stage that allows scoring of anomalies to not be dependent on prior information such as assumed distribution on the data itself or domain expert. This was done by allowing the discovery of the threshold value, a parameter that has traditionally been set manually by neural network based models, to be naturally formed by the distribution of reconstruction errors which assumes to exhibit a long tail distribution. Compared to existing methods that have been tested throughout this study, the proposed method performed comparatively well in an identified domain of negatively scored semantic variation value of anomaly detection datasets suggesting that it works well in scenarios with more variation in the anomalies present. This has been proven through experimentation on the MCC metric where in most cases, the proposed method performed significantly better if the dataset has a score of -0.5 or less in terms of semantic variation even when parameters of the existing methods have been optimized whereas the proposed method had its parameter configured automatically all throughout. Apart from the family of ensemble method where the performance of the proposed method works better in the range of -0.75 to 0.60, as well as the positively scored datasets, it still remains a case to case basis as other methods work better or worse than the proposed method without any evident generalization. This could be up for investigation in future work.

Anomaly detection in general is still an open problem without any standard objective definition. But with the case of this study, the narrative can be progressed towards examining behavior of methods given categorical measures of the datasets themselves to constitute to a working quantitative definition of an anomaly detection problem.

Acknowledgment

The authors would like to thank the Ateneo de Manila University and the University Research Council for supporting this study.

References

[1] M. Salehi and L. Rashidi, "A survey on anomaly detection in evolving data: [with application to forest fire risk prediction]," *SIGKDD Explor. Newsl.*, vol. 20, no. 1, pp. 13–23, May 2018. [Online]. Available: http://doi.acm.org/10.1145/3229329.3229332

[2] A. Emmott, S. Das, T. Dietterich, A. Fern, and W.-K. Wong, "Systematic construction of anomaly detection benchmarks from real data," 08 2013, pp. 16–21.

[3] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ser. ICDM '08. USA: IEEE Computer Society, 2008, p. 413–422. [Online]. Available: https://doi.org/10.1109/ICDM.2008.17

[4] M. Breunig, H.-P. Kriegel, R. Ng, and J. Sander, "Lof: Identifying density-based local outliers." vol. 29, 06 2000, pp. 93–104.

[5] P. Rousseeuw and K. Driessen, “A fast algorithm for the minimum covariance determinant estimator,” *Technometrics*, vol. 41, pp. 212–223, 08 1999.

[6] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep 1995. [Online]. Available: https://doi.org/10.1007/BF00994018

[7] B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt, “Support vector method for novelty detection,” vol. 12, 01 1999, pp. 582–588.

[8] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM Comput. Surv.*, vol. 41, 07 2009.

[9] Y. Ishii and M. Takanashi, “Low-cost unsupervised outlier detection by autoencoders with robust estimation,” *Journal of Information Processing*, vol. 27, pp. 335–339, 01 2019.

[10] R. Chalapathy, A. Menon, and S. Chawla, “Anomaly detection using one-class neural networks,” 02 2018.

[11] T. Amarbayasgalan, B. Jargalsaikhan, and K. Ryu, “Unsupervised novelty detection using deep autoencoders with density based clustering,” *Applied Sciences*, vol. 8, 08 2018.

[12] E. Acuna and C. Rodriguez, “An empirical study of the effect of outliers on the misclassification error rate,” 11 2004.

[13] T. Pevný, “Loda: Lightweight on-line detector of anomalies,” *Mach. Learn.*, vol. 102, no. 2, p. 275–304, feb 2016. [Online]. Available: https://doi.org/10.1007/s10994-015-5521-0

[14] Y. Zhao, Z. Nasrullah, M. Hryniewicki, and Z. Li, “Lscp: Locally selective combination in parallel outlier ensembles,” 01 2019.

[15] A. Stromberg, “Robust covariance estimates based on resampling,” pp. 321–334, 02 1997. [Online]. Available: https://doi.org/10.1016/S0378-3758(96)00051-1Get

[16] X. Li, J. C. Lv, and D. Cheng, “Angle-based outlier detection algorithm with more stable relationships,” in *Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems, Volume 1*, H. Handa, H. Ishibuchi, Y.-S. Ong, and K. C. Tan, Eds. Cham: Springer International Publishing, 2015, pp. 433–446.

[17] J. Janssens, “Outlier selection and one-class classification,” Ph.D. dissertation, 2013, series: TiCC Ph.D. Series Volume: 27.

[18] Z. Li, Y. Zhao, N. Botta, C. Ionescu, and X. Hu, “COPOD: Copula-based outlier detection,” in *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE, nov 2020. [Online]. Available: https://doi.org/10.1109%2Ficdm50108.2020.00135

[19] Z. He, X. Xu, and S. Deng, “Discovering cluster-based local outliers,” *Pattern Recognition Letters*, vol. 24, no. 9, pp. 1641–1650, 2003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167865503000035

[20] M. Goldstein and A. Dengel, “Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm,” 09 2012.

[21] D. Chicco and G. Jurman, “The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation,” *BMC Genomics*, vol. 21, 12 2020.

APPENDIX

TABLE II. ANNTHYROID DATASETS

Dataset	Contamination Ratio	Semantic Variation Score
Annthyroid1	1%	0.508125181
Annthyroid2	2%	0.1707903902
Annthyroid3	3%	0.2287771725
Annthyroid4	4%	0.08389630917
Annthyroid5	5%	0.06585469485

TABLE III. BACKDOOR DATASETS

Dataset	Contamination Ratio	Semantic Variation Score
Backdoor1	1%	0.5475187707
Backdoor2	2%	0.1404898671
Backdoor3	3%	0.1396325904
Backdoor4	4%	0.3682554791
Backdoor5	5%	0.4539561087

TABLE IV. BANK NOTES DATASETS

Dataset	Contamination Ratio	Semantic Variation Score
Bank1	1%	-0.06208764389
Bank2	2%	-0.06267816812
Bank3	3%	-0.07630856422
Bank4	4%	-0.07151140273
Bank5	5%	-0.09810198756

TABLE V. COVER DATASETS

Dataset	Contamination Ratio	Semantic Variation Score
Cover1	1%	0.46640986
Cover2	2%	0.3869515162
Cover3	3%	0.4684336152
Cover4	4%	0.6339205037
Cover5	5%	0.4937923986

TABLE VI. CREDIT CARD FRAUD DATASETS

Dataset	Contamination Ratio	Semantic Variation Score
CCF1	1%	-1.71859332
CCF2	2%	-1.971678786
CCF3	3%	-1.721024117
CCF4	4%	-1.36247447
CCF5	5%	-2.139169117

TABLE VII. DONORS DATASETS

Dataset	Contamination Ratio	Semantic Variation Score
Donors1	1%	1.262872064
Donors2	2%	1.184351675
Donors3	3%	1.243335783
Donors4	4%	1.212204939
Donors5	5%	1.194491206

TABLE VIII. KDDCUP99 DATASETS

Dataset	Contamination Ratio	Semantic Variation Score
KDDCup991	1%	0.06888742085
KDDCup992	2%	-1.653239859
KDDCup993	3%	-1.669393574
KDDCup994	4%	-1.730795094
KDDCup995	5%	-1.501027102

TABLE IX. MAGIC04 DATASETS

Dataset	Contamination Ratio	Semantic Variation Score
Magic041	1%	-0.4032003924
Magic042	2%	-0.5770861584
Magic043	3%	-0.5031427544
Magic044	4%	-0.4501034267
Magic045	5%	-0.3297699596

TABLE X. MAGIC04 DATASETS

Dataset	Contamination Ratio	Semantic Variation Score
Mammography1	1%	1.179545636
Mammography2	2%	0.6250250795
Mammography3	3%	1.003284824
Mammography4	4%	1.341220359
Mammography5	5%	1.493847255

TABLE XI. MUSK DATASETS

Dataset	Contamination Ratio	Semantic Variation Score
Musk1	1%	0.6492543029
Musk2	2%	0.2927728505
Musk3	3%	0.4260056274
Musk4	4%	0.3493787179
Musk5	5%	0.3849209583

TABLE XII. Pageblocks Datasets

Dataset	Contamination Ratio	Semantic Variation Score
Pageblocks1	1%	-0.9167593238
Pageblocks2	2%	-1.335170795
Pageblocks3	3%	-1.41523494
Pageblocks4	4%	-1.1950763
Pageblocks5	5%	-1.816508555

TABLE XIII. Seismic Datasets

Dataset	Contamination Ratio	Semantic Variation Score
Seismic1	1%	-0.5731575761
Seismic2	2%	0.2270706794
Seismic3	3%	-0.07664012976
Seismic4	4%	0.1222953365
Seismic5	5%	-0.1572111807

TABLE XIV. Shuttle Datasets

Dataset	Contamination Ratio	Semantic Variation Score
Shuttle1	1%	-0.5363501714
Shuttle2	2%	-0.5423016379
Shuttle3	3%	-0.7827652121
Shuttle4	4%	-0.3966784238
Shuttle5	5%	-0.7382668293

TABLE XV. Speech Datasets

Dataset	Contamination Ratio	Semantic Variation Score
Speech1	1%	0.05144099108
Speech2	2%	0.05144099108
Speech3	3%	-0.2575904149
Speech4	4%	0.1041183734
Speech5	5%	-0.2727381015

TABLE XVI. Synthetic Datasets

Dataset	Contamination Ratio	Semantic Variation Score
Synthetic1	1%	-0.2289063831
Synthetic2	2%	0.03732289785
Synthetic3	3%	0.1082732279
Synthetic4	4%	0.1454660726
Synthetic5	5%	-0.03449716229

TABLE XVII. Waveform Datasets

Dataset	Contamination Ratio	Semantic Variation Score
Waveform1	1%	0.1132031272
Waveform2	2%	0.2289010261
Waveform3	3%	-0.3424085047
Waveform4	4%	0.5371335689
Waveform5	5%	0.1755232835