

# Machine Learning Based Outlier Detection for Data Certification

Patomporn Payoungkhamdee

Mahidol University

*patomporn.pay@gmail.com*

2 August 2019

# Overview

① Background

② Objective

③ Datasets

④ Model

One-Class SVM

Isolation Forest

Autoencoder

⑤ Results and Interpretation

⑥ Summary

# Data Quality Monitoring (DQM)

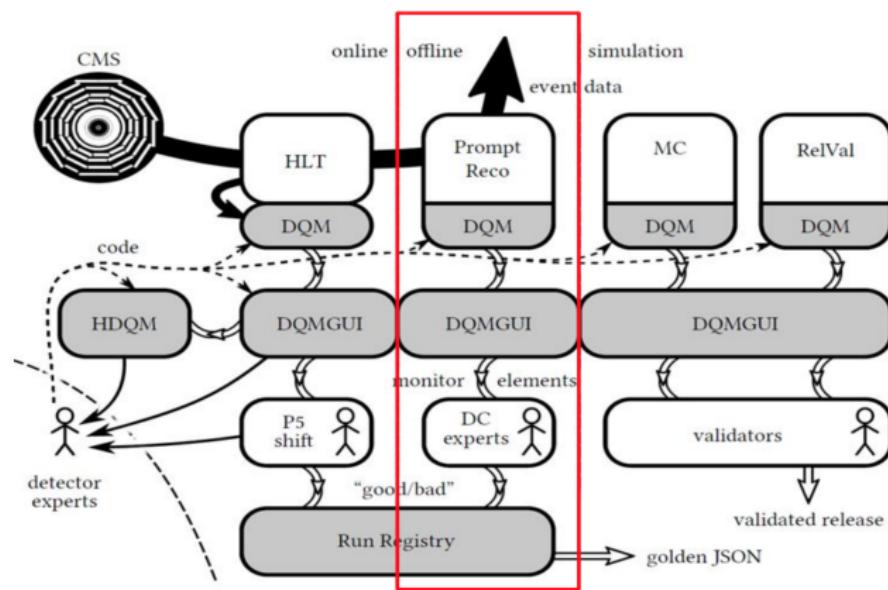


Figure: Tools and Processes of DQM, retrieved from M. Schneider, CHEP 2018

## Data granularity in CMS (Offline)

- Reconstruct physics quantity 48 Hours after collision
- Offline shifters and detector experts check the dozens of distribution histograms to define goodness of data
- Certification is made on Run and Lumisection levels
- Lumisection(LS) is taken around 23 seconds
- Criteria for bad LS
  - ① Runs tagged as bad by human (whole run)
  - ② Automatically filter by DCS bits, beam status and etc. (LS levels)
  - ③ In rare cases are marked by DC experts (LS levels)
- The Golden JSON contains the list of all good LS

# Objective

- **Certify data quality in lumisection granularity**
  - Classification on the basis of actual data distributions per LS
- Reduce manual work of DC Experts

# Expectation

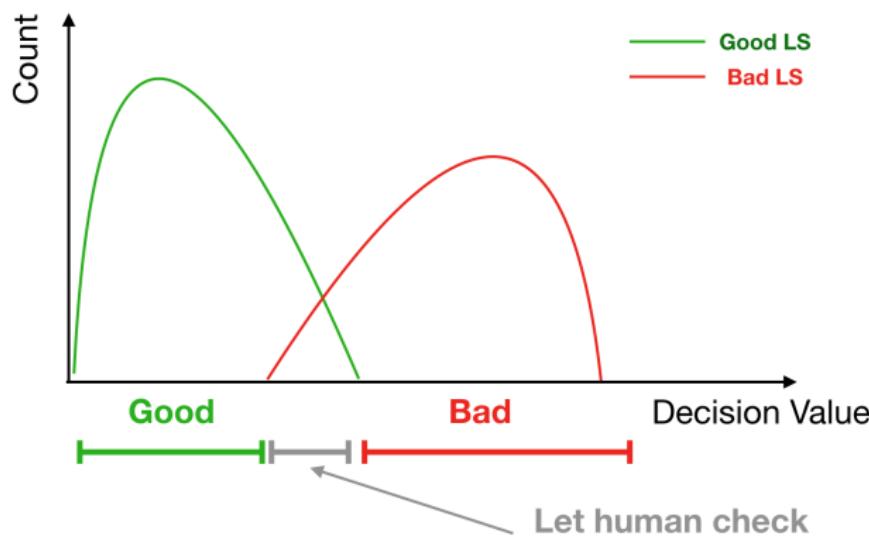
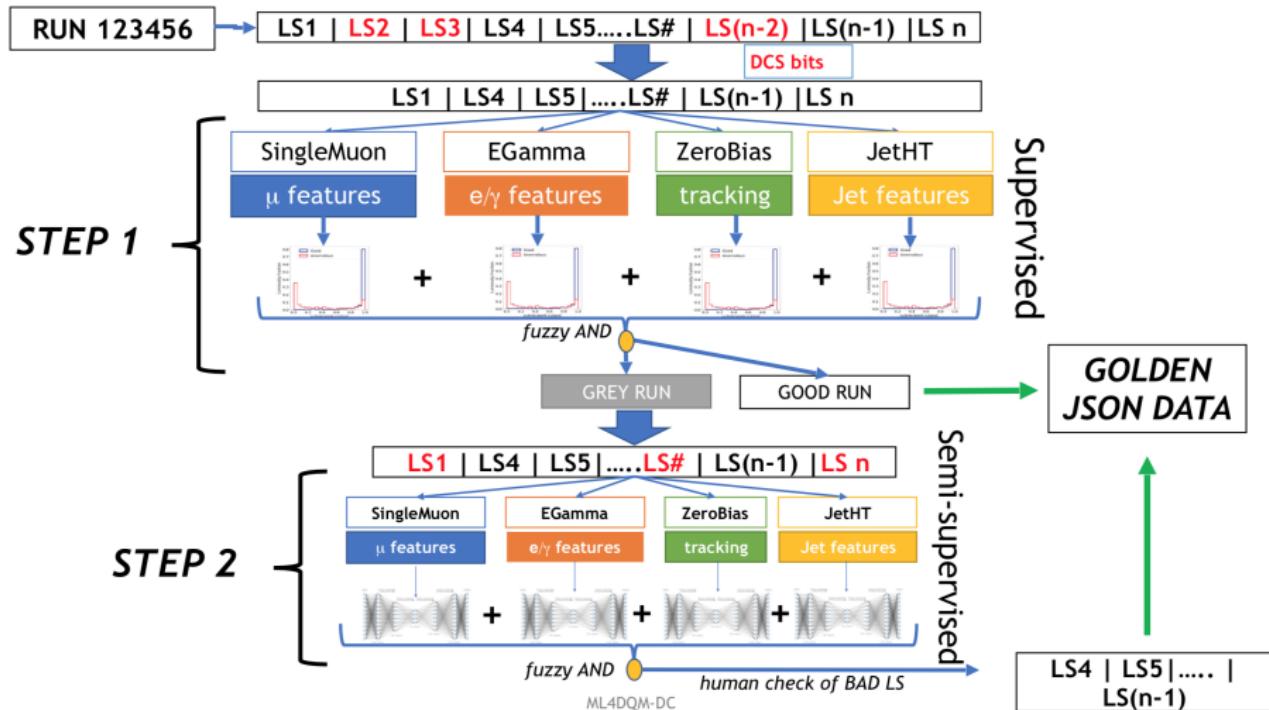


Figure: Three possible regions of prediction

## Proposal For An Alternative approach: two steps

- The automatic DCS bit flagging will stay, ML applied on top of it
- Automatize the Data Certification procedure in two steps
  - ① Provide a reliable quality **flag per Run using** grey-zone approach and **Supervised** models (artificial BAD data can be used for training)
  - ② **Use Autoencoders** only on the grey-zone with the goal **to search for anomalous LS** and flag them automatically, human double check at this stage
- Using physical quantities as
  - **features** (pT, eta, etc) and
  - **objects** mapped to the relevant Primary Dataset (i.e tracks to ZeroBias, muons to SingleMuon ... etc)  
to better mimic the current DC procedure
- In this work, I will focus only on second step



# Datasets

- pp collisions, 2016 data, PromptReco, JetHT
- Each lumisection (datapoint) contains
  - 39 histogram of physics quantity e.g. JetPt, JetEta, JetPhi, etc.
  - Represent one histogram with 7 numbers
  - 259 Features ( $39 \times 7$ )
- Good LS defined in Golden JSON else Bad LS
- Data splitting
  - 60% good LSs for training
  - 20% good LSs for validation
  - 20% good LSs combine with bad LS for testing

# Histogram representation

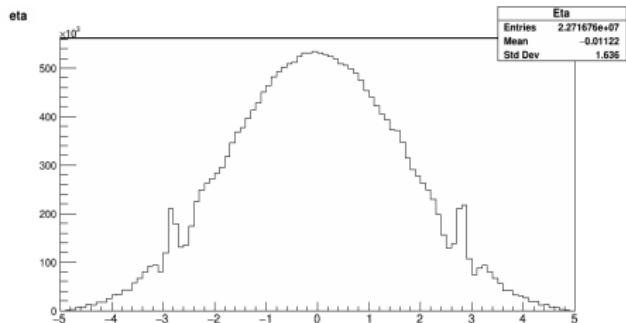


Figure: Example of Eta distribution

- Collection of physics objects e.g. photons, muons and so on
  - Measurement quantity:  
Transverse momentum, eta, phi, etc. [▶ Click here to see all selected features](#)
- 
- ① Quantize [0%, 25%, 50%, 75%, 100%] of the histogram
  - ② Combine mean and rms
  - ③ Use these **7 values to represent one histogram**

# Data Preprocessing

- MinMaxScalar Transformation
- Consider Lumisection  $i$  and Feature  $j$

$$x'_{ij} \leftarrow \frac{x_{ij} - \min_{\forall i \in S_{\text{train}}} \{x_{ij}\}}{\max_{\forall i \in S_{\text{train}}} \{x_{ij}\} - \min_{\forall i \in S_{\text{train}}} \{x_{ij}\}} \quad (1)$$

- Then our datapoint should be in range  $[0, 1]$

# Semi-supervised Learning

- Unsupervised Models
  - Schölkopf's One-Class SVM
  - Isolation Forest
  - 4 Flavours of Autoencoder
- Feed only good LS for train and validate the model
- Testing with good LS and bad LS
- Consequently, it's falling into **Semi-supervised Learning** category

## One-Class SVM

## Schölkopf's One-Class SVM

- Minimize (Soft Margin)

$$\frac{\|w\|^2}{2} + \frac{1}{\nu l} \sum_{i=1}^l \xi_i - \rho \quad (2)$$

- Under

$$w \cdot \Phi(x_i) \geq \rho - \xi_i, \quad \xi_i \geq 0 \quad (3)$$

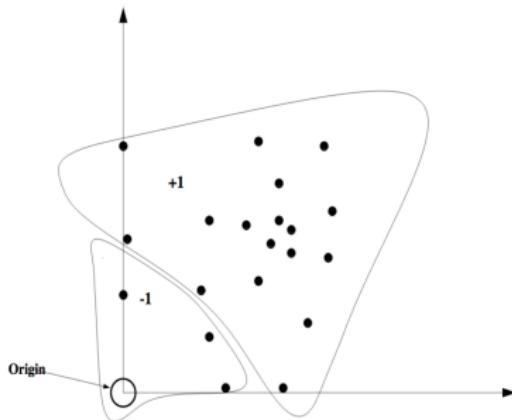


Figure: Scattering in latent space: retrieved from

<http://www.jmlr.org/papers/volume2/manevitz01a/manevitz01a.pdf>

- Kernel:** Gaussian Base Radial function (GBF)
- Determine tangent distance from hyperplane

## Isolation Forest

# Isolation Forest

- Ensemble Forest from tree by subsampling ( $\Psi$ )
  - Iteratively picking up features and random value to construct the node (equivalent to step function)
  - Anomaly score evaluate from average depth of the instance over forest

$$s(x, \Psi) = \exp^{-\langle h(x) \rangle / c(\Psi)} \quad (4)$$

- where
  - $h(x)$  is the depth in tree  $h$
  - $c(\Psi)$  normalization factor growing as  $\log_2(\Psi)$  from branching

[1] <https://cs.nju.edu.cn/zhouzh/zhouzh.files/publication/icdm08b.pdf?q=isolation-forest>

## Autoencoder

## Vanilla Autoencoder

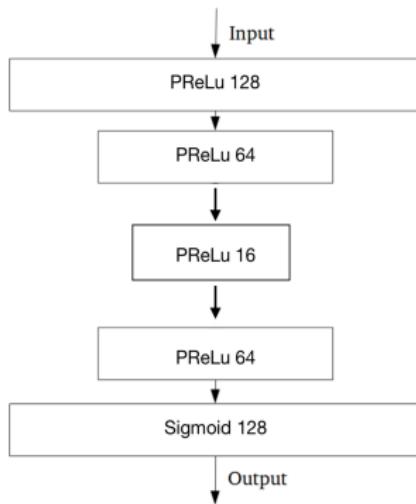


Figure: Body of Vanilla AE

- Concise the information into small latent space and reconstruct
- Loss function

$$\mathcal{L}_{\text{tot}} \equiv \frac{1}{N} \sum_i^N |x_i - \tilde{x}_i|^2 \quad (5)$$

- Truncated normal initializer
- Adam optimizer

## Autoencoder

## Sparse Autoencoder

- Similar to Vanilla AE
- Tweak by L1 Regularization (Prevent overfitting)
- Loss function

$$\mathcal{L}_{\text{tot}} \equiv \frac{1}{N} \sum_i^N |x_i - \tilde{x}_i|^2 + \lambda_s \sum_j ||w_j|| \quad (6)$$

- where  $\lambda_s = 10^{-5}$

## Autoencoder

## Contractive Autoencoder

- Tweak by Jacobi Matrix (Prevent variation in dataset)
- Loss function

$$\mathcal{L}_{\text{tot}} \equiv \frac{1}{N} \sum_i^N |x_i - \tilde{x}_i|^2 + \lambda_c ||J_h(x)||^2 \quad (7)$$

- where  $\lambda_c = 10^{-5}$

## Autoencoder

## Contractive Autoencoder

- Definition

$$\|J_h(x)\|^2 \equiv \frac{1}{N} \sum_{ij} \left( \frac{\partial h_j}{\partial x_i} \right)^2 \quad (8)$$

- where  $h_j$  is activation function
- In our cases
  - PReLU activation function

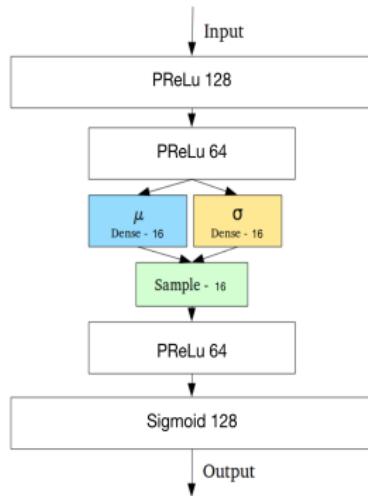
$$\|J_h(x)\|^2 = \frac{1}{N} \sum_j [\alpha_j H(-(w_{ji}x^i + b_j)) + H(w_{ji}x^i + b_j)] \sum_i (w_{ji})^2 \quad (9)$$

- Sigmoid activation function

$$\|J_h(x)\|^2 = \frac{1}{N} \sum_j [h_j * (1 - h_j)] \sum_i (w_{ji})^2 \quad (10)$$

## Autoencoder

## Variational Autoencoder



**Figure:** Body of Variational AE retrieved from  
<https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf>

- Random “new sampling” in latent space by gaussian random generator

$$\mathcal{Z} \equiv \mathcal{N}(\mu_i, \sigma_i) \quad (11)$$

- Tweak by reduce discontinuity in latent space
- Loss function

$$\mathcal{L}_{\text{tot}} = \frac{1}{N} \sum_i^N |x_i - \tilde{x}_i|^2 + \mathcal{D}_{\text{KL}}(p|q) \quad (12)$$

## Autoencoder

## Variational Autoencoder

## Theorem (Kullback-Leibler Divergence)

- “How much information is loss after represent data with function”

$$\mathcal{D}_{KL} \equiv <\log p - \log q> \quad (13)$$

- Where  $p$  is observed value and  $q$  is approximation function
- Since our  $q$  is Gaussian function

$$\mathcal{D}_{KL,i} = \frac{1}{2} \sum_k^{n_{latent}} (\mu_{ik}^2 + \sigma_{ik}^2 - 2 \log \sigma_{ik} - 1) \quad (14)$$

$$\mathcal{L}_{\text{tot}} = \frac{1}{N} \sum_i^N |x_i - \tilde{x}_i|^2 + \frac{1}{2N} \sum_i^N \sum_k^{n_{latent}} (\mu_{ik}^2 + \sigma_{ik}^2 - 2 \log \sigma_{ik} - 1) \quad (15)$$

# Performance

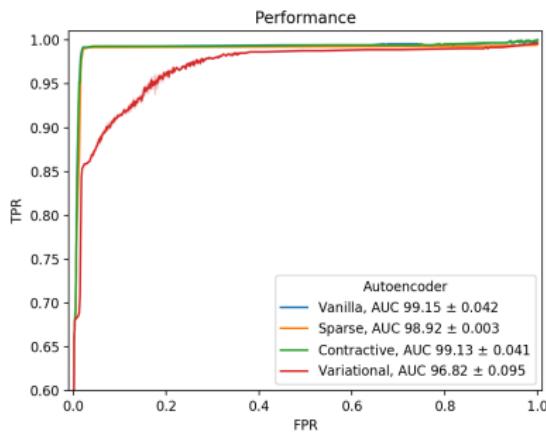


Figure: Various AE

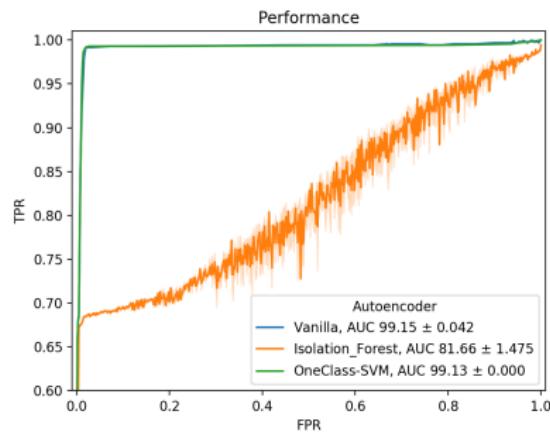


Figure: Vanilla vs SVM vs Forest

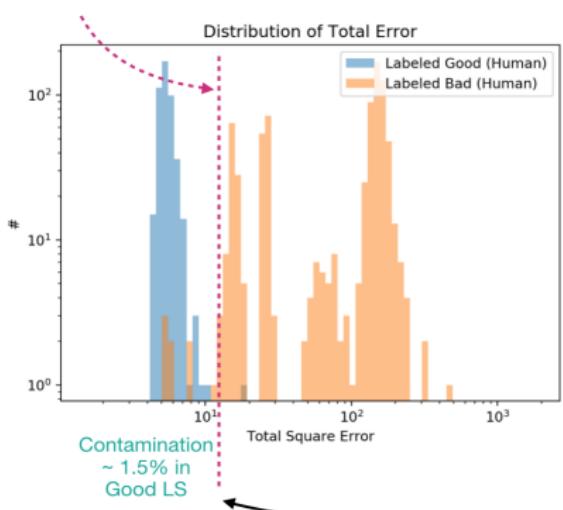
Under configuration

- **Isolation Forest**  $N_{tree} = 200$ ,  $\Psi = 512$
- **OneClass-SVM**  $\nu = 0.1$ ,  $\gamma = 0.1$ (Inverse gaussian width)

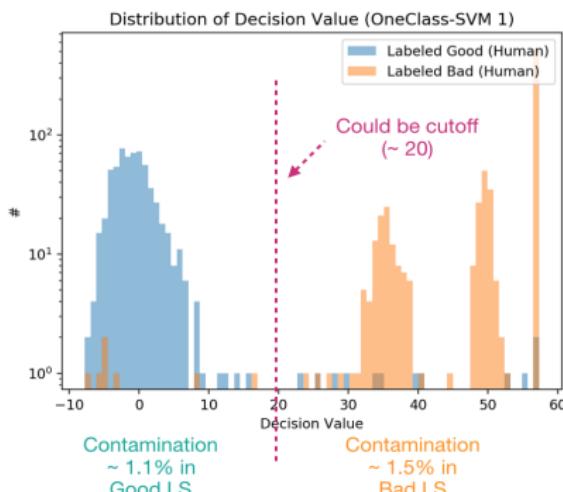
# Find the cutoff

## Vanilla AE

Could be cutoff  
(~ 10)



## One-Class SVM



Spot the same Bad => Good LS

# Example of Reconstruction

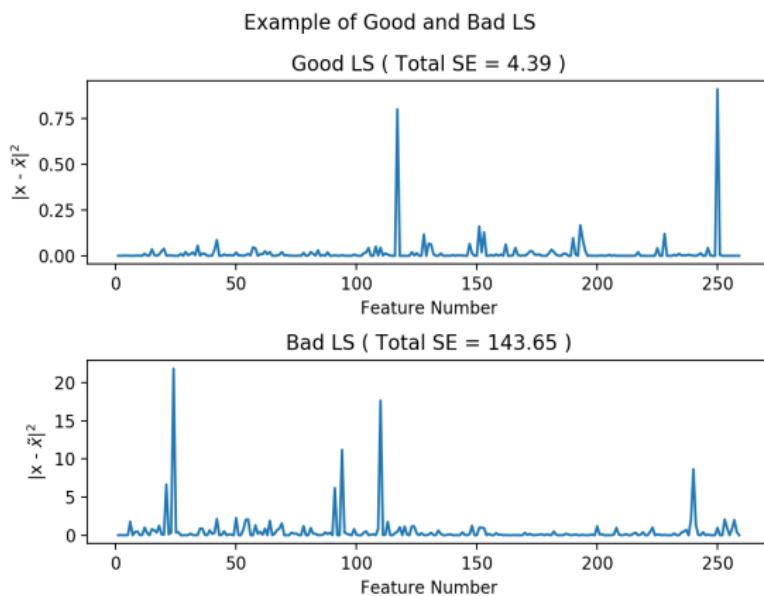
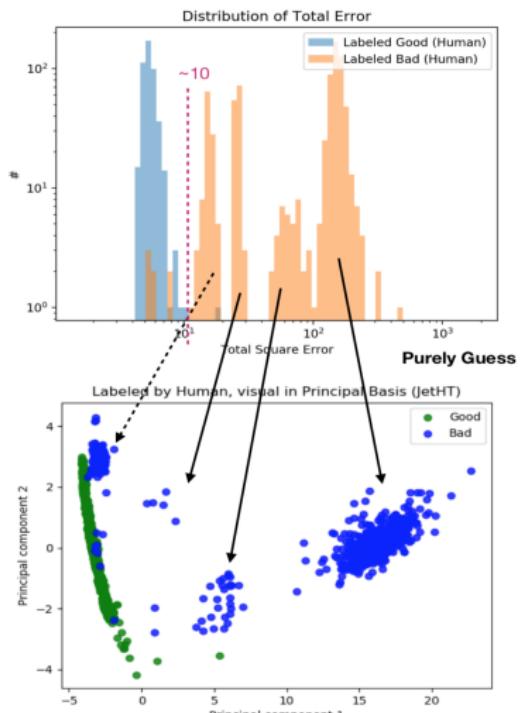
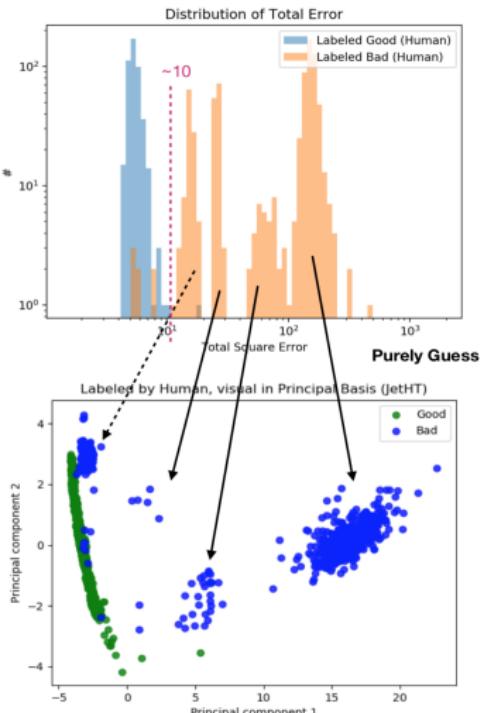
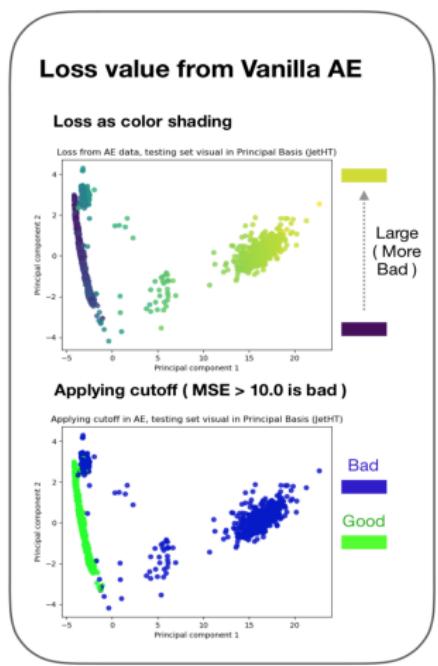


Figure: Reconstruction error from Vanilla AE

# Extended Investigation

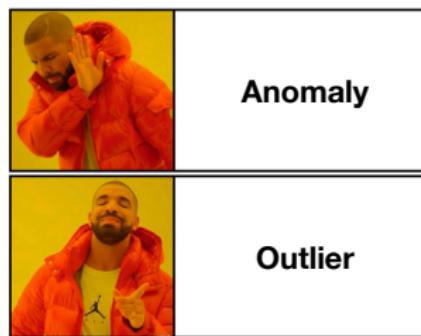


# Extended Investigation



# Summary

- Semi-supervised learning yield a remarkable result
- There is no grey zone from our model for this dataset
- Bad LS could be divided into two parts
  - Bad with some pattern
  - Anomaly



## Future work

- Exploit the same technique in 2018 dataset
- 2018 data preparation still in progress
- Remove trivial bad LS (run tag and DCS bits) but take consider only bad LS that came from DC Experts

# Future work

Let's have a look at principal component for JetHT PD

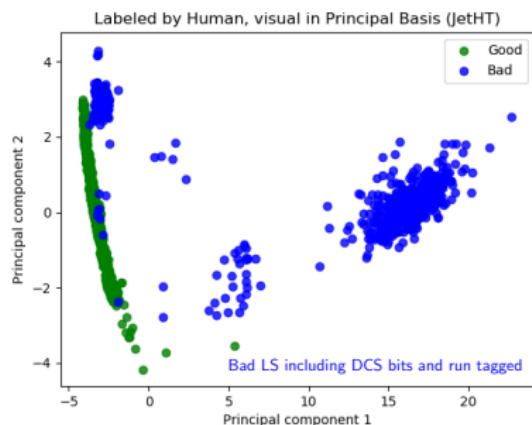


Figure: 2016 data

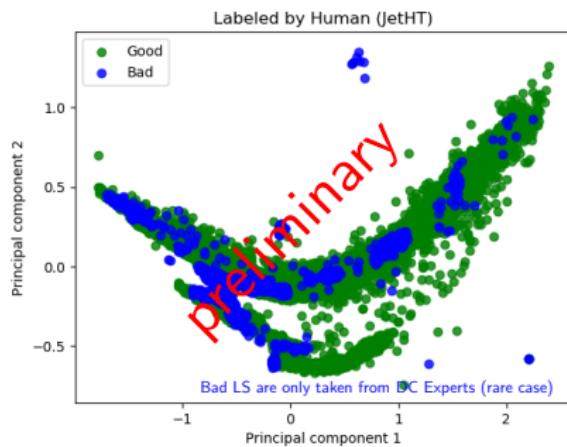


Figure: 2018 data

# Acknowledgement

- CERN Summer Student program 2019
- Especially
  - **Marcel Andre Schneider**
  - Francesco Fiori
  - Kaori Maeshima
  - Adrian Alan Pol
  - Countless CMS DQM people :)
- GPU resources from IBM in collaboration with CERN Openlab





# Question?

Background  
oo

Objective  
oooo

Datasets  
ooo

Model  
o  
o  
oooooo

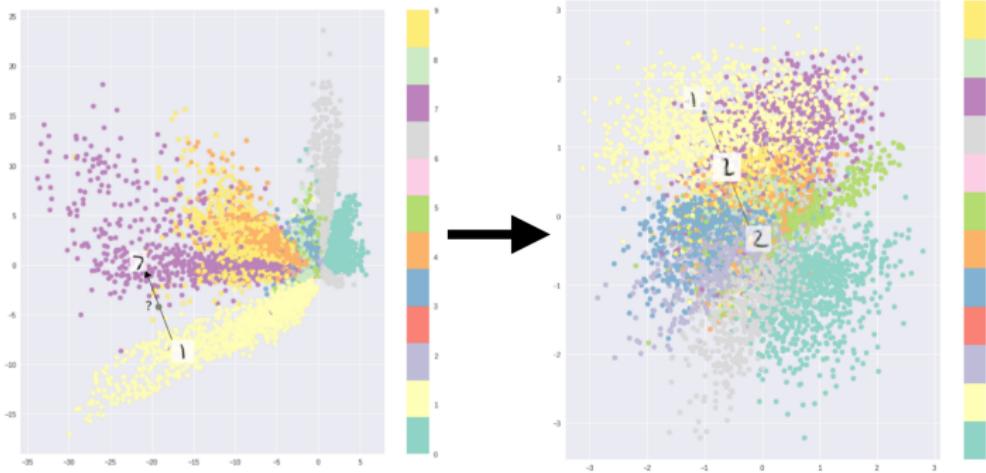
Results and Interpretation  
ooooo

Summary

# Back up

## Example of removing discontinuity in latent space

### Ex: Latent space in MNIST



[1] <https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf>