DOPING: Generative Data Augmentation for Unsupervised Anomaly Detection with GAN

Swee Kiat Lim, Yi Loo, Ngoc-Trung Tran, Ngai-Man Cheung, Gemma Roig, Yuval Elovici
ST Electronics - SUTD Cyber Security Laboratory, Singapore University of Technology and Design
Singapore
{sweekiat_lim, loo_yi, ngoctrung_tran, ngaiman_cheung, gemma_roig, yuval_elovici}@sutd.edu.sg

***Abstract*—Recently, the introduction of the generative adversarial network (GAN) and its variants has enabled the generation of realistic synthetic samples, which has been used for enlarging training sets. Previous work primarily focused on data augmentation for semi-supervised and supervised tasks. In this paper, we instead focus on unsupervised anomaly detection and propose a novel generative data augmentation framework optimized for this task. By using a GAN variant known as the adversarial autoencoder (AAE), we impose a distribution on the latent space of the dataset and systematically sample the latent space to generate artificial samples. To the best of our knowledge, our method is the first data augmentation technique focused on improving performance in unsupervised anomaly detection. We validate our method by demonstrating consistent improvements across several real-world datasets[1].**

***Index Terms*—unsupervised learning, anomaly detection, generative adversarial network, adversarial autoencoders, data augmentation**

I. Introduction

Data augmentation and oversampling are important approaches to handle imbalanced data in machine learning. Previous data augmentation and oversampling approaches have focused on the supervised setting. For example, Synthetic Minority Over-sampling Technique (SMOTE) [1] and Structure Preserving Oversampling (SPO) [2] use labeled minority class samples to generate more samples.

In this work, we focus on unsupervised anomaly detection, where previous data argumentation approaches are ineffective. This is because labels and knowledge of minority classes are unavailable (Figure 1).

A key challenge of anomaly detection is the pervasiveness of false positives [3] (positives are predicted anomalies). High false positive rates can be due to the difficulty in defining a distribution which contains all of the normal data. In particular, *infrequent normal samples*, defined here as normal samples that occur with very small probability, are poorly characterized in many density-based anomaly detectors [3]–[5].

We hypothesize that in unsupervised anomaly detection, targeted oversampling of infrequent normal samples will be most effective, since such samples seem to be primarily responsible for false positive errors (Figure 1). In this paper, we first proceed to validate this hypothesis in our study. Based on this idea, we then introduce a novel data augmentation method for systematically generating such infrequent normal samples without using class labels.

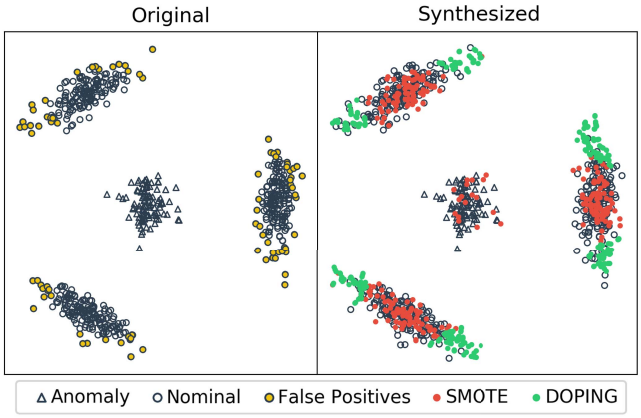

Fig. 1: Our DOPING method is a form of unsupervised data augmentation. Consider the above 2D dataset with 5% contamination, where the anomalies are clustered in the middle. Note that we focus on unsupervised setting, and the class labels are unknown to the methods. Using an interpolation method such as in SMOTE [1], we can randomly synthesize samples across the training set (red), but without labels, this increases the density of all samples indiscriminately, which is not helpful for most unsupervised anomaly detection algorithms. Instead, we propose DOPING, a method that strategically synthesizes samples at the edge of the normal distribution (green). Our DOPING samples help to reduce false positive rates by increasing the density of infrequent normal samples. In the above example, both the false positives and DOPING samples occur in overlapping areas.

Specifically, we propose a data augmentation method targeted at unsupervised anomaly detection, which we coin DOPING. Generating infrequent normal samples can be difficult when the data distributions are complex, e.g., multi-modal (Figure 1). We propose to overcome this challenge by using the adversarial autoencoder (AAE) [6], a variant of the generative adversarial network (GAN) [7]. Using the AAE, we impose a multivariate Gaussian on the latent space of the training data, thereby transforming different data distributions into a unimodal distribution with well-defined tail probability in the

[1]Corresponding author: Ngai-Man Cheung (ngaiman_cheung@sutd.edu.sg). Our code will be published here: https://github.com/greentfrapp/doping. An extended version of the paper can be found at arxiv.org/abs/1808.07632.

978-1-5386-9159-5/18/$31.00 ©2018 IEEE
DOI 10.1109/ICDM.2018.00146

IEEE computer society

latent space. Then, infrequent normal samples can be easily generated by sampling this unimodal latent distribution. In summary, our DOPING technique differs both from previous works where GAN architectures were used in anomaly detection [8]–[10], as well as previous works on data augmentation for supervised classification [11]–[16]:

- It is an unsupervised technique. Our method trains the AAE on the unlabeled dataset and uses general knowledge of the latent distribution to generate desired samples.
- It is a form of unsupervised data augmentation. This makes it complimentary with any anomaly detection algorithm, by augmenting the dataset before training.
- It uses a systematic sampling strategy that has been validated in our analysis to be effective for anomaly detection. In contrast, previous methods mainly focused on generating artificial samples in a minority class.

In the following sections, we discuss details of our DOPING method for unsupervised anomaly detection. We perform studies to validate the effectiveness of oversampling infrequent normal samples. Finally, we compare DOPING to other data augmentation techniques and validate our method on several real-world datasets.

II. Related Work

Autoencoders have previously been used directly for anomaly detection [17]–[21]. Munawar *et al.* [17] trained an autoencoder by minimizing/maximizing reconstruction loss of normal/anomalous data. The trained autoencoder then detects anomalies using reconstruction loss, which is larger for anomalies. A recent work by Zhou *et al.* [19] also introduced robust deep autoencoders (RDA), an unsupervised anomaly detection method that combined the autoencoder and Robust Principal Component Analysis (PCA). The model is trained to decompose the dataset as the sum of two matrices - a normal component that can be reconstructed with little loss via an autoencoder and a sparse matrix consisting of anomalies.

Recent anomaly detection methods have also utilized the GAN [8]–[10]. These methods are typically similar to the approaches above, first modeling the normal data and then using the model to detect anomalies. For instance, Zenati *et al.* [10] utilized the BiGAN [22] for anomaly detection by training on normal data and then defining a score function based on reconstruction loss and discriminator-based loss.

GANs have also been used for data augmentation, generating samples in imbalanced or small datasets, particularly in supervised classification [11]–[14]. More traditional data augmentation techniques for synthesizing data samples include geometric transformation or oversampling [15], [16].

Several oversampling techniques have been proposed: SMOTE [1], Borderline-SMOTE [23] and SPO [2] and Integrated Oversampling (INOS) [24]. However, these techniques focus on oversampling the minority class for use in supervised classification. In contrast, our method focuses on unsupervised anomaly detection, where we do not have access to label information of the minority samples, which is required for the aforementioned methods.

In this paper, we propose a new mechanism to generate synthetic data for improving anomaly detection systems in a purely unsupervised setting. To the best of our knowledge, our method is the first data augmentation technique focused on improving performance and reducing false positive rates in unsupervised anomaly detection.

III. DOPING

We hereby introduce Doping with Infrequent Normal Generator (DOPING), a data augmentation technique for anomaly detection algorithms. In physics, doping is a process whereby a material is introduced into a semiconductor, typically to improve its electrical conductivity. Similarly, our method introduces artificial samples to the original training set, to improve the performance of anomaly detection algorithms.

Adversarial Autoencoders (AAE) [6]. Consider a dataset with data distribution $p_d(\mathbf{x})$ and an encoding function $q(\mathbf{z}|\mathbf{x})$ from the autoencoder. The aggregated posterior distribution $q(\mathbf{z})$ on the latent vector is then defined as:

$$q(\mathbf{z}) = \int_{\mathbf{x}} q(\mathbf{z}|\mathbf{x}) p_d(\mathbf{x}) \mathrm{d}\mathbf{x}$$

The AAE tries to match $q(\mathbf{z})$ to an arbitrary prior $p(\mathbf{z})$, similar to the variational autoencoder (VAE) [25]. While the VAE uses the KL-divergence for regularization, the AAE borrows the adversarial concept from the GAN and attaches a discriminator network that discriminates between outputs from the prior $p(\mathbf{z})$ and the encoder $q(\mathbf{z})$. After training, $q(\mathbf{z})$ then matches $p(\mathbf{z})$.

A. Data augmentation with DOPING: Details

Train Unlabeled AAE In DOPING, we apply the vanilla Unlabeled AAE architecture [6]. The encoder, decoder and discriminator each have two layers of 1000 hidden units with ReLU activation function, except for the activation of the last layer of the decoder, which is linear. *No label is used in the training of the Unlabeled AAE*. We use the ADAM optimizer [26] with a learning rate of 10^{-4} for all networks. We use a 2-D Gaussian with mean 0.0 and standard deviation 10.0 on both dimensions as the prior $p(z)$.

Sample Latent Space After training the AAE, we selectively sample and decode latent vectors to generate synthetic samples. As will be discussed, our analysis finds improved performance when we decode latent vectors at the boundary of the normal latent distribution. Hence, we formalize the DOPING technique as in Algorithm 2 and use an edge-based sampling approach to sample the latent vectors for decoding.

Specifically, we use the AAE's encoder $E(\cdot)$ to encode the entire training set X and generate the corresponding set of latent vectors Z. Z is then filtered by the norm of the latent vectors to form the subset Z_{edge} (see Equation 1), which contains latent vectors near the tail-end of the latent distribution.

$$Z_{\text{edge}} = \{\mathbf{z} \in Z \mid \alpha < \|\mathbf{z}\| < \beta\} \tag{1}$$

For all experiments in this paper, we set β as 3 standard deviations larger than the mean of the latent vector norms for

the training set and α as the 90th percentile norm from the remaining vectors.

Interpolate Sampled Vectors Thereafter, we randomly sample latent vectors from Z_{edge} and new latent vectors are generated by interpolating these selected latent vectors with their nearest neighbor in the latent space (Algorithm 1). These new latent vectors form the set Z_{synth}.

Decode and Synthesize Samples We then decode the set of new latent vectors Z_{synth} to generate synthetic samples X_{synth} for data augmentation. The synthetic samples X_{synth} are added to the original dataset X and the augmented dataset is used to train the anomaly detector algorithm.

Algorithm 1 InterNN (interpolate with nearest neighbor)

Require: $\mathbf{z}_{\text{sample}}$: latent vector to interpolate
Sample $\alpha \sim U(0, 1)$
$\mathbf{z}_{\text{NN}} \leftarrow$ nearest neighbor of $\mathbf{z}_{\text{sample}}$ in latent space
return $\alpha(\mathbf{z}_{\text{NN}} - \mathbf{z}_{\text{sample}}) + \mathbf{z}_{\text{sample}}$

Algorithm 2 DOPING

Require: K: no. of samples to synthesize, X: set of training samples, $\{\alpha, \beta\}$: hyperparameters for edge-based sampling (see Equation 1)
Train AAE on X, with multivariate Gaussian as prior
$E(\cdot) \leftarrow$ encoder in AAE
$D(\cdot) \leftarrow$ decoder in AAE
$Z \leftarrow E(X)$
$Z_{\text{edge}} \leftarrow \{\mathbf{z} \in Z \mid \alpha < \|\mathbf{z}\| < \beta\}$ (see Section III-A)
Initialize list of new latent vectors $Z_{\text{synth}} \leftarrow []$
for $i = 1, 2, ...K$ **do**
  Sample infrequent latent vector $\mathbf{z}_{\text{edge}} \in Z_{\text{edge}}$
  $\mathbf{z}_{\text{new}} \leftarrow$ **InterNN**$(\mathbf{z}_{\text{edge}})$
  Z_{synth} **append** $\mathbf{z}_{\text{new}}$
end for
$X_{\text{synth}} \leftarrow D(Z_{\text{synth}})$
Train anomaly detector on augmented dataset $(X + X_{\text{synth}})$

B. Use of Adversarial Autoencoders

We specifically choose to adopt AAE over other GAN variants in our framework for the following reasons:

Explicit Encoding to Latent Space While other GAN variants are able to synthesize new samples from a latent space after training, it is non-trivial to encode an arbitrary sample back into the latent space, e.g., requiring iterative estimation [8]. In contrast, the AAE incorporates an autoencoder architecture, which allows explicit decoding from and encoding to the latent space. The encoding network is crucial for our analysis of samples in the latent space, while the decoding network enables generation of synthetic samples from specific regions in the latent space based on our analysis. The encoding and decoding are tightly coupled in AAE, enabling our analysis outcomes to directly guide sample synthesis.

Explicit Control of Latent Space Using the AAE, we are able to impose a variety of prior distributions on the latent space, similar to the VAE [25]. However, the VAE requires access to the exact functional form of the prior distribution. In contrast, the adversarial method used by AAE only requires samples from the desired prior, which allows us to impose more complex distributions. By imposing the same prior distribution and systematically decoding from this consistent latent space, we have a general method that can be applied to any dataset.

In our method, we use the AAE to impose a multivariate Gaussian on the latent space of the normal data and treat that as a proxy for the data distribution. Our proposed use of multivariate Gaussian as the prior transforms different complex data distributions (e.g. multimodal, skewed) into *unimodal ones with well-defined tail probability in the latent space*. This enables decoding from the edge of the latent Gaussian distribution to generate synthetic samples that are infrequent normals. As a data augmentation technique, DOPING is used with another anomaly detection algorithm, such as Isolation Forest (iForest) [5].

IV. Analysis of DOPING with Synthetic Datasets

In this section, we analyze and validate fundamental components of DOPING with synthetic datasets. We first demonstrate that the use of the AAE in our framework enables mapping of diverse data distributions to a consistent latent distribution, which can be systematically sampled.

A. Dataset and Experimental Design

Dataset We introduce three synthetic datasets that each comprise samples from a normal distribution and an anomalous distribution (Figure 2a). Each dataset contains 1000 training samples and 1000 test samples, with 95% of the samples belonging to the normal distribution and 5% of the samples belonging to the anomalous distribution.

- **Dataset A** The normal distribution is a 2-D Gaussian with mean at origin and standard deviation 10.0 on each dimension, while the anomalous distribution is a 2-D Gaussian with mean [30.0, 0.0] and standard deviation 5.0 on each dimension.
- **Dataset B** A 3-D version of Dataset A.
- **Dataset C** The normal distribution follows a ring with radius of mean 30.0 and standard deviation 5.0, while the anomalous distribution is a 2-D Gaussian with mean at origin and standard deviation 5.0 on each dimension. This is meant to be a challenging dataset, with the anomalies surrounded by normal samples, making it more difficult for anomaly detection algorithms to recognize anomalies.

Baseline Method In this experiment, we use the popular iForest anomaly detection algorithm. We use the implementation in version 0.19.1 of the scikit-learn package [27] and vary the contamination hyperparameter from 0.01 to 0.69 in increments of 0.04 to generate the receiver operating characteristic (ROC) curve for evaluation.

Experiment Details In this section, we first adopt a *Labeled AAE* architecture to analyze our method. It is similar to the Unlabeled AAE, with the exception of label information

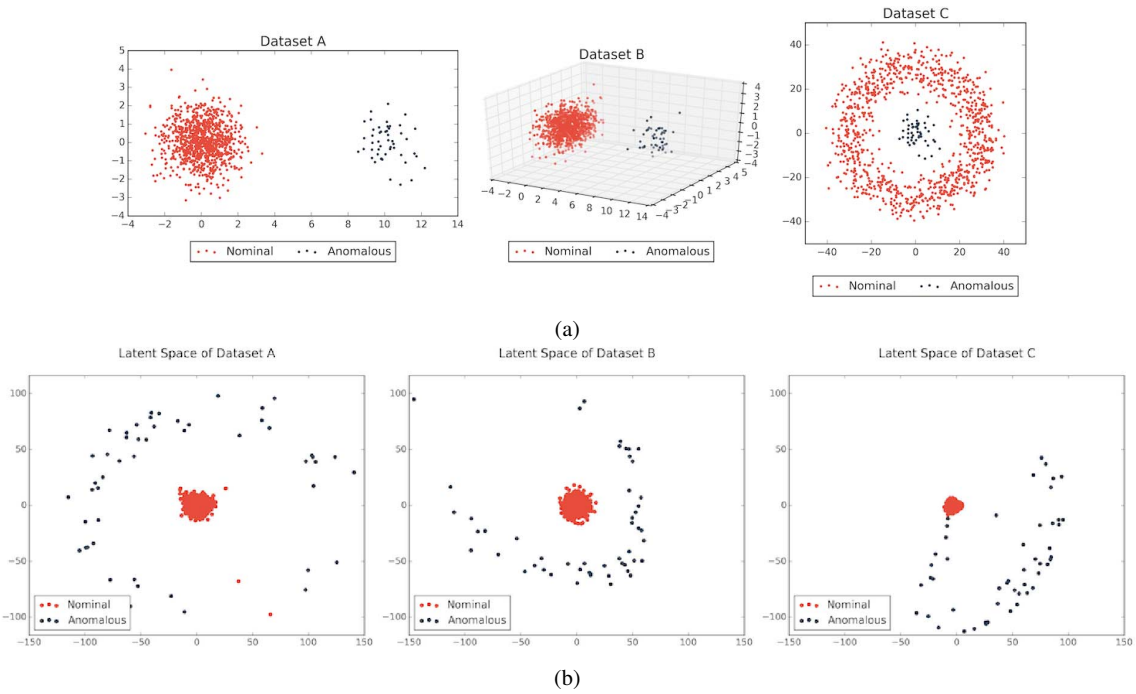

Fig. 2: (a) The three synthetic datasets used and (b) the corresponding latent spaces after training with the AAE.

provided to the discriminator network in the form of a one-hot vector. In our case, the label indicates if the sample is normal or anomalous. If the sample is anomalous, we use a ring with radius 100.0 as the prior $p(z)$, by randomly sampling a latent vector with l^2-norm of 100.0. If the sample is normal, we follow the prior used in the Unlabeled AAE. All other conditions are the same as described in the Unlabeled AAE. This Labeled AAE is used *only in this analysis* to understand the effectiveness of sampling at different latent regions. In practice, Unlabeled AAE is used, as will be further discussed.

In this section, we also use a magnitude-based sampling method for synthesizing new samples, in order to demonstrate the effects of sampling from different regions in the latent distribution. In magnitude-based sampling, we sample the latent vectors from an n-dimensional spherical surface of increasing magnitudes. In this case, with a 2-D latent space, we essentially sample from rings of increasing magnitudes.

We sample latent vectors at different l^2-norms from 5.0 to 100.0 at increments of 5.0. At each l^2-norm, we decode and add the 100 random samples to the original dataset. Each augmented dataset of 1100 samples is then used to train the anomaly detectors and tested on the test set, to generate the ROC curves. We then measure the area-under-curve (AUC) for each ROC curve and plot the AUC against the l^2-norm, as shown in Figure 3. We treat predictions of anomalies as positive instances for the calculations of the AUC.

B. Optimal Latent Space Sampling for Synthesizing Samples

In this section, we test our hypothesis that decoding from the boundary of the normal latent distribution to generate additional infrequent normal samples improves anomaly detection performance.

All three graphs in Figure 3 clearly show the same trends, with significantly better AUC when DOPING with latent vectors of l^2-norms in the range of 15.0 to 20.0. With reference to Figure 2b, this region corresponds to the boundary of the normal latent distribution, which agrees with our hypothesis.

The optimal samples for data augmentation is not necessarily intuitive when viewed in the data space. For instance, in Dataset C, it is not immediately clear if the ideal synthetic samples should lie in the inner or outer boundary of the normal distribution. Despite that, Figure 3 shows that an improvement can be consistently achieved with our method, when decoding from the boundary of the latent distribution.

Thereafter, as l^2-norm increases, the AUC falls to below that of the original dataset, implying that addition of samples in this region actually worsens the performance of the anomaly detectors. In the context of our hypothesis, samples decoded from latent vectors in this region corresponds to anomalous samples. By increasing the density of anomalies in the training set, the anomaly detector algorithms are more likely to misclassify anomalies as normal, which worsens performance.

Unsupervised Data Augmentation Since samples at the boundary of the normal latent distribution provide improvement in anomaly detection performance, in practice, the Labeled AAE can be replaced with the unsupervised Unlabeled AAE. We reiterate that the nature of anomaly detection tasks is that "anomalies are 'few and different', which make them more susceptible to isolation than normal points" [5]. With low contamination, an Unlabeled AAE trained on the entire dataset

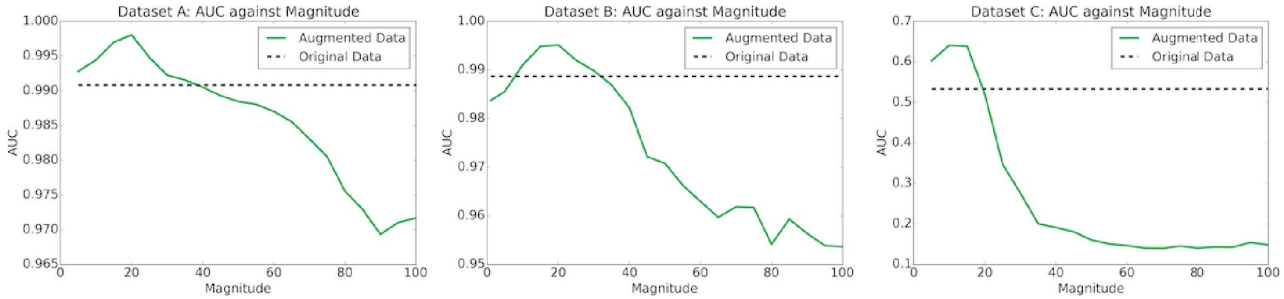

Fig. 3: Graphs of AUC against l^2-norm/magnitude for the different synthetic datasets, showing a consistent trend.

can approximate the normal latent distribution in a Labeled AAE and the contamination due to the 'few and different' anomalies can be neglected. We demonstrate this in the next section using the Unlabeled AAE.

V. Evaluation with Real-world Datasets

Here, we evaluate the performance of the formal DOPING algorithm, as described in Section III-A and Algorithm 2, on several real-world datasets.

A. Dataset and Experimental Setup

Datasets We show the results of DOPING on four anomaly detection datasets from the Outlier Detection DataSets (ODDS) repository[2].

- **Mammography** The Mammography dataset [28] is obtained from the ODDS repository, with each sample having 6 features. The dataset comprises of 11183 samples, including 260 anomalies (2.3%contamination).
- **Thyroid** The Thyroid dataset [28] is obtained from the ODDS repository, with each sample having 6 features. The dataset comprises of 3772 samples, including 93 anomalies (2.5% contamination).
- **Lymphography** The Lymphography dataset [28] is obtained from the ODDS repository, with each sample having 18 features. The dataset comprises of 148 samples, including 6 anomalies (4.0% contamination).
- **Cardiotocography** The Cardiotocography dataset [28] is obtained from the ODDS repository, with each sample having 21 features. The dataset comprises of 1831 samples, including 176 anomalies (9.6% contamination).

For all datasets, we follow the settings in [29], [30] with completely clean training data: in each run, we randomly sample 50% of the data for training with the remaining 50% reserved for testing, and only data samples from the normal class are used for training models.

Baseline Method iForest [5] is used as the anomaly detector. We use the implementation in version 0.19.1 of the scikit-learn package [27] and vary the contamination hyperparameter from 0.01 to 0.69 in increments of 0.02 to calculate the AUC and search for the best F1.

For each dataset, we train iForest with four variants: the original training set and three training sets augmented with DOPING, a SMOTE variant [1] and an INOS variant [24] respectively. For each augmentation method, we synthesize an additional 10% of the original training set size.

[2]http://odds.cs.stonybrook.edu/

TABLE I: Comparing best F1 on real-world datasets.

Best F1	iForest [5]	iForest w. DOPING	iForest w. SMOTE	iForest w. INOS
Mammo.	0.304	0.345	0.319	**0.351**
Cardio.	0.795	**0.812**	0.771	0.773
Thyroid	0.744	**0.771**	0.758	0.730
Lympho.	0.614	**0.720**	0.657	0.655
Average	0.614	**0.662**	0.626	0.627

TABLE II: Comparing G-measure on real-world datasets.

G-measure	iForest [5]	iForest w. DOPING	iForest w. SMOTE	iForest w. INOS
Mammo.	0.318	0.345	0.323	**0.361**
Cardio.	0.787	**0.807**	0.779	0.780
Thyroid	0.754	**0.775**	0.753	0.737
Lympho.	0.665	**0.750**	0.700	0.707
Average	0.631	**0.669**	0.639	0.646

Experiment Details When implementing DOPING on these datasets, we first train an Unlabeled AAE on the training sets with a 2-D regular Gaussian prior with standard deviation 10.0 on both dimensions. We then synthesize samples using edge-based sampling as described in Section III-A.

For comparison, we also show the results of iForest with two other data augmentation methods - SMOTE [1] and INOS [24]. For both of the variants, we randomly select samples from the training set and apply interpolation in the data space. In the original SMOTE and INOS algorithms, only the minority class is oversampled in this manner. Here we randomly sample from the entire training set since we do not have labels in unsupervised anomaly detection.

For the SMOTE variant, we implement interpolation with nearest neighbor (Algorithm 1) on randomly selected samples across the whole training set. For INOS, we implement the technique using the OSTSC package [31] and set the r variable as 0.8 where 80% of the synthetic samples are generated using ESPO and 20% using ADASYN.

B. Results and Discussion

Tables I and II show the performance of Isolation Forest [5] with and without augmentation, via F1 score and G-measure,

where F1 is the harmonic mean of precision and recall and G-measure is the geometric mean.

We first observe that training with the augmented datasets mostly give better results than baseline iForest, implying that data augmentation has a positive effect on anomaly detection. This can be attributed to the increase in density and variety of normal data samples in the training set.

Furthermore, we see that iForest with DOPING demonstrates the best performance across all datasets, except Mammography. Rather than randomly synthesizing training samples as with other augmentation methods, DOPING purposefully synthesizes infrequent normal samples and the better performance may be attributed to the anomaly detector better recognizing these infrequent normal samples in the dataset.

VI. Conclusion

This paper proposes a novel form of data augmentation designed to tackle the problem where infrequent normal instances are misclassified, thereby reducing false positives. This is done by using an AAE to impose a multivariate Gaussian on the latent space and subsequently decoding from the edge of this latent distribution to generate infrequent normal samples.

In this paper we explained the intuition behind why DOPING helps to improve the performance of anomaly detectors and analyzed the performance of DOPING on several datasets. We show that the AUC against l^2-norm/magnitude trend is consistent across different datasets and DOPING makes use of this consistent trend to synthesize specific samples for dataset augmentation. Our experiments demonstrate empirically that our data augmentation technique helps to improve anomaly detection performance when applied across a variety of datasets.

Acknowledgment

This work was supported by both ST Electronics and the National Research Foundation (NRF), Prime Minister's Office, Singapore under Corporate Laboratory @ University Scheme (Programme Title: STEE Infosec - SUTD Corporate Laboratory). The authors would also like to thank the anonymous reviewers for their valuable comments and helpful suggestions.

References

[1] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[2] H. Cao, X.-L. Li, Y.-K. Woon, and S.-K. Ng, "Spo: Structure preserving oversampling for imbalanced time series classification," in *Data Mining (ICDM), 2011 IEEE 11th International Conference on*. IEEE, 2011, pp. 1008–1013.

[3] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[4] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: Identifying density-based local outliers," in *ACM sigmod record*, vol. 29, no. 2. ACM, 2000, pp. 93–104.

[5] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. IEEE, 2008, pp. 413–422.

[6] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," *arXiv preprint arXiv:1511.05644*, 2015.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[8] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," *CoRR*, vol. abs/1703.05921, 2017. [Online]. Available: http://arxiv.org/abs/1703.05921

[9] M. Ravanbakhsh, E. Sangineto, M. Nabi, and N. Sebe, "Training adversarial discriminators for cross-channel abnormal event detection in crowds," *CoRR*, vol. abs/1706.07680, 2017.

[10] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar, "Efficient gan-based anomaly detection," *arXiv preprint arXiv:1802.06222*, 2018.

[11] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *arXiv preprint arXiv:1712.04621*, 2017.

[12] A. Antoniou, A. Storkey, and H. Edwards, "Data augmentation generative adversarial networks," *arXiv preprint arXiv:1711.04340*, 2017.

[13] X. Zhu, Y. Liu, Z. Qin, and J. Li, "Data augmentation in emotion classification using generative adversarial networks," *CoRR*, vol. abs/1711.00648, 2017. [Online]. Available: http://arxiv.org/abs/1711.00648

[14] L. Sixt, B. Wild, and T. Landgraf, "Rendergan: Generating realistic labeled data," *arXiv preprint arXiv:1611.01331*, 2016.

[15] S. C. Wong, A. Gatt, V. Stamatescu, and M. D. McDonnell, "Understanding data augmentation for classification: when to warp?" in *Digital Image Computing: Techniques and Applications (DICTA), 2016 International Conference on*. IEEE, 2016, pp. 1–6.

[16] P. Y. Simard, D. Steinkraus, J. C. Platt *et al.*, "Best practices for convolutional neural networks applied to visual document analysis." in *ICDAR*, vol. 3, 2003, pp. 958–962.

[17] A. Munawar, P. Vinayavekhin, and G. D. Magistris, "Limiting the reconstruction capability of generative neural network using negative learning," *CoRR*, vol. abs/1708.08985, 2017.

[18] M. Hasan, J. Choi, j. Neumann, A. K. Roy-Chowdhury, and L. Davis, "Learning temporal regularity in video sequences," in *CVPR*, 2016.

[19] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 665–674.

[20] P. Seeböck, S. M. Waldstein, S. Klimscha, B. S. Gerendas, R. Donner, T. Schlegl, U. Schmidt-Erfurth, and G. Langs, "Identifying and categorizing anomalies in retinal imaging data," *CoRR*, vol. abs/1612.00686, 2016.

[21] D. Xu, E. Ricci, Y. Yan, J. Song, and N. Sebe, "Learning deep representations of appearance and motion for anomalous event detection," *arXiv preprint arXiv:1510.01553*, 2015.

[22] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," *arXiv preprint arXiv:1605.09782*, 2016.

[23] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-smote: a new over-sampling method in imbalanced data sets learning," in *International Conference on Intelligent Computing*. Springer, 2005, pp. 878–887.

[24] H. Cao, X.-L. Li, D. Y.-K. Woon, and S.-K. Ng, "Integrated oversampling for imbalanced time series classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 12, pp. 2809–2822, 2013.

[25] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[26] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[28] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[29] S. Zhai, Y. Cheng, W. Lu, and Z. Zhang, "Deep structured energy based models for anomaly detection," in *International Conference on Machine Learning*, 2016, pp. 1100–1109.

[30] B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. Cho, and H. Chen, "Deep autoencoding gaussian mixture model for unsupervised anomaly detection," in *International Conference on Learning Representations*, 2018.

[31] M. Dixon, D. Klabjan, and L. Wei, "Ostsc: Over sampling for time series classification in r," 2017.