Anomaly Detection - Algorithms and Applications

Abstract

It has become increasingly important to study anomalies as they play an important role in different use cases such as detecting fraud, cyberattacks, and etc. We access the performance of different models which can be used for anomaly detection such as Generative Adversarial Networks, Autoencoders, Isolation Forests, and PIDForest for detecting anomalies and train them in an unsupervised approach. For each model, we perform oversampling and undersampling of our datasets to achieve a desired percentage of anomalies within the dataset. We notice that [...]

1 Introduction

Anomaly detection or outlier detection can be defined as the process of identifying instances generated by mechanisms that differ from those generating normal instances (Faria & Vistulo de Abreu, 2019). As defined by Grubbs, "An outlying observation, or outlier, is one that appears to deviate markedly from other members of the sample in which it occurs" (1969). It has become increasingly important to study anomalies further since they are often associated with particular interesting events or suspicious data records (Goldstein & Uchida, 2016). For example, anomalies can indicate insider threats, cyberattacks, machine component failures, or the emergence of cancer cells in normal tissue (Emmott et al., 2013). Depending on the availability of labels in the dataset, supervised, semi-supervised, and unsupervised machine learning techniques have been employed to detect anomalies (Goldstein & Uchida, 2016). More recently, it was shown that deep learning techniques outperform traditional machine learning as the scale of data increases (Chalapathy & Chawla, 2019). To assess the performance of these techniques, a score is often used as an output to rank the observations (Goldstein & Uchida, 2016). A novel deep learning approach to anomaly detection uses Generative Adversarial Networks (GANs) – these have achieved state-of-the-art performance in high-dimensional generative modeling (Deecke et al., 2018). GANs aims to learn the underlying distribtion of the observations in order to generate new samples that fall within the same distibution as the observations. Another deep learning structure which can be used for anomaly detection includes autoencoders (AEs). AEs are an unsupervised approach that constructs representation based on non-linear combinations of input features, and where the non-linearity is introduced by way of some non-linear activation function (LeCun et al., 2015; Zhou & Paffenroth, 2017). We will also evaluate the performance of Isolation Forests (iForests) and PIDForests, two common non-deep learning based approaches for anomaly detection. While various comparative studies have demonstrated the strengths and weaknesses of anomaly detection algorithms (Liu et al., 2008; Emmott et al., 2013; Chalapathy & Chawla, 2019), none, to our knowledge, have compared GANs, AEs, iForests and PIDForests for anomaly detection. As such, our study aims to compare the performance of deep learning approaches (i.e., GANs and AEs) with the likes of iForests and PIDForests on anomaly detection in different types of datasets.

1.1 Algorithms

1.1.1 Generative Adversarial Networks

Generative Adversarial Networks(GANs) are a type of neural network used for generating new samples based off a distribution, thus it is a generative model. Generative models are models in which the goal is to, given training data, learn the distribution of the data and use that distribution to generate new samples following the same distribution as the training samples.

The framework behind GANs is that there are two net-

works that are trained jointly to minimize an objective function as shown in equation 1. The first network is the generative network G(**z**), which given some noise **z**, will output fake samples (we define fake samples as ones generated from the generator network). The second network is the discriminative network D(**a**) which will to learn to distinguish between real and fake samples. The discriminative network will take as input both real and fake samples and will output a likelihood in (0,1) that the sample was real. In other words, D(**a**) represents the probability that **a** came from the data rather than from the generator.

We can see that a logical way to proceed is that we would like to make the generator fool the discriminator by generating real-looking samples. GANs framework are analogous to minimax two-player game and have the following objective function:

$$\min_{\theta_g} \max_{\theta_d} {}_{x\sim p_{data}(x)}[log(D_{\theta_d}(x))] +_{z\sim p_z(z)}[log(1-D_{\theta_d}(G_{\theta_g}(z)))] \tag{1}$$

where,

θ_g is the weights used for the generator
θ_d is the weights used for the discriminator.
x is a real sample from the desired distribution.
z is the random noise.

For anomaly detection, GANs can be trained by feeding the training observations (without the anomolous indicator label) through the discriminative network as real samples and after iterations of training, can can simply keep the discriminative network. With the discriminative network, we can pass an observation through it and if the network outputs a low score, then we can assume that the observation was anomolous as we would like to believe that the discriminative network learned the distribution of an "normal" observation as is able to distinguish it from an anomolous observation. This approach works best when we can think that anomolous observations have an inherently different different distribution from "normal" observations and thus the discriminative model would assign a low likelihood that the anomolous observations is in fact "normal".

Papers are in 2 columns with the overall line width of 6.75 inches (41 picas). Each column is 3.25 inches wide (19.5 picas). The space between the columns is .25 inches wide (1.5 picas). The left margin is 1 inch (6 picas). Use 10 point type with a vertical spacing of 11 points. Times Roman is the preferred typeface throughout.

Paper title is 16 point, caps/lc, bold, centered between 2 horizontal rules. Top rule is 4 points thick and bottom rule is 1 point thick. Allow 1/4 inch space above and below title to rules.

Reviewing is double-blind, so do not include author names, affiliations, or any other identifying information in the original submission. If you include urls to supplementary material, make sure the urls also do not disclose your identity.

After a paper is accepted, for the camera-ready submission, Authors' names are centered, initial caps. The lead author's name is to be listed first (left-most), and the Co-authors' names (if different address) are set to follow. If only one co-author, center both the author and co-author, side-by-side.

One-half line space between paragraphs, with no indent.

1.1.2 Isolation Forests

A common approach to outlier detection revolves around seperating outliers from the rest of the data. Isolation Forests (iForest) does exactly this by isolating anomalies. Developed by Liu et al., iForest computes an isolation score for each observation by constructing random forests. The trees within the random forest are generated recursively by random splits along a chosen attribute. The isolation score is calculated as the average path length from the root to the node containing the single observation. As anomalies are susceptible to isolation, they are more likely to be closer to the root node whereas "normal" observations tend to be to be deeper within the tree as they tend to be more similar in values. According to Liu et al., iForest provides linear time complexity and works well in high-dimensional problems.

1.1.3 PIDForest

PIDForest is also a random forest based approaches that attempts to distinguish anomalies from the "normal" observations of points by relatively few attribute values. PIDForest does this by calculating a PIDScore for each observation, this score measures the minimum density of data points over all subcubes containing the point. In a sense it is very similar to iForest as both are random forest based approaches which assign an anomaly score, the main difference is that iForest chooses which attribute we split on as well as the breakpoint at random, while PIDForest choses the attribute to split on in a non-random fashion.

2 Related Works

3 FIRST LEVEL HEADINGS

First level headings are all caps, flush left, bold and in point size 12. One line space before the first level heading and 1/2 line space after the first level heading.

3.1 SECOND LEVEL HEADING

Second level headings must be flush left, all caps, bold and in point size 10. One line space before the second level heading and 1/2 line space after the second level heading.

3.1.1 Third Level Heading

Third level headings must be flush left, initial caps, bold, and in point size 10. One line space before the third level heading and 1/2 line space after the third level heading.

Fourth Level Heading

Fourth level headings must be flush left and initial caps. One line space before the fourth level heading and 1/2 line space after the fourth level heading.

3.2 CITATIONS, FIGURES, REFERENCES

3.2.1 Citations in Text

Citations within the text should include the author's last name and year, e.g., (Cheesman, 1985). Reference style should follow the style that you are used to using, as long as the citation style is consistent.

For the original submission, take care not to reveal the authors' identity through the manner in which one's own previous work is cited. For example, writing "In (Bovik, 1970), we studied the problem of AI" would be inappropriate, as it reveals the author's identity. Instead, write "(Bovik, 1970) studied the problem of AI."

3.2.2 Footnotes

Indicate footnotes with a number[1] in the text. Use 8 point type for footnotes. Place the footnotes at the bottom of the page on which they appear. Precede the footnote with a 0.5 point horizontal rule 1 inch (6 picas) long.[2]

3.2.3 Figures

All artwork must be centered, neat, clean, and legible. Figure number and caption always appear below the figure. Leave 2 line spaces between the figure and the caption. The figure caption is initial caps and each figure numbered consecutively.

Make sure that the figure caption does not get separated from the figure. Leave extra white space at the bottom of the page rather than splitting the figure and figure caption.

[1] Sample of the first footnote

[2] Sample of the second footnote

Figure 1: Sample Figure Caption

3.2.4 Tables

All tables must be centered, neat, clean, and legible. Table number and title always appear above the table. See Table 1.

One line space before the table title, one line space after the table title, and one line space after the table. The table title must be initial caps and each table numbered consecutively.

Table 1: Sample Table Title

PART	DESCRIPTION
Dendrite	Input terminal
Axon	Output terminal
Soma	Cell body (contains cell nucleus)

Acknowledgements

Use unnumbered third level headings for the acknowledgements title. All acknowledgements go at the end of the paper.

References

References follow the acknowledgements. Use unnumbered third level heading for the references title. Any choice of citation style is acceptable as long as you are consistent.

J. Alspector, B. Gupta, and R. B. Allen (1989). Performance of a stochastic learning microchip. In D. S. Touretzky (ed.), *Advances in Neural Information Processing Systems 1*, 748-760. San Mateo, Calif.: Morgan Kaufmann.

F. Rosenblatt (1962). *Principles of Neurodynamics.* Washington, D.C.: Spartan Books.

G. Tesauro (1989). Neurogammon wins computer Olympiad. *Neural Computation* **1**(3):321-323.