Reliable Anomaly Detection using Variational Autoencoders

Stéphane Caron *[1], Thierry Duchesne[1], François-Michel De Rainville[2], and Samuel Perreault[1]

[1] *Département de mathématiques et de statistique, Université Laval, Québec, Canada*
[2] *Intact Insurance, Québec, Canada*

April 20, 2020

Abstract

In this paper, we propose an anomaly detection methodology that aims to detect anomalies among complex data, such as images, and that is based on a confidence level rather than on a certain threshold or metric. In order to do that, we demonstrate the usefulness of using variational autoencoders (VAE) to deal with complex data and also to have some kind of representation from which we can apply hypothesis testings. From our experiments, we can show that our approach is able to detect images that are outliers in a given dataset by only specifying a certain confidence level. By using this approach, the anomaly detection becomes feasible for real-world complex images and is also easier to maintain and interpret considering the detection is made on a confidence level rather than on a vague and possibly changing metric.

Keywords: autoencoder, hypothesis testing, confidence level, unsupervised.

*Corresponding author: stephane.caron.9@ulaval.ca

1 Introduction

Anomaly detection is a challenging topic that generated a lot of research in statistics, machine learning and more recently in computer vision. There are many applications of anomaly detection in fields such as cyber-intrusion, financial and insurance fraud detection, medical anomaly detection or industrial damage identification (Chandola et al. (2007)). An anomaly is defined by Zimek and Schubert (2017) as events, items or observations that differ significantly from the majority of the data. An anomaly, or what can also be called an outlier, is something intrinsic to many fields related with data because it is by nature, something interesting to extract or to remove from a given source of data. It can be interesting to extract because it is what we are looking for in the problem or it could interesting to remove prior to another learning problem.

One challenge with anomaly detection is that we often deal with unlabeled data, that means the problem usually needs to be tackled with an unsupervised approach. Another important challenge, especially with unsupervised approaches, is that those detection algorithms often need a threshold. That threshold allows us to take a decision regarding a certain anomaly. Finally, those challenges become even more problematic when dealing with complex data such as images.

In that context highly complex data, neural networks are commonly used because their stacked layers are able to start from a complex input, like an image, and compress that information into smaller and richer representations. To deal with data without labels, a family of neural networks called the autoencoders are often used. Many applications of autoencoders exists in anomaly detection where the reconstruction error can used as a indicator of anomaly. However, those methodologies require a specific threshold, that can be difficult to define or can change in time. That is what An and Cho (2015) tackled in their paper where the authors suggest a reconstruction probability, a measure that is more objective than the reconstruction error and does not require a threshold. However, the measure is still based on the reconstruction, where that could be problematic in the context of complex images compared to use a measure based on a more compressed representation.

In this study, we propose an approach that aims to simplify the thresh-

old determination in the case of unsupervised anomaly detection. With this contribution, we are not proposing new methods, but simply put togheter the statistical theory behind hypotheses testing and autoencoders, more specifically variational autoencoders, to learn complex data structures and encode them into simpler representations. Those representations can then be tested with some confidence level, instead of a threshold based on a metric, to determine which observations are anomalies.

2 Background

First, we will briefly describe the theory behind autoencoders and how it can be used in the context of anomaly detection. Then, we will describe one family of autoencoders, the variational autoencoders, and how its outputs can be used along with statistical hypothesis testing to detect anomalies.

2.1 Autoencoders

An autoencoder is an unsupervised neural network technique that aims to learn an efficient intermediate representation of an input (Goodfellow et al. (2016)). To achieve this objective, the autoencoder has 2 components: an encoder and a decoder. The encoder receives an input x and and converts it to a hidden representation z. The decoder receives a representation z and decodes it back to retrieve as much as possible the input x. This structure is illustrated in the figure 1. Historically, autoencoders were known as a dimensionnalty reduction method, but it has now more applications by learning latent variables rich in informations.

Autoencoders intuition is to build back the input x by passing through the 2 components (encoder and decoder). As such, this kind of model does not need any target, so we say it is an unsupervised method. The training of the parameters is mainly done by minimizing the reconstruction error. The loss could then be given by a function in the form of:

$$L(x, p_\phi\{q_\theta(x)\})$$

where $q(x, \theta)$ is the encoder and $p(z, \phi)$ is the decoder function. The minimization of that loss function is done by gradient descent. For instance, the encoder

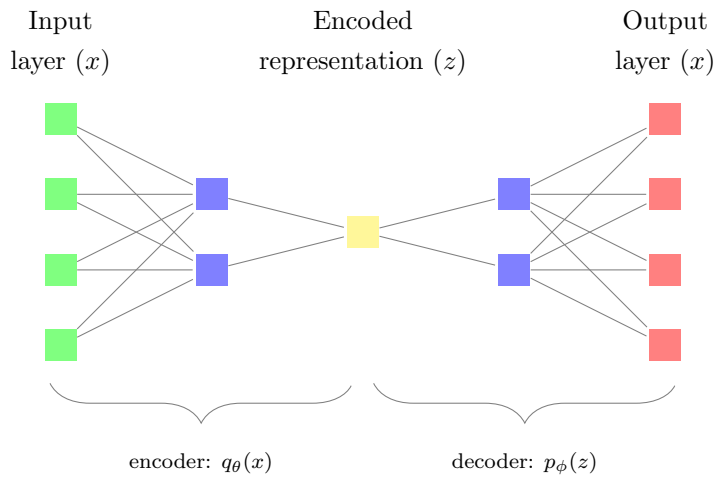

Figure 1: Basic autoencoder structure example

and decoder parameters are gradually updated by taking the derivatives of the loss functions regarding to the different parameters:

$$\Theta \leftarrow \Theta - \epsilon * \frac{\partial L}{\partial \Theta} \tag{1}$$

where ϵ is a learning weight that aims to control the size of the learning steps and $\Theta : \{\theta, \phi\}$ includes both encoder and decoder parameters.

Once the autoencoder is adequately trained, we can use the reconstruction error as an anomaly score. In Aggarwal (2016), that anomaly detection method is part of a catogory of algorithms based on linear or non-linear models. In this group of methods, we first fit a linear or non-linear model to the data and we use the reconstruction error, or the residual, as an anomaly score. Methods based on linear regression, principal components analysis (PCA), or matrix factorization are also part of this broad family of anomaly detection approaches. However, reconstruction is not the only criteria that can be used from autoencoders. In the next section, we cover a specific type of autoencoder, the variational autoencoder, and see how could we leverage another component as an anomaly score.

2.2 Variational autoencoders

The variational autoencoders (VAE) Kingma and Welling (2013) have a slightly different approach than other kinds of autoencoders. In fact, instead of en-

coding a hidden representation of size n, it outputs two vectors of size n: a vector of means $\boldsymbol{\mu}$ and a vector of standard deviations $\boldsymbol{\sigma}$. Those two vectors are then used to generate a hidden representation that is sampled from a Gaussian distribution that gives this specific property to VAE, which is to have a continous latent representation. This is the main difference with other kinds of autoencoders. In other words, the basic autoencoders learn a reprensetation that "points" somewhere in the latent space, while VAE learns a representation that points to an "area", an area that is defined by the mean $\boldsymbol{\mu}$ and the standard deviation $\boldsymbol{\sigma}$ of the latent space. The figure 2 illustrates the basic structure of a variational autoencoder. In the figure, we can see that the input data first pass through some layers (fully-connected or convolutional). At some point near the encoded representation, the layers are split into 2 components (μ and σ). The z representation is sampled for a Gaussian distribution using sample the parameters μ and σ corresponding to the values of those layers. Once we have the sampled representation, it is decoded back to the same size as the input.

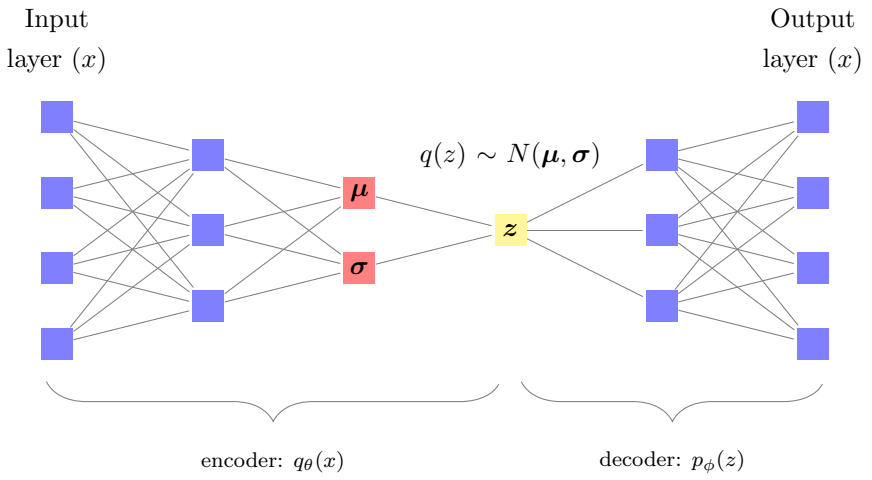

Figure 2: Variational autoencoder structure.

When it comes to minimizing the loss, the variational autoencoder is also different from other autoencoders. In fact, there is another component added to basic reconstruction criterion. The loss function is then given by the sum of 2 components:

$$L(x, p_\phi\{q_\theta(x)\}) + D_{KL}\big[q_\theta(z|x)||p(z)\big]$$

where D_{KL} is the Kullback-Leibler divergence. Its goal is to ensure that the encoded distribution $q(z)$ and a target distribution $p(z)$ are similar. The function $p(z)$ is a Gaussian distribution $N(0, I)$. In this kind of autoencoder, the hidden representation is stochastic because it is coming from a probability distribution. In order to simplify the derivatives in the backpropagation, we do a clever trick called the "reparametrization trick". In fact, it is possible for some distribution (such as the Gaussian), to separate the parameters (μ and σ) from the stochasticity. Concretely, we can express a normally-distributed variable as:

$$z = \mu + \sigma \odot \epsilon$$

where $\epsilon \sim N(0, 1)$. In brief, that means that the z layer in the figure 2 is generated from the 2 parameters layers μ and σ and a normal sample. The backpropagation then ignores the stochastic component, and derivates the parameters layers only, which simplifies at lot the optimization process. At the end, that Kullbach-Leibler loss component acts as a regularizer in the optimisation (Kingma and Welling (2013)).

2.3 Hypothesis testing

Hypothesis testing is a method of statistical inference. It aims to test a null hypothesis (H_0) versus an alternative hypothesis (H_1). In the context of anomaly detection, we could define hypotheses that allow us to test if a certain observation is coming from an expected population or not:

$\boldsymbol{H_0}$: x_i comes from the population P
$\boldsymbol{H_1}$: x_i does not come from the population P

Once we have defined our hypothesis, we need a test method that will includes statistical assumptions about our sample. That generally consists of two components: an expected distribution and a metric. Having those two components, we can compute what we call a p-value. A p-value is defined as the probability, under null hypothesis, to observe the test statistic from the assumed distribution. When we have a small the p-value, it means that the null hypothesis may not adequatly explain the observation. The null hypothesis is then rejected when the value is less than a certain threshold α, which is referred to as the level of significance. When the null hypothesis is true and the underlying random variable is continuous, then the probability distribution of the p-value is

uniform on the interval [0, 1]. In this project, we are aiming to benefit from that level of significance rather than on a certain metric to detect anomalies.

3 Methodology

We propose an anomaly detection framework where we use the hidden representation of a variational autoencoder and transform that representation to a metric we can apply hypothesis testing. Ultimately, that hypothesis testing allows us to detect anomalies with a certain level of significance.

3.1 Approach

In our approach, we suppose we have access to a dataset that contains mostly "normal" observations. Considering we have access to such dataset, we use a variational autoencoder to learn the distribution of that "normal" population. That distribution is essentially contained in the hidden representation, or more specifically in the layers μ and σ of the VAE. Like we described in the section 2.2, VAE has the particularity of having a loss component applied to the latent representation, ensuring that this latent encoding follows a prior distribution, which is $N(0, I)$ in our case. Once we have trained the autoencoder, we can use the parameters of the encoder part of the model to encode each instances of our dataset into a μ and σ vectors. As a simple approach, we can average these two vectors to summarize the dataset information into two new vectors: $\hat{\mu}$ and $\hat{\sigma}$. Suppose our dataset contains n observations and we encoded our latent representation so that we have k latent dimensions, the i element of the $\hat{\mu}$ and $\hat{\sigma}$ vectors are given by:

$$\mu^{(i)} = \frac{1}{n} \sum_{k=1}^{n} \mu^{(k)}$$
$$\sigma^{(i)} = \frac{1}{n} \sum_{k=1}^{n} \sigma^{(k)}$$

Because of that, we can expect the μ and σ layers of new "normal" instances to have a small Kullbach-Leibler distance from a $N(0, I)$ distribution. Kullbach-Leibler (KL) distance is a measure a distance between 2 distributions. At the opposite, new outlier instances should have μ and σ layers that are further from the prior distribution, so a greater KL distance. The proposed methodologie to

compute p-values is describe in the algorithm 1.

Algorithm 1: VAE anomaly detection algorithm

Input: Inliers dataset $x^{(j)}, j = 1, ..., m$,
Testing dataset with unknown anomalies $y^{(i)}, i = 1, ..., n$
Output: p-values for all test instances $p^{(i)}, i = 1, ..., n$
θ, $\phi \leftarrow$ train encoder ($q_\theta(x)$) and decoder ($p_\phi(z)$) VAE parameters;
for *j=1 to m* **do**
  $\mu^{(j)} = p_\theta(x^{(j)})["mu"]$;
  $\sigma^{(j)} = p_\theta(x^{(j)})["sd"]$;
  $kl^{(j)} = kl\_distance(\mu^{(j)}, \sigma^{(j)})$
end
$kl\_sorted = sort(kl)$;
for *i=1 to n* **do**
  $\mu^{(i)} = p_\theta(y^{(i)})["mu"]$;
  $\sigma^{(i)} = p_\theta(y^{(i)})["sd"]$;
  $kl_test^{(j)} = kl\_distance(\mu^{(i)}, \sigma^{(i)})$;
  $p^{(i)} = Q_{kl\_sorted}(kl\_test)$ where Q is the quantile
end
return $\boldsymbol{p}$

In our proposed approach, the p-value is computed from an empirical distribution, which is the Kullbach-Leibler distances of all training instances. At the end, we are testing if a new observation is coming from the inliers population :

$$\boldsymbol{H_0}\text{: } y^{(i)} \text{ comes from the population } X$$
$$\boldsymbol{H_1}\text{: } y^{(i)} \text{ does not come from the population } X$$

Because we made the hypothesis that X is mostly, if not entirely, inliers, we could rephrase our test :

$$\boldsymbol{H_0}\text{: } y^{(i)} \text{ is an inlier}$$
$$\boldsymbol{H_1}\text{: } y^{(i)} \text{ is an outlier}$$

Finally, once we have the p-values of all instances of test dataset, we can use a level of significance α to conclude if it's an outlier (see the algorithm 2).

Algorithm 2: Outlier decision algorithm

Input: p-values for all test instances $p^{(i)}, i = 1, ..., n$,
level of significance α
Output: outlier indicators $o^{(i)}, i = 1, ..., n$
for *i=1 to n* **do**
  if $p^{(i)} < \alpha$ **then**
    $o^{(i)} = true$
  else
    $o^{(i)} = false$
  end
end
return ***o***

3.2 Adapt for complex data

Parler de pourquoi utiliser les representation encoder pour le test au lieu de l'erreur de reconstruction. Peut-etre parler de la perceptual loss pour dealer avec une reconstruction difficile pour des images complex.

3.3 Hypothesis testing advantages

Parler du fait que c'est simple de definir un level of significance versus trouver une metric quelconque.

4 Experiments

4.1 Datasets

4.2 Models tested

4.3 Discussion

5 Conclusion

6 Acknowledgment

References

Aggarwal, C. C. (2016). *Outlier Analysis*. Springer Publishing Company, Incorporated, 2nd edition.

An, J. and Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability.

Chandola, V., Banerjee, A., and Kumar, V. (2007). Anomaly detection: A survey.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. cite arxiv:1312.6114.

Zimek, A. and Schubert, E. (2017). *Outlier Detection*, pages 1–5. Springer New York, New York, NY.