CRAYFIS
cosmic rays found in smartphones

SCHOOL OF DATA ANALYSIS

CMS DQM

experiments with individual channels

Maxim Borisyak, Fedor Ratnikov, Denis Derkach, Andrey Ustyuzhanin

Yandex School of Data Analysis

Motivation

Main goal

Study how anomalies affect individual channels.

Examples.

- What channels are responsible for anomalies?
- If only photons were affected is it possible to save muonic data?
- Which plots should receive more attention from DQ experts?

Outline

Outline:

- data overview
- unsupervised approach
- supervised approach

Data overview

Data

The same settings as for the anomaly detection study:

- 2010B open data;
- data from three streams:
  - photons,
  - muons,
  - minibias,
- 4 channels:
  - photons,
  - muons,
  - PF,
  - calo.

Data

This study was performed on:

- photon stream, photon channel,
- muon stream, muon channel,
- minibias stream, PF channel,
- minibias stream, calo channel.

Unsupervised approach

AutoEncoder

AutoEncoder (AE) is a type of Neural Network for dimensionality reduction. AE typically consists of two parts, each represented by a Neural Network:

- encoder (f) - a transformation from original feature space into reduced space (code space):

$$f : \mathbb{R}^n \to \mathbb{R}^m$$

- decoder (g) - reverse transformation ($g \approx f^{-1}$):

$$g : \mathbb{R}^m \to \mathbb{R}^n$$

Principal Component Analysis is a special form of AE.

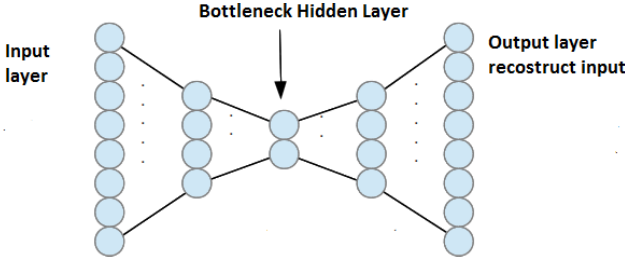
AutoEncoder

Typically, either:

- $m \ll n$;
- f is heavily restricted (e.g. for sparsity).

AE is trained to find such encoder and decoder to efficiently reconstruct original input, e.g.:

$$\mathbb{E}_X \left(X - g(f(X))\right)^2 \to \min$$

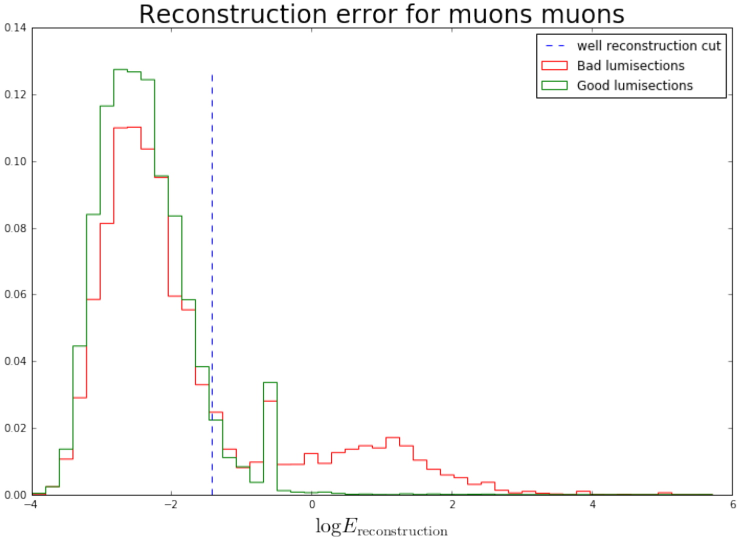


Unsupervised approach

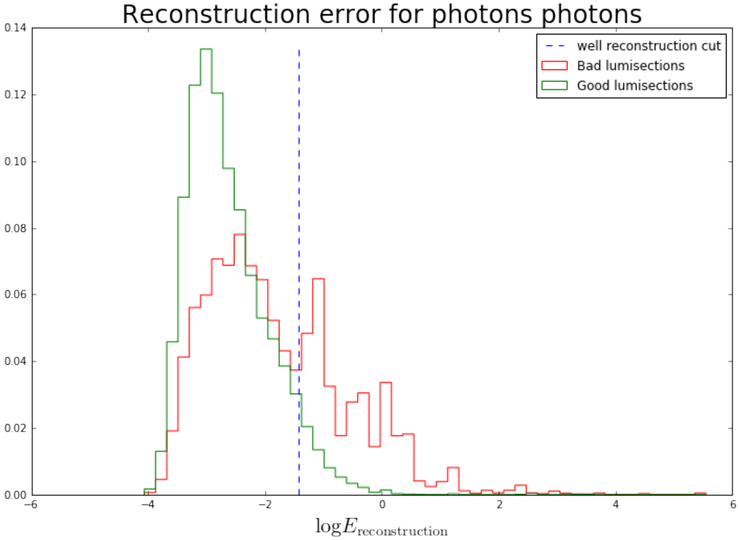
For each channel individually:

1. train AutoEncoder (AE) on good lumisections:
   - from 250+ features to 20;
2. compare distribution of reconstruction error with these of anomalous lumisections:
   - AE has to learn statistical relations within features;
   - anomalous data has a little chance to be reconstructed well,
   - unless anomaly did not affect this channel.

AE results

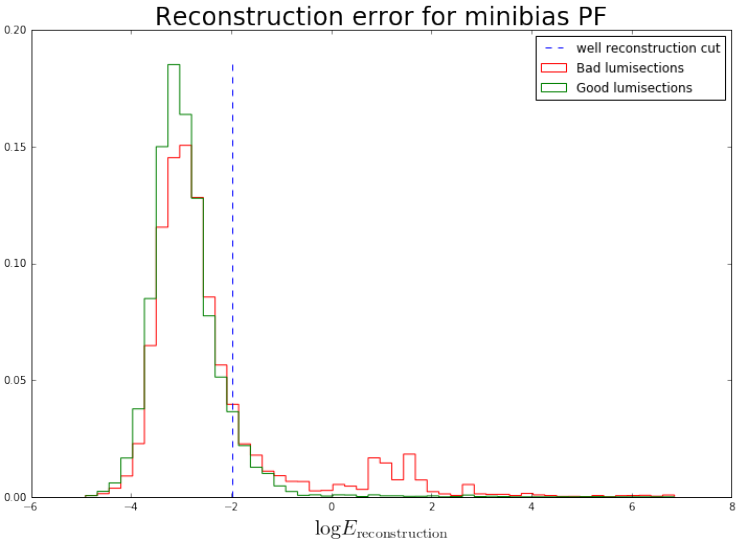

Reconstruction error for muons muons
well reconstruction cut
Bad lumisections
Good lumisections
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
−4
−2
0
2
4
6
$\log E_{\mathrm{reconstruction}}$

AE results



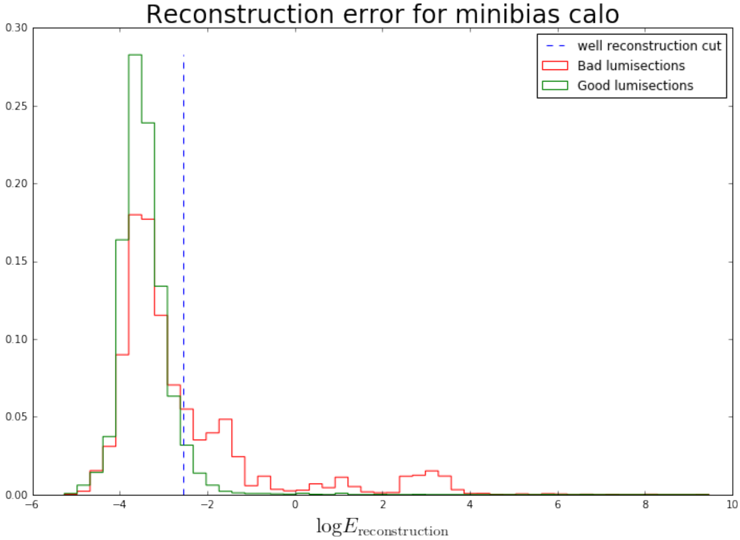
Reconstruction error for photons photons
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
−6
−4
−2
0
2
4
6
$\log E_{\mathrm{reconstruction}}$
well reconstruction cut
Bad lumisections
Good lumisections

AE results



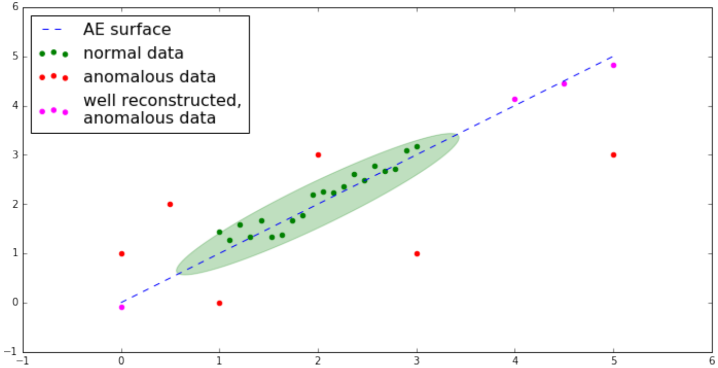
Reconstruction error for minibias PF
well reconstruction cut
Bad lumisections
Good lumisections
0.20
0.15
0.10
0.05
0.00
−6
−4
−2
0
2
4
6
8
$\log E_{\mathrm{reconstruction}}$

AE results


Reconstruction error for minibias calo
well reconstruction cut
Bad lumisections
Good lumisections
0.30
0.25
0.20
0.15
0.10
0.05
0.00
−6
−4
−2
0
2
4
6
8
10
$\log E_{\mathrm{reconstruction}}$

Unsupervised approach

AE has a small chance of learning to compress well some anomalous data.



An example of anomalous data being reconstructed well.

One-class SVM

To enhance our estimations, distribution support estimation is performed. One-class Support Vector Machine (ν-SVM) is a special case of SVM to estimate support of given data.

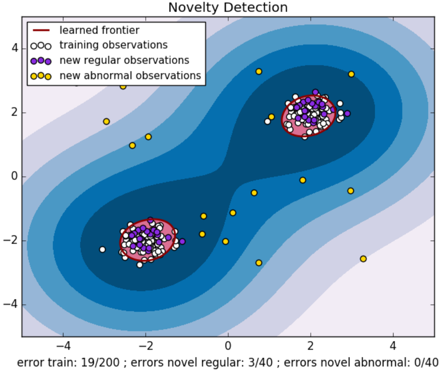
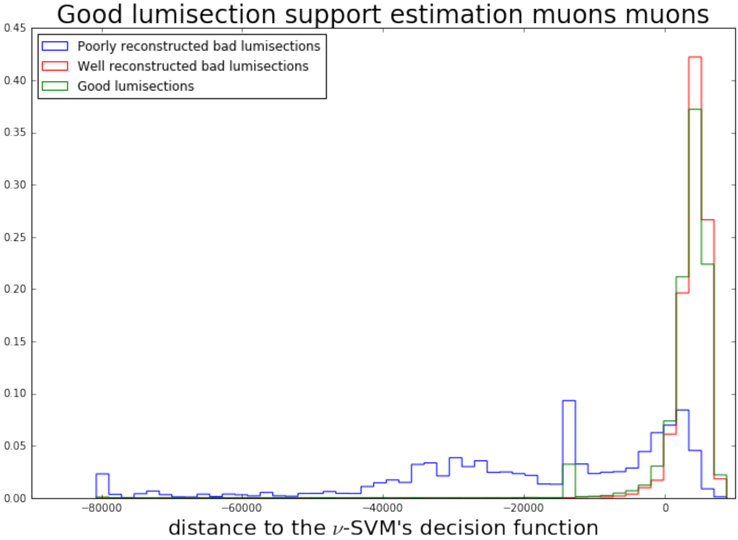


Figure from http://scikit-learn.org

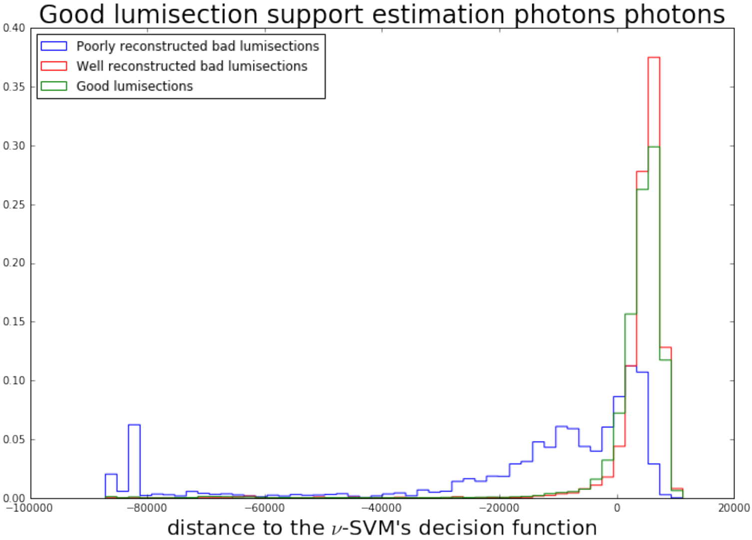
Unsupervised approach

1. in AE's code space train a one-class SVM on good lumisections to estimate support of normal data distribution;
2. compare distributions of distances to the support bounds:
   - positive distances correspond to the interior of the support.

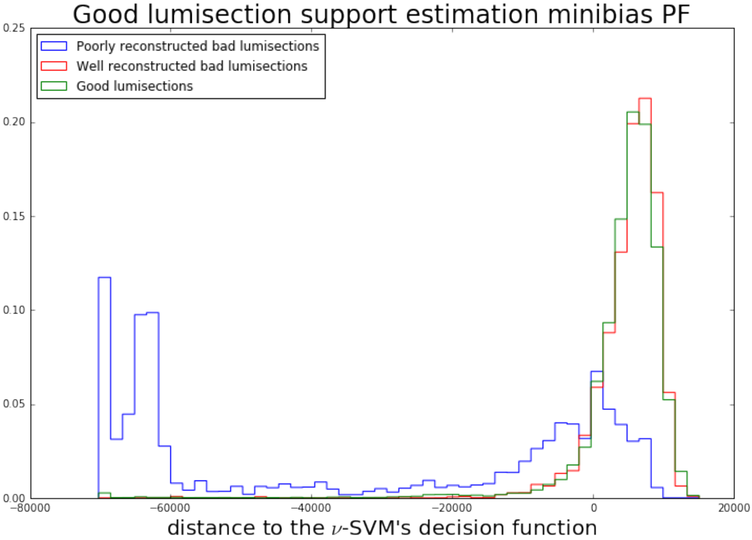
ν-SVM results



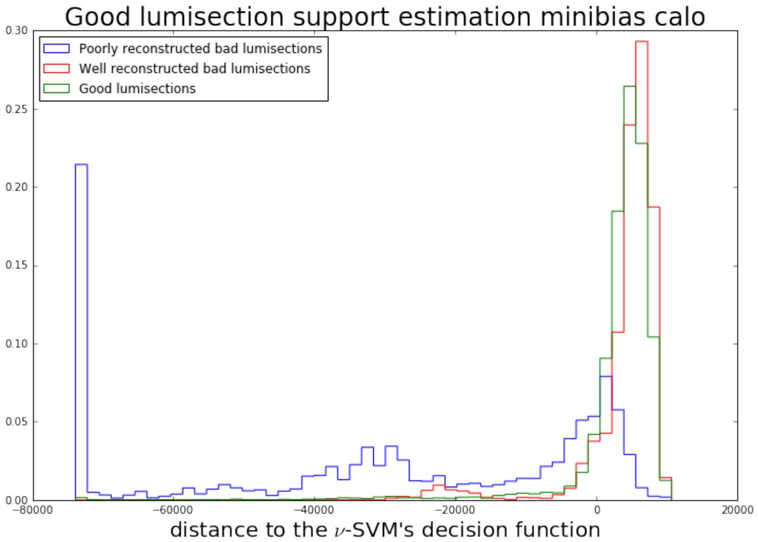
ν-SVM results



ν-SVM results



ν-SVM results



Supervised approach

Supervised approach

1. On features from each channel build a neural network;
2. each channel network returns a score for its channel;
3. connect networks by:
   - logistic regression,
   - `min` operator (with dropout),
   - a sort of fuzzy `and` operator;
4. train network to recover global labels;
5. define estimation of score for each channel as corresponding network output.

Discussion

Consider set of channels $\mathcal{C}$, an anomaly A affecting channels $C \subseteq \mathcal{C}$.

Assumption 1

Anomaly A can be detected independenly from data of any channel from C.

Assumption 2

Anomaly A can not be detected from data of channels other than C.

Discussion

Corollary 1

If an anomaly can be detected from a channel's features, the channel's data is anomalous.

'Theorem' 1

Under assumptions 1 and 2
for all described above networks
having enough degrees of freedom and data samples for training
each subnetwork has high descriminative power against anomalies affecting its channel.

'Proof' of `fuzzy-and` network

The idea of the 'proof' is to show that global minimum of loss function corresponds to the state where each subnetwork 'reacts' only on anomalies affecting its channel.

- cross-entropy loss (1 - normal lumisections, 0 - anomaly);
- outputs $f^i_{\mathrm{subnetwork}}$ of i-th subnetwork are bounded:

$$f^i_{\mathrm{subnetwork}} \in (0, 1)$$

- activation function for the whole network:

$$f_{\mathrm{network}} = \phi\left(\sum_{i=1}^{4} f^i_{\mathrm{subnetwork}}\right)$$

$$\phi(x) = \mathsf{exp}(x - 4)$$

'Proof' of `fuzzy-and` network

- consider channel $c \in \mathcal{C}$:
  - $\mathcal{A}_c$ all anomalies that does affect c;
  - $\bar{\mathcal{A}}_c$ all anomalies that does not affect c;

'Lemma' 1

Under assumptions 1 and 2 and the theorem's conditions in case of anomaly from $\mathcal{A}_c$ output of subnetworks corresponding to channel c is as close to 0 (anomaly) as possible.

'Proof' of `fuzzy-and` network

- relative to a subnetwork there are 3 cases:
  - no anomalies;
  - anomaly 'visible' from its channel ($\mathcal{A}_c$);
  - anomalies 'invisible' from its channel ($\bar{\mathcal{A}}_c$);
- with respect to these cases, loss of the whole network can be decomposed into:

$$\mathcal{L} = \mathcal{L}_{\text{normal}} + \mathcal{L}_{\bar{\mathcal{A}}_c} + \mathcal{L}_{\mathcal{A}_c}$$

'Proof' of `fuzzy-and` network

- $\mathcal{L}_{\mathcal{A}_c}$ and $\mathcal{L}_{\mathrm{normal}} + \mathcal{L}_{\bar{\mathcal{A}}_c}$ can be optimized independently:
  - since structure of subnetwork is sufficient to learn to separate these cases by assumptions;

$$\mathcal{L}_{\mathcal{A}_c} = -\sum_j \log\left[1 - \exp\left(\sum_{i=1}^{4} f^i_{\mathrm{subnetwork}}(X_j) - 4\right)\right]$$

- where the first sum is over samples X_j with anomalies from $\mathcal{A}_c$;
- since subnetworks are independent:
  - $\mathcal{L}_{\mathcal{A}_c}$ is minimized when output of the subnetwork built on channel c is as close to 0 as possible.
- This proves 'Lemma' 1.

'Proof' of `fuzzy-and` network

- subnetwork can not distinguish normal cases and $\bar{\mathcal{A}}_c$;
- nevertheless, since $\bar{\mathcal{A}}_c$ is still an anomaly, subnetwork receives punishment either for:
  - predicting low score for normal cases;
  - predicting large score for cases from $\bar{\mathcal{A}}_c$.
- this may result in some bias relative to the presence of anomalies from $\mathcal{A}_c$.

'Lemma' 2

Under assumptions and theorem 1 conditions, all subnetwork are unbiased, i.e. for normal cases and anomalies from $\bar{\mathcal{A}}_c$ output of subnetwork for channel c is close to 1.

'Proof' of `fuzzy-and` network

Let X be output of subnetwork for channel c under normal cases and anomalies from $\bar{\mathcal{A}}_c$, ϵ_i and ϵ'_i - sum of outputs from the rest of subnetworks, α - fraction of good lumisections, β - fraction of anomalies from $\bar{\mathcal{A}}_c$:

$$\begin{aligned}
\mathcal{L} &= \mathcal{L}_{\mathrm{normal}} + \mathcal{L}_{\bar{\mathcal{A}}_c} + \mathcal{L}_{\mathcal{A}_c} \\
&= -\frac{\alpha}{n_1}\sum_{i=1}^{n_1} \log \exp\left(X + \epsilon'_i - 4\right) \\
&- \frac{\beta}{n_2}\sum_{i=1}^{n_2} \log\left(1 - \exp\left(X + \epsilon_i - 4\right)\right) \\
&+ \mathcal{L}_{\mathcal{A}_c}
\end{aligned}$$

'Proof' of `fuzzy-and` network

In the worst case scenario and by 'Lemma' 1 (at least one network reports anomaly with score $\delta \ll 1$):

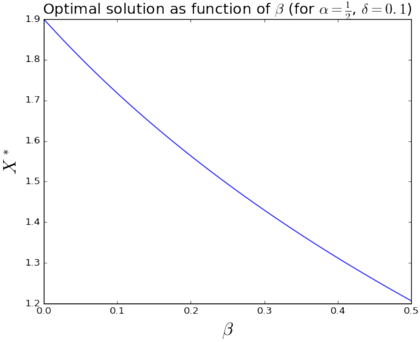
$$\epsilon < 2 + \delta$$

Solving for lower bound on optimal X:

$$\begin{aligned} \frac{\partial \mathcal{L}_{\text{worst case}}}{\partial X} &= -\alpha + \beta \frac{\exp(X + \delta - 2)}{1 - \exp(X + \delta - 2)} = 0 \\ &\Rightarrow X^* = 2 - \delta + \log \frac{\alpha}{\alpha + \beta} \end{aligned}$$

'Proof' of `fuzzy-and` network

Dataset is reweighted so that $\alpha = \frac{1}{2}$. Thus, $\beta \in [0, \frac{1}{2}]$.



X is restricted to be in range $(0, 1)$, thus minimum of $\mathcal{L}$ is achieved for X as close to 1 as possible, hence **subnetwork is unbiased**.

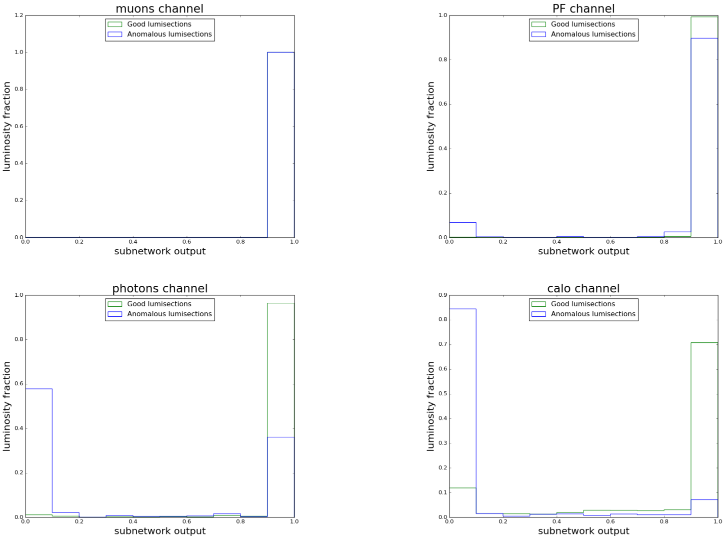
'Proof' of `fuzzy-and` network

To summarize, each subnetwork return score:

- close to 1 for normal lumisections;
- close to 1 for anomalies 'invisible' from subnetwork's channel;
- close to 0 for anomalies 'visible' from subnetwork's channel.

Thus, whole network 'decompose anomalies by channels'.

Results



muons channel
PF channel
photons channel
calo channel
Good lumisections
Anomalous lumisections
luminosity fraction
subnetwork output

Discussion

- results heavily rely on the assumptions;
- the only evidence for assumption 1 is good quality of the whole network;
- thus, estimated by this method amount of data that can be saved is only an upper bound;
- assumption 2 is quite reasonable by itself;
- results for different decision rules are also present (see github);
- results from all methods are consistent.

Summary

Summary

- two methods for upper bound for amount of data that can be saved from individual channels.
- all methods suggest that:
  - a lot of muonic data can be saved;
  - some of photon and PF data can be saved;
  - most of anomalies are caused (or at least, best detected) by calo channel.

Summary

Results can be found on `cms-dqm` git repository:

- unsupervised approach:
  - `https://github.com/yandexdataschool/cms-dqm/blob/master/notebooks/CMS-AE.ipynb`
- supervised approach:
  - `https://github.com/yandexdataschool/cms-dqm/blob/master/notebooks/CMS-NN-logreg.ipynb`
  - `https://github.com/yandexdataschool/cms-dqm/blob/master/notebooks/CMS-NN-minpool.ipynb`
  - `https://github.com/yandexdataschool/cms-dqm/blob/master/notebooks/CMS-NN-fuzzyand.ipynb`