Coupled IGMM-GANs for improved generative adversarial anomaly detection

Kathryn Gray[1], Daniel Smolyak[2], Sarkhan Badirli[3], and George Mohler[4]
[1]Dept. of Applied Math. and Computer Science, University of Colorado Boulder
[2]Dept. of Computer Science, University of Maryland
[3,4]Dept. of Computer Science, Indiana University—Purdue University, Indianapolis
[1]kathryn.gray@colorado.edu, [2]dsmolyak@umd.edu,[3,4]{sbadirli, gmohler}@iu.edu

***Abstract*—Detecting anomalies and outliers in data has a number of applications including hazard sensing, fraud detection, and systems management. While generative adversarial networks seem like a natural fit for addressing these challenges, we find that existing GAN based anomaly detection algorithms perform poorly due to their inability to handle multimodal patterns. For this purpose we introduce an infinite Gaussian mixture model coupled with (bi-directional) generative adversarial networks, IGMM-GAN, that facilitates multimodal anomaly detection. We illustrate our methodology and its improvement over existing GAN anomaly detection on the MNIST dataset.**

I. Introduction

In many anomaly detection studies, significant pre-processing and feature engineering is used prior to classification or similarity comparisons. Furthermore, anomaly definition is often vague and subjective. With a lack of ground truth datasets, it is difficult to compare benchmark models available for detecting anomalies. Several recent studies [Schlegl et al.2017], [Zenati et al.2018] have successfully applied GANs for the purpose of anomaly detection to overcome these challenges while also providing a generative method for augmenting anomaly detection data sets. By making use of Bidirectional GAN (BiGAN) [Donahue, Krähenbühl, and Darrell2016], these methods have fared favorably in anomaly detection compared to other deep embedding methods such as variational auto-encoders.

However, existing GAN based anomaly detection methods, in particular GANomaly [Schlegl et al.2017] and Efficient GAN Anomaly Detection [Zenati et al.2018], have difficulties when the data is multimodal. These methods, which assume that the latent noise and encoded data in the BiGAN are unimodal Gaussians, are unable to accurately detect anomalies when multiple classes with multiple modes or clusters are present in the data. In this paper we propose using a GAN coupled with an Infinite Gaussian Mixture Model [Rasmussen1999] that can simultaneously generate realistic data as well as detect anomalies in multimodal data. In Figure 1, we provide an overview of our coupled IGMM-GAN model. We use a Bidirectional GAN (BiGAN) that learns an encoder in addition to a generator neural network for transforming data into a latent space where outliers may be detected. Unlike previous unimodal GAN based anomaly detection [Schlegl et al.2017], [Zenati et al.2018], we use an Infinite Gaussian Mixture Model to detect anomalies in the latent space through a multi-modal Mahalanobis metric. We find this approach significantly improves the accuracy of previous GAN based anomaly detection algorithms on the MNIST dataset.

The outline of the paper is as follows. In Section 2, we provide details on the IGMM-GAN model. In Section 3, we present experimental results applying our model to MNIST. We compare AUC scores of the IGMM-GAN against several recently proposed GAN based anomaly detection algorithms.

II. Methodology

A. GANs

GANs, first proposed in [Goodfellow et al.2014], consist of a generator (G) network and a discriminator (D) network: the two follow the below minimax game, where the generator tries to minimize the $\log(1 - D(G(z)))$ term and the discriminator tries to maximize the $\log(D(x))$ term.

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$$

The discriminator network improves the loss when it classifies a sample x correctly and $D(x)$ is the probability that x is real rather than generated data. Meanwhile, the generator network maps Gaussian samples z into synthetic data samples $G(z)$ (e.g. image). The generator attempts to minimize the discriminator loss by generating a fake sample $G(z)$ such that the discriminator labels the sample as real (hence the $1 - D(G(z))$ term).

B. BiGANs

Bidirectional GANs, first proposed by [Donahue, Krähenbühl, and Darrell2016], include an encoder (E) that learns the inverse of the generator. While the generator will learn a mapping from the latent dimension to data, the encoder will learn a mapping from data to the latent dimension. The discriminator then must classify pairs of the form $(G(z), z)$ or $(x, E(x))$ as real or synthetic, where z is noise from a standard distribution and x is real data.

$$\min_{G,E} \max_D V(D,G,E) = \mathbb{E}_{x\sim p_{data}(x)}\left[\mathbb{E}_{z\sim p_E(\cdot|x)}[\log D(x,z)]\right] + \mathbb{E}_{z\sim p_z(z)}\left[\mathbb{E}_{x\sim p_G(\cdot|z)}[\log(1 - D(x,z))]\right]$$

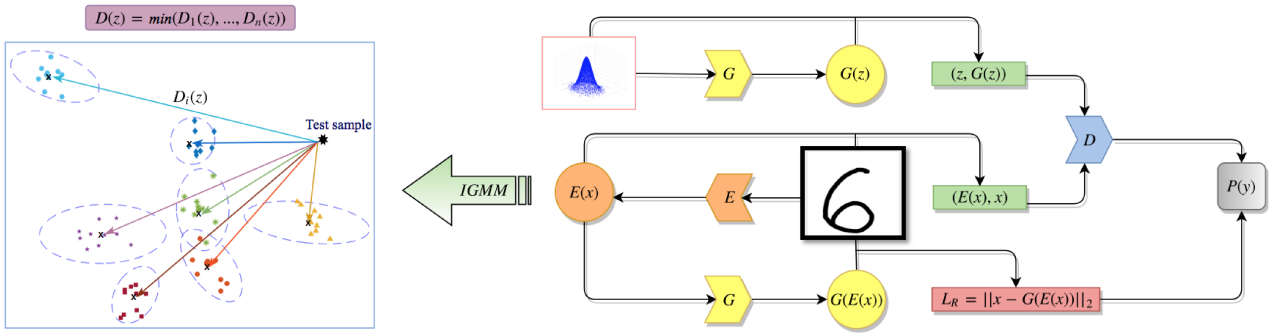

Fig. 1: The IGMM-GAN architecture. The generator network learns to transform Gaussian samples into synthetic images, while the discriminator network learns to distinguish real from fake images. Simultaneously, and encoder network learns the inverse mapping of the generator for image embedding in the latent space. Finally, a multimodal Mahalanobis distance metric from the IGMM is used to detect outliers in unseen test data.

C. Anomaly Detection with BiGANs

As first proposed in the work by [Schlegl et al.2017], and further developed by [Zenati et al.2018], variants of GANs that also learn an inverse of the generator can be used to detect anomalous data. Specifically, after training a generator, discriminator, and encoder, an anomaly score can be calculated for each data sample, where a higher score indicates greater likelihood of belonging to the anomalous class. In the current state-of-the-art GAN based anomaly detection [Zenati et al.2018], a combination of a reconstruction loss L_G and discriminator-based loss L_D is used to determine the anomaly score,

$$A(x) = \alpha L_G(x) + (1 - \alpha) L_D(x), \tag{1}$$

where the reconstruction loss is given by $L_G(x) = \|x - G(E(x))\|$ and the discriminator loss is given by the cross-entropy, $L_D(x) = \sigma(D(x, E(x)), 1)$. We refer to this algorithm as EGBAD (Efficient GAN based anomaly detection) and in [Zenati et al.2018] the method is shown to outperform a variety of deep unsupervised models including anogan, variational auto-encoders, and deep auto-encoder GMM.

One drawback of GAN based anomaly detection such as EGBAD and GANomaly is that they are not detecting anomalies in multimodal datasets (as we will show experimentally in the next section). Our approach is therefore not to view the latent variable z as a single model Gaussian, but as a mixture of several Gaussians with means and covariances (μ_i, Σ_i). Outliers can then be detected using a multimodal Mahalanobis distance for the anomaly score. In particular, for a new data point in latent space, $z = E(x)$, the Mahalanobis distance to cluster i is given by,

$$D_M^i(z) = \sqrt{(z - \mu_i)^T \Sigma_i^{-1} (z - \mu_i)} \tag{2}$$

The anomaly score is then given by the minimum distance, $D(x) = \min_i D_M^i$. We note that the Mahalanobis based anomaly score produces not only improved anomaly detection results, as will be seen in the next section, but also up to 4x faster inference time over the cross-entropy loss in Equation 1.

D. Infinite Gaussian Mixture Model

Because our goal is end-to-end learning for anomaly detection, we use an infinite Gaussian mixture model (IGMM) [Rasmussen1999] to automatically learn the number of clusters as well as the cluster means and covariances (μ_i, Σ_i) for the anomaly score defined by Equation 2.

IGMM [Rasmussen1999] is a Dirichlet Process Mixture Model in which the number of components can grow arbitrarily as data allows, hence the name Infinite Gaussian Mixture Model. IGMM assumes each cluster is modeled by a single Gaussian component and the base Dirichlet distribution serves as a prior for the parameters of these components (cluster mean μ and cluster covariance Σ). As the name Gaussian mixture suggests, the bi-variate prior, H, involves a Gaussian prior over mean vectors and Inverse-Wishart over covariance matrices. More precisely H can be written as follows,

$$H = N(\mu|\mu_0, \Sigma_0 \kappa_0^{-1}) W^{-1}(\Sigma|\Sigma_0, m) \tag{3}$$

where μ_0 is the mean of Gaussian prior, κ_0 is scaling constant that adjusts the dispersion of cluster center and parameter m dictates the expected shapes of clusters. Note that Normal and Inverse-Wishart distributions are conjugate, thus the posterior predictive distribution can be analytically derived, in the form of a multivariate Student-t distribution, by integrating out the component parameters $\{\mu_i, \Sigma_i\}$. For inference we utilize Collapsed Gibbs Sampling [Rasmussen1999] due to the conjugacy between the model (Gaussian) and the prior (NIW).

The generative model is illustrated in (4)

$$\begin{aligned} z_i &\sim N(z_i|\mu_i, \Sigma_i) \\ \{\mu_i, \Sigma_i\} &\sim G \\ G &\sim DP(\alpha H) \end{aligned} \tag{4}$$

where H is defined by Equation (3), z_i is the data point from cluster i and α is the concentration parameter of the Dirichlet process.

III. Experimental results

A. Datasets

Previous GAN based anomaly detection studies have used MNIST (a dataset of handwritten numbers) [LeCun, Cortes, and Burges2010] for bench-marking competing methods. Anomalies are defined by leaving out a digit from training and assessing the AUC (or other classification metric) of the anomaly score on a test data set which includes the held-out digit.

B. Architecture

As mentioned previously, our model architecture is based on that of the BiGAN in [Donahue, Krähenbühl, and Darrell2016] and [Zenati et al.2018]. The architecture for the model is given in Table I. The encoder consists of an input layer taking in an $N \times N$ image. Whereas the encoder consists of several convolution and dense layers, the generator makes use of convolution transpose layers to facilitate learning of the inverse of the encoder. The 2D convolution layers in the model are each followed by batch normalization and "Leaky ReLu" activation. The discriminator is slightly more complex, beginning as two separate models, one composed similarly to the encoder which takes the real and generated data as input, and one containing dense layers which takes the latent representation as input. These two networks are then concatenated, ending in two final dense layers and a sigmoid activation.

Furthermore, combining the ideas from [Akcay, Atapour-Abarghouei, and Breckon2018] and [Donahue, Krähenbühl, and Darrell2016], we add onto the existing architecture a reconstruction loss term, taking into account the ability of the encoder and generator to reproduce a real image. This loss term helps ensure that not only can the generator's images fool the discriminator, but also that the encoder and generator function as closely as possible to inverses of one another. This loss is defined as:

$$L_R = ||x - G(E(x))||_2$$

We use an Adam optimizer [Kingma and Ba2014] with a learning rate of $lr = 1e^{-5}$ and $\beta = 0.5$. These parameters are sufficient for the generator and discriminator loss for our model to converge, similarly to the other models. In Figure 3 we display sample generated digits after 20000 epochs to verify the model is learning a good represenation of the data.

Layer	Units	BN	Activation	Kernel
$E(x)$				
Dense	768		ReLU	
Convolution	32	✓	ReLU	3×2
Convolution	64	✓	ReLU	3×2
Convolution	128	✓	ReLU	3×2
Dense	100			
$G(z)$				
Conv. Transpose	128	✓	ReLU	3×2
Conv. Transpose	64	✓	ReLU	3×2
Conv. Transpose	32			3×2
Dense	1		Linear	
$D(x)$				
Convolution	64		Leaky ReLU	3×2
Convolution	64	✓	Leaky ReLU	3×1
$D(z)$				
Dense	512		Leaky ReLU	
Concatenate				
$D(x, z)$				
Dense	1		Leaky ReLU	

TABLE I: The architecture for our model, layer by layer. Units refer to number of filters in the case of convolution layers, and BN is Batch Normalization abbreviated.

C. Hyper-parameter tuning for IGMM

The hyper-parameters of IGMM are coarsely tuned to maximize the macro-f1 score. As the data is not well balanced, macro-f1 was chosen to suppress the dominance of large classes. IGMM has 4 hyper-parameters, $\{\kappa_0, m, \mu_0, \Sigma_0\}$ to be tuned. To simplify the tuning process, the prior mean, μ_0, is set to the mean of data and we set Σ_0 to an identity matrix scaled by a parameter s. This left us with 3 parameters, $\{\kappa_0, m, s\}$ to tune. Parameter ranges and best triples are illustrated in Table II. The number of sweeps in the inference is fixed at 500, with 300 used for the burn-in period. Label samples are collected in every 50 iteration after burn-in and aligned by the Hungarian method to render final cluster labels.

HP	**Range**
κ_0	$0.01; 0.1; 1; 10; 100$
m	$d+10; d+15; d+20; 5d; 10d; 100d$
s	$1; 3; 5; 7; 9$

(a) Parameter ranges used in tuning

HP	**MNIST**
κ_0	0.1
m	$d+20$
s	7

(b) Best triples from tuning

TABLE II: Ranges for tuning and best triples used in experiments. HP stands for hyper-parameters

We restricted created clusters to ones with more than 50 points as IGMM may generate artificial small clusters to fit in distribution.

D. Determining Anomaly Scores

Anomaly scores were determined by using IGMM on the encoded training data to determine the cluster means and covariance matrices. From there, an anomaly score was determined by the Mahalanobis distance to the nearest cluster. Figure 4 shows an example TSNE visualization

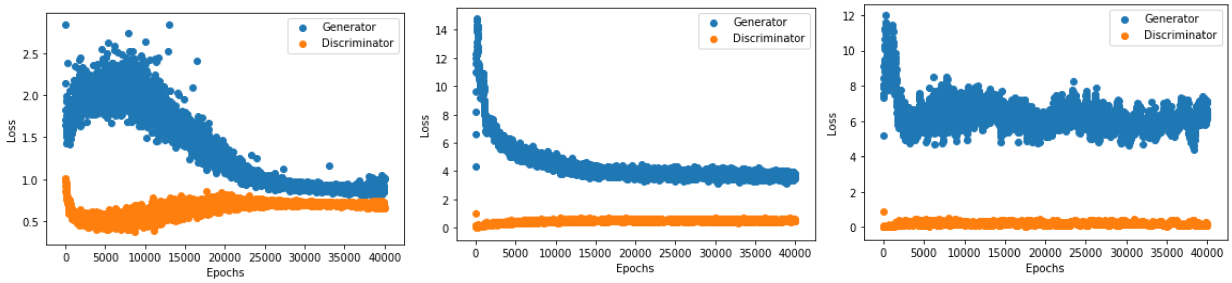

Fig. 2: Generator and discriminator loss by epoch for MNIST, digit 9 (Order: Ganomaly, EGBAD, Our Method)

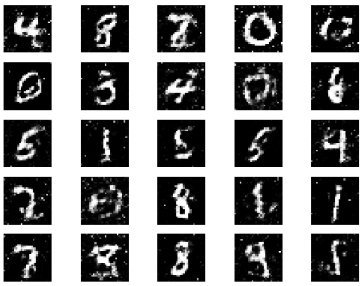

Fig. 3: Sample of generated images after 20000 epochs.

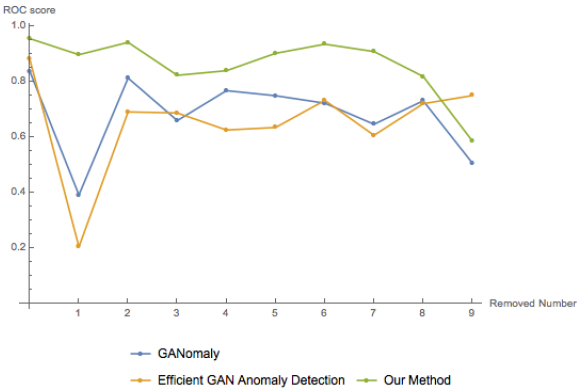

Fig. 5: ROC AUC scores with MNIST data

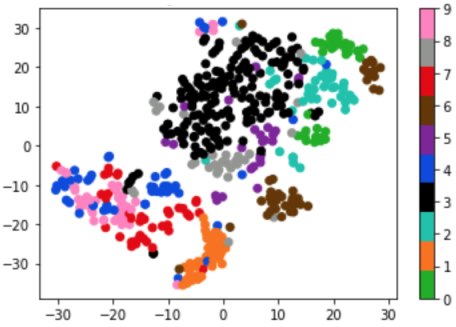

Fig. 4: TSNE visualization of the latent dimension with MNIST data (held out class in black).

[Maaten and Hinton2008] of the MNIST colored by the class labels. In this case the held out class is in black, illustrating the need for multi-modal anomaly detection.

E. Improving MNIST Benchmarks with IGMM

Following the approach of [Zenati et al.2018], we start by designating one digit as an anomalous class and remove it from the training dataset. For the remaining data we perform an 80/20 split into training and test sets and train the models for 40,000 epochs (where each epoch involves training on a random batch of 128 images). We repeat this process for each digit and for each anomaly detection method, scoring each method on its ROC AUC score. In Figure 5 we display the AUC scores of IGMM-GAN against GANomaly and EGBAD for each digit held out of testing. The IGMM-GAN significantly improves the AUC scores for the majority of digits held out, especially for digits 1 and 7.

IV. Conclusion

In this paper we improved GAN based anomaly detection through the introduction of the multimodal IGMM-GAN. We believe that the IGMM-GAN will serve as a complimentary tool to existing algorithms for anomaly detection that require significant feature engineering. Our method may also find application to anomaly detection in other domains beside computer vision where the data is multimodal.

V. Acknowledgements

This work was supported in part by NSF grants ATD-1737996, REU-1343123, and SCC-1737585.

References

[Akcay, Atapour-Abarghouei, and Breckon2018] Akcay, S.; Atapour-Abarghouei, A.; and Breckon, T. P. 2018. Ganomaly: Semi-supervised anomaly detection via adversarial training. *arXiv preprint arXiv:1805.06725*.

[Donahue, Krähenbühl, and Darrell2016] Donahue, J.; Krähenbühl, P.; and Darrell, T. 2016. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*.

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

[Kingma and Ba2014] Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[LeCun, Cortes, and Burges2010] LeCun, Y.; Cortes, C.; and Burges, C. J. 2010. Mnist handwritten digit database. 2010. *URL http://yann. lecun. com/exdb/mnist*.

[Maaten and Hinton2008] Maaten, L. v. d., and Hinton, G. 2008. Visualizing data using t-sne. *Journal of machine learning research* 9(Nov):2579–2605.

[Rasmussen1999] Rasmussen, C. E. 1999. The infinite gaussian mixture model. In *Advances in neural information processing systems*, 554–560.

[Schlegl et al.2017] Schlegl, T.; Seeböck, P.; Waldstein, S. M.; Schmidt-Erfurth, U.; and Langs, G. 2017. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging*, 146–157. Springer.

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