

# Détection d'anomalies basée sur les représentations latentes d'un autoencodeur variationnel

## Résumé de mon mémoire de maîtrise

Stéphane Caron

Université Laval

18 avril 2021

# Contenu

## 1 Introduction

## 2 Mise en contexte

- Méthodes de détections d'anomalies
- Les autoencodeurs
- Les autoencodeurs variationnels

## 3 Méthode proposée

## 4 Expérimentations

- Jeux de données et méthodes testées
- Résultats

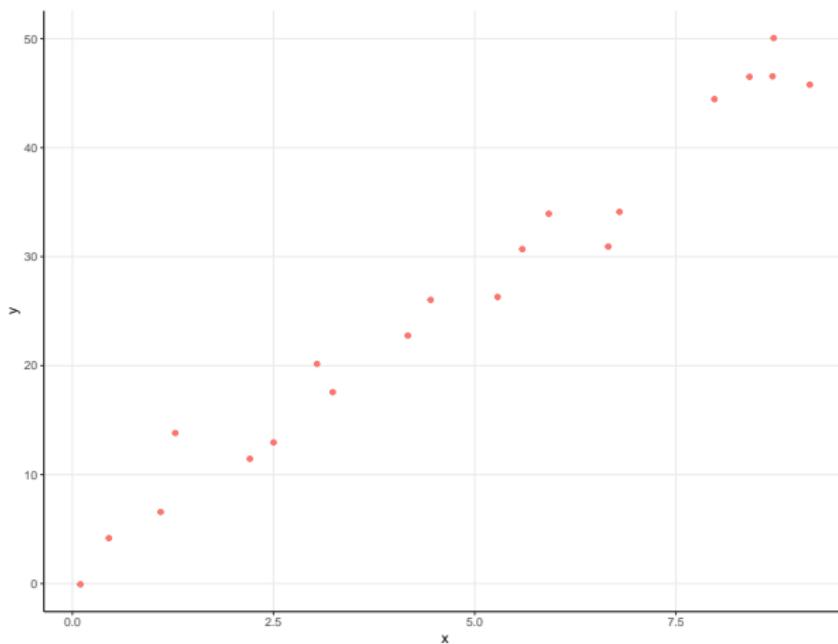
## 5 Conclusion

## Contexte

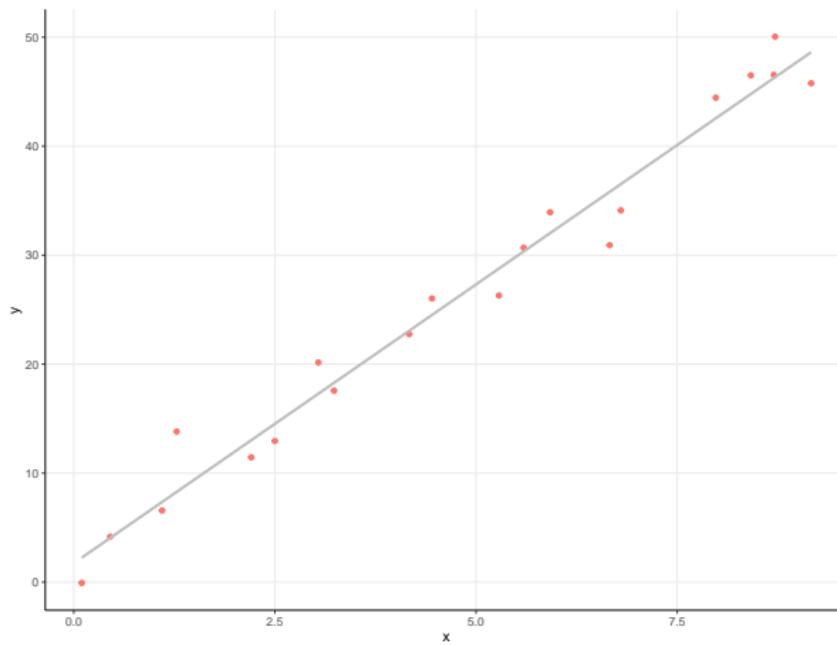
- Établissement : Université Laval
- Programme : Maîtrise en statistique avec mémoire
- Directeur : Thierry Duchesne, Professeur titulaire
- Co-directeur : François-Michel De Rainville, Ph.D.
- Conseiller en recherche : Samuel Perreault, Ph.D.

## Introduction

# Introduction



# Introduction



# Introduction

Call:

```
lm(formula = y ~ x, data = data1)
```

Residuals:

Min	1Q	Median	3Q	Max
-4.8911	-1.7580	-0.0998	1.7552	5.5365

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	1.7322	1.1205	1.546	0.14
x	5.1169	0.2003	25.551	1.35e-15 ***

---

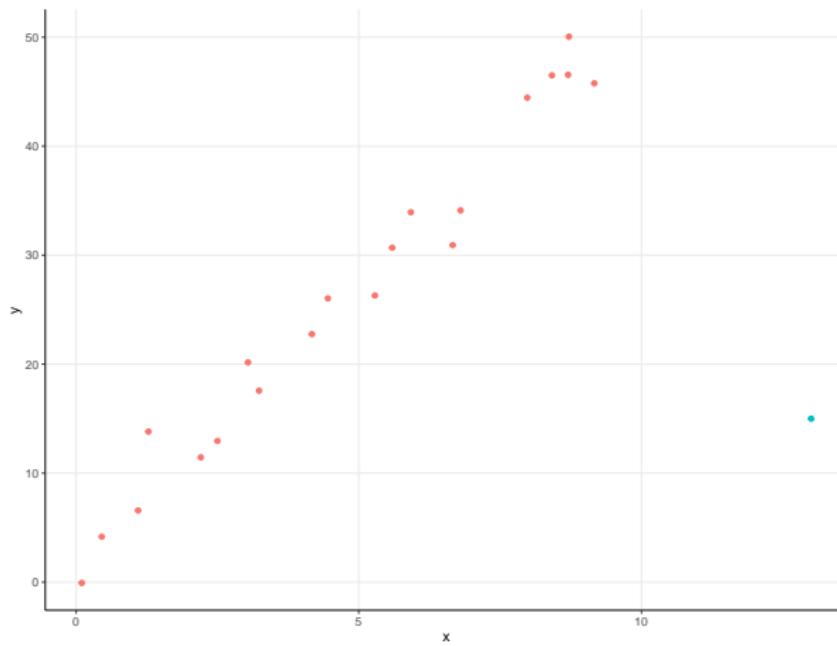
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.592 on 18 degrees of freedom

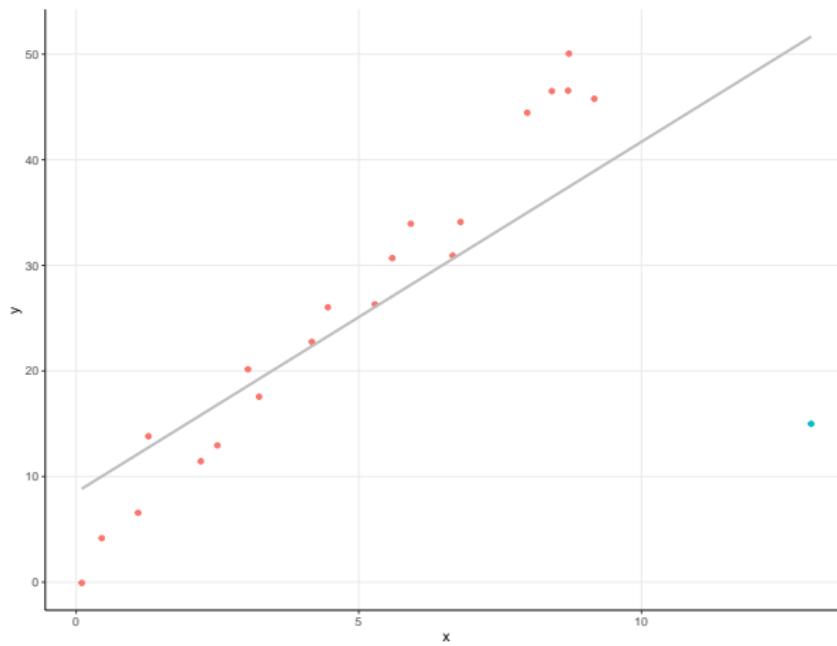
Multiple R-squared: 0.9732, Adjusted R-squared: 0.9717

F-statistic: 652.8 on 1 and 18 DF, p-value: 1.353e-15

# Introduction



# Introduction



# Introduction

Call:

```
lm(formula = y ~ x, data = data2)
```

Residuals:

	Min	1Q	Median	3Q	Max
-36.662	-3.851	1.063	5.779	12.617	

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	8.5046	4.2224	2.014	0.058374 .
x	3.3198	0.6862	4.838	0.000114 ***

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.45 on 19 degrees of freedom

Multiple R-squared: 0.5519, Adjusted R-squared: 0.5284

F-statistic: 23.41 on 1 and 19 DF, p-value: 0.0001143

## Mise en contexte

## L'importance du modèle

Le choix du modèle est crucial, car si celui-ci ne s'ajuste pas bien aux données, il pourrait nous induire en erreur sur "l'anormalité" d'une observation.

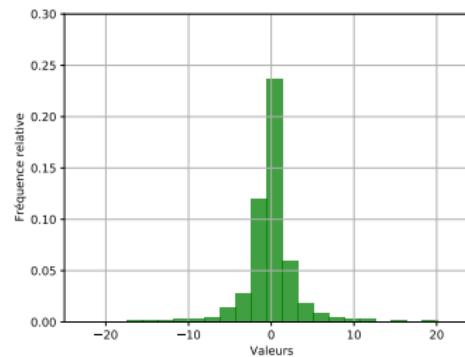
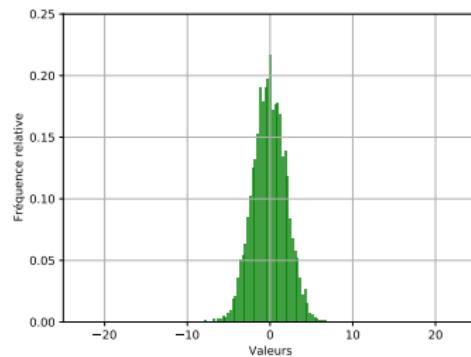


Figure – Distribution de 5000 simulations d'une loi normale (gauche) et d'une loi standard Cauchy (droite).

# Approches de détection d'anomalies

Il est possible définir 4 catégories d'approches (Aggarwal, 2016) :

- Analyse des valeurs extrêmes
- Les modèles probabilistes
- Les modèles linéaires et non-linéaires
- Les méthodes basées sur les distances

# Autoencodeurs

Un autoencodeur est un réseau de neurones qui a comme objectif d'apprendre une représentation intermédiaire d'une entrée de manière non-supervisée (Goodfellow et al., 2016).

# Autoencodeurs

- On apprend à encoder ( $q$ ) et décoder ( $p$ ) des observations  $x$  de dimensions  $D$ .
- Les représentations latentes  $z$ , de dimensions  $m$ , sont généralement de complexité moindre que l'entrée  $x$  ( $m \ll D$ ).
- Notations : l'encoder  $p$  est une fonction déterministe qui encode l'entrée  $x$  ( $q(x) = z$ ), le décodeur est une fonction déterministe qui permet de retrouver les dimensions initiales ( $p(z) = \hat{x}$ ).

# Autoencodeurs

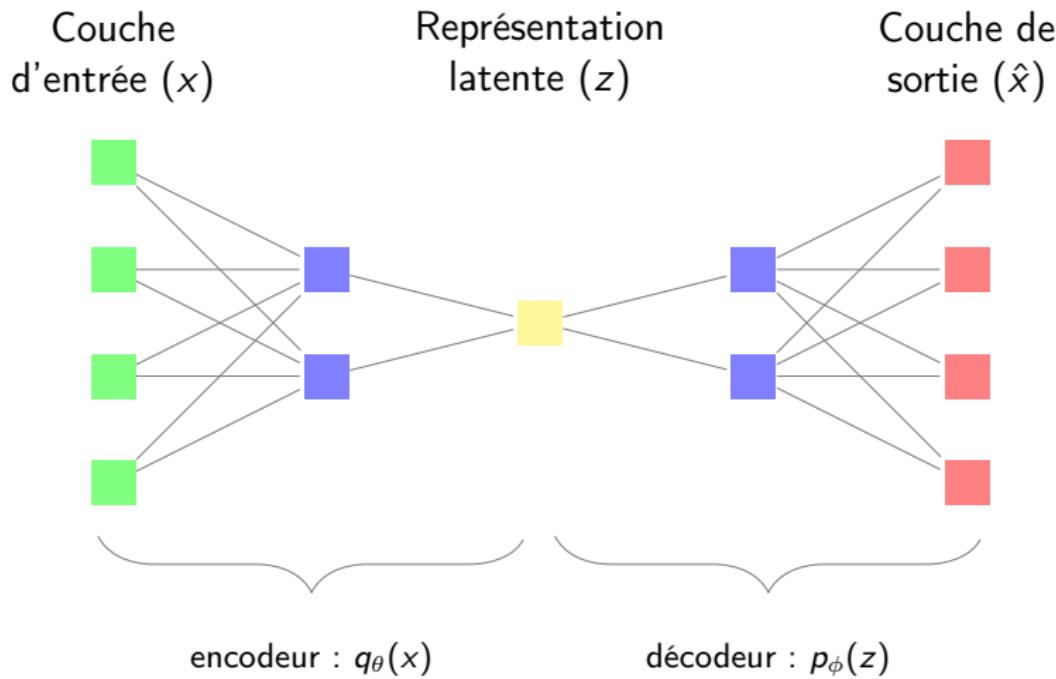


Figure – Exemple illustrant la structure de base d'un autoencodeur.

# Optimisation d'un autoencodeur

Les paramètres de l'autoencodeur sont généralement optimisés par descente du gradient sur la fonction de perte.

- $\hat{x} = p_\phi(q_\theta(x))$
- $L(x, \hat{x}) = L(x, p_\phi(q_\theta(x)))$
- $\Theta = \{\theta, \phi\}$
- $\Theta' \leftarrow \Theta - \epsilon * \frac{\partial L}{\partial \Theta}$ , où  $\epsilon$  est le taux d'apprentissage

# Autoencodeurs variationnels (VAE)

Les VAE (Kingma and Welling, 2013) sont similaires aux AE par rapport au concept d'encodage/décodage, mais ils ont une notion probabiliste additionnelle qui les différencient.

- On souhaite avoir un *a priori* sur  $z$
- $z$  est donc définie de manière continue et non discrète
- $p(z) \sim N(0, I)$
- $q_\theta(z|x) \sim N(\mu, \sigma)$
- Les paramètres  $\mu$  et  $\sigma$  sont définis par les couches du réseau précédents la représentation  $z$

# Autoencodeurs variationnels (VAE)

Couche  
d'entrée

Couche  
de sortie

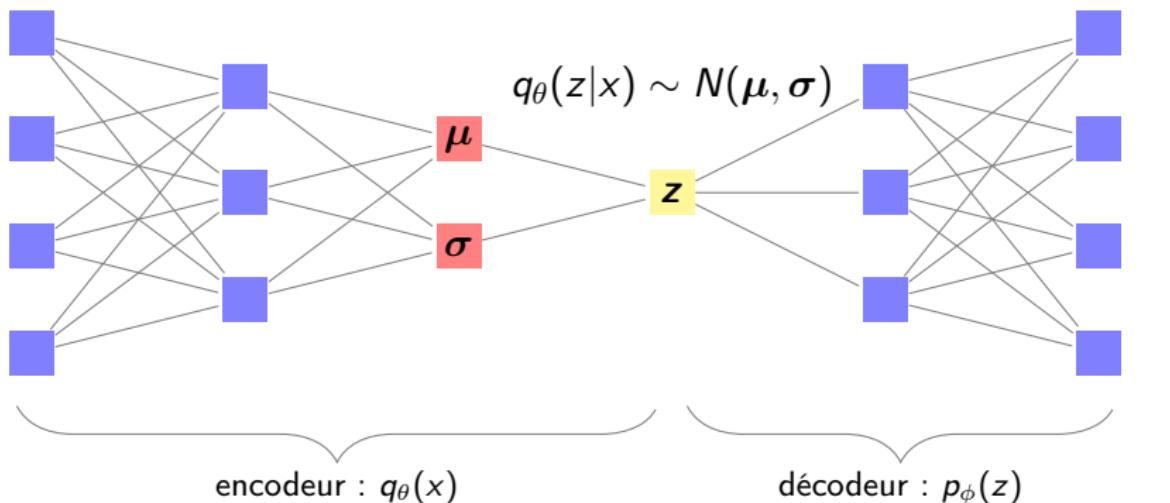


Figure – Structure de base d'un autoencodeur variationnel.

# L'optimisation du VAE

Pour s'assurer que  $z \sim N(0, I)$ , la fonction perte contient un terme additionnel.

- $L(x, \hat{x}) = L(x, p_\phi(q_\theta(x))) + D_{KL}[q_\theta(z|x)||p(z)]$
- $D_{KL}$  correspond au critère de divergence de Kullback-Leibler, qui quantifie la différence entre 2 distributions
- La perte est donc constituée de 2 composantes
  - critère de reconstruction :  $L(x, p_\phi(q_\theta(x)))$
  - critère sur la représentation latente :  $D_{KL}[q_\theta(z|x)||p(z)]$
- L'optimisation de  $\Theta = \{\theta, \phi\}$  se fait également par descente du gradient

Mise en contexte

Les autoencodeurs variationnels

## L'optimisation du VAE

Comment est-il possible de faire propager le gradient sachant que la représentation latente  $z$  est simulée d'une  $N(\mu, \sigma)$  ?

# L'optimisation du VAE

Comment est-il possible de faire propager le gradient sachant que la représentation latente  $z$  est simulée d'une  $N(\mu, \sigma)$  ?

- Réponse : *reparamétrisation trick*

# L'optimisation du VAE

Comment est-il possible de faire propager le gradient sachant que la représentation latente  $z$  est simulée d'une  $N(\mu, \sigma)$  ?

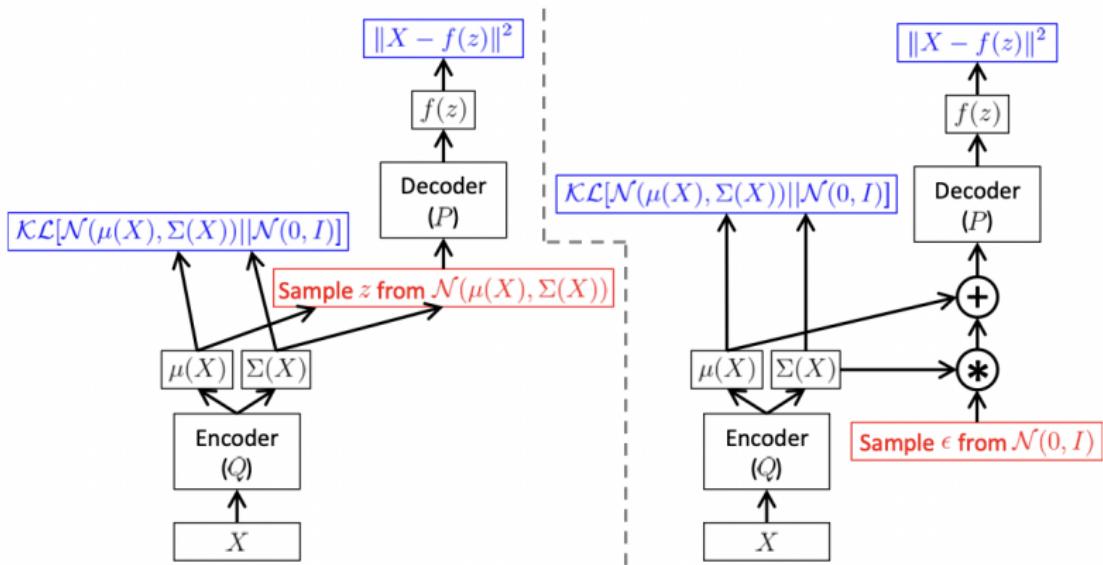
- Réponse : *reparamétrisation trick*
- L'objectif : on sépare les paramètres  $(\mu, \sigma)$  et sa composante stochastique  $\epsilon$

# L'optimisation du VAE

Comment est-il possible de faire propager le gradient sachant que la représentation latente  $z$  est simulée d'une  $N(\mu, \sigma)$  ?

- Réponse : *reparamétrisation trick*
- L'objectif : on sépare les paramètres  $(\mu, \sigma)$  et sa composante stochastique  $\epsilon$
- $z = \mu + \sigma * \epsilon$ , où  $\epsilon \sim N(0, I)$

## *Reparametrisation trick*



## Méthode proposée

## Les objectifs

- ➊ Faire la détection d'anomalies sur des données complexes, par exemple des images.
- ➋ Avoir un score d'anomalie simple à interpréter et à mettre en pratique, comme un seuil qui s'apparante à un niveau de confiance.

## Les hypothèses

- Ensemble d'entraînement  $\mathcal{X} = \{\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(n)}\}$ 
  - $n$  observations indépendantes de  $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}$
  - $n$  observations proviennent d'un mélange où  $(1 - p) \in \mathcal{N}$  et  $p \notin \mathcal{N}$
  - $\mathcal{N}$  : population "normale"
  - on suppose que  $p$  est faible (ex : < 5%)
  - on ne connaît pas la valeur exacte de  $p$
- Ensemble de test  $\mathcal{X}^* = \{\mathbf{X}^{*(1)}, \dots, \mathbf{X}^{*(k)}\}$ 
  - $k$  observations indépendantes de  $\mathbf{X}^* \in \mathbb{R}^{d_1 \times d_2}$
  - $k$  observations proviennent d'un mélange où  $(1 - p^*) \in \mathcal{N}$  et  $p^* \notin \mathcal{N}$
  - on suppose que  $p^*$  est faible et peut être similaire ou différent de  $p$
  - on ne connaît pas la valeur exacte de  $p^*$

# Description de l'approche

- ① Entraîner un autoencodeur variationnel.
- ② Définir un cadre décisionnel pour discriminer les observations "normales" des observations "anormales" à partir des représentations latentes et d'un seuil  $\alpha$  appelé, **niveau de filtration**.

# Entraînement du VAE

Comment orienter l'apprentissage du VAE vers le contenu ?



Images "normales"



Images "anormales"



Images "normales"

## Perceptual loss

Pour orienter la perte vers le contenu, on utilise le concept de *perceptual loss* (Johnson et al., 2016).

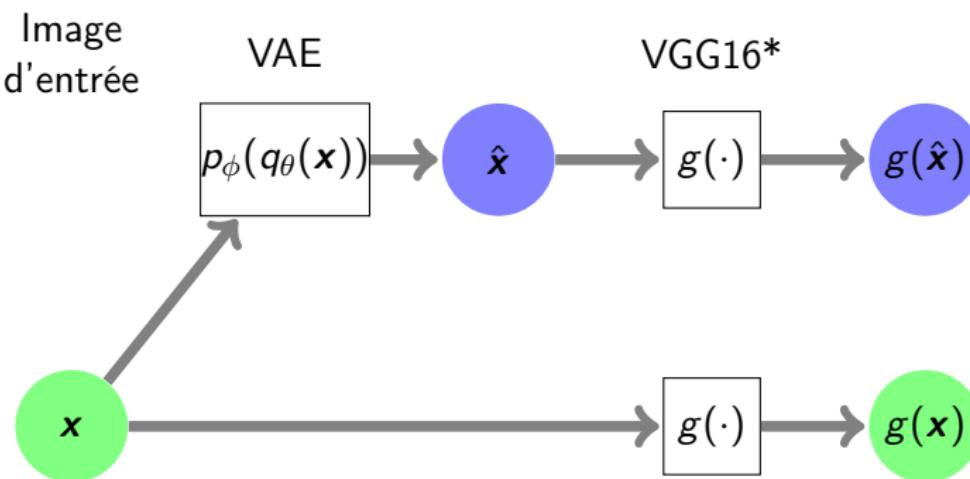


Figure – Mécanisme de la *perceptual loss*

## Définir un cadre décisionnel

On cherche à savoir si les représentations latentes apprises par le VAE nous permettent de discriminer les observations "normales" des observations "anormales".

## Définir un cadre décisionnel

Le cadre décisionnel se divise en 3 étapes :

- ① On encode les représentations latentes vers des statistiques de distance que l'on peut calculer avec une fonction  $T(\cdot)$ .
- ② On utilise ces statistiques de distance afin d'obtenir un score d'anomalie  $\gamma$ .
- ③ On compare  $\gamma$  avec un niveau de filtration  $\alpha$  pour savoir si l'observation semble "normale" ou "anormale".

## Définir la statistique de distance

Étant donné que nos représentations latentes sont des vecteurs  $(\mu, \sigma)$ , on utilise une distance de Kullback-Leibler pour définir nos statistiques.

- $T^{(i)} = D_{KL}[N(\mu^{(i)}, \sigma^{(i)}) || N(0, I)]$
- $T_{\mathcal{X}} = \{T^{(1)}, \dots, T^{(n)}\}$

2 scénarios sont possibles :

- ① **Près de la  $N(0, I)$**  : Les valeurs des statistiques de distance correspondant aux observations "normales" sont faibles
- ② **Éloigné de la  $N(0, I)$**  : Les valeurs des statistiques de distance correspondant aux observations "normales" sont élevées

# Définir la statistique de distance

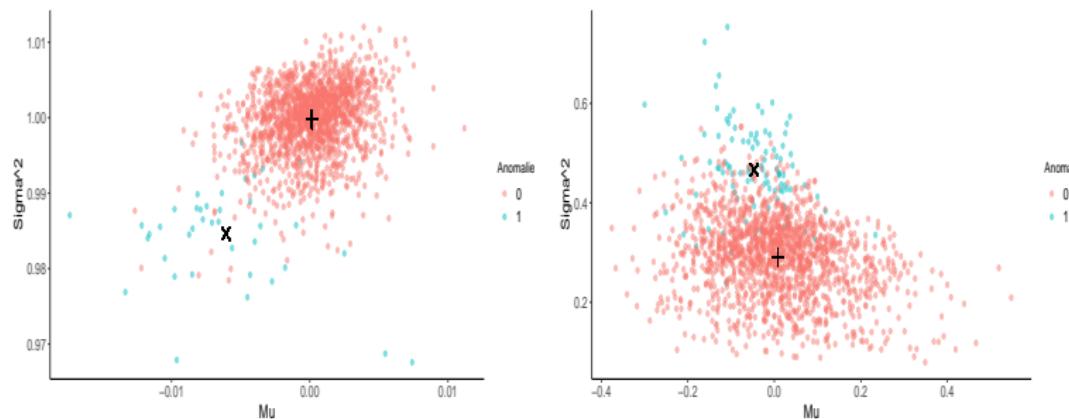


Figure – Scénario 1 (gauche) et scénario 2 (droite)

## Définir le score d'anomalie $\gamma$

Pour trouver le score d'anomalie  $\gamma$ , on ordonne les statistiques distance selon le scénario et on calcule le rang.

- $T'_{\mathcal{X}}$  : l'ensemble  $T_{\mathcal{X}}$  ordonné selon le scénario 1 ou 2
  - Scénario 1 :  $T_{\mathcal{X}}$  en ordre croissant
  - Scénario 2 :  $T_{\mathcal{X}}$  en ordre décroissant
- $\gamma(\mathbf{x}^{(j)}) = \frac{rang_{T'_{\mathcal{X}}}(T^{(j)})}{n}$
- $rang_{T'_{\mathcal{X}}}(T^{(j)})$  correspond au rang de la statistique de distance  $T^{(j)}$  dans l'ensemble ordonné  $T'_{\mathcal{X}}$

# Résumé de l'approche

## Algorithm 1: Algorithme de détection d'anomalies

**Input:** Ensemble de données d'entraînement  $x^{(i)}, i = 1, \dots, n$   
 Ensemble de données de test  $x^{*(j)}, j = 1, \dots, k$

**Output:** Scores d'anomalie  $\gamma^{(j)}, j = 1, \dots, k$

$\theta, \phi \leftarrow$  paramètres de l'encodeur et du décodeur du VAE entraîné;

**for**  $i=1$  to  $n$  **do**

$(\mu^{(i)}, \sigma^{(i)}) = q_\theta(x^{(i)})$

$T_{\mathcal{X}}^{(i)} = D_{KL}[N(\mu^{(i)}, \sigma^{(i)}) || N(0, I)]$

**end**

Ordonner  $T_{\mathcal{X}}$  selon le scénario identifié pour obtenir  $T'_{\mathcal{X}}$  ;

**for**  $j=1$  to  $k$  **do**

$(\mu^{(j)}, \sigma^{(j)}) = q_\theta(x^{*(j)})$

$T_{\mathcal{X}^*}^{(j)} = D_{KL}[N(\mu^{(j)}, \sigma^{(j)}) || N(0, I)]$

$\gamma(x^{*(j)}) = \text{rang}_{T'_{\mathcal{X}}} (T_{\mathcal{X}^*}^{(j)}) / n$

**end**

**return**  $\gamma^{(j)}, j = 1, \dots, k$

## Filtrer les anomalies

La dernière étape consiste à comparer ces scores avec notre niveau de filtration  $\alpha$ .

- $\gamma^{(j)} > (1 - \alpha)$  : observation  $\alpha$ -anormale
- $\gamma^{(j)} \leq (1 - \alpha)$  : observation  $\alpha$ -normale

## Expérimentations

Expérimentations

Jeux de données et méthodes testées

## 2 jeux de données

Voici les 2 jeux de données sur lesquels nous avons testé notre approche :

- MNIST
- ImageNet

## Expérimentations

Jeux de données et méthodes testées

## Exemples d'images - MNIST

Objectif : Trouver des anomalies dans des chiffres écrits à la main.

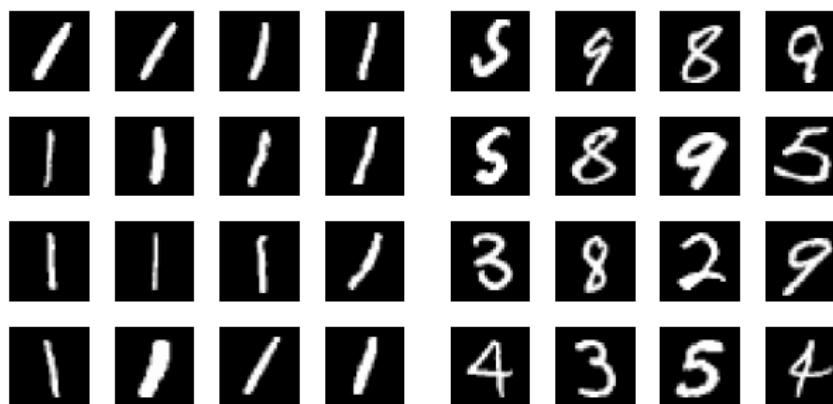


Figure – La catégorie "normale" (gauche) et la catégorie "anormale" (droite).

## Expérimentations

Jeux de données et méthodes testées

## Exemple d'images - ImageNet

Objectif : Trouver des anomalies dans un jeu de données d'images réelles plus complexes.

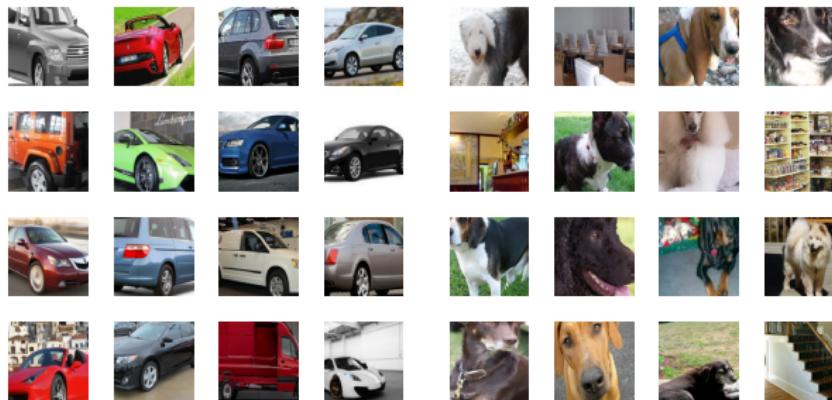


Figure – La catégorie "normale" (gauche) et la catégorie "anormale" (droite).

## Expérimentations

Jeux de données et méthodes testées

## Scénarios testés - MNIST

Scénario de test	Chiffre "normal"	Chiffres "anormaux"
1	1	5
2	1	5,9
3	1	0,2,3,4,5,6,7,8,9
4	6	8
5	6	3,8
6	6	0,1,2,3,4,5,7,8,9

## Expérimentations

Jeux de données et méthodes testées

## Différents niveaux de contamination - MNIST

Contamination	Ensemble de données	Nombre d'instances	Pourcentage d'anomalies
Moins	$\mathcal{X}$	4000	10%
	$\mathcal{X}^*$	800	1%
Égal	$\mathcal{X}$	4000	5%
	$\mathcal{X}^*$	800	5%
Plus	$\mathcal{X}$	4000	1%
	$\mathcal{X}^*$	800	10%

## Expérimentations

Jeux de données et méthodes testées

## Différents niveaux de contamination - ImageNet

Contamination	Ensemble de données	Nombre d'instances	Pourcentage d'anomalies
Moins	$\mathcal{X}$	10000	10%
	$\mathcal{X}^*$	1000	1%
Égal	$\mathcal{X}$	10000	5%
	$\mathcal{X}^*$	1000	5%
Plus	$\mathcal{X}$	10000	1%
	$\mathcal{X}^*$	1000	10%

## Méthodes testées

Nous avons testé différentes méthodes sur ces scénarios de test :

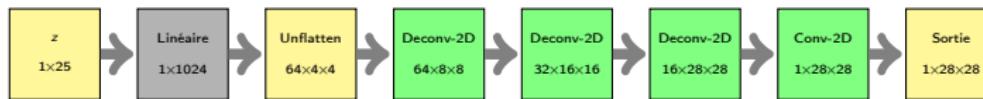
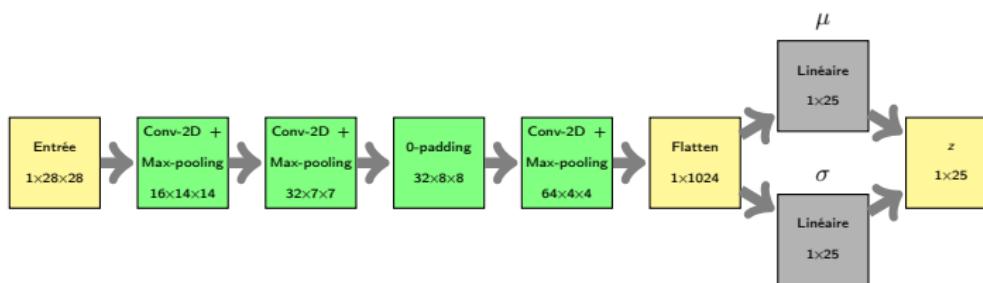
- PCA : analyse en composantes principales
- AE : autoencodeur traditionnel
- ISOF-VAE : Isolation Forest appliqué sur la représentation latente du VAE
- DA-VAE : méthode que nous proposons

Plus d'informations sur les autres méthodes, voir [l'appendice](#).

## Expérimentations

Jeux de données et méthodes testées

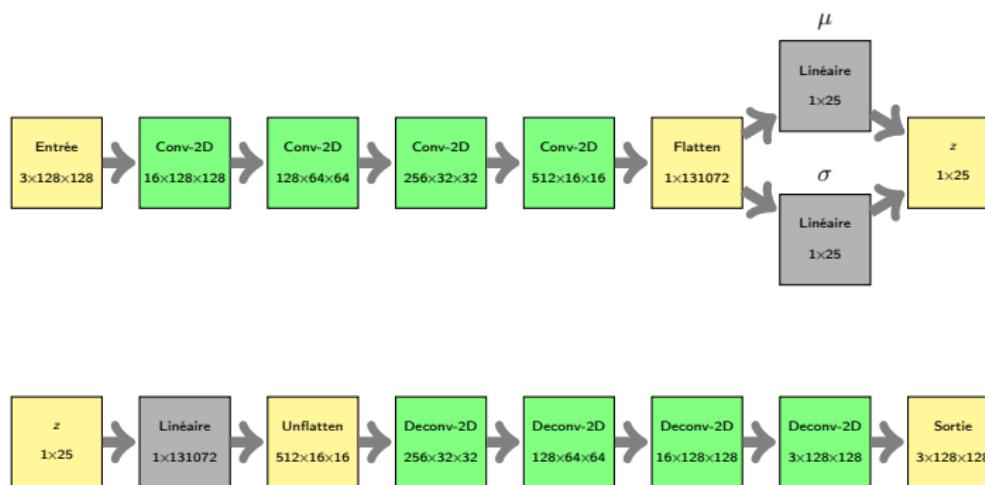
## Détails sur DA-VAE - MNIST



## Expérimentations

Jeux de données et méthodes testées

## Détails sur DA-VAE - ImageNet

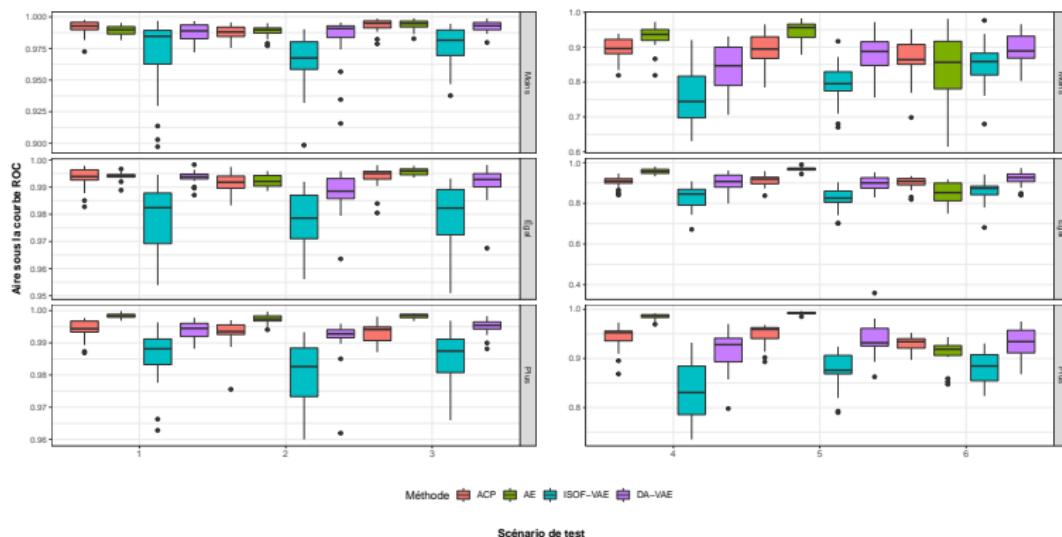


## Expérimentations

## Résultats

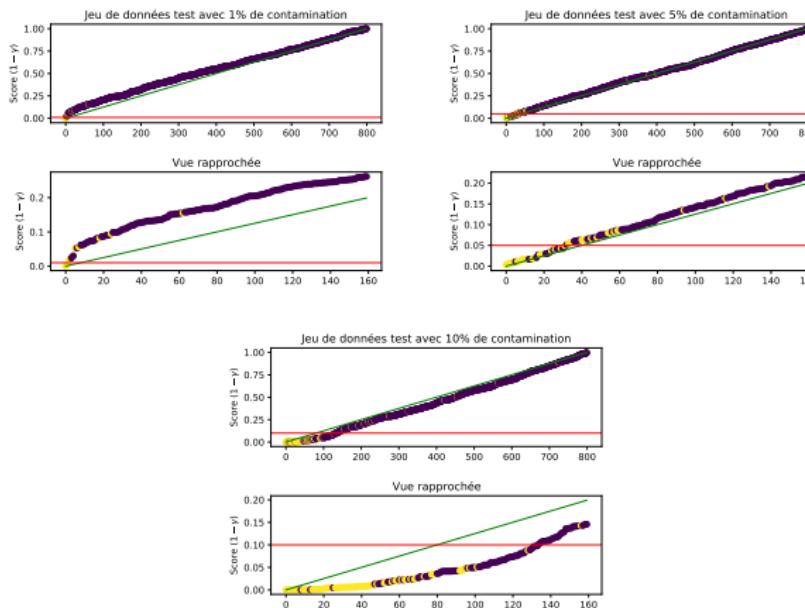
## Résultats - MNIST

Résultats en aire sous la courbe ROC (AUC).



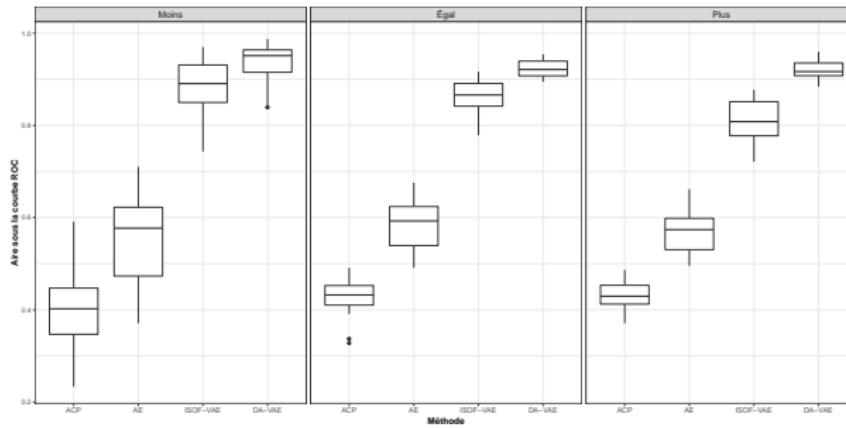
# Résultats - MNIST

Résultats sur les anomalies (jaunes) et les observations "normales" (mauvres) du jeu de données  $\mathcal{X}^*$ .



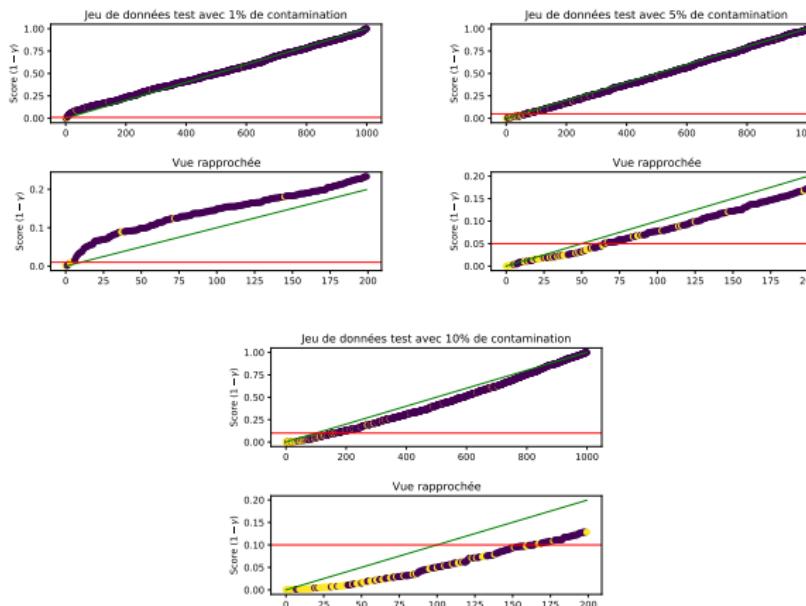
# Résultats - ImageNet

Résultats en aire sous la courbe ROC (AUC).



# Résultats - ImageNet

Résultats sur les anomalies (jaunes) et les observations "normales" (mauvres) du jeu de données  $\mathcal{X}^*$ .



# Analyse des résultats

Si on regarde quelques résultats des méthodes basées sur la reconstruction, on remarque que le contenu de l'image n'est pas le facteur déterminant dans la détection d'anomalie.

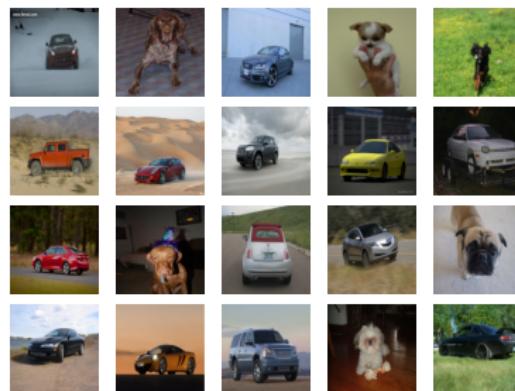


Figure – Bonnes reconstructions

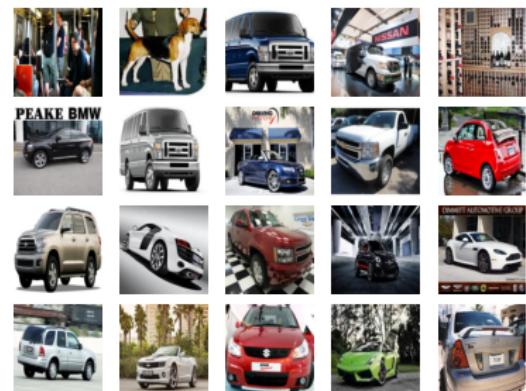
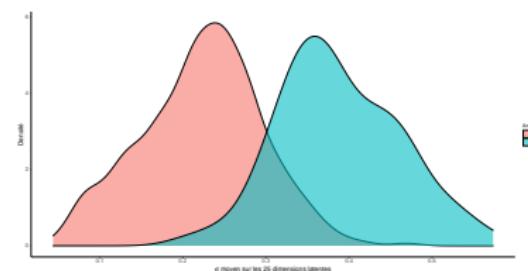
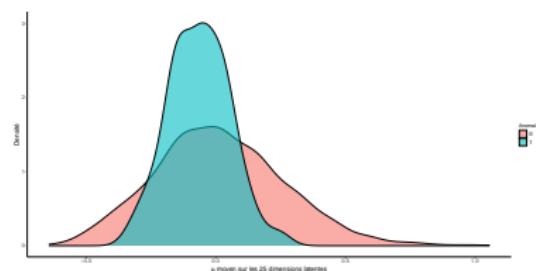


Figure – Mauvaises reconstructions

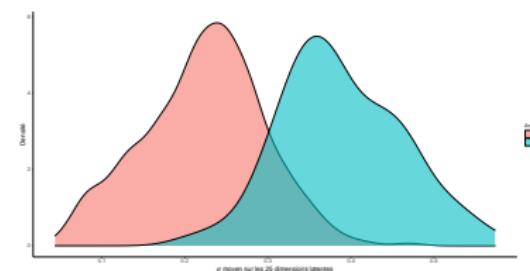
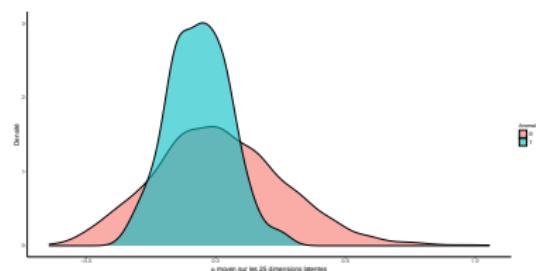
# Analyse des résultats

Si on regarde notre modèle, la représentation latente semble bien différente entre les anomalies et les observations "normales".



# Analyse des résultats

Si on regarde notre modèle, la représentation latente semble bien différente entre les anomalies et les observations "normales".



## Conclusion

## Conclusion

# Exemple d'application

On pourrait imaginer un exemple d'application pour nettoyer un jeu de données qui peut contenir des images abérantes.

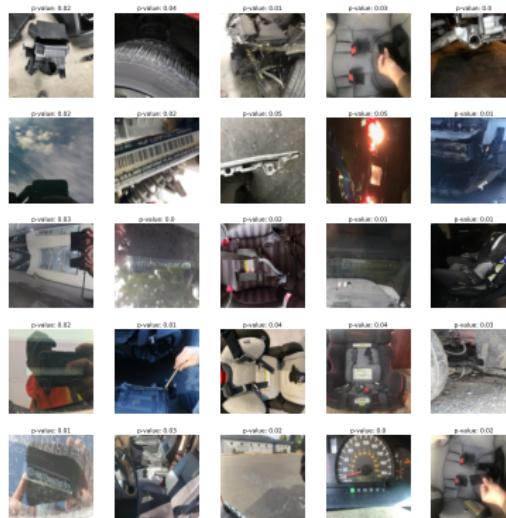


Figure – Exemples avec  $\gamma > (1 - \alpha)$

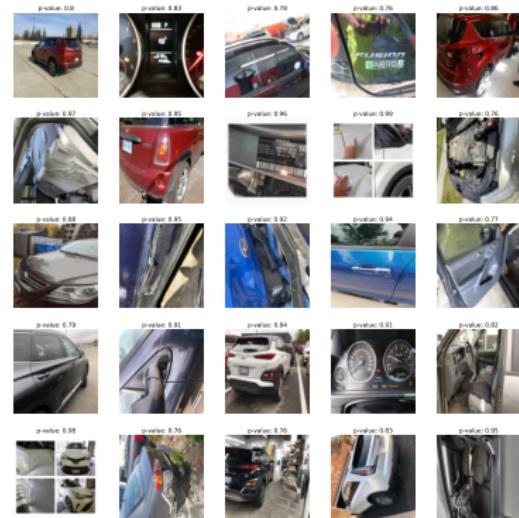


Figure – Exemples avec  $\gamma \leq (1 - \alpha)$

## Points à retenir

- Pour des images complexes, les autoencodeurs permettent de bien capturer l'information des données et sont utiles dans un contexte de détection d'anomalies.
- L'utilisation des représentations latentes permet de mieux modéliser le contenu d'une image, contrairement à la reconstruction qui peut être davantage affectée par la complexité variante des images dans un jeu de données.
- Pour obtenir une représentation latente qui répond à certains objectifs, les  $\beta$ -VAE sont des modèles difficiles à entraîner et à ajuster (beaucoup d'essais-erreurs).

## Aspects à explorer

- Voir si d'autres lois de probabilité, autre que la  $N(0, I)$ , pourraient être utilisées comme loi *a priori* pour régulariser la représentation latente.
- Mieux comprendre les comportements possibles de la représentation latente (les 2 scénarios expliqués dans la méthodologie).

## Bibliographie

- Aggarwal, C. C. (2016). *Outlier Analysis*. Springer Publishing Company, Incorporated. 2nd Edition. New York.
- Chen, H., Ma, H., Chu, X., and Xue, D. (2020). Anomaly detection and critical attributes identification for products with multiple operating conditions based on isolation forest. *Advanced Engineering Informatics*, 46 :101139.
- Doersch, C. (2016). Tutorial on variational autoencoders. cite arxiv :1606.05908.
- Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. <http://www.deeplearningbook.org>.
- Johnson, J., Alahi, A., and Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer.
- Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. arxiv :1312.6114.

## Appendice

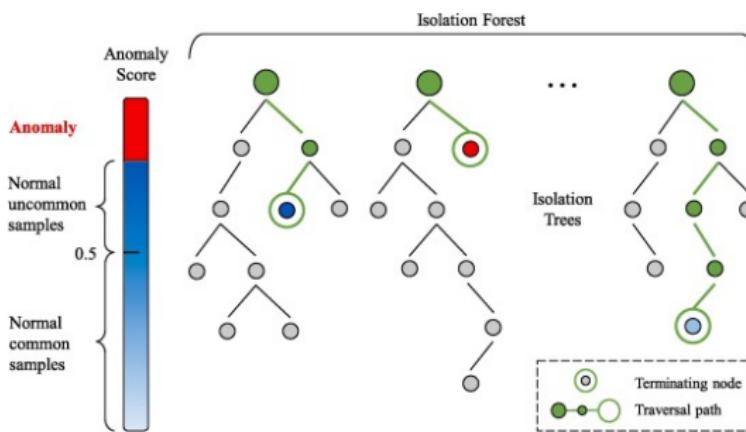
## Score d'anomalie basé sur la reconstruction

Pour les méthodes PCA et AE, nous avons appliqué une approche de détection d'anomalie basée sur la capacité de reconstruire l'entrée  $x$  avec un niveau de filtration  $\alpha$  :

$$I(x^{*(j)}, \hat{x}^{*(j)}) > t_{1-\alpha}(L(\mathcal{X}, \hat{\mathcal{X}}))$$

On suppose toujours que  $\alpha$  est égale au niveau de contamination dans  $\mathcal{X}^*$ , soit  $p^*$ .

## Détails sur ISOF-VAE



. Image tirée de Chen et al. (2020)

## Détails sur ISOF-VAE

Pour la méthode basée sur les Isolation Forest, nous avons utilisé le score d'anomalie calculé par l'algorithme :

- $S$  : score d'anomalie calculé par l'algorithme
- $S_{\mathcal{X}} = \{S^{(1)}, \dots, S^{(n)}\}$
- $S'_{\mathcal{X}}$  :  $S_{\mathcal{X}}$  ordonné en ordre croissant
- $S^*(x^{*(j)}) = \frac{\text{rang}_{S'_{\mathcal{X}}}(S^{(j)})}{n}$

On suppose toujours que  $\alpha$  est égale au niveau de contamination dans  $\mathcal{X}^*$ , soit  $p^*$ .