



# Anomaly Detection

## Sber, April 26, 2021

Alexey Zaytsev

Assistant professor, Lab Head  
Skolkovo Institute of Science and Technology



# Laboratory of applied research LARSS Sberbank-Skoltech

Founded in 2019

- A group of employees, master students, and bachelor students
- 3 Q1 paper in 2019-2020
- Joint industrial projects with major industrial companies
- Educational program
- Work with students
- Science projects in various areas

Head of the lab:

Alexey Zaytsev



# Lecture plan

- Intro to Anomaly detection
- Unsupervised approaches for Anomaly detection. General idea
- Autoencoders for Anomaly detection
- GAN-based Anomaly detection
- Anomaly detection for Time Series









# Intro to Anomaly Detection

# Problem statement

The problem is to find objects that anomalous given training data

Normal data

Image Type	No Cat	Cat not on approach	Cat on approach	Cat with prey
Count of Images	6,542	9,504	6,689	260
Example				

Anomaly data







<https://www.theverge.com/tldr/2019/6/30/19102430/amazon-engineer-ai-powered-catflap-prey-ben-hamm>



# Problem statement

The problem is to find objects that anomalous given training data

Normal data

Image Type	No Cat	Cat not on approach	Cat on approach	Cat with prey
Count of Images	6,542	9,504	6,689	260
Example				

Anomaly data



<https://www.theverge.com/tldr/2019/6/30/19102430/amazon-engineer-ai-powered-catflap-prey-ben-hamm>

# Problem examples

- Fraud detection 🕵️
- Failure detection for an airplane ✈️
- Intrusion detection 😱
- Earthquake prediction 💥

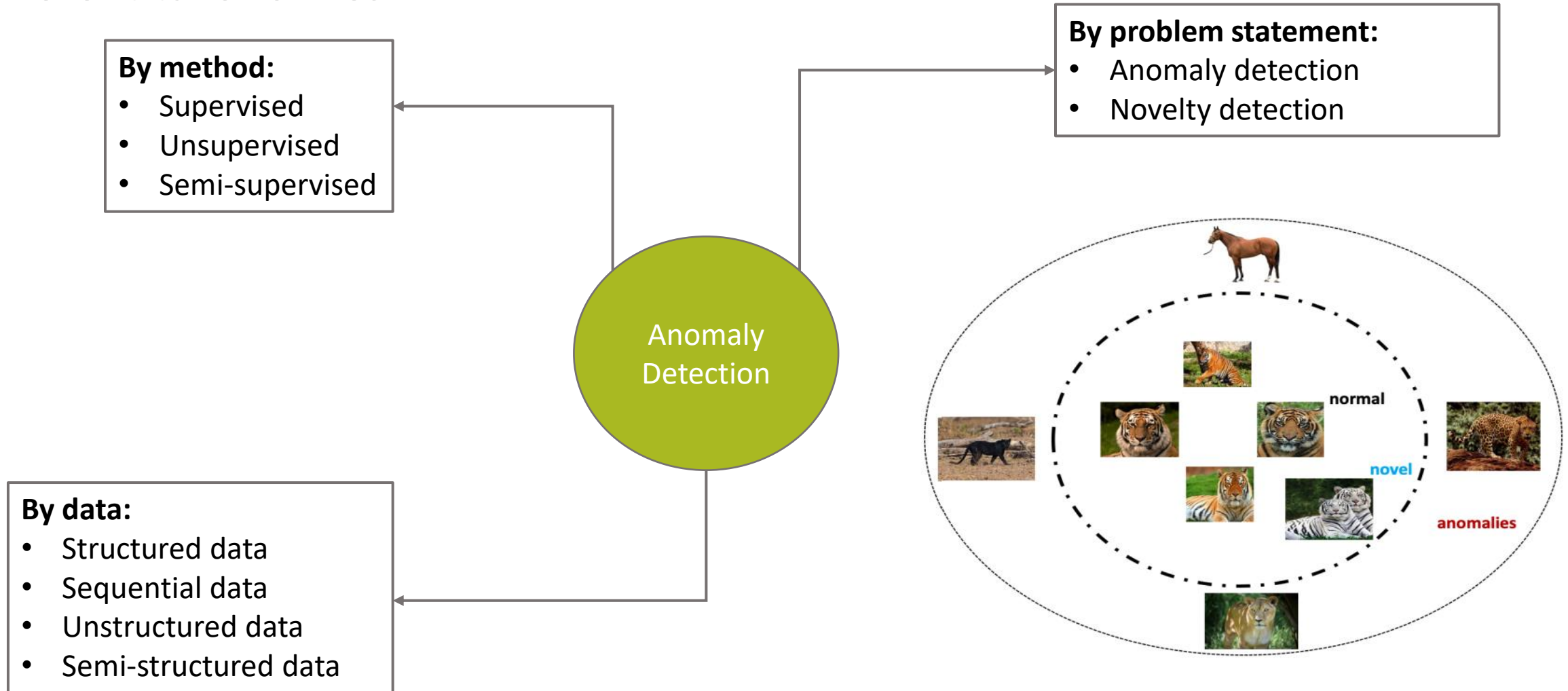


## Typical challenges:

- Requires problem-specific knowledge => new problem – new approach
- Hard to identify something we don't see
- Bunch of various problem statements => how to define what is anomaly?



# Different taxonomies

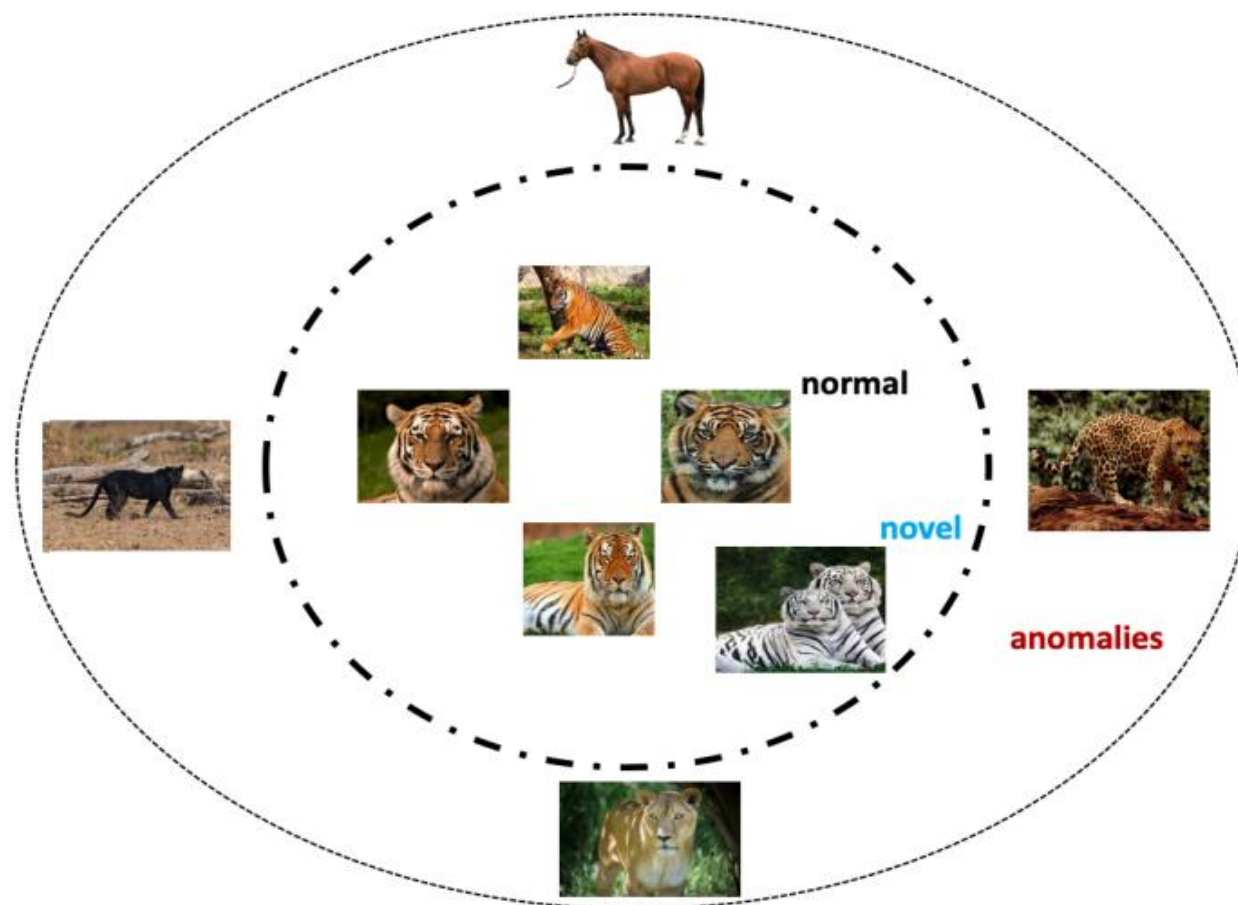


<https://arxiv.org/pdf/1901.03407.pdf>



# Taxonomy with respect to problem statement





- Novelty detection
- Anomaly detection



<https://arxiv.org/pdf/1901.03407.pdf>

# Anomaly type taxonomy

Normal data

Image Type	No Cat	Cat not on approach	Cat on approach	Cat with prey
Count of Images	6,542	9,504	6,689	260
Example				

Point Anomaly



Group Anomaly



Contextual Anomaly



In 2019




He is mad! We should avoid him. (anomaly)

In 2020



He takes care of himself and others. Well done! (normal)



# Approaches to Anomaly Detection

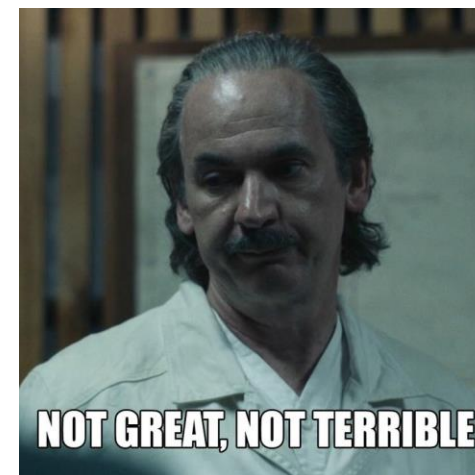
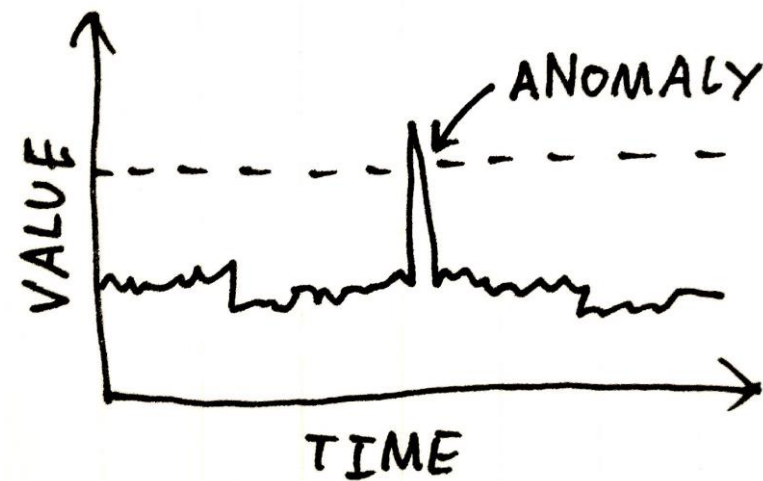
# Classic approach to anomaly detection

- Construct anomaly score  $s(x)$  using data
- Signal about anomaly if anomaly score is greater than some threshold  $\tau$

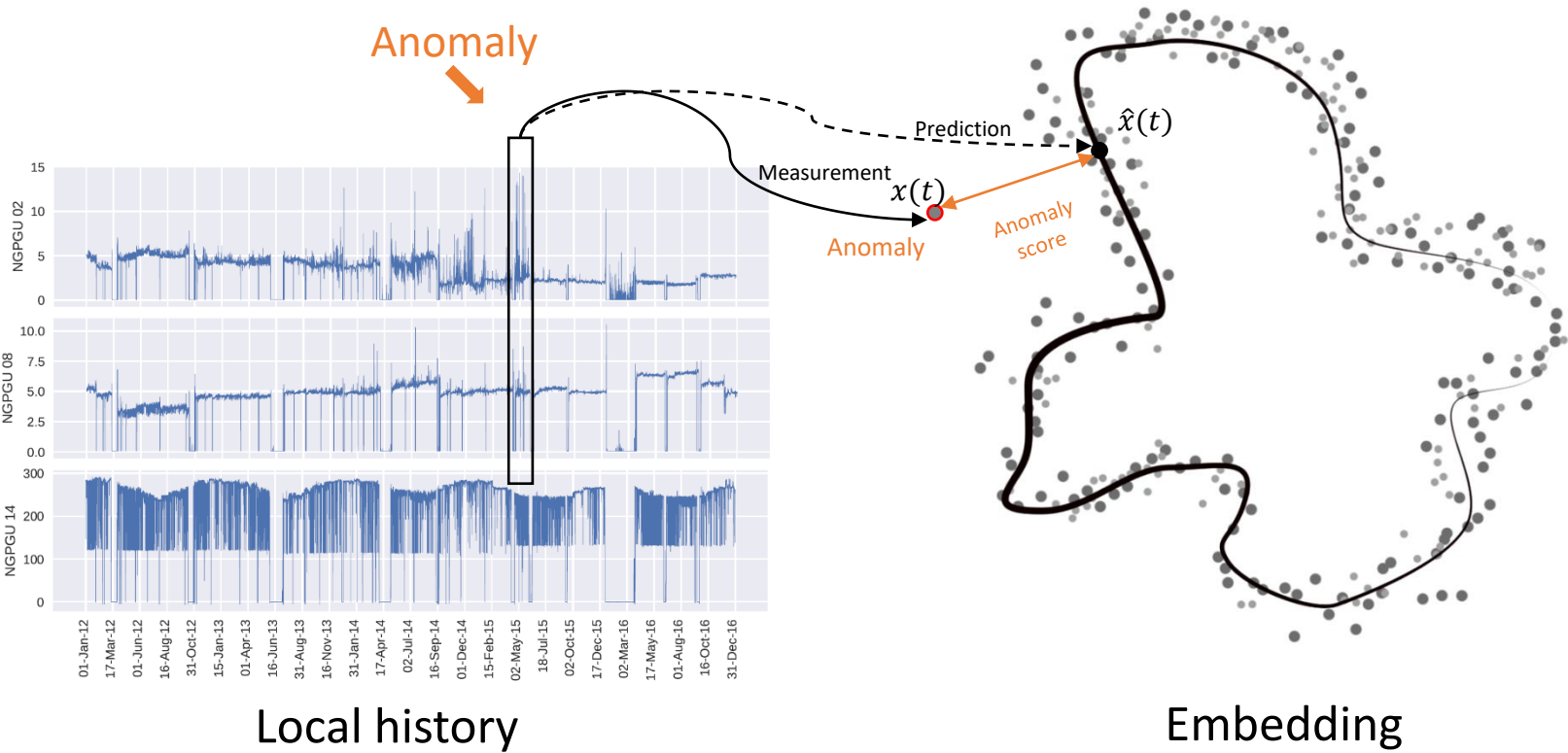
## Problems:

- In terms of sequential data, this approach mostly work well only with numeric time series
- Usually require the data to have “good” properties

Threshold selection  $\tau$  is a separate problem, as we often have only positive examples



# Unsupervised anomaly detection. General approach





## Classic approach revisited

- A sample  $D = \{\mathbf{x}_i\}_{i=1}^n$  is given, each  $\mathbf{x} \in \mathbb{R}^d$ .
- Construct models

$$\hat{x}_1 = f_1(x_2, x_3, \dots, x_d),$$

...

$$\hat{x}_d = f_d(x_1, x_2, \dots, x_{d-1}).$$

- We have  $d$  anomaly scores for  $\mathbf{x} = \{x_1, x_2, \dots, x_d\}$ :

$$s_i(\mathbf{x}) = |\hat{x}_i - x_i|, i = \overline{1, d}.$$

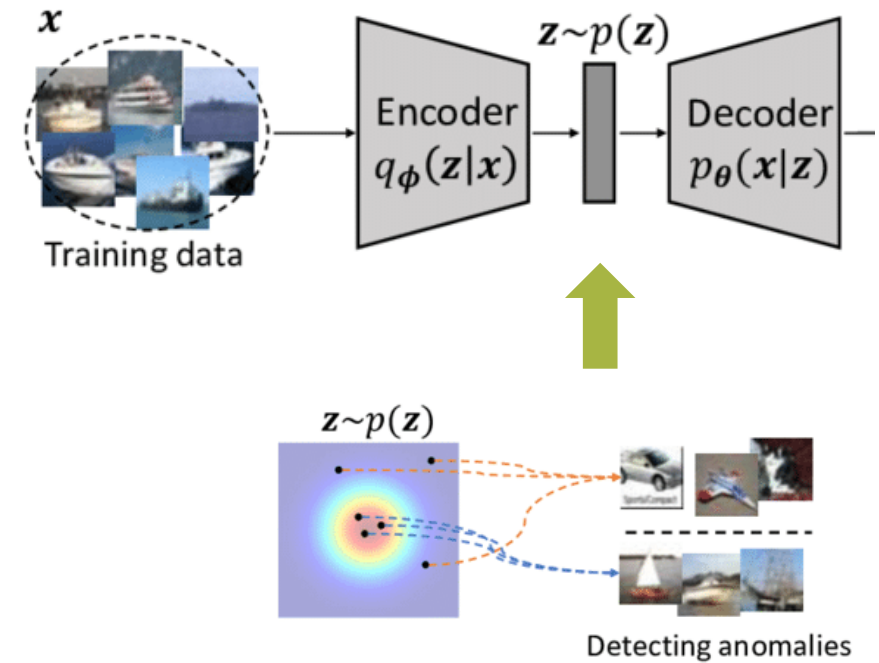
# Unsupervised anomaly detection. General approach

- A sample  $D = \{\mathbf{x}_i\}_{i=1}^n$  is given, each  $\mathbf{x} \in \mathbb{R}^d$ .
- Construct encoder and decoder model

$$\mathbf{z}_i = e(\mathbf{x}_i),$$
$$\mathbf{x}_i \approx \hat{\mathbf{x}}_i = d(\mathbf{z}_i) = d(e(\mathbf{x}_i)).$$

- We have an anomaly score  $s(\mathbf{x})$  for any  $\mathbf{x}$ :

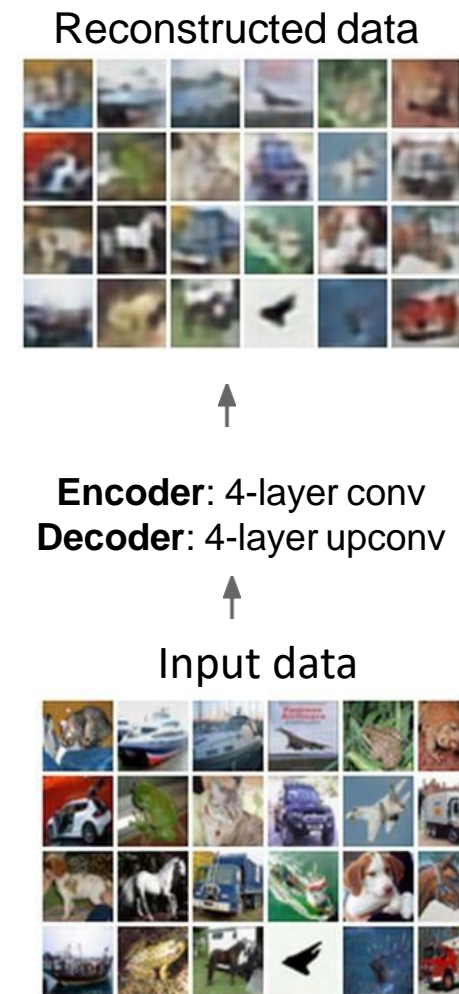
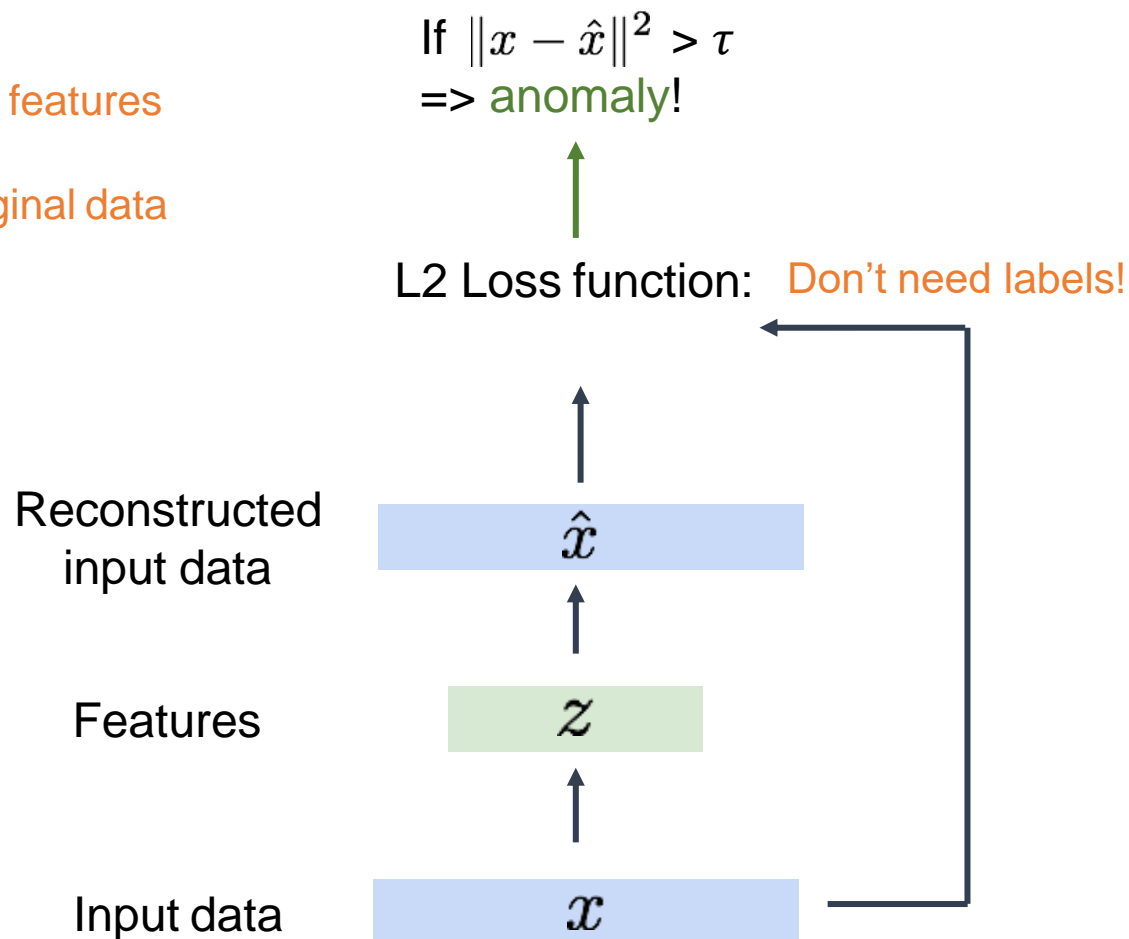
$$s(\mathbf{x}) = \|\hat{\mathbf{x}}_i - \mathbf{x}_i\|.$$



Encoder, decoder examples: PCA, Autoencoder

# Autoencoder. General idea

Train such that features  
can be used to  
reconstruct original data



Slides were adapted from lecture by Fei-Fei Li & Justin Johnson & Serena Yeung

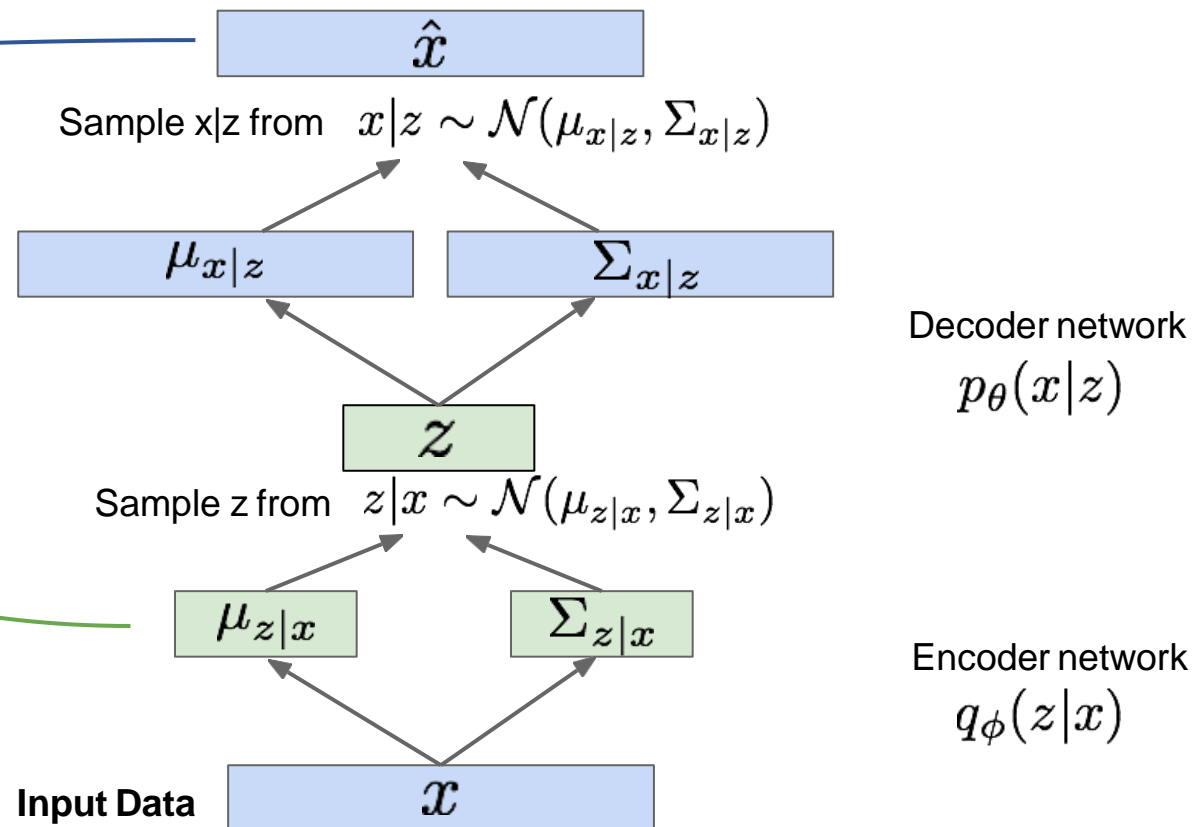
# Variational Autoencoder

We maximize the likelihood lower bound

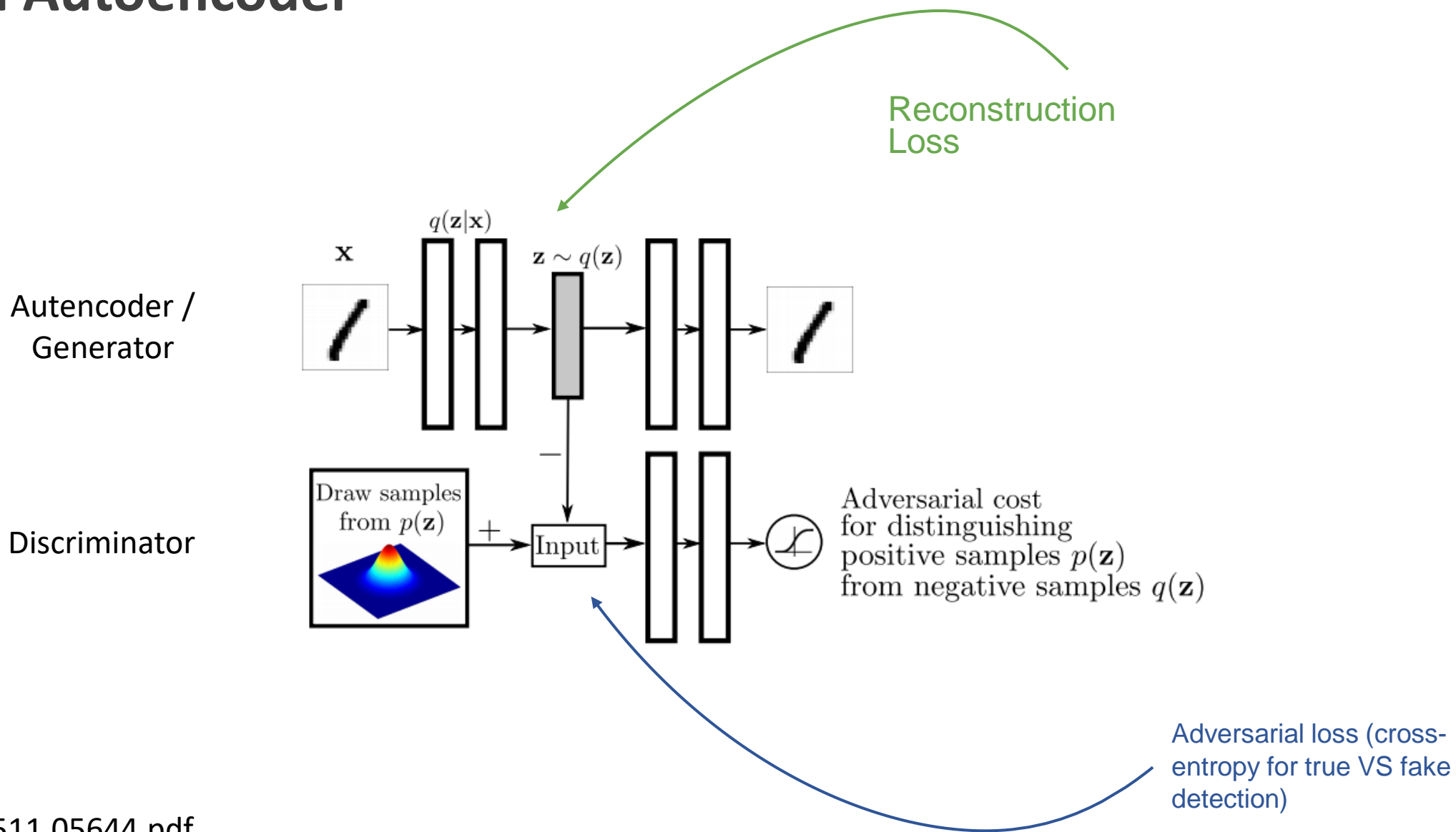
Maximize likelihood of original input being reconstructed

$$\underbrace{\mathbb{E}_z \left[ \log p_\theta(x^{(i)} | z) \right] - D_{KL}(q_\phi(z | x^{(i)}) || p_\theta(z))}_{\mathcal{L}(x^{(i)}, \theta, \phi)}$$

Make approximate posterior distribution close to prior



# Adversarial Autoencoder



<https://arxiv.org/pdf/1511.05644.pdf>





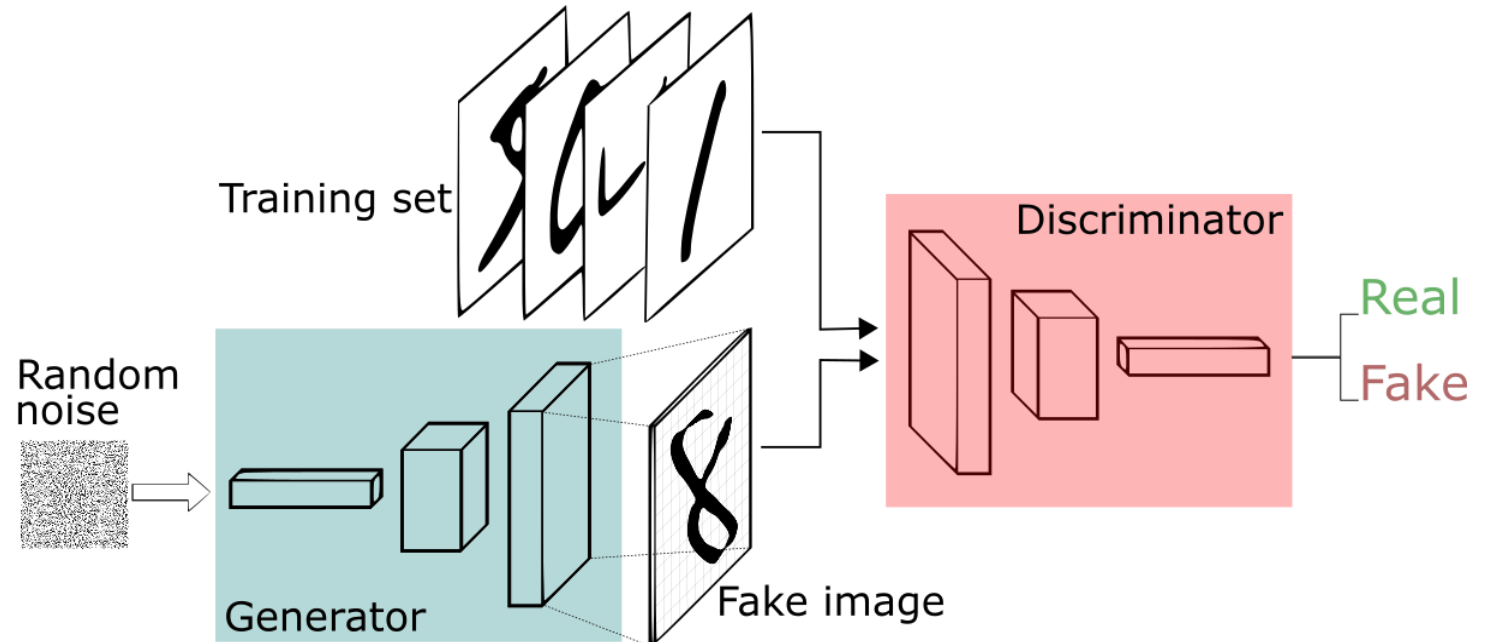
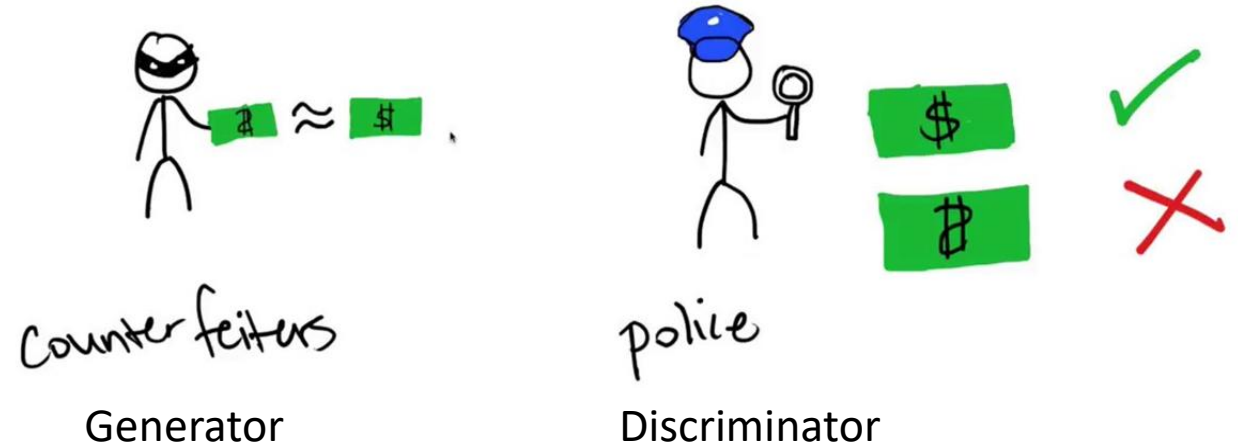
# GANs for Anomaly Detection

# Generative Adversarial Network

**Generator network:** try to fool the discriminator by generating real-looking images

**Discriminator network:** try to distinguish between real and fake images

Train jointly in **minimax game!**



# Generative Adversarial Network: formulas

Minimax objective function:

$$\min_{\theta_g} \max_{\theta_d} \left[ \mathbb{E}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

Discriminator outputs likelihood in (0,1) of real image

Discriminator output for real data x      Discriminator output for generated fake data G(z)

- *Discriminator* with parameters  $\theta_d$  wants to **maximize objective** such that  $D(x)$  is close to 1 (real) and  $D(G(z))$  is close to 0 (fake)
- *Generator* with parameters  $\theta_g$  wants to **minimize objective** such that  $D(G(z))$  is close to 1: the discriminator is fooled into thinking generated  $G(z)$  is real

# GAN: example for novelty detection

The model is trained using images of penguins



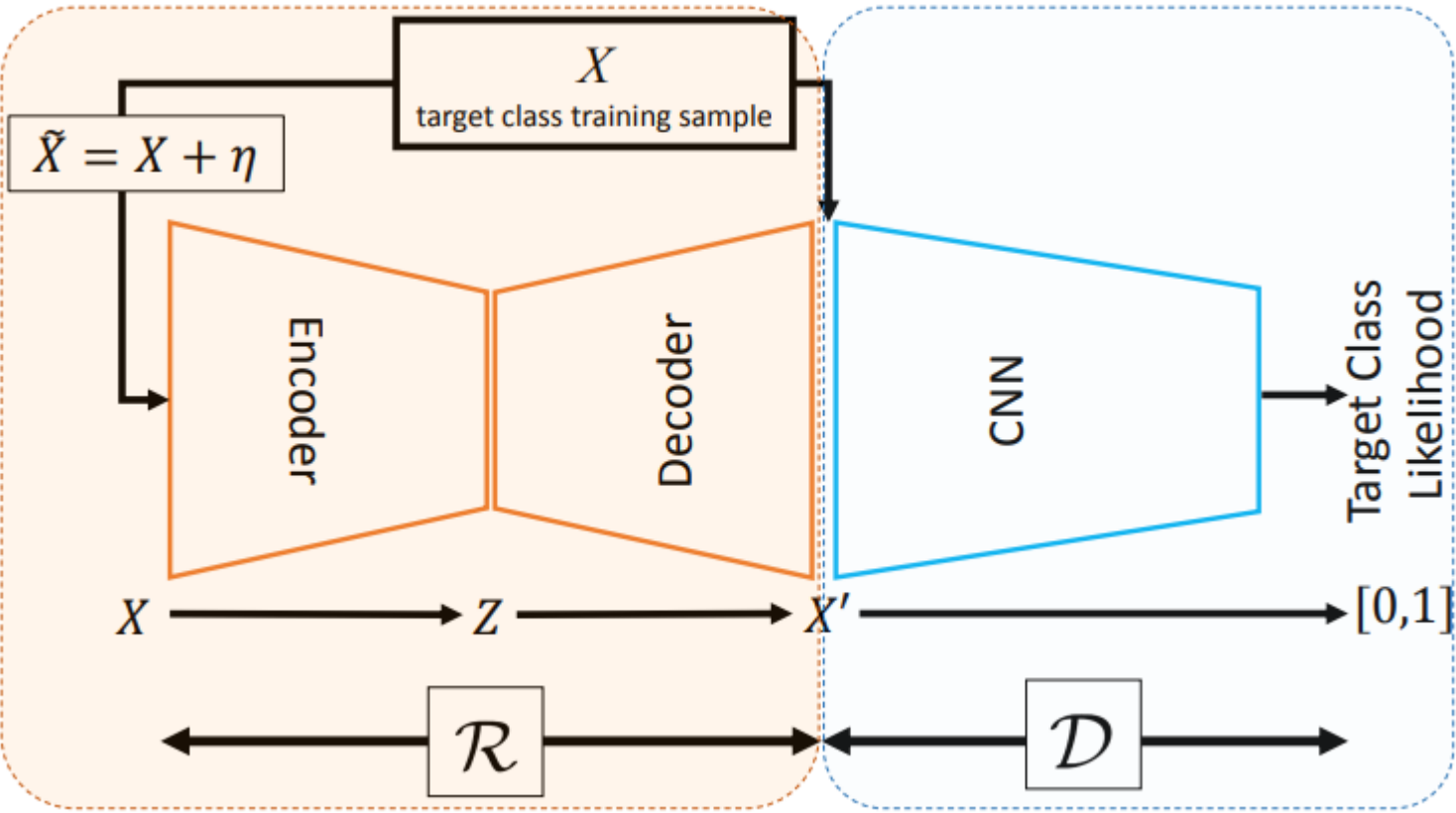
- If we use noisy inliers and pass them to  $\mathcal{R}$  NN, we get enhanced images as the output
- If we use outlier sample instead, the output of  $\mathcal{R}$  is distorted

	Noisy Inlier Samples		Outlier Samples	
$X$				
$\mathcal{R}(X)$				
$\mathcal{D}(X)$	0.75	0.72	0.53	0.27
$\mathcal{D}(\mathcal{R}(X))$	<b>0.85</b>	<b>0.91</b>	0.25	0.10

M. Sabokrou et al. *Adversarially Learned One-Class Classifier for Novelty Detection*, CVPR, 2018

# GAN: example for novelty detection

Architecture



Autoencoder generator

Discriminator gives  
anomaly score

$$\mathcal{L} = \mathcal{L}_{\mathcal{R}+\mathcal{D}} + \lambda \mathcal{L}_{\mathcal{R}}$$

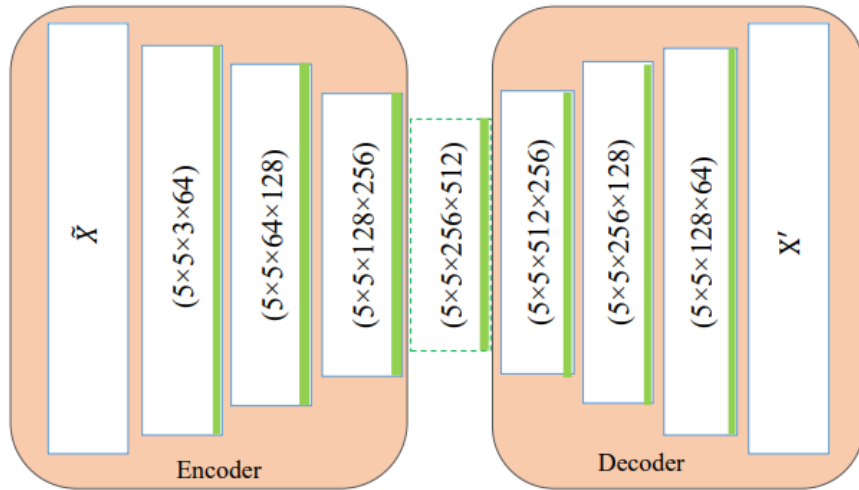
GAN loss

$$\mathcal{L}_{\mathcal{R}} = \|X - X'\|^2$$

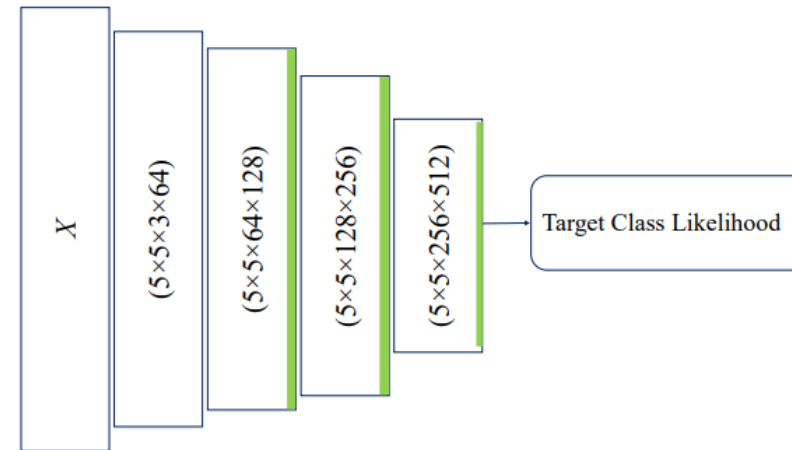
<https://arxiv.org/abs/1802.09088>



# Internal architectures



Autoencoder generator



Discriminator gives  
anomaly score

<https://arxiv.org/abs/1802.09088>

## GAN: example for novelty detection










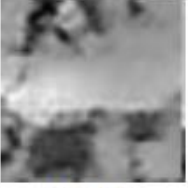
Anomaly score with state-of-the-art performance:

$$\text{OCC}_1(X) = \begin{cases} \text{Target Class} & \text{if } \mathcal{D}(X) > \tau, \\ \text{Novelty (Outlier)} & \text{otherwise,} \end{cases} \quad \text{PCA, VAE, AAE}$$

Anomaly score that utilizes encoder-decoder

$$\text{OCC}_2(X) = \begin{cases} \text{Target Class} & \text{if } \mathcal{D}(\mathcal{R}(X)) > \tau, \\ \text{Novelty (Outlier)} & \text{otherwise.} \end{cases}$$

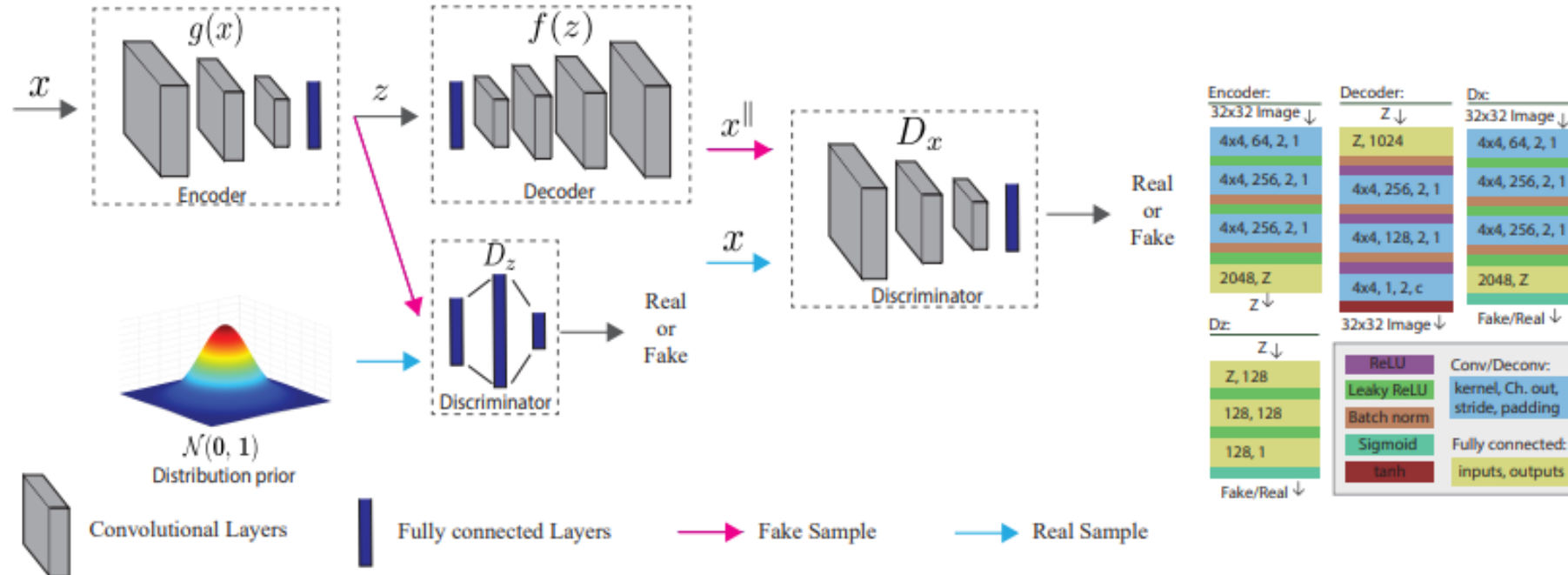
# Model quality

	Normal Patches			Anomaly Patches	
$X$					
$\mathcal{R}(X)$					
$\mathcal{D}(X)$	0.15	0.19	0.32	0.35	0.44
$\mathcal{D}(\mathcal{R}(X))$	<b>0.44</b>	<b>0.64</b>	<b>0.56</b>	0.20	0.30

	CoP [32]	REAPER [22]	OutlierPursuit [50]	LRR [24]	DPCP [45]	R-graph [52]	Ours $\mathcal{D}(X)$	Ours $\mathcal{D}(\mathcal{R}(X))$
AUC	0.905	0.816	0.837	0.907	0.783	<b>0.948</b>	0.932	0.942
$F_1$	0.880	0.808	0.823	0.893	0.785	0.914	0.916	<b>0.928</b>
AUC	0.676	0.796	0.788	0.479	0.798	0.929	0.930	<b>0.938</b>
$F_1$	0.718	0.784	0.779	0.671	0.777	0.880	0.902	<b>0.913</b>
AUC	0.487	0.657	0.629	0.337	0.676	0.913	0.913	<b>0.923</b>
$F_1$	0.672	0.716	0.711	0.667	0.715	0.858	0.890	<b>0.905</b>

# Adversarial autoencoders help

- Construct anomaly score  $s(x)$  using data
- Signal about anomaly if anomaly score is greater than some threshold  $t$



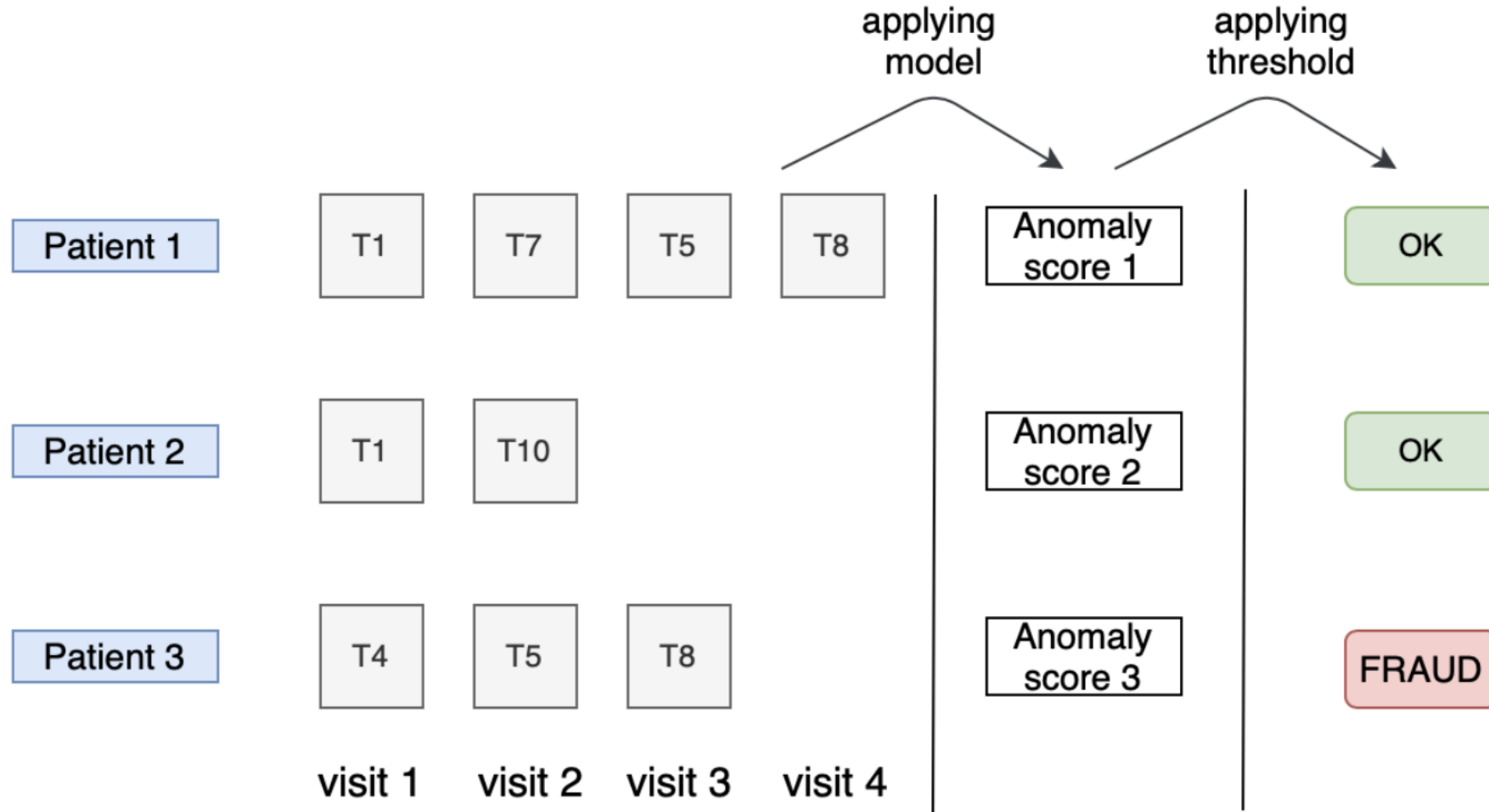
<https://papers.nips.cc/paper/7915-generative-probabilistic-novelty-detection-with-adversarial-autoencoders.pdf>



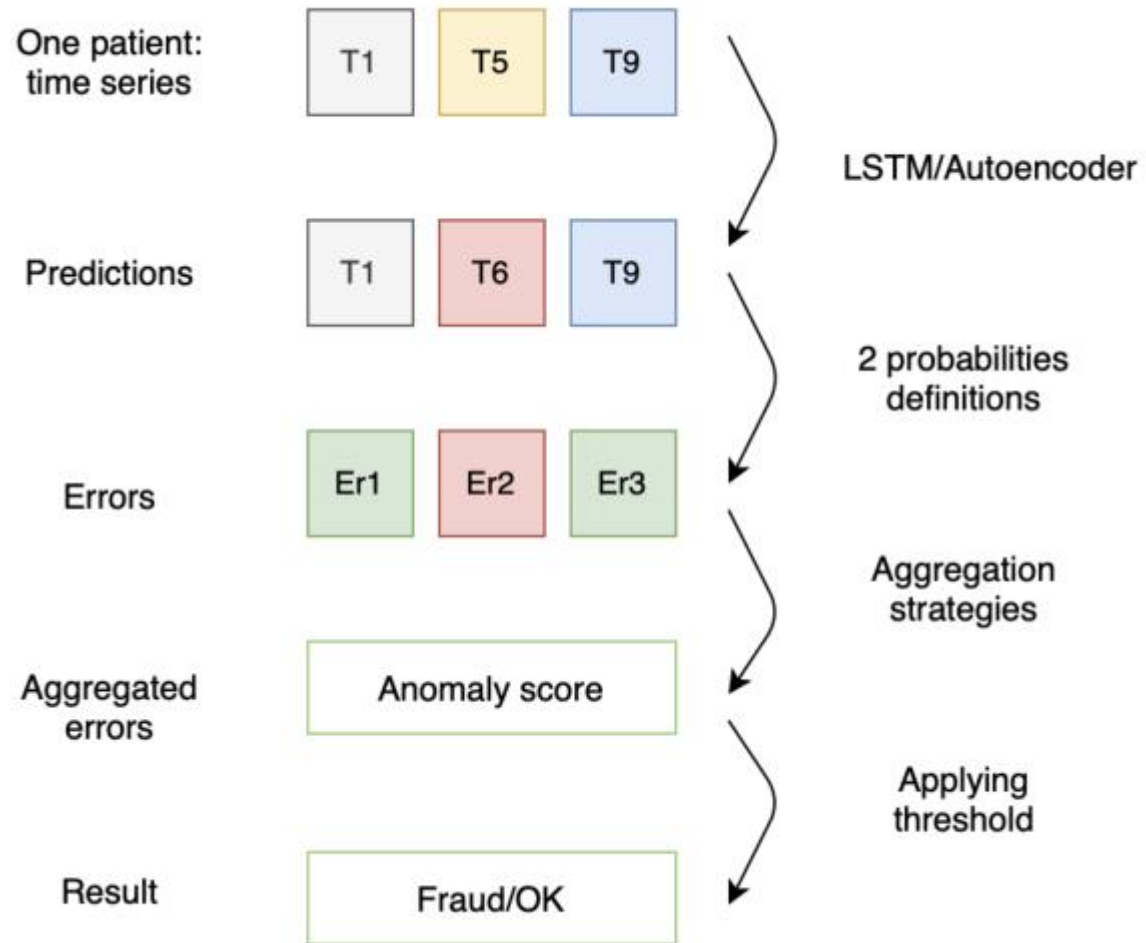
# Anomaly Detection for Time Series



# Anomaly detection for sequential data: healthcare insurance



# Anomaly detection for sequential data: healthcare insurance



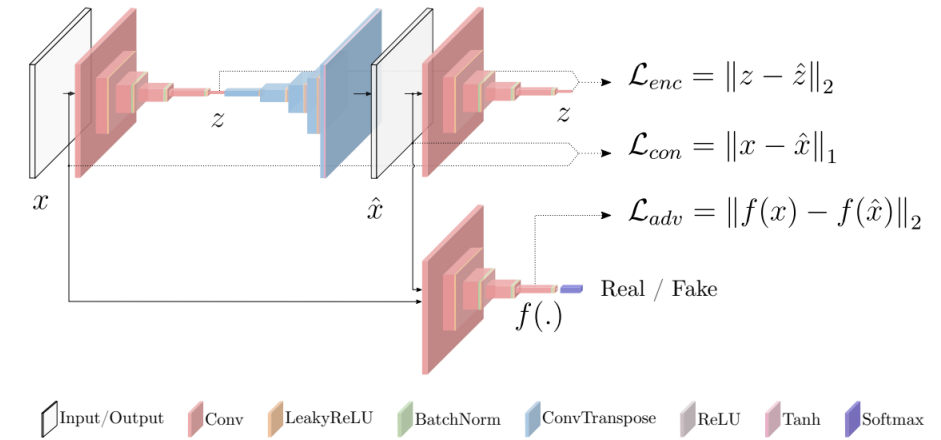
# Anomaly and novelty for Time Series

## Problems:

1. SotA techniques of anomaly detection are rarely used for classic time series
2. Available solutions don't take into account the statistical nature of Time Series

## Proposed solution:

1. To develop a loss function for GAN-based anomaly detection for time series
2. To take into account requirements of statistical change point detection models: low number of false alarms and small detection delay
3. To develop new resampling techniques by learning data distribution



*Li D. et al. MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks // arXiv:1901.04997. – 2019.*

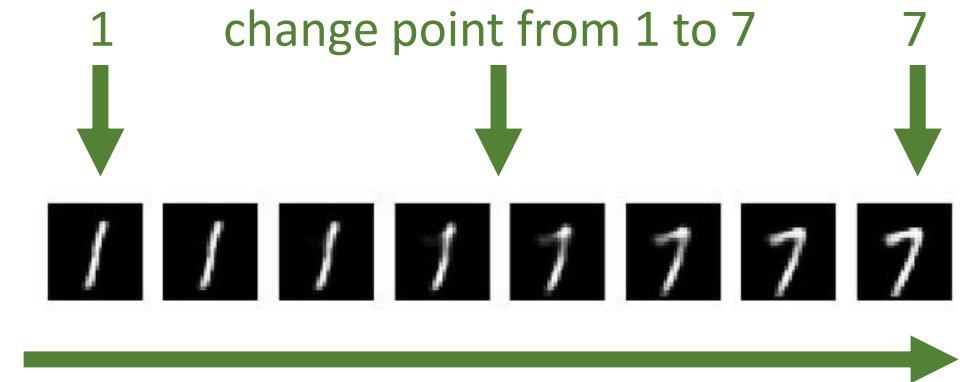
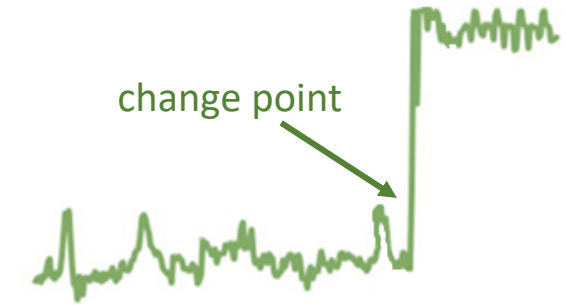
# Change detection in semi-structured data

**The change-point detection** (CPD) model signals about time of change in the data distribution

**Semi-structured data** – sequences of semi-structured data (images, texts)

**Goal:** minimize Detection Delay & minimize number of False Alarms

**Problems:** Can't apply classic method for semi-structured data

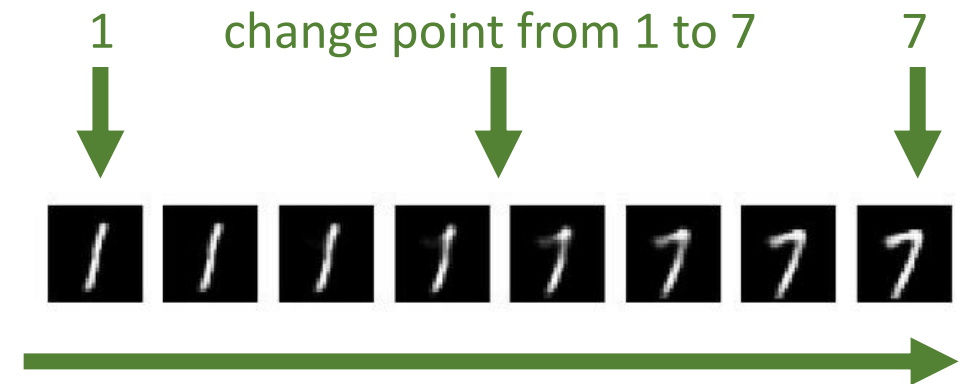
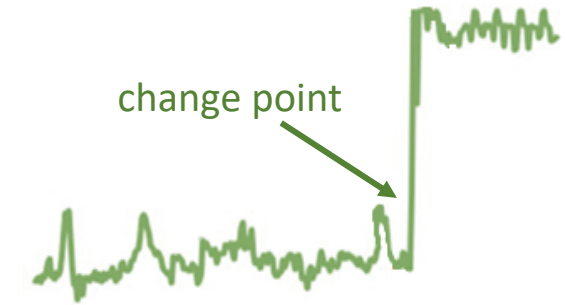


A sequence of MNIST images is an example of semi-structured data

# Change detection in semi-structured data

## Proposed solution:

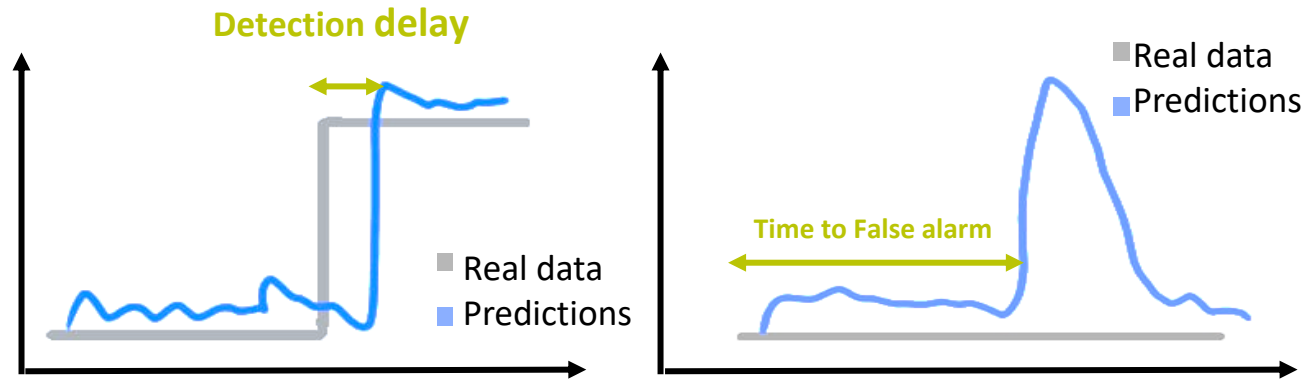
1. Develop a data embedding procedure
2. End2end methods based on statistical tests or detection outliers in embedded space – unsupervised anomaly detection
3. Develop new loss function for direct minimization of the problem specific metrics



A sequence of MNIST images is an example of semi-structured data

# Our end2end approach

We concentrate on typical quality metrics for change-point detection: delay detection and mean time to False alarm.



We optimize lower bounds for these metrics:

- $p_k$  is the model's change point probability at moment  $k$ ,
- $T$  – hyperparameter that restricts the length of the considered sequence.

$$Loss_{detection\_delay} = \sum_{t=\theta}^T (t - \theta) p_t \prod_{k=\theta}^{t-1} (1 - p_k) + (T + 1) \prod_{k=\theta}^T (1 - p_k),$$

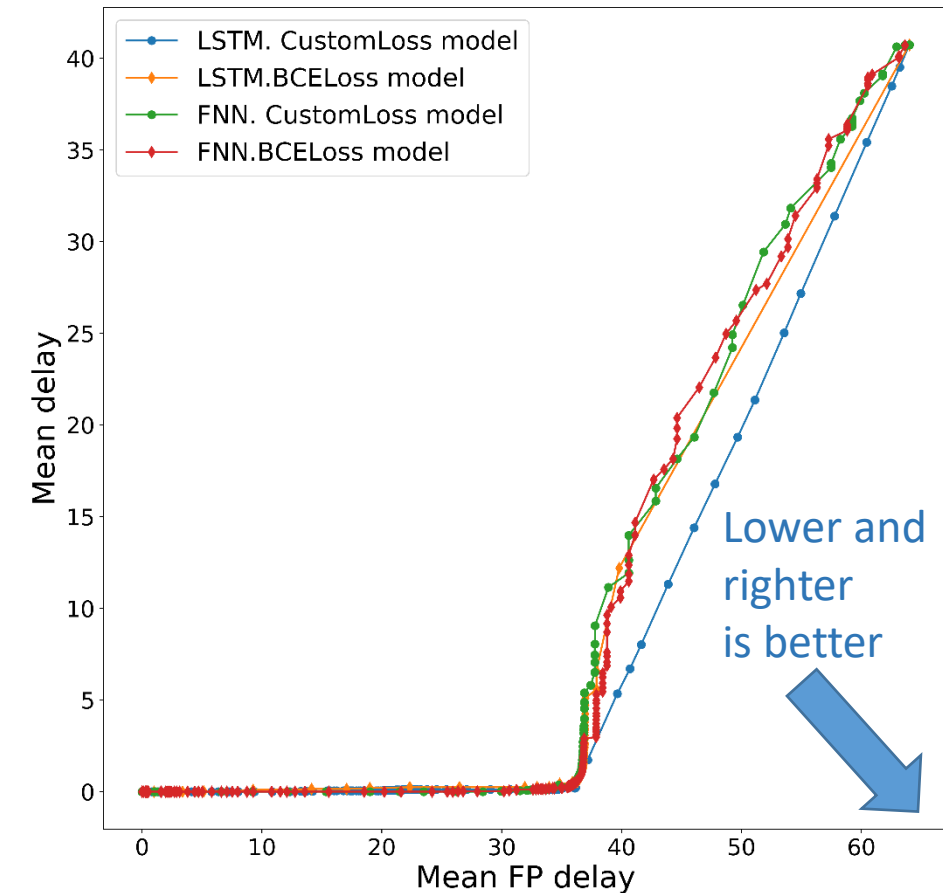
$$Loss_{FP\_delay} = 1 - \sum_{t=0}^{\theta} (t - \theta) p_t \prod_{k=0}^{\theta} (1 - p_k)$$

# Results

**Dataset:** sequences with images from MNIST. There are sequences with (e.g. from 1 to 7) and without (e.g. from 1 to 1) change point.

We compare LSTM and fully connected neural network (FNN) architectures, as well as binary cross entropy loss (BCELoss) and our proposed loss.

**LSTM with proposed loss function** has a better Pareto frontier with respect to the mean detection delay and the mean false positive delay.





# Future work

- Try proposed approach for other datasets of semi-structured data
- Consider different neural network architectures for processing of semi-structured sequential data
- Combine representation learning and statistical change point detection procedures

# References

Change point detection (CPD). Basic knowledge and main statistical approaches:

- Shiryaev A. N. Stochastic disorder problems. – Springer International Publishing, 2019.
- Romanenkova E. et al. Real-Time Data-Driven Detection of the Rock-Type Alteration During a Directional Drilling //IEEE Geoscience and Remote Sensing Letters. – 2019.

Supervised CPD:

- Malhotra P. et al. Long short term memory networks for anomaly detection in time series //ESANN proceedings, 2015.
- Hundman K. et al. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding //Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. – 2018

# Factors to consider when choosing an Anomaly Detector

- Few parameters
  - parameter-free the best
  - easy to tune; not too sensitive to parameter setting
- Fast runtime: can scale up to large datasets and high dimensional datasets
- Low space complexity
- Known behaviours under different data properties
- Can deal with different types of anomalies
- Its ability to deal with high dimensional problems
- Understand the nature of anomalies and the best match algorithm

[https://federation.edu.au/\\_data/assets/pdf\\_file/0011/443666/ICDM2018-Tutorial-Final.pdf](https://federation.edu.au/_data/assets/pdf_file/0011/443666/ICDM2018-Tutorial-Final.pdf)

# Take-home messages

- Anomaly detection is a challenging problem
- Often problem-specific knowledge helps
- Common approaches are Autoencoder-based and Isolation forest
- There are some time-series specific approaches: the problem is similar to the change point detection problem

# More references?

See

- <https://awesomeopensource.com/project/hoya012/awesome-anomaly-detection?categoryPage=18>
- <https://github.com/yzhao062/anomaly-detection-resources>
- <https://github.com/zhuyiche/awesome-anomaly-detection>

# Contacts

Alexey Zaytsev

[a.zaytsev@skoltech.ru](mailto:a.zaytsev@skoltech.ru)

Telegram: @likzet

