# Anomaly Detection for imbalanced datasets with Deep Generative Models

Nazly Rocio Santos Buitrago[1], Loek Tonnaer[1], Vlado Menkovski[1], and Dimitrios Mavroeidis[2]

[1] Eindhoven University of Technology, Eindhoven, The Netherlands
[2] Royal Philips B. V., Eindhoven, The Netherlands

**Abstract.** Many important data analysis applications present with severely imbalanced datasets with respect to the target variable. A typical example is medical image analysis, where positive samples are scarce, while performance is commonly estimated against the correct detection of these positive examples. We approach this challenge by formulating the problem as anomaly detection with generative models. We train a generative model without supervision on the 'negative' (common) datapoints and use this model to estimate the likelihood of unseen data. A successful model allows us to detect the 'positive' case as low likelihood datapoints.
In this position paper, we present the use of state-of-the-art deep generative models (GAN and VAE) for the estimation of a likelihood of the data. Our results show that on the one hand both GANs and VAEs are able to separate the 'positive' and 'negative' samples in the MNIST case. On the other hand, for the NLST case, neither GANs nor VAEs were able to capture the complexity of the data and discriminate anomalies at the level that this task requires. These results show that even though there are a number of successes presented in the literature for using generative models in similar applications, there remain further challenges for broad successful implementation.

**Keywords:** Anomaly Detection · Generative Models · Variational Autoencoder · Generative Adversarial Network.

## 1 Introduction

A long-standing challenge for Machine Learning is to deal with small datasets and an insufficient amount of labeled data[16]. This is particularly true when there is a significant imbalance in the data with respect to the class (or target variable). We address this challenge by formulating it as an anomaly detection task. Specifically, we train a generative model in an unsupervised fashion with the samples from only one class. We treat the other class as an anomaly, such that our model is expected to produce low likelihood of samples from the other class.

In other words, we consider a Probability Density Estimation process in which the goal is to discover the probability distribution of the *normal* data $p_{data}$, by defining a parametric distribution $p_{model}$ and finding the optimal parameters to approximate $p_{data}$. Computing these optimal parameters $\theta$, means getting the values that maximize the likelihood of the observed data.

Given a set of training datapoints $X = \{x_1, x_2, ..., x_n\}$, we train a generative model to learn the probability distribution $p(x)$. The model inference is based on Maximum Likelihood Estimation (MLE) for the parameters $\theta$. Having the likelihood function $p_{model}(X|\theta)$, the MLE is defined by:

$$\theta_{MLE} = \arg\max_{\theta} p_{model}(X|\theta) = \arg\max_{\theta} \prod_{i} p_{model}(x_i|\theta) \tag{1}$$

When computing the MLE, we find the parameters that maximize the likelihood of the data given our model $p_{model}$.

More specifically, we define the optimization as a minimization of a negative log likelihood given by[17]:

$$E(w) = -\sum_{i} \log p[x_i|f(x_i; w)], \tag{2}$$

where the model $f(.; w)$ is a type of neural network with parameters $w$ defined by the specific generative model. By having this form, the task becomes an optimization process that can be solved using Stochastic Gradient Descent (SGD).

Furthermore, we need to develop a boundary to distinguish anomalies by developing a threshold $\epsilon$, with respect to the learned likelihood. It is also the case that this cut-off is not obvious to identify and relies entirely on experts' opinions[11].

Deep Generative Models are the current unsupervised methods with strong capacity for feature representation, data generation and learning of the data distribution. Their structure, using neural networks, allows them to construct powerful functions from the training and generate new *alike* samples, particularly for high dimensional data, for which density estimation is a long standing problem.

Our particular goal is to apply this approach to difficult applications such as lung cancer screening. Lung cancer alone was responsible for 1.69 million deaths in 2015[3]. Early cancer detection and diagnosis of abnormal anatomies, by means of Computer Tomography (CT), has been a recurrent research topic specially in the Computer Vision domain[14][5].

## 2 Related Work

Two main frameworks gained popularity and acceptance in the deep learning community: Generative Adversarial Networks[4] (GAN) and Variational AutoEncoders[7] (VAE). Since their appearance in 2013-2014, strong research moved into their

[3] http://www.who.int/news-room/fact-sheets/detail/cancer

interpretation, application and development. Currently there are more 200 variations of GANs in terms of training, architecture, loss function, objective and applications[4]. GANs are known for being unstable to train, with several hyperparameters to tune. However, the results are sharp, and could fool the human eye when producing new image samples. VAEs are known for producing blurry results in the new samples. However, their training setup is well defined, and they feature explicit sampling from the learned probability distribution. Due to their proven performance when dealing with high-dimensional datasets in an unsupervised setup, deep generative models are suitable for the design of the anomaly detection framework.

Other recent and important density estimation methods are the autoregressive model (with normalizing flows), the Neural Autoregressive Distribution Estimator (NADE)[9], the real-valued neural autoregressive density-estimator (RNADE)[18], the real-valued non-volume preserving method[2], and the Masked Autoregressive Flow estimation[12]. An improved VAE approach with inverse autoregressive flows[8] has also demonstrated strong capacity for density estimation. In the current version of the work we have not evaluated the autoregressive flow models.

## 3 Applications

### 3.1 MNIST dataset, 2D benchmark setup

We presented an evaluation to the anomaly detection over the benchmark dataset MNIST[5]. For our experiments we split the dataset in a binary classification problem, having an imbalanced setup. We trained only using the Negative Samples (a subset of 9216 images containing equal samples of 0-8). Then we tested the approach using some Positive samples (images of 9).

### 3.2 Lung cancer detection, nodules from NLST 3D dataset

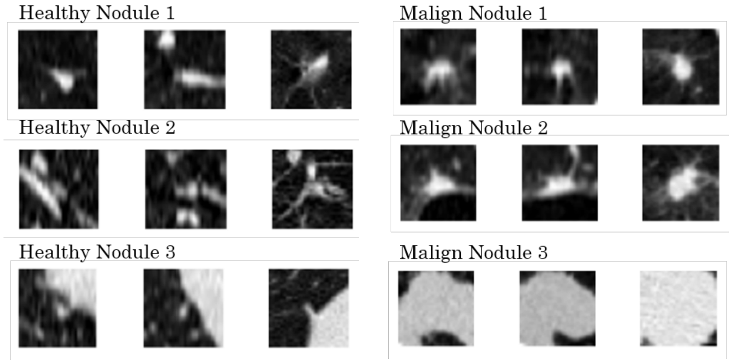
Lung cancer detection usually requires annotated images (cancer, non-cancer) at a nodule (tumor) level, with its additional information such as malignancy, diameter, spiculation or lobulation, and a preferably amount of samples of each class. Recent efforts[6] leveraged from the use of the publicly available datasets with considerable nodule annotations, achieving good performance. However, this supervised approach does not seem to be easily scalable due to the lack of new, equally rich data. In this particular application, the benign nodules of the lung do not share specific characteristics. They are diverse in size, texture, shape, and location. As a consequence, the differentiation between malign nodules is not evident for human perception. Due to the high complexity of the data, we are

[4] https://github.com/hindupuravinash/the-gan-zoo

[5] http://yann.lecun.com/exdb/mnist/

[6] https://github.com/dhammack/DSB2017/blob/master/dsb_2017_daniel_hammack.pdf

not sure how far the abnormal samples are from the normal samples. We would like to test our anomaly detection framework in this scope and evaluate whether the generative models are able to understand class related particularities such as shapes, edges or spatial position, plus additional hidden features. The raw dataset is provided by the NLST (National Lung Screening Trial), consisting of high resolution chest tomographies. The input for our models is the result of a nodule detector. We are dealing with 3D cubes of 32x32x32 mm$^3$ with a voxel size of 1mm$^3$. In our research we designed and implemented 3D models for handling the data, and investigated whether this approach helped for the robust estimation of the probability density. Figure 1 shows some nodule examples, the variation between the data and the difficulty for humans to discriminate healthy from abnormal samples.



(a) Healthy Samples (b) Abnormal Samples

Fig. 1: Examples of samples in the dataset with their axial, coronal and sagital perspective. Figure 1a shows 3 different healthy nodules. Figure 1b shows 3 different nodules identified as abnormal (positive for cancer).

For our experiments the input of the models is a 3D cube of 28x28x28 pixels, the result of a data augmentation process that produces sub patches from the original shape of 32x32x32. For convenience in display, figure 2 shows the 3D image as a set of 25 slices of 28x28 pixels. Table 1 shows the details of the how we organized the data.

# 4 Anomaly Detection Framework

## 4.1 Anomaly Detection with GANs

The reference paper for Anomaly Detection[15], based on work from[19], proposed a framework composed of three steps: (1) learn a manifold $\mathcal{X}$ of a corpus

| | Label 0<br>Healthy | Label 1<br>Cancer | Total |
|---|---|---|---|
| Training | 1722 | 460 | 2182 |
| Validation | 431 | 115 | 546 |
| Testing | 539 | 143 | 682 |

Table 1: Lung nodule dataset after data augmentation

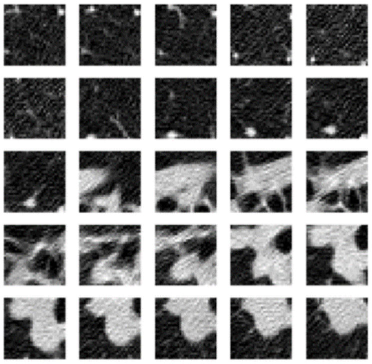


Fig. 2: Displaying 25 slices of 28x28 pixels, as a representation of the cube of 28x28x28 pixels used for training the models.

of *normal* images, (2) map images back to the latent space, and (3) detect abnormal samples using a visual and perceptual component.

The model used to learn the manifold for step (1) is a GAN[4], consisting of a generator $G$ that generates images given latent space samples $z$, and a discriminator $D$ that is trained to distinguish generated images from real data. Both $G$ and $D$ are neural networks.

Mapping an image $x$ back to the latent space in step (2) entails finding some $z_\gamma$ in latent space such that $G(z_\gamma)$ is as similar as possible to $x$.

The visual component of step (3) is the residual loss, which compares similarity of images at pixel level through the generator $G$. The residual loss is defined by

$$\mathcal{L}_R(z_\gamma) = \sum |x - G(z_\gamma)|, \tag{3}$$

where $x$ is the query image and $G(z_\gamma)$ is the most similar generated image. If the generator is able to generate a perfect looking image with respect to the query, the residual loss is $\mathcal{L}_R(z_\gamma) = 0$. The perceptual component is defined as a discriminator loss, based on the discriminator $D$:

$$\mathcal{L}_D(z_\gamma) = \sum |f(x) - f(G(z_\gamma))|, \tag{4}$$

where $f$ is a hidden layer from the discriminator. The features learned from the query image $f(x)$ are compared to the ones of the most similar generated image $f(G(z_\gamma))$.

The method for detecting an abnormal sample consists of using the overall loss composed by a weighted sum of the residual and the discriminator loss. A parameter $\lambda$ sets the relative importance of each loss component:

$$\mathcal{L}(z_\gamma) = (1-\lambda)\mathcal{L}_R(z_\gamma) + \lambda\mathcal{L}_D(z_\gamma). \tag{5}$$

An iterative procedure[19] is used to find $z_\gamma$; starting with a random point $z_1$ in latent space that generates an image $G(z_1)$, and then using equation 5 to find more suitable $z_2, z_3, \dots, z_\gamma$ through stochastic gradient descent (SGD) with momentum. After $\gamma$ steps, if the query image $x$ belongs to the learned distribution of the model, we would expect $G(z_\gamma) \approx x$.

After training, we obtain the closest image to the query $x$, generated by $G(z_\gamma)$ and the loss value $\mathcal{L}(z_\gamma)$. As suggested in the paper, we use equation 5 to set a threshold $\epsilon$ on $\mathcal{L}(z_\gamma)$ for Anomaly Detection. The reasoning is that if the query image $x$ is close to the learned representation, it is consider *normal* and will have a low loss. If $x$ is *abnormal*, it will have a higher loss, above the defined threshold.

### 4.2 Anomaly Detection with VAEs

A Variational Autoencoder[7] (VAE) is a latent variable model that uses neural networks to express the parameters $\phi$ of an approximate posterior distribution $q_\phi(z|x)$ over the latent variables $z$ (the encoder), as well as for the parameters $\theta$ of a generative model $p_\theta(x|z)$ (the decoder), given some prior distribution $p(z)$ for the latent variables. It is trained on maximizing the Evidence Lower Bound (ELBO), a lower bound to the log likelihood $\log p(x)$ of the data. The ELBO can be formulated as:

$$ELBO(\phi, \theta; x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - KL(q_\theta(z|x)||p(z)), \tag{6}$$

where $KL(\cdot||\cdot)$ is the KL divergence between two probability distributions. The first term in equation 6 can be interpreted as a reconstruction error on pixel level, whereas the second term acts as a regularizer. We also use this ELBO as an approximation to the likelihood, for use in our anomaly detection framework.

In our experiments, we use multivariate Gaussians with diagonal covariance:

$$\begin{aligned} q(\boldsymbol{z}|\boldsymbol{x}) &= \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_{enc}(\boldsymbol{z}), \boldsymbol{\sigma}_{enc}(\boldsymbol{z})), & (7)\\ p(\boldsymbol{x}|\boldsymbol{z}) &= \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_{dec}(\boldsymbol{x}), \sigma_{dec} \cdot \boldsymbol{I}), & (8)\\ p(\boldsymbol{z}) &= \mathcal{N}(\boldsymbol{z}|\boldsymbol{0}, \boldsymbol{I}). & (9) \end{aligned}$$

In this case, the KL divergence term from equation 6 can be computed analytically, whereas the expectation in the first term can be approximated efficiently by means of Monte Carlo sampling. During training we use a single sample, but for evaluation of the ELBO in our anomaly detection framework we sample 100 instances from the approximate posterior, in order to find a reliable estimate of the likelihood of a data point. We used a fixed value $\sigma_{dec} = \frac{1}{\sqrt{2}}$, whereas $\boldsymbol{\mu}_{enc}(\boldsymbol{z}), \boldsymbol{\sigma}_{enc}(\boldsymbol{z})$, and $\boldsymbol{\mu}_{dec}(\boldsymbol{x})$ are all expressed by neural networks.

Similar approaches[1] use only the reconstruction error (the first term in equation 6) for anomaly detection, from the perspective of image segmentation. However, by using just that part of the VAE loss function, we are not really estimating the true likelihood $p(x)$. The goal of our Anomaly Detection framework is to estimate how likely it is that an image query belongs to the learned distribution.

### 4.3 Evaluation

After training the generative models, we expect that the model learned specific features from the data and the resultant loss values can be seen as a likelihood value for each data point, measuring how likely it is for that sample to belong to the distribution of normal data. We then expect that the likelihood of normal samples is far greater than for anomalous data. To evaluate this assumption, we took an equal number of *normal* and *anomaly* samples and computed the likelihood value for all the datapoints, using a trained GAN and VAE.

## 5 Results

We present results for the Anomaly Detection framework with GAN architectures (GAN-AD) and VAE architectures (VAE-AD). For both phases the tests were performed for the application cases described in section 3, over the 2D MNIST and 3D NLST datasets. As a general structure, we present:

- High level defined architecture for the GAN-AD and VAN-AD over the NLST dataset,
- Qualitative performance of the models in terms of the generation/reconstruction of samples,
- Visual evaluation of the Anomaly Detector output using plot of density distributions,
- The AUROC score obtained after thresholding the Anomaly Detector output to separate the anomalies based on their likelihood measure.

### 5.1 MNIST 2D setup

For both models we trained with a subset of 9216 images containing normal samples, images of digits 0 to 8. Then we evaluated the approach using an equal number of samples from both classes: 450 positive samples (images of number 9) and 450 additional normal samples.

**GAN-AD** For the MNIST dataset dimensions, we used the proposed DCGAN[13], with similar configuration of the convolutional layers, and the same recommendations for training. These type of implementations are widely explored by the community, so there was no need to tune the hyper-parameters with rigor. We trained for 50 epochs and we computed the anomaly scores using 100 backpropagation steps for finding the optimal $z$ mapping back to latent space. We chose $\lambda = 0.5$ in equation 5, after empirical experimentation.

After the training, we performed the Anomaly Detection framework, obtaining the anomaly scores (equation 5). Figure 3 shows the distribution of the scores corresponding to each class. We found high fragility in the results when we: increased the trained epochs, increased the number of backpropagation steps for finding the optimal $z$ in latent space, and, changed the number of samples used both for training and for evaluation.

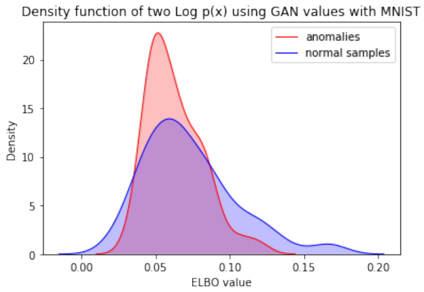


Fig. 3: Density distribution for the anomaly scores obtained with GAN-AD with MNIST dataset. We can perceive that there is no clear separation or threshold between normal and abnormal samples.

For quantitative evaluation, we plot a ROC curve based on thresholding the anomaly score on different values. The result is shown in figure 4. The Area Under Curve (AUROC) value of 0.66 shows that the classifier is somewhat able to separate normal from abnormal samples, although not in a powerful way. As explained before, this value was fragile for the number of samples used in the evaluation and the previous training.

**VAE-AD** For the 2D context of MNIST, we use a simple VAE architecture with 2D convolutions and 2D upsamplings. For this dataset there was no need for a deep level of convolutions or number of units. We trained the VAE for 30 epochs, using the same 9216 images labeled as *normal* (digits from 0 to 8). The training lasted approximately 1 min. The model is able to visually reconstruct a normal sample with high quality, and tries to approximate an abnormal sample with the information it got during training.

The metrics of our model were used for the computation of the likelihood lower bound. For the VAE-AD, we took the trained VAE and passed new samples

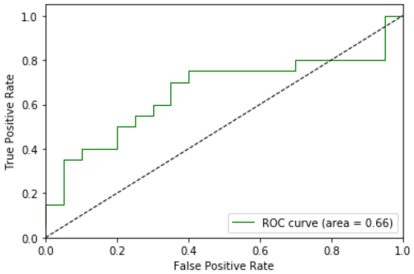


Fig. 4: ROC curve with AUROC score for GAN-AD with MNIST.

through it. For this step we used 450 normal samples and 450 abnormal samples. The result is a density graph composed by the ELBO values. Figure 5 shows the distribution of the results for both types of samples.

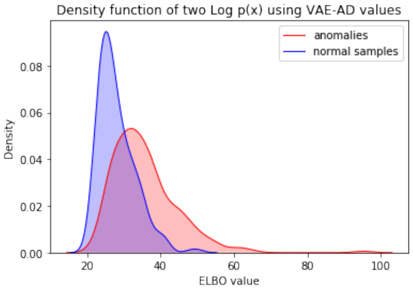


Fig. 5: Density distribution of VAE-AD with MNIST dataset. We can see how the values create a differentiation in the densities. Values greater than 50 are highly probable to be anomalies.

Figure 6 shows the ROC curve and AUROC score results based on thresholding the ELBO scores on different values. The AUROC score of 0.84 shows high potential for differentiation between normal and anomalous samples.

### 5.2 NLST 3D nodules dataset

**GAN-AD** After exhaustive parameter tuning and attempts in training, figure 9 shows the 3D WGAN-GP[6] architecture that was able to learn from the nodule data and produced some visually understandable results.

Training was configured with a seed $z$ of size 100 following a uniform distribution. We trained for 100 epochs, since the loss function for the critic showed optimization around epoch 50 and stops learning from epoch 60. The samples, however, keep improving visually until 100 epochs. Figure 7 shows examples of new data produced by the GAN.

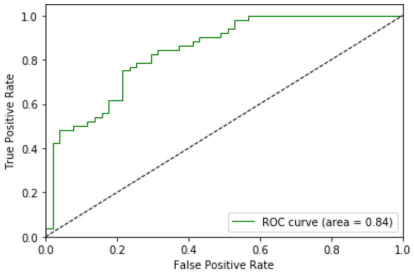
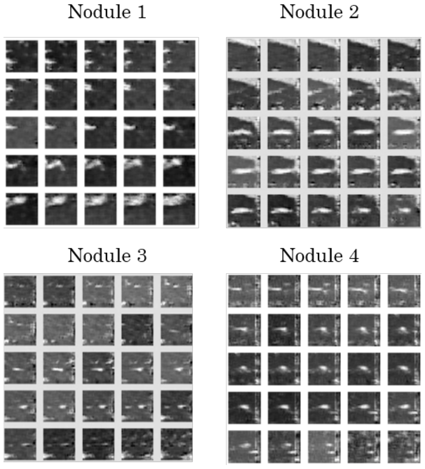


Fig. 6: ROC curve with AUROC score for VAE-AD for MNIST. The result implies that the framework has the potential to discriminate normal from abnormal samples.


Nodule 1

Nodule 2

Nodule 3

Nodule 4

Fig. 7: Four nodules generated by the 3DGAN

While comparing different generated images from the variations in training of WGAN-GP, we notice a partial mode collapse[3] in the samples. We can see that the images look similar and they are not able to create complex shapes as seen in the training data. This was the case for less sharp images generated from simpler architectures or with change on the random seed $z$. Even when the generator is able to construct simple shapes, they are very similar to each other.

With the final trained architecture as shown in figure 9, we compute the metrics proposed in our methodology.

We used the test split, 120 normal samples and 120 abnormal samples, for calculation of the loss score. We ran 100 backpropagation steps for mapping images into the latent space, and we chose $\lambda = 0.5$ in equation 5, after empirical experimentation. The experiment setup showed that backpropagating in the latent space was resource consuming, taking almost 30 seconds per image for

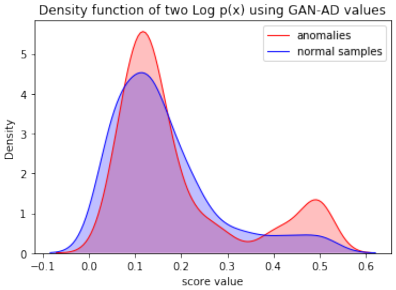


Fig. 8: Density distribution of values using GAN-AD with NLST

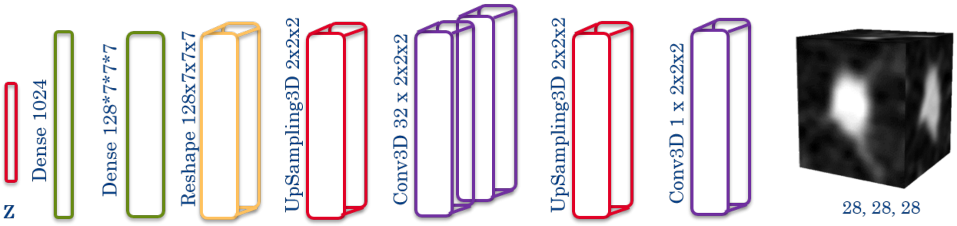


Fig. 9: Trained 3D WGAN-GP architecture for Generator

100 steps. Also, giving more weight $\lambda$ to one loss did not improve the resulting optimization. Figure 8 shows the distribution of the results. Visually, it is clear that the model is not able to differentiate the distribution of normal samples from abnormal, as they overlap.

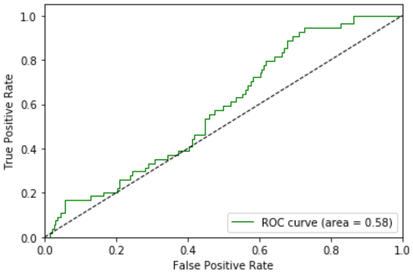


Fig. 10: ROC curve with AUROC score for GAN-AD with NLST. The result implies the classifier was not able to discriminate any feature from normal to abnormal samples.

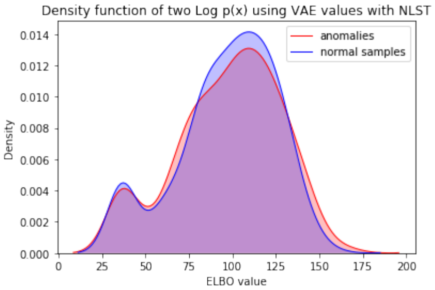


Fig. 11: Density distributions of VAE-AD outputs for NLST dataset

Figure 10 shows the ROC curve and AUROC score results. A value of 0.58 implies that the input features were not relevant enough, and the classifier was able to perform just better than random guessing.

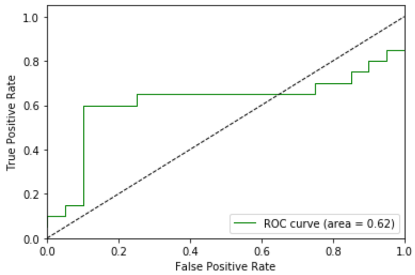


Fig. 12: ROC curve with AUROC score for VAE-AD with NLST. The result implies the classifier performs better than random choice, but still, it does not have the capacity for discriminating normal from abnormal samples.

**VAE-AD** Based on the performance of the 3D WGAN-GP architecture, we trained a 3D VAE using a similar setup of 3D convolutional layers and Upsampling3D. Figure 13 shows the architecture used for the encoder.

Using the same 1722 normal nodules as for GANs, we trained the model for 100 epochs. As for the VAE-AD, we used the resultant metrics for computation of the likelihood lower bound. We used the trained 3D VAE and passed new samples through it. We used 115 normal samples and 115 anomaly samples. The distribution of the values is shown in figure 11. Visually it is clear that the distributions overlap, not making an ideal separation between normal and anomalous samples.

Figure 12 shows the resulting ROC curve for thresholding on different values. We can see that even if we perform better than random guessing, the given

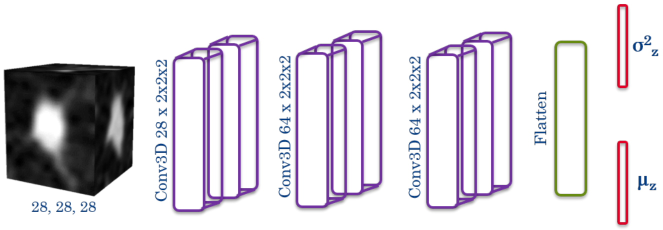


Fig. 13: Trained 3D VAE architecture for Encoder

representation was not enough to make a clear distinction between normal and abnormal samples. Empirically, we noticed that increasing the number of samples could improve this score. We used samples from the validation split to perform more experiments, but the AUROC score was not greater than 0.62. In presence of additional data, more experimentation could give better performance.

## 6 Conclusion

This work defined a comparative Anomaly Detection framework for two state-of-the-art deep generative models. We used a metric based on likelihood estimation, and created an evaluation protocol for the identification of anomalies. The concept of likelihood estimation is closer related to the VAE framework. GAN computes a loss score that has no direct link to probability theory, but that can be interpreted as an anomaly score.

For the first use case with MNIST, the GAN approach is fragile and it is dependent on hyperparameter tuning and the number of training samples. When evaluated with ROC curves, it did not show the expected performance as for the reference paper[15]. Results showed an AUROC score of 0.66 when training with fewer than 10.000 samples of the normal class. Since our scope was imbalance and scarcity of samples, this was a realistic scenario for evaluating the model. Regarding the VAE approach, it is easy to train, not time consuming and the scores are obtained in a straightforward manner. The resulting ROC curve shows potential for the separation of abnormal samples, with a value of 0.84.

The use case of lung cancer detection at a 3D image nodule level showed that neither of the generative models are able to capture the feature complexity of the data. The GAN approach evaluation showed a performance just better than random with an AUROC score of 0.58. With a VAE we obtained an AUROC score of 0.62, which we consider not significantly relevant due to the importance of the abnormal samples.

Previous work[10] showed that GAN-AD did not perform well in an NLST 2D setup. We performed experiments over 3D architectures, expecting a richer model. However, we saw that deep generative models are still not robust enough in cases such as CT data of lung cancer at a nodule level.

## 7 Discussion

The current results showed that deep generative models are a suitable approach for anomaly detection and developing models in highly imbalanced settings. However, their applicability depends on the complexity of the dataset. Particularly, for cancer detection at a nodule level we have yet to develop models that precisely model the distribution of the images to a level that the malignant tumors can be distinguished from the benign lesions. Recent developments of Autoregressive Models with normalizing flows for density estimation presented in the background section (See section 2) offer significant advances to current generative models and are hence a strong candidate for a solution in this domain.

## Acknowledgements

The authors thank the National Cancer Institute for access to NCIs data collected by the National Lung Screening Trial (NLST). The statements contained herein are solely those of the authors and do not represent or imply concurrence or endorsement by NCI.

## References

1. Baur, C., Wiestler, B., Albarqouni, S., Navab, N.: Deep autoencoding models for unsupervised anomaly segmentation in brain MR images. CoRR **abs/1804.04488** (2018), http://arxiv.org/abs/1804.04488
2. Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using real NVP. CoRR **abs/1605.08803** (2016), http://arxiv.org/abs/1605.08803
3. Goodfellow, I.J.: NIPS 2016 tutorial: Generative adversarial networks. CoRR (2017), http://arxiv.org/abs/1701.00160
4. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Bing Xu, D.W.F., Ozair, S., Courville, A., Bengio, Y.: Generative Adversarial Nets (2014), https://arxiv.org/abs/1406.2661
5. Greenspan, H., van Ginneken, B., Summers, R.M.: Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique. IEEE Transactions on Medical Imaging **35**(5), 1153–1159 (May 2016). https://doi.org/10.1109/TMI.2016.2553401
6. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. CoRR **abs/1704.00028** (2017), http://arxiv.org/abs/1704.00028
7. Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes. ArXiv e-prints (Dec 2013)
8. Kingma, D.P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., Welling, M.: Improved variational inference with inverse autoregressive flow. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29, pp. 4743–4751. Curran Associates, Inc. (2016), http://papers.nips.cc/paper/6581-improved-variational-inference-with-inverse-autoregressive-flow.pdf

9. Larochelle, H., Murray, I.: The neural autoregressive distribution estimator. In: Gordon, G., Dunson, D., Dudk, M. (eds.) Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 15, pp. 29–37. PMLR, Fort Lauderdale, FL, USA (11–13 Apr 2011), http://proceedings.mlr.press/v15/larochelle11a.html
10. Mendoza, R.: Anomaly Detection with generative models. Master's thesis, Eindhoven University of Technology (2018)
11. Papadimitriou, S., Kitagawa, H., Gibbons, P.B., Faloutsos, C.: Loci: fast outlier detection using the local correlation integral. In: Proceedings 19th International Conference on Data Engineering (Cat. No.03CH37405). pp. 315–326 (March 2003). https://doi.org/10.1109/ICDE.2003.1260802
12. Papamakarios, G., Pavlakou, T., Murray, I.: Masked Autoregressive Flow for Density Estimation. ArXiv e-prints (May 2017)
13. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. CoRR **abs/1511.06434** (2015), http://arxiv.org/abs/1511.06434
14. Roth, H.R., Lu, L., Liu, J., Yao, J., Seff, A., Cherry, K.M., Kim, L., Summers, R.M.: Improving computer-aided detection using convolutional neural networks and random view aggregation. CoRR **abs/1505.03046** (2015), http://arxiv.org/abs/1505.03046
15. Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. CoRR **abs/1703.05921** (2017), http://arxiv.org/abs/1703.05921
16. Shi, J., Zhou, S., Liu, X., Zhang, Q., Lu, M., Wang, T.: Stacked deep polynomial network based representation learning for tumor classification with small ultrasound image dataset. Neurocomputing **194**, 87 – 94 (2016). https://doi.org/https://doi.org/10.1016/j.neucom.2016.01.074, http://www.sciencedirect.com/science/article/pii/S0925231216002344
17. Sunehag, P., Trumpf, J., Vishwanathan, S.V.N., Schraudolph, N.N.: Variable metric stochastic approximation theory. In: van Dyk, D., Welling, M. (eds.) Proc. 12-th Intl. Conf. Artificial Intelligence and Statistics (AIstats). Workshop and Conference Proceedings, vol. 5, pp. 560–566. Clearwater Beach, Florida (2009)
18. Uria, B., Murray, I., Larochelle, H.: RNADE: The real-valued neural autoregressive density-estimator. ArXiv e-prints (Jun 2013)
19. Yeh, R., Chen, C., Lim, T.Y., Hasegawa-Johnson, M., Do, M.N.: Semantic Image Inpainting with Deep Generative Models. arXiv:1607.07539 (2016), https://arxiv.org/abs/1607.07539