Poster CAP

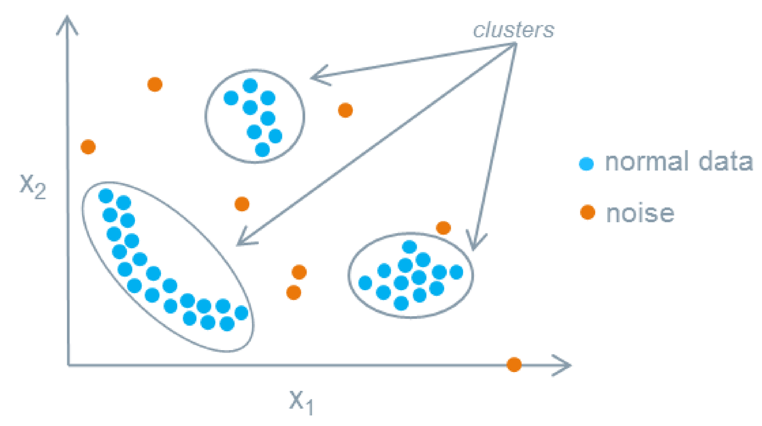
# An Encoding Adversarial Network for Anomaly Detection (Construction d'espace latent pour la detection d'anomalies par apprentissage adversarial)


**Elies Gherbi**[1,2], Blaise Hanczar[1], Jean-Christophe Janodet [1], Witold Klaude[3]

[1]IBISC, Univ Evry, Universite Paris-Saclay , [2]IRT SystemX, [3]Renault

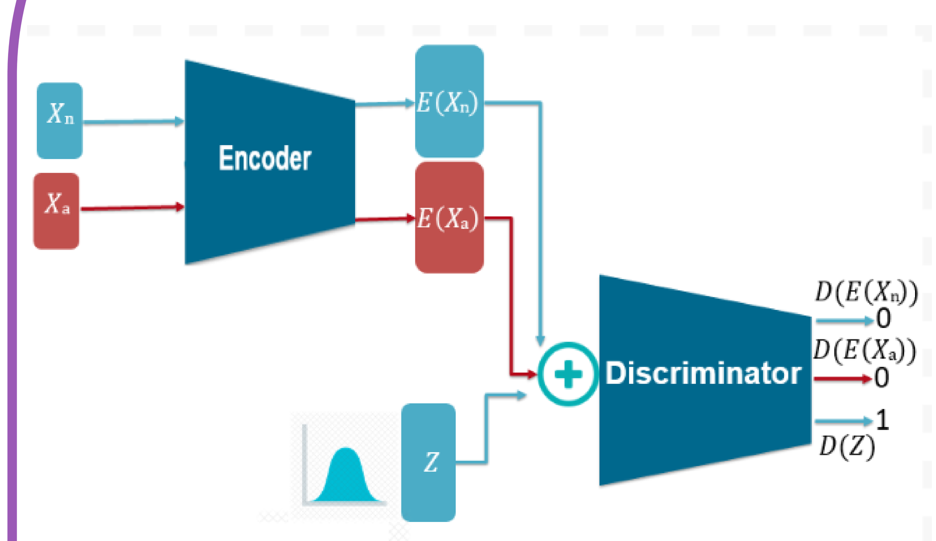

## Issues Overview

We can formulate the anomaly detection problem as follow. Let $D$ be a data set containing a large number of normal examples (the normal states of the system). A model M must learn the distribution function $P_X$ over the normal data during training. Then, given any test example $x$ , it must determine whether x deviates from the learned distribution $P_X$ by using an anomaly score function $a(x)$. In this work "Encoding Adversarial Network", consists to project the example $X_n$ and $X_a$ into a small space, called decision space.
Where given an example $x$ we measure the degree of anomaly of an example $a(x)$.



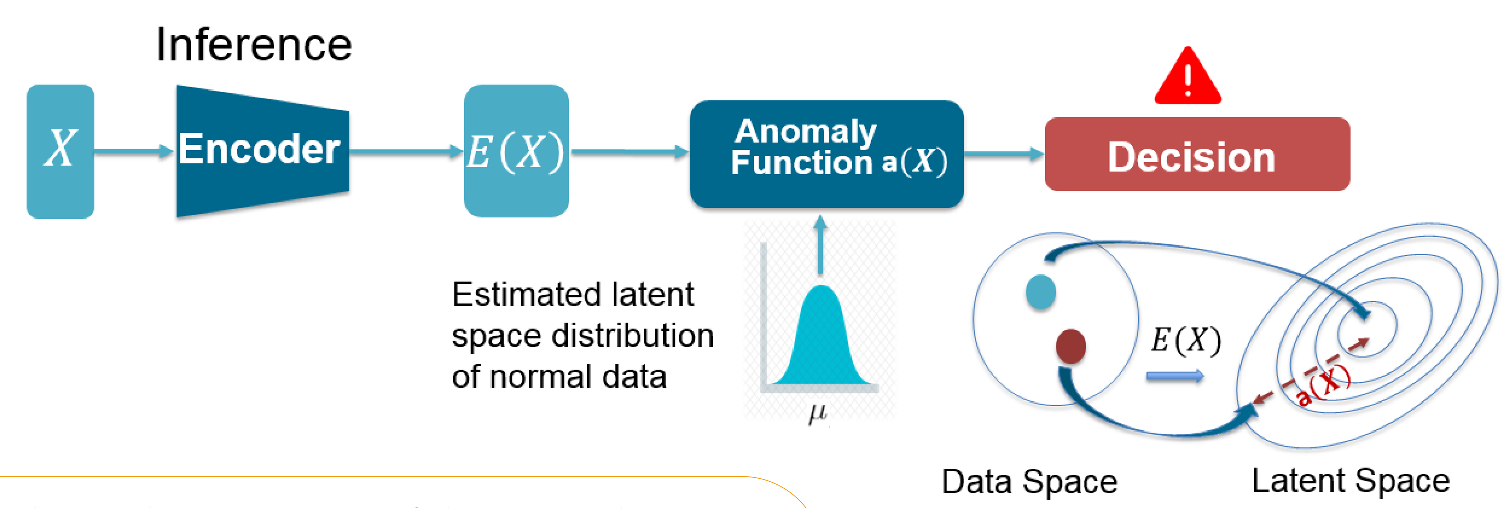

## Encoder Adversarial Network

### Architecture




AnoEAN is composed of two neural networks.

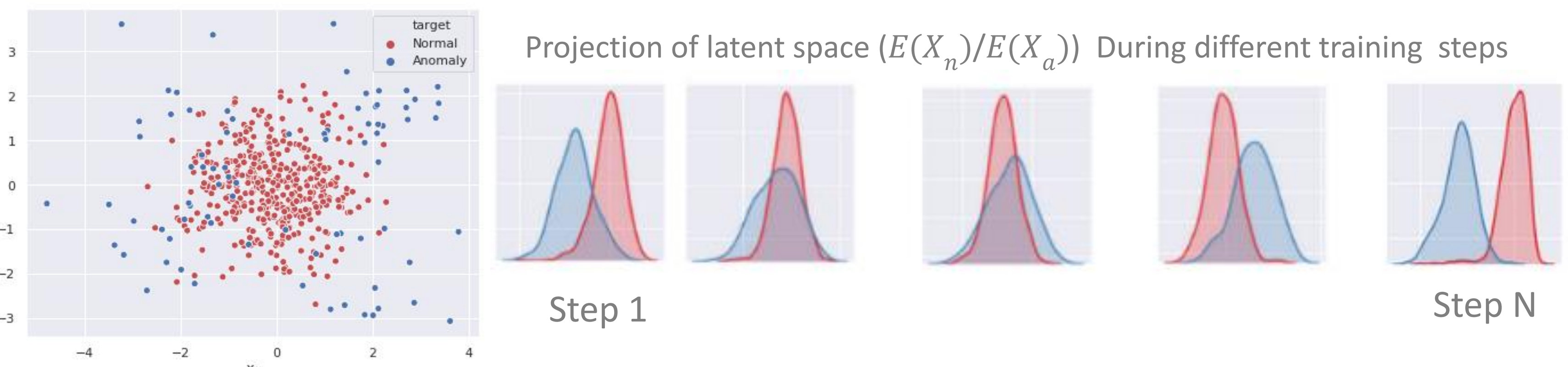
- We call *encoder*, the neural network that projects the examples from the original space into the decision space $E(x)$ by projecting normal examples in $P_z$ and anomalies outside $P_z$.
- we use a second network called discriminator. In the same way as GANs, which receives as input a vector of the decision space $Z$ and predicts if this vector comes from $P_z$ or from the encoder by returning a probability




In the inference phase, the prediction of the anomalies requires only the use of the encoder.
The anomaly score is the Mahalanobis distance between $E(x)$ and $\mu$.

This distribution is supposed to tend to $E(x) = N(0; I)$. Because of the finite size of the training set, our experiments showed that the projection distribution of normal examples could diverge slightly from $P_z$. We represent this distribution by a Gaussian distribution $N(\mu; \sum)$ whose parameters are assessed with a validation base.

### Algorithm and Theoretical analysis



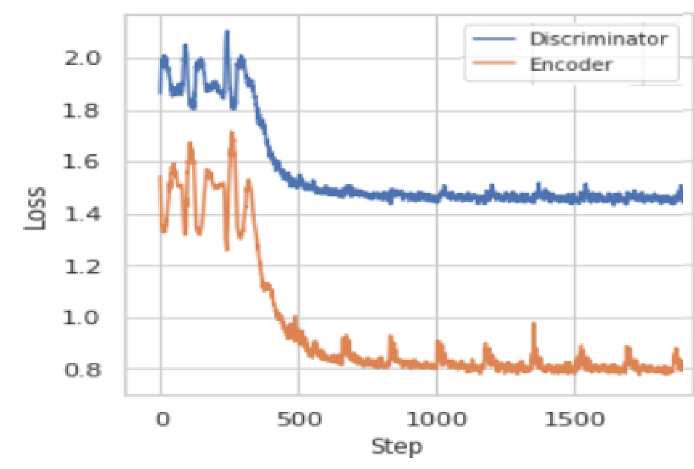

We added an new constraint on the loss function compared to original GANs. By doing this The encoder must therefore both misleads the discriminator on the normal example projection $E(X_n)$ to make it believe that it comes from $P_z$ and help the discriminator differentiate $P_z$ from the projection of the anomalies $E(X_a)$.

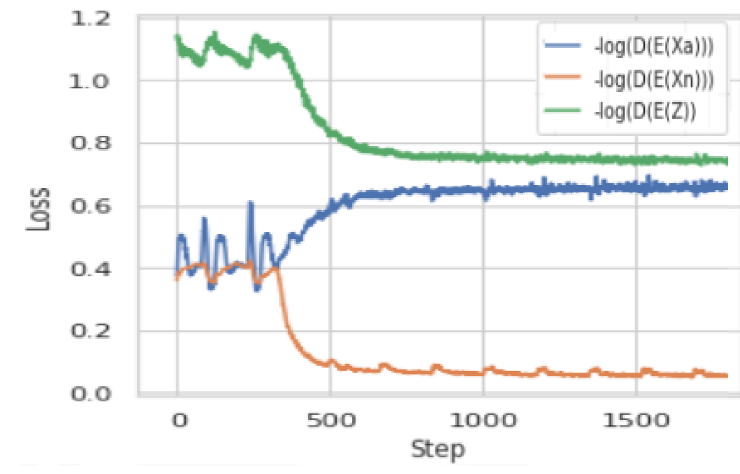
$$L_D = -\mathbf{E}_{z\sim p_z}[\log(D(z))] - \mathbf{E}_{x_n\sim p_{x_n}}[\log(1-D(E(x_n)))] - \mathbf{E}_{x_a\sim p_{x_a}}[\log(1-D(E(x_a)))]$$

$$L_E = \mathbf{E}_{x_n\sim p_{x_n}}[\log(1-D(E(x_n)))] + \mathbf{E}_{x_a\sim p_{x_a}}[\log(D(E(x_a)))]$$

$$a(x) = \sqrt{(E(x)-\mu)^T\Sigma^{-1}(E(x)-\mu)}$$

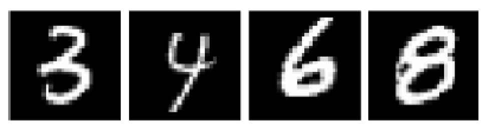
## Experimentation and results



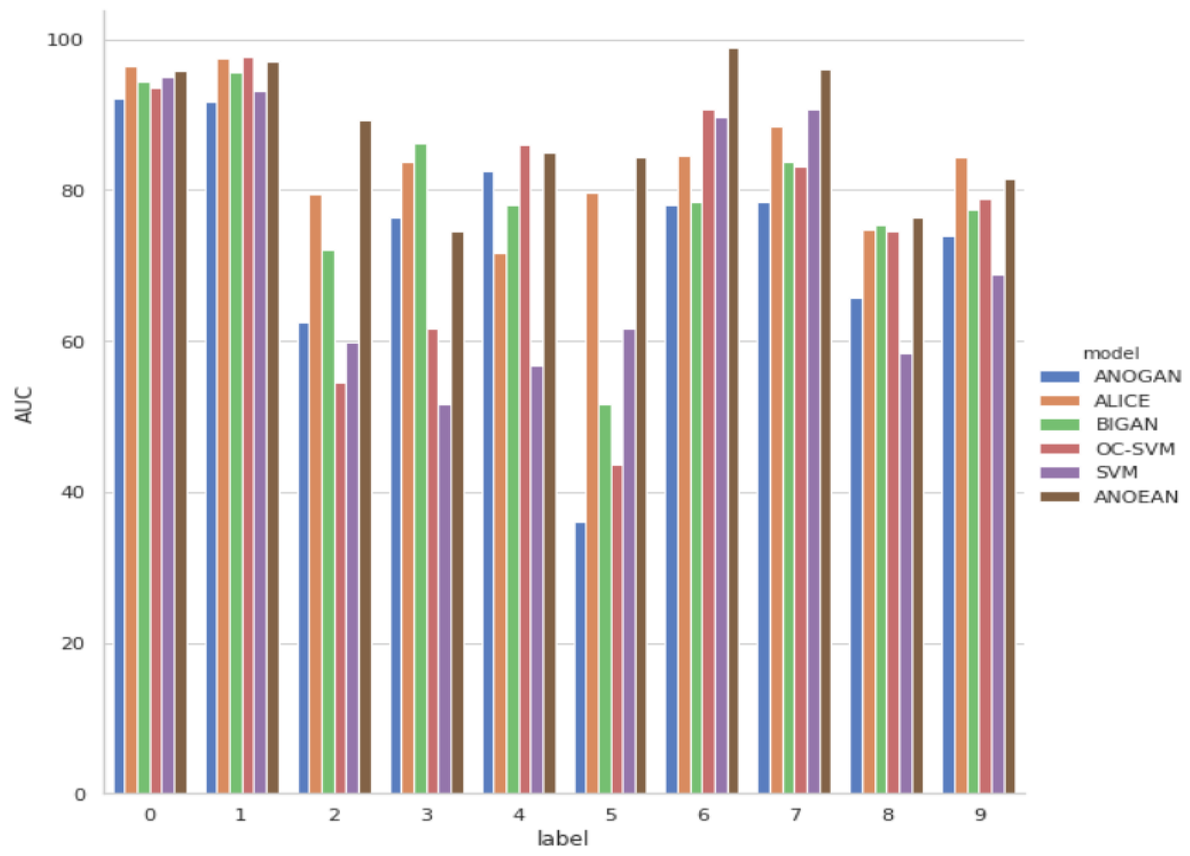




The green and blue curves show that the discriminator is confused through the learning steps: it cannot differentiate between the distributions over z and $E(X_n)$. Therefore, the encoder is getting better with respect to the approximation of $P_z$. At the same time, we see that the orange loss curve keeps decreasing, which means that the distribution of anomalous examples $E(X_a)$. diverges from $P_z$

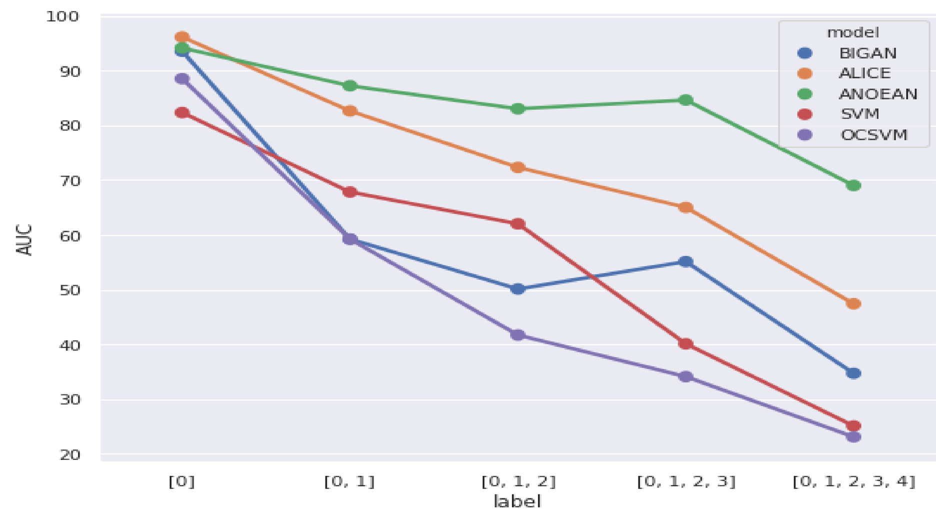
### Image Dataset (Mnist)



Each digit is successively considered as the normal class, the others being seen as anomalies..




We are interested in the problem of identifying anomalies in a dataset where the normal class is heterogeneous (composed of several digits).




### Network Dataset (KDD)

| AUC | F1 | ROC | accuracy | Modle |
|---|---|---|---|---|
| 79.6 | 80.5 | 82.3 | 81.9 | AnoGAN |
| 93.8 | 87.1 | 95.4 | 97.3 | EGBAD |
| 93.7 | 88.1 | 95.7 | 97.0 | ALAD |
| 94.1 | 89.3 | 97.0 | 97.2 | OCSVM |
| **98.0** | 95.0 | 99.1 | 98.0 | **AnoEAN** |
| 97.3 | 93.3 | 98.9 | 97.2 | SVM |

Table 1: NSL-KDD

| AUC | F1 | ROC | accuracy | Modle |
|---|---|---|---|---|
| 72.9 | 73.4 | 79.3 | 77.1 | AnoGAN |
| 95.4 | 96.6 | 98.4 | 98.6 | EGBAD |
| 89.1 | 93.9 | 97.9 | 97.6 | ALAD |
| 73.8 | 87.0 | 88.3 | 88.1 | OCSVM |
| **97.5** | 96.3 | 99.1 | 98.5 | **AnoEAN** |
| 97.2 | **98.7** | 98.7 | 99.1 | SVM |

Table 2: KDD99

## Conclusion and future work

- Adapt our method to the case where no anomalies are available in the training set.
- Time series approaches to learn the normal behavior of linux kernel embedded in autonomous cars