NATIONAL RESEARCH UNIVERSITY
HIGHER SCHOOL OF ECONOMICS

as a manuscript

Maksim Alexandrovich Borisiak

MACHINE LEARNING METHODS FOR DATA QUALITY MONITORING IN NATURAL SCIENCES

PhD Dissertation Summary

for the purpose of obtaining academic degree
Doctor of Philosophy in Computer Science

Moscow — 2020

The PhD dissertation was prepared at National Research University Higher School of Economics.

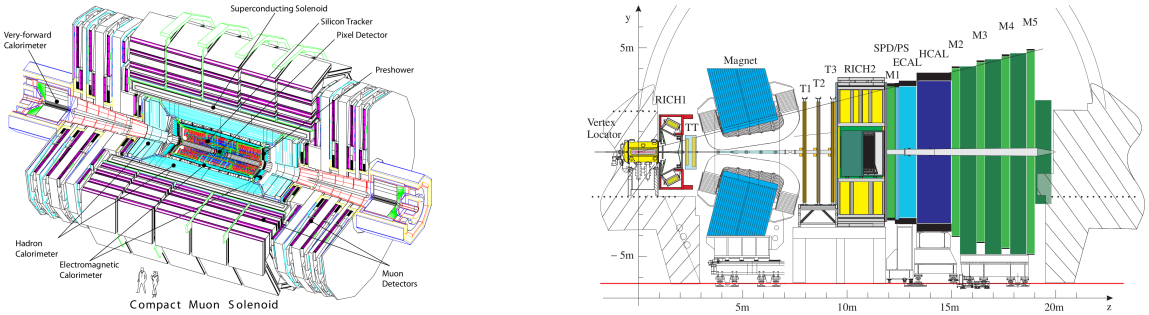
Academic Supervisor: Andrey E. Ustyuzhanin, Candidate of Sciences, Associate Professor at Joint Department with Yandex, Big Data and Information Retrieval School, Head of Laboratory of Methods for Big Data Analysis.

Contents

1 Introduction

Dissertation relevance. Data acquisition and data processing are essential steps in all scientific experiments. In many areas of natural sciences, modern experiments increasingly rely on complex detectors and automated processing pipelines. For instance, in High Energy Physics (HEP) and astrophysics, data gathering and processing, at least, its initial stages, are performed solely in an automatic manner, that involve large computing farms — Large Hadron Collider is capable of producing millions of events per second, each of which requires complex analysis and must be processed immediately [1, 2], modern observatories rely on a large number of detectors and produce significant amounts of data, e.g., the Square Kilometre Array [3] employs computing farms with around 100 PFLOPS of processing power [4].

Data collected in modern experiments are complex and often involve thousands or more dimensions. Figure 1 demonstrates structure of some LHC detectors, for example, CERN CMS detector [1] consists of multiple subdetectors, each employing complicated electronics and software; the typical raw size of an event in the detector is around 0.5 Mb [2] with the event rate exceeding 1 GHz. The Square Kilometre Array radio telescope employs more than 250 000 dual dipole antennas, which produce more than 2.5 Pb/s of raw data [5]. Machine Learning, with its profound ability to efficiently handle complex data, became an essential tool in data processing [6–12].



(a) CERN CMS [1]. (b) CERN LHCb [13].

Figure 1: Examples of High Energy Physics detector layouts.

Data quality monitoring (DQM) is an integral part of data acquisition. The main goal of DQM is to verify the validity of the collected data, i.e., ensuring that data are collected under the nominal conditions determined by the experiment. In this work, deviations from these nominal conditions are referred

to as anomalies and include human errors, detector malfunctions [14, 15], and external events, such as seismic activity [16] or even clouds [17, 18]. Not accounting for such abnormal states of operation leads to corrupted data, which, in turn, might alter conclusions of the experiment or even lead to false discoveries[1] completely undermining the primary purpose of the experiment [22]. For instance, the Laser Interferometer Gravitational-Wave Observatory [16] uses an extremely sensitive optical setup; therefore, it has to account for various types of noise, including environmental ones [23], in addition to "glitches" in the setup [24]. In geoscience, man-made objects can alter results of hyper-spectral imaging complicating analysis of soil composition [25]. Data quality monitoring extends beyond natural sciences; for example, in medicine, various artifacts present in MR spectroscopic images interfere with the automatic processing of these images, leading to unreliable diagnoses [26]. In climatology, unsuitable configuration, poor maintenance of observation stations, instrument misreading, inaccurate data digitization, and post-processing were identified as causes of misleading and erroneous results [27]. As with data processing in general, data quality monitoring increasingly relies on Machine Learning methods since accounting for anomalous behavior can only increase the complexity of data analysis [14, 24, 28–30].

In practice, data quality monitoring is often split into two tasks: online and offline DQM. Online DQM tends to focus on anomalies associated with the machinery and operates on raw or minimally processed data [14, 15, 31, 32]. The structure of data depends on the detector and varies significantly from one experiment to another. Offline DQM checks more subtle irregularities, including inspection of results of data processing pipelines [30, 33, 34]. Offline DQM typically analyses processed and aggregated data[2]. This division, however, is not strict, and some experiments might employ additional stages.

Moreover, a similar task is considered — discovery of differences between observations/experimental data and expected outcomes/theory. From the perspective of Machine Learning, such a task is the same as DQM, since disagreement between observations and a theory is an anomaly with respect to the

[1]For example, claims by the OPERA experiment [19] about neutrinos traveling at superluminal speed [20] were explained by instrumental sources afterward [21].

[2]One of the popular methods in offline DQM is to compare estimates of well-known quantities against their nominal values [22, 35].

theoretical predictions. For example, observations of the Higgs boson [36, 37] are an irregularity in the invariant mass distribution of so-called background (predictions of the best theoretical model not accounting for the Higgs boson). Black-hole mergers [38] are observed as oscillations that are unexpectedly strong under the background-noise model. Machine Learning becomes especially relevant for searching anomalies without a concrete underlying hypothesis, e.g., search for new physics [39–45].

Search for disagreements between theory and observations is usually treated separately from DQM due to differences in nature of the causes of deviations and different levels of data processing[3]. As the primary concern of this work is Machine Learning methods, we do not make distinctions between data quality monitoring and search for disagreements between theory and observations, treating both tasks as anomaly detection problems [40–45].

Terminology. In this work, any state of operation that deviates from the nominal conditions determined by the experiment is referred to as an anomalous state, and data observed during such state — as anomalous or, simply, an anomaly. Additionally, we consider any discrepancy between observations and theoretical predictions as an anomaly.

Note that this terminology is slightly different from definitions used in areas of Machine Learning, such as Outlier Detection. The latter defines anomalies or outliers as observations that appear to be inconsistent with the remainder of the set in which it occurs [46, 47], in other words, outliers are significantly different from normal samples by definition. However, in the case of data quality monitoring, anomalous status is not defined relative to normal data but by the state of operation, including the state of the detector and the environment. Thus, while anomalous states are quite likely to produce observations that are significantly different from normal data, i.e., outliers, they can also be potentially indistinguishable from normal samples. For example, the CERN LHCb experiment employs an array of silicon microstrips that registers energetic particles passing through [48]: if a portion of these strips becomes unresponsive, observations might still be consistent with observations obtained under the nominal

[3]For example, in Higgs boson analysis numerous events, each containing around 0.5 Mb of information, are reduced into several one-dimensional histograms [36]. At the same time, monitoring of the same detector operates with much more granular data [14].

conditions because, in some rare but possible events, particle trajectories do not intersect these unresponsive strips.

To avoid ambiguity, when it is not clear from the context, we refer to the task of detecting anomalies as defined in the previous paragraph simply as anomaly detection, including both: anomaly detection in data quality monitoring and search for disagreements between theory and observations.

Object and goals of the dissertation. The main difficulty behind data quality monitoring lies in the properties of anomalous data. Some anomalies might not be distinguishable from normal samples, especially considering that data quality monitoring is often performed on a reduced set of measurements (features in Machine Learning terminology) or over a set of aggregated statistics [27, 30, 34]. It is essential that DQM algorithms account for such cases by assigning proper class probability estimates or scores lower than those for unambiguously normal data. Moreover, it is often possible to label such data correctly upon examining additional information. This difficulty is especially pronounced when searching deviations from theoretical predictions as discrepancies are expected to be minor [39].

Additionally, some types of anomalies or alternative hypothesis might be known in advance, and, therefore, must be accounted for to address the previous issue regarding ambiguous samples adequately [30]. At the same time, even if a sample of anomalies is available, it is often not possible to assume that this sample is statistically representative, as taking into account all sources of anomalous behavior is impossible in practice [47]. Thus, anomaly detection algorithms should be robust to novel types of anomalies when it is possible. From the perspective of Machine Learning, this often puts anomaly detection problems into the limbo between supervised and unsupervised learning [49].

As was mentioned above, due to the nature of anomalies, data quality monitoring systems tend to operate on raw or minimally processed data. Many modern detectors have a unique setup and, thus, a unique structure of the collected data. It leads to another practically important task — collecting data for training anomaly detection algorithms. Two potential approaches can be employed:

- manual labeling;

- automatic sample generation, most often, by means of computer simulations;

or a combination of both.

The first approach often requires a large amount of manual labor [30, 34]; thus, algorithms capable of assisting experts are often desirable. Such algorithms can perform a significant portion of the work, thus allowing either to reduce costs of manual labeling or to increase the number of labeled samples.

The second approach exploits the fact that a large number of experiments, especially in natural sciences, employ computer simulations [50–55]. Such simulators are usually based on physics laws expressed in a computational form like differential or stochastic equations. Those equations relate input or initial conditions to the observable quantities under conditions of parameters that define physics laws, geometry, or other valuable property of the simulation. Computer simulations are capable of producing vast numbers of examples of nominal behavior (and, potentially, simulate some known instances of abnormal behavior), which can be used for training anomaly detection algorithms. Computer simulations are especially relevant for searching for minor differences between theoretical predictions (in this case, outputs of the simulation) and observations [42, 45].

Nevertheless, parameters of these simulations often require fine-tuning — search for parameters such that outputs of the simulation match values observed in practice [56–58]. The major challenge of fine-tuning computer simulations is computational cost as often fine-tuning procedures require large sample sizes, while computer simulations tend to be computationally demanding [59].

The goal of this dissertation is to develop Machine Learning algorithms to address major tasks of data quality monitoring and anomaly detection, namely:

- data collection:
  - reducing human labor;
  - assisting manual labeling;
  - fine-tuning of computer simulations;
- anomaly detection that takes into account known anomalies.

In order to achieve these goals, the following stages have to be completed:

- demonstrating that Machine Learning methods can be successfully applied for assisting manual labeling in DQM settings and evaluating these methods on data from large experimental setups;
- developing methods for assisting manual analysis of anomalous samples and evaluating these methods on data from large experimental setups;
- reducing computational costs of general-purpose fine-tuning methods;
- developing anomaly detection methods that combine properties of binary and one-class classification approaches and comparing their performance to that of state-of-the-art algorithms.

Figure 2 depicts relations between methods considered in this work.

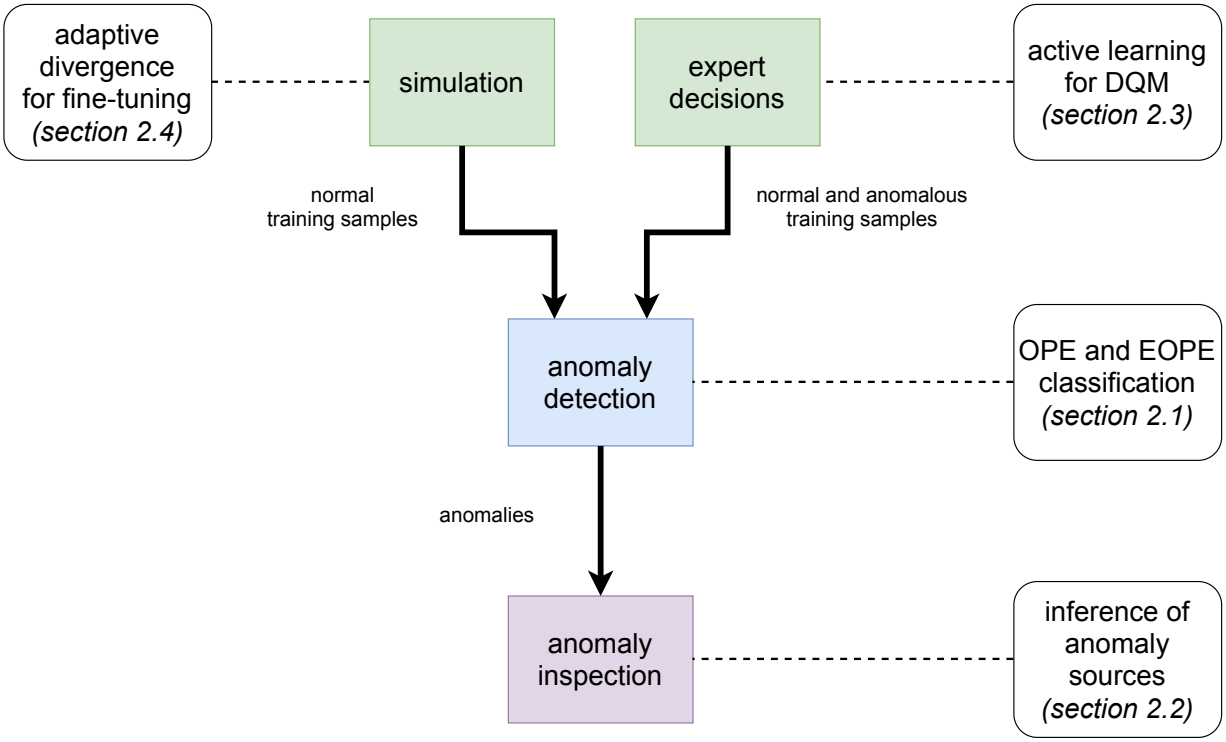


Figure 2: Main steps of data quality monitoring systems and corresponding contributions.

Structure of the dissertation. The second chapter provides a detailed overview of the main results. In Section 2.1, anomaly detection algorithms are considered, and the author introduces a novel family of general-purpose anomaly detection methods capable of operating under constraints and assumptions that are frequently imposed by DQM. First, it is argued that current

state-of-the-art Machine Learning methods do not adequately address the most common case of DQM: a large, statistically representative set of nominal examples and either non-representative or small set of anomalous samples. A family of methods is introduced to combine the main features of two-class and one-class classification methods. Proposed methods cover a wide range of problems: traditional binary classification, traditional one-class classification, and the intermediate cases, including highly imbalanced classification problems, making it perfectly suitable for DQM-related problems. Additionally, the proposed methods' main properties are strictly proven, and their performance is evaluated on a number of popular benchmark data sets. This contribution corresponds to the "anomaly detection" step in Figure 2.

In Section 2.2, the author proposes a novel Deep Learning algorithm that, under some assumptions, infers sources of anomalies, e.g., can point to a particular subsystem that displays faulty behavior. The main advantage of the proposed method is that it does not require labels for each subsystem and relies only on global labels, i.e., does not need any additional preparations for training. Such an algorithm further improves the quality of DQM as these algorithms assist investigations into the potential causes of anomalies. This contribution corresponds to the "anomaly inspection" step in Figure 2.

Section 2.3 considers manual data labeling for training anomaly detection algorithms, and an active learning algorithm for assisting experts is introduced. The proposed algorithm gradually learns on the manually labeled data and makes automatic decisions for samples similar to those with an expert label. The performance of the method is evaluated on a real case that involves DQM data from the CERN CMS experiment. This contribution corresponds to the "expert decisions" step in Figure 2.

Section 2.4 is dedicated to the major issue behind the automatic generation of nominal samples through computer simulations — fine-tuning of the simulations. The attention is focused on the high computational costs of fine-tuning procedures. The author introduces a novel family of adaptive divergences, and a novel class of fine-tuning algorithms based on these divergences formulated explicitly to reduce the computational burden. The performance of the proposed methods is evaluated on various tasks, including a realistic example with Pythia event generator. This contribution corresponds to the "simulation" step

in Figure 2.

Related work. Anomaly detection is the cornerstone of data quality monitoring. Current state-of-the-art methods can be divided into three categories, namely, supervised and unsupervised approaches and learning from positive and unlabeled data (PU learning).

Supervised approaches consider anomaly detection as a binary classification problem[4]. Such methods demonstrate good performance in cases with relatively frequent anomalies [14,28,30]. However, as shown in our recent work [49], binary classification methods are unreliable when supplied with small or unrepresentative training samples.

Unsupervised one-class classification methods [60–64] are widely used for anomaly detection when anomalies are rare or available training samples are not representative, i.e., do not cover the whole range of possible anomalies. Some unsupervised methods are based on reconstruction error [30, 63] with the main idea that a model trained to reconstruct normal samples is unlikely to properly reconstruct anomalies, especially, if the model is trained as a generative one [65–67]. Other one-class classification methods make use of restricted classifiers [60–62, 64]. Support Vector Data Description (SVDD [68]) and a related method, one-class Support Vector Machine (one-class SVM [62]), employ a soft-margin objective similar to that of conventional SVM but additionally minimize the area classified as the normal class. As for all kernel-based methods, the major downside of SVDD and one-class SVM is their high computational complexity [69], which makes them impractical to train on large data sets[5]. Several anomaly detection methods are based on similar ideas: Deep SVDD [61] employs a severely restricted neural network to learn a non-trivial basis for the linear (non-kernel) version of SVDD, likewise, one-class Neural Network [60] learns the basis by training an auto-encoder. Methods based on decision trees [70] employ heuristics associated with decision tree training procedures, and, like all decision-tree based algorithms, struggle in cases with a

[4]In some cases, the anomalous class is divided into several classes (see, for example, [29]), which, technically, results in a multiclass classification problem. This work focuses only on two classes: normal and anomalous. Nevertheless, our methods can be easily adapted to multiclass cases by introducing an additional classifier for anomalous instances.

[5]For instance, two benchmark data sets considered in our work [49] contain more than 10^6 samples.

high degree of dependencies between features (a relevant comparison can be found in [60, 61, 71]).

One-class classification methods tend to show good performance on data sets with non-overlapping or insignificantly overlapping classes. However, the main disadvantage of one-class classification methods for anomaly detection is that they ignore available anomalous samples; thus, they are unable to make reliable predictions in cases when supports of classes are significantly overlapping, labeling ambiguous samples as normal ones.

Learning from positive and unlabeled data [72] is a field closely related to anomaly detection. The problem statement of PU learning is somewhat similar to that of DQM — binary classification with labeled positive samples and an unlabeled mixture of negative and positive samples. However, there are substantial differences between OPE and PU learning settings: this dissertation focuses primarily on the case of a non-representative anomalous sample rather than on incomplete label information; nevertheless, some analogies might be drawn. Most notably, some PU learning approaches consider unlabeled part of the data set as the negative class, which resembles 'one against everything' approach considered in this work [73, 74].

Another primary task of data quality monitoring is the analysis of anomalies. In this dissertation, the author considers determining the origin of anomalies, i.e., identifying subsystems that display faulty behavior. Generally, such tasks are in the domain of causal inference; a comprehensive overview of causal inference can be found in [75]). As noted in the overview: "behind every causal conclusion there must lie some causal assumption that is not testable in observational studies." To the best of the author's knowledge, assumptions considered in this dissertation are not addressed anywhere else in the literature, mostly because these assumptions include the absence of subsystem-level labels.

The third primary DQM-related task is collecting training data for anomaly detection algorithms, for which two approaches are considered: manual labeling and the use of computer simulations. The problem of minimization of human labor in manual data labeling belongs to the domain of active learning — an area of Machine Learning concerned with training on a stream of data or with expert feedback. Active learning considers a wide range of problem statements varying by, e.g., available sampling procedures or underlying data model [76].

A general overview of active learning can be found in [76]. In the context of data quality monitoring, the most relevant approach is the so-called minimization of data collection. The core idea behind this technique is to make decisions for unambiguous samples automatically, request expert labels for others, subsequently updating the model [77]. The ambiguity of a sample is determent by various heuristics, e.g., measuring disagreement of a committee of classifier [78], by using "conflict" and "ignorance" metrics [79] or employing fuzzy classifiers [80].

Computer simulations often require adjustment of their parameters to a particular experimental setup, i.e., fine-tuning. Fine-tuning methods can be split into several categories. The first category employs heuristics for matching ground-truth distributions and output of the simulation [56, 57, 81]. The major drawback of these methods is the need for special features that are carefully constructed to satisfy assumptions behind a particular heuristic, which might not always be possible in practice. The second category is closely related to generative models, in particular, to Generative Adversarial Networks [82], and likelihood-free inference [58,83–86]. This category includes general-purpose methods, which can be applied practically to any simulation. However, these methods generally rely on adversarial learning [58] or similar approaches [83], which makes them computationally expensive. To the best of our knowledge, our work [87] is the first one that explicitly addresses the computational complexity of fine-tuning methods, in particular, for cases with non-differentiable computationally heavy simulations.

Scientific novelty. The main contributions of this dissertation are the following.

- A novel family of algorithms for anomaly detection is introduced. Unlike traditional one-class classification methods, proposed methods combine properties of two-class and one-class methods and are capable of addressing problems under a wide range of assumptions on the nature of anomalies.

- A novel method for inferring sources of anomalies is introduced and evaluated on data from a large experimental setup, namely, the CERN CMS experiment. The algorithm relies on assumptions that are often met for

DQM systems and do not require additional subsystem-level labels for training.

- The considered active learning approach for assisting manual labeling is demonstrated to significantly reduce the amount of human labor on data from a large experimental setup, namely, the CERN CMS experiment;
- A novel family of divergences is introduced, allowing for a significant acceleration of fine-tuning procedures with respect to the number of calls to the target simulation.

It should also be noted that the main results of this work can be applied or easily adopted to settings outside DQM.

- Novel anomaly detection methods introduced in this dissertation, namely $(1 + \varepsilon)$-class classification, are general-purpose methods designed to address a wide range of problems, for instance, they can be easily adapted for tasks outside DQM, for training on imbalanced data sets [49] or for increasing robustness of classification methods [88].
- The proposed method for inferring sources of anomalies is a general-purpose method that relies on assumptions non-specific to DQM and can be applied in industrial settings that are consistent with these assumptions.
- Adaptive divergences are not inherently dependent on the absence of gradient information or computational complexity of simulation; thus, they can be employed in general-purpose adversarial learning — consider, for instance, [89–91].

Practical value. The results of this work are directly applicable to data quality monitoring systems and allow for:

- improving quality of anomaly detection by taking into account known anomalous samples;
- solving a wide range of anomaly detection problems;
- automatic assistance in analyzing anomalies;

- significant reduction in computational costs of fine-tuning algorithms.
- considerable reduction of human labor required for manual data quality monitoring systems;

Methodology and research methods. The research methods involved probability and statistics, functional analysis, application and analysis of Machine Learning methods, knowledge of Machine Learning methods in particle physics, and astrophysics. The algorithms were developed in Python programming language [92], using numpy [93], scipy [94], scikit-learn [95], tensorflow [96], pytorch [97] and many other packages. All numerical experiments are reproducible, and the code of the experiments is available publicly; references are provided in the corresponding works.

Publications and approbation of the research. The results presented in this dissertation are based on the following publications.

First-tier publications:

- (1 + epsilon)-class Classification: an Anomaly Detection Method for Highly Imbalanced or Incomplete Data Sets / M. Borisyak, A. Ryzhikov, A. Ustyuzhanin, D. Derkach, F. Ratnikov, O. Mineeva // Journal of Machine Learning Research. — 2020. — Vol. 21, no. 72. — P. 1–22. (Scopus Q1);
  Contributions of the dissertation's author: combination of main properties of binary and one-class classification methods and the corresponding loss function, derivation of energy approximation of the loss function, theoretical proofs for asymptotic cases, efficient training algorithms, experimental studies on various benchmark data sets. The dissertation's author is the main author of the publication.
- Adaptive divergence for rapid adversarial optimization / M. Borisyak, T. Gaintseva, A. Ustyuzhanin // PeerJ Computer Science. — 2020. — May. — Vol. 6. — P. e274. (Scopus Q1);
  Contributions of the dissertation's author: introduction of the adaptive divergences and formulation of several instances of adaptive divergences, efficient training algorithms for several widely used classification models,

theoretical proofs, experimental studies for several realistic scenarios. The dissertation's author is the main author of the publication.

Second-tier publications:

- Deep learning for inferring cause of data anomalies / V. Azzolini, M. Borisyak, G. Cerminara, D. Derkach, G. Franzoni, F. De Guio, O. Koval, M. Pierini, A. Pol, F. Ratnikov, F. Siroky, A. Ustyuzhanin, J-R. Vlimant. // Journal of Physics: Conference Series. — 2018. — sep. — Vol. 1085. — P. 042015. (Scopus Q3);
  Contributions of the dissertation's author: introduction of the loss function for the "fuzzy-and" network, theoretical proof, the preliminary experimental study on a CERN CMS data set.

- Towards automation of data quality system for CERN CMS experiment / M. Borisyak, F. Ratnikov, D. Derkach, A. Ustyuzhanin // Journal of Physics: Conference Series. — 2017. — oct. — Vol. 898. — P. 092041. (Scopus Q3).
  Contributions of the dissertation's author: application of active learning to data quality monitoring systems, the experimental study on a CERN CMS data set. The dissertation's author is the main author of the publication.

2 Main results

2.1 Anomaly detection

Anomaly detection algorithms are at the core of any data quality monitoring system. As was discussed above, anomaly here means any deviation from the nominal conditions of the experiment. Note that this definition is different from the one that is the most widely used in unsupervised learning. The key difference is that an anomaly here can be potentially represented by the same feature vector as some nominal state; in other words, distributions of anomalous and nominal conditions potentially have overlapping supports.

In our recent work [49], we argue that such relaxed assumptions are more relevant for practical settings than the traditional ones [47]. For instance, data

quality monitoring is often performed on a reduced set of observable quantities, for example, in another work [34], anomaly detection algorithm receives several statistics aggregated over a considerable number of events, which makes anomalies present only in a few events practically indistinguishable from an unlikely but possible series of events under nominal conditions. Additionally, in some instances, conditions of the experiment are not fully observed, which potentially leads to some anomalies producing the same values for observable quantities as under some nominal conditions. For example, the CERN LHCb tracker consists of a large number of silicon microstrips that record particles traveling through; if a small group of these strips becomes unresponsive (which is an anomaly), readings from the tracker might be consistent with some rare but possible events, e.g., with ones that do not interact with this group of microstrips.

Nominal conditions of experiments in natural sciences are often defined by a narrow set of states. Moreover, examples of nominal behavior are plentiful, while anomalies are relatively rare; for instance, the CERN CMS experiment reports 2% of anomalous samples [30]. Combined with the previous statements, this leads to the following assumptions under which we consider generalized anomaly detection problems:

- the distribution of anomalies might not be perfectly separable from the distribution of nominal states;
- some types of anomalies might not be present in the training sample.

Under settings described above, it is crucial to correctly identify ambiguous instances as data quality monitoring policy might require an additional examination of such cases, e.g., taking into account more detailed information or delegating the decision to an expert.

Here, we consider neural networks as the primary model for our methods. Among other popular methods, algorithms based on decision trees are known to struggle in cases with a high degree of dependencies between features (a relevant comparison can be found in [60, 61, 71]), which is a typical case in DQM settings; SVM-based methods [98] are practically challenging to apply due to their high computational complexity [69].

Technically, the anomaly detection problem, as stated above, is a classification problem. Let $\mathcal{X}$ be a Banach space representing space of possible observable states of the experiment. A Bayes optimal classifier is given by $f^* : \mathcal{X} \to [0, 1]$:

$$f^*(x) = \frac{P(x \mid \mathcal{C}^+)P(\mathcal{C}^+)}{P(x \mid \mathcal{C}^+)P(\mathcal{C}^+) + P(x \mid \mathcal{C}^-)P(\mathcal{C}^-)}; \tag{1}$$

where $\mathcal{C}^+$, $\mathcal{C}^-$ denote nominal and anomalous classes and (with the abuse of notation) corresponding posterior distributions.

Traditionally, binary classifiers are trained by minimizing cross-entropy loss function:

$$\mathcal{L}_2(f) = -P(\mathcal{C}^+) \mathop{\mathbb{E}}_{x\sim\mathcal{C}^+} \log f(x) - P(\mathcal{C}^-) \mathop{\mathbb{E}}_{x\sim\mathcal{C}^-} \log\left(1 - f(x)\right); \tag{2}$$

where $\mathbb{E}_{x\sim\mathcal{C}}$ denotes conditional average — $\mathbb{E}_x\left[\cdot \mid \mathcal{C}\right]$.

Notice, that optimal classifier (1) is undefined outside $\operatorname{supp}\mathcal{C}^+ \cup \operatorname{supp}\mathcal{C}^-$, and, in general, a function f^* that minimizes $\mathcal{L}_2$ can take any value for $x \notin \operatorname{supp}\mathcal{C}^+ \cup \operatorname{supp}\mathcal{C}^-$. In case of limited sample sizes, this leads to a complete lack of guarantees on outputs of a trained classifier in areas not covered by the training set. Predictions in such areas might depend on a particular architecture of the neural network, initial weights of the network, or even on a particular sequence of mini-batches.

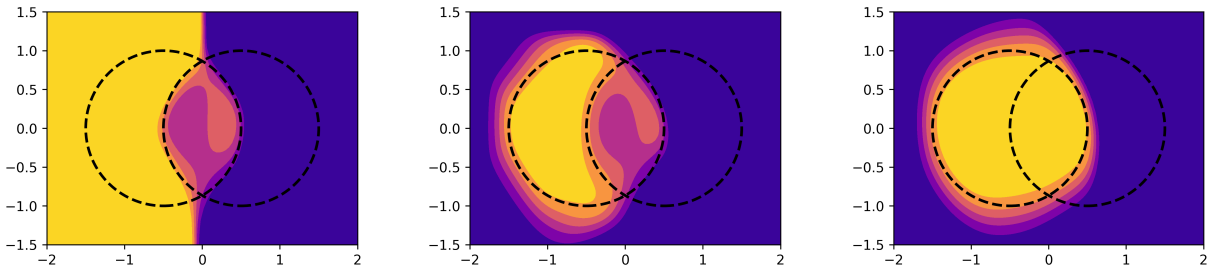
This behavior goes against our assumptions that some types of anomalies might not be present in the training set and, therefore, any $x \notin \mathcal{C}^+$ must be classified as an anomaly, i.e., $f^*(x) = 0$.

In work [49] we propose adding a uniform distribution U to the anomalous class, where $\operatorname{supp} U \subseteq \mathcal{X}$ is a compact set that covers supports of both original classes. In this case, loss function (2) becomes:

$$\begin{aligned}
\mathcal{L}_{1+\varepsilon}(f) &= \frac{1}{2}\left(L^+(f) + \gamma\, L^-(f) + (1-\varepsilon)\, L^0(f)\right); \\
L^+(f) &= -\mathop{\mathbb{E}}_{x\sim\mathcal{C}^+} \log f(x); \\
L^-(f) &= -\mathop{\mathbb{E}}_{x\sim\mathcal{C}^-} \log(1 - f(x)); \\
L^0(f) &= -\mathop{\mathbb{E}}_{x\sim U[\Omega]} \log(1 - f(x));
\end{aligned} \tag{3}$$

with the solution:

$$f^*_{1+\varepsilon}(x) = \frac{P(x \mid \mathcal{C}^+)}{P(x \mid \mathcal{C}^+) + (1-\varepsilon)\, C + \gamma\, P(x \mid \mathcal{C}^-)}; \tag{4}$$



(a) two-class classification (b) OPE classification (c) one-class classification.

Figure 3: Demonstration of the main idea behind OPE loss. Samples are uniformly distributed within areas bounded by the circles: the left one as positive class, the right one as negative. One-class solution was obtained by setting $\gamma = 1$, and $\varepsilon = 0$. Training samples are not shown for visual clarity.

where: $C = \text{const}$ is a probability density of the uniform distribution U, $\varepsilon \in [0, 1]$ regulates the impact of the regularization term L^0, and γ should be ideally such that

$$\gamma + (1 - \varepsilon) = 2 \cdot \frac{P(\mathcal{C}^-)}{P(\mathcal{C}^+)};$$

if the prior probabilities are known. We refer to loss function (3) as One Plus Epsilon loss or OPE loss.

It can be seen from Equation (4), that for $\varepsilon = 1$, optimal classifier recovers binary solution (1), while under $\varepsilon = 0$ and $\gamma = 0$, optimal classifier becomes a monotonous transformation of $P(x \mid \mathcal{C}^+)$ which corresponds to a one-class solution. Intermediate values of ε allow us to introduce desirable property of one-class classification, $f(x) = 0$ for $x \notin \operatorname{supp} \mathcal{C}^+$, into solution of the binary classification. In other words, regularization term L^0 introduces a bias towards a one-class solution, which makes the optimal classifier properly defined outside supports of both original classes, at the same time, for ε close to 1, has little impact otherwise. Figure 3 illustrates this phenomenon on a synthetic data set: while two-class classification correctly estimates probabilities in the intersection of two classes, it also assigns a positive label to instances outside the support of the positive class, at the same time, one-class classification provides correct predictions outside the support of the positive class, but completely ignores the negative class; a classifier trained with OPE loss provides correct predictions everywhere.

Algorithm 1 shows training procedure based on OPE loss, to which we

refer as brute-force OPE. While OPE loss theoretically leads to a solution with desirable properties, brute-force OPE shows the main weakness of this approach: mini-batch estimations of $\nabla L^0(f)$ can be extremely noisy if $\mathcal{X}$ is a high-dimensional space or if f is a high-capacity model, which is usually the case for neural networks. The variance of gradient estimations directly influences convergences of stochastic gradient optimization procedure, which might render regularization term L^0 practically ineffective.

Algorithm 1: Brute-force OPE

Input: normal data, anomalous data—samples from $\mathcal{C}^+$, $\mathcal{C}^-$, the latter might be absent; f_θ—a classifier with parameters θ.

Hyper-parameters: γ—the ratio of class priors; ε—the strength of the regularization.

while *not converged* **do**

- sample normal data $\{x_i^+ \sim \text{normal data}\}_{i=1}^m$;
- sample known anomalies $\{x_i^- \sim \text{anomalous data}\}_{i=1}^m$;
- sample pseudo-negative samples $\{x_i^0 \sim U[\Omega]\}_{i=1}^m$;
- $\nabla L^+ \leftarrow -\sum_i \nabla_\theta \log f_\theta(x_i^+)$;
- $\nabla L^- \leftarrow -\sum_i \nabla_\theta \log(1 - f_\theta(x_i^-))$;
- $\nabla L^0 \leftarrow -\sum_i \nabla_\theta \log(1 - f_\theta(x_i^0))$;
- $\theta \leftarrow \text{Adam}\left(\nabla L^+ + \gamma \nabla L^- + (1-\varepsilon)\nabla L^0\right)$

end

To combat high variance of gradient estimations for the regularization term, we propose a different regularization term:

$$\begin{aligned} L^E(g) &= \int_\Omega \exp(g(x))dx; \\ \text{where: } g(x) &= \sigma^{-1}(f(x)); \\ \sigma(\chi) &= \frac{1}{1+\exp(-\chi)}; \end{aligned}$$

to which we refer as energy regularization, and to the corresponding loss function as energy OPE or EOPE.

In work [49], we show that L^E leads to solutions with the same desirable properties, i.e. the following loss function:

$$\mathcal{L}_1^E(g) = \frac{1}{2}\left[\mathop{\mathbb{E}}_{x\sim\mathcal{C}^+} \log\left(1+\exp(-g(x))\right) + (1-\varepsilon)L^E(g)\right] \tag{5}$$

leads to a one-class solution.

More formally, this property is captured by the following theorem.

Theorem 1 *Let $(\mathcal{X}, \|\cdot\|)$ be a Banach space, $P(x)$—a continuous probability density function such that $\Omega = \operatorname{supp} P$ is an open set in $\mathcal{X}$. If continuous function $g^* : \Omega \to \mathbb{R}$ minimizes $\mathcal{L}_1^E$ (defined by Equation 5) with $P(x \mid \mathcal{C}^+) = P(x)$, then there exists a strictly increasing function $s : \mathbb{R} \to \mathbb{R}$, such that $g^*(x) = s(P(x))$. Moreover, $\lim_{y\to 0} s(y) = -\infty$ if $\inf_\Omega P = 0$.*

Proof of Theorem 1 can be found in the corresponding contribution [49].

The major advantage of L^E regularization is that, unlike L^0, gradients of L^E can be estimated much more precisely:

$$\nabla L^E(g) = \frac{1}{Z}\int_\Omega \exp(g(x))\nabla g(x) = \mathop{\mathbb{E}}_{x\sim P_g} \nabla g(x). \tag{6}$$

Equation (6) is based on a property that is widely used in application to energy models. Therefore, most of the algorithms for training energy models are also applicable here, most notably, contrastive divergence with Markov Chain Monte-Carlo sampling and Deep Directed Generated Networks [99]. This effectively introduces a family of methods. Algorithm 2 outlines the general procedure for training models with EOPE loss, Figures 4 and 5 demonstrate the results of the proposed OPE algorithms on a toy data set.

Additionally, we notice that EOPE is robust to imperfect sampling procedures, and propose an approximate but computationally cheap sampling procedure, that shows performance comparable to exact MCMC procedures.

Proposed algorithms were evaluated on a number of popular benchmark tasks, including natural images (MNIST, CIFAR-10, Omniglot), anomaly detection data sets (KDD-99), and High Energy Physics data sets (HIGGS, SUSY). All data sets were augmented to reflect considered anomaly detection problems: for multiclass problems, one of the classes was selected as normal, the rest were labeled as anomalous, and only some of the anomalous classes were present in the training data set. For binary classification problems, the number of anomalous samples used for training was varied. The performance of proposed algorithms was compared to several state-of-the-art one-class classification, semi-supervised, and traditional binary classification methods. The results are presented in Figures 6 – 8. Experiments indicate that proposed

Algorithm 2: Energy OPE

Input: normal data, anomalous data—samples from $\mathcal{C}^+$, $\mathcal{C}^-$, the latter might be absent; g_θ—a classifier with parameters θ.

Hyper-parameters: γ—the ratio of class priors; ε—the strength of the regularization; MCMC—a Monte-Carlo sampling procedure.

while *not converged* **do**

sample normal data $\{x_i^+ \sim \text{normal data}\}_{i=1}^m$;

sample known anomalies $\{x_i^- \sim \text{anomalous data}\}_{i=1}^m$;

sample pseudo-negative examples $\{x_i^0 \sim \text{MCMC}\left[x \mapsto \exp(g(x))\right]\}_{i=1}^m$;

$\nabla L^+ \leftarrow \sum_i \nabla_\theta \log(1 + \exp(-g_\theta(x_i^+))$;

$\nabla L^- \leftarrow \sum_i \nabla_\theta \log(1 + \exp(g_\theta(x_i^-)))$;

$\nabla L^E \leftarrow \sum_i \nabla_\theta g_\theta(x_i^0)$;

$\theta \leftarrow \text{Adam}\left(\nabla L^+ + \gamma \nabla L^- + (1 - \varepsilon)\nabla L^E\right)$

end

methods either outperform baseline methods or achieve comparable results. Firstly, as expected, the performance of OPE and EOPE methods improves with the addition of known negative samples and quickly approaches the performance of binary classification on full (balanced) datasets. Secondly, OPE and EOPE methods demonstrate the best relative performance for data set with significantly overlapping classes (Figures 6 and 7). Overall results show that OPE and EOPE are well suited for solving anomaly detection problems in data quality monitoring.

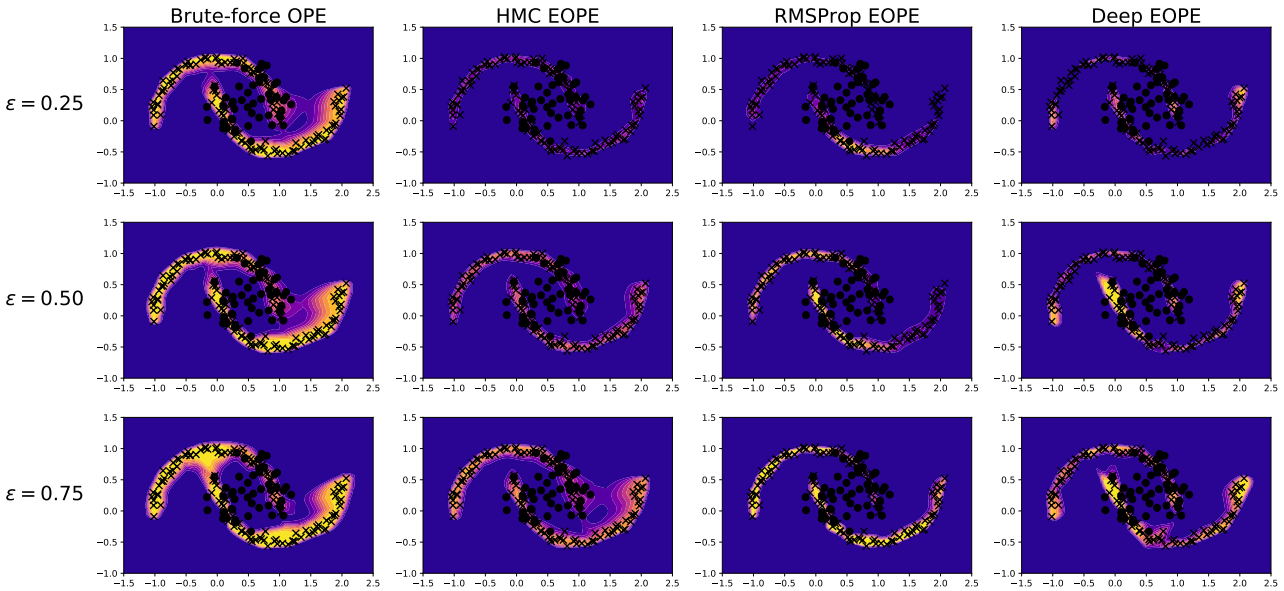
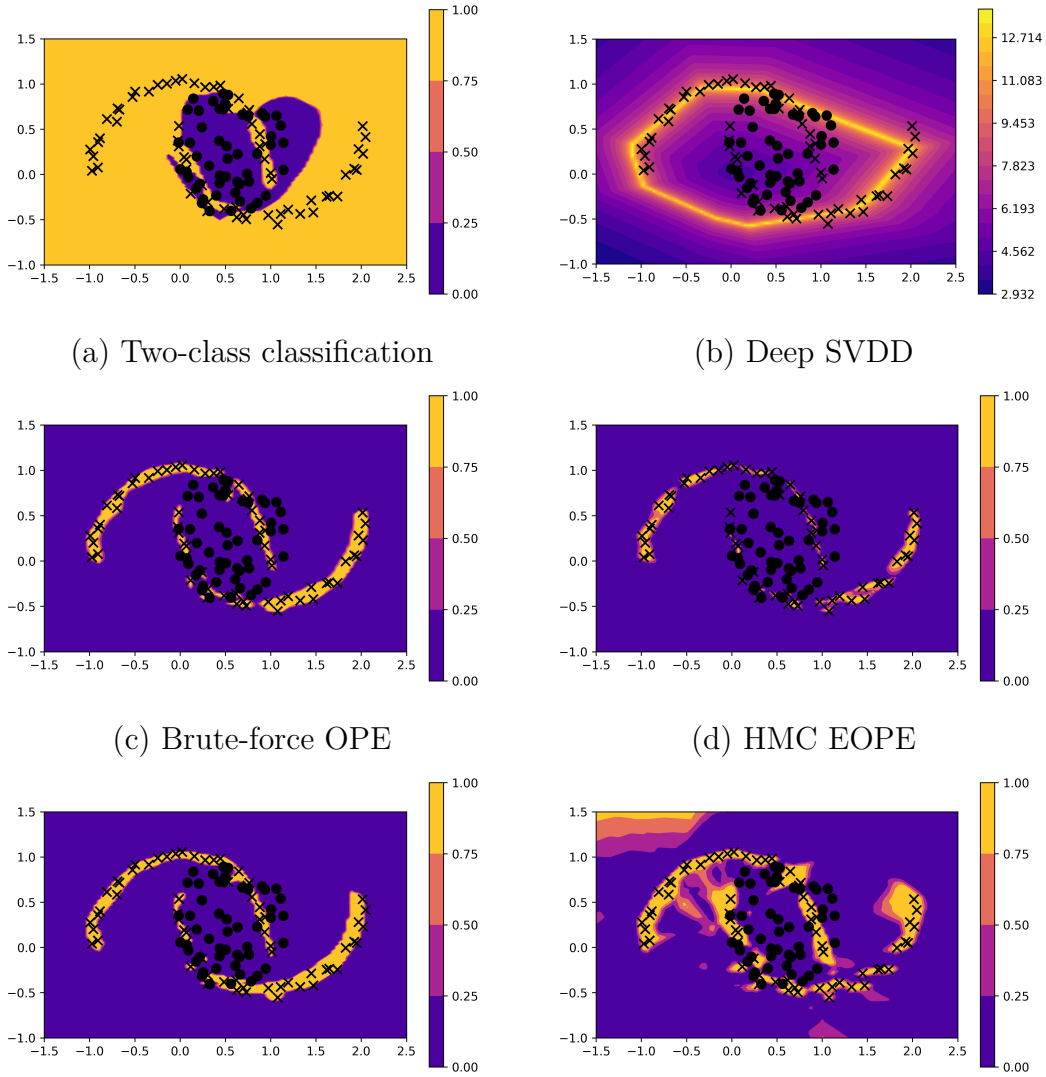


Figure 4: Comparison of OPE and EOPE losses with varying ε, and, for illustration purposes, $\gamma = 1 - \varepsilon$. For $\varepsilon < 1$, all losses lead to similar solutions. It appears that EOPE penalizes positive predictions stronger than OPE loss.



(a) Two-class classification

(b) Deep SVDD

(c) Brute-force OPE

(d) HMC EOPE

(e) RMSProp EOPE

(f) Deep EOPE

Figure 5: Comparison of different methods on a toy example: positive examples (marked as 'x') are sampled from the Moons data set, negative examples (marked by black circles) are sampled uniformly from a circle of radius $\frac{1}{2}$. For visual consistency negative logarithm of Deep SVDD output is displayed.

	one class	100	1000	10000	1000000
Robust AE	0.530 ± 0.002	0.530 ± 0.002	0.530 ± 0.002	0.530 ± 0.002	0.530 ± 0.002
Deep SVDD	0.497 ± 0.006	0.497 ± 0.006	0.497 ± 0.006	0.497 ± 0.006	0.497 ± 0.006
cross-entropy	-	0.496 ± 0.017	0.529 ± 0.007	0.566 ± 0.006	0.858 ± 0.002
semi-supervised	-	0.498 ± 0.003	0.522 ± 0.003	0.603 ± 0.002	0.745 ± 0.005
brute-force OPE	0.499 ± 0.009	0.500 ± 0.009	0.520 ± 0.003	0.572 ± 0.005	0.859 ± 0.001
HMC EOPE	0.491 ± 0.000	0.523 ± 0.005	$\mathbf{0.567} \pm 0.008$	$\mathbf{0.648} \pm 0.005$	0.848 ± 0.001
RMSProp EOPE	0.498 ± 0.002	0.494 ± 0.008	0.531 ± 0.008	0.593 ± 0.011	$\mathbf{0.861} \pm 0.000$
Deep EOPE	$\mathbf{0.531} \pm 0.000$	$\mathbf{0.537} \pm 0.011$	0.560 ± 0.008	0.628 ± 0.005	0.860 ± 0.001

Figure 6: Results on HIGGS data set. The first row indicates the number of anomalous samples used in training.

	one class	100	1000	10000	1000000
Robust AE	0.394 ± 0.012	0.394 ± 0.012	0.394 ± 0.012	0.394 ± 0.012	0.394 ± 0.012
Deep SVDD	0.541 ± 0.022	0.541 ± 0.022	0.541 ± 0.022	0.541 ± 0.022	0.541 ± 0.022
cross-entropy	-	0.658 ± 0.033	0.736 ± 0.021	0.757 ± 0.036	0.871 ± 0.006
semi-supervised	-	0.715 ± 0.020	0.766 ± 0.009	$\mathbf{0.847} \pm 0.002$	0.876 ± 0.000
brute-force OPE	$\mathbf{0.648} \pm 0.035$	0.678 ± 0.025	0.729 ± 0.029	0.757 ± 0.036	0.871 ± 0.006
HMC EOPE	0.472 ± 0.000	$\mathbf{0.738} \pm 0.019$	$\mathbf{0.770} \pm 0.012$	0.816 ± 0.006	0.877 ± 0.000
RMSProp EOPE	0.443 ± 0.038	0.714 ± 0.019	0.760 ± 0.016	0.807 ± 0.004	0.877 ± 0.000
Deep EOPE	0.468 ± 0.118	0.670 ± 0.054	0.746 ± 0.024	0.813 ± 0.003	$\mathbf{0.878} \pm 0.000$

Figure 7: Results on SUSY data set. The first row indicates the number of anomalous samples used in training.

	one class	1	2	4	8
Robust AE	$\mathbf{0.972} \pm 0.006$	$\mathbf{0.972} \pm 0.006$	$\mathbf{0.972} \pm 0.006$	$\mathbf{0.972} \pm 0.006$	$\mathbf{0.972} \pm 0.006$
Deep SVDD	0.939 ± 0.014	0.939 ± 0.014	0.939 ± 0.014	0.939 ± 0.014	0.939 ± 0.014
cross-entropy	-	0.571 ± 0.213	0.300 ± 0.182	0.687 ± 0.268	0.619 ± 0.257
semi-supervised	-	0.315 ± 0.258	0.469 ± 0.286	0.758 ± 0.171	0.865 ± 0.087
brute-force OPE	0.398 ± 0.108	0.667 ± 0.175	0.394 ± 0.261	0.737 ± 0.187	0.541 ± 0.257
HMC EOPE	0.786 ± 0.200	0.885 ± 0.152	0.919 ± 0.055	0.863 ± 0.094	0.958 ± 0.023
RMSProp EOPE	0.765 ± 0.216	0.824 ± 0.237	0.770 ± 0.213	0.941 ± 0.048	0.960 ± 0.021
Deep EOPE	0.602 ± 0.279	0.767 ± 0.245	0.548 ± 0.279	0.763 ± 0.217	0.786 ± 0.267

Figure 8: Results on KDD-99 data set. The first row indicates the number of original classes randomly selected as the anomalous class, at most 1000 examples are sampled from each original class.

2.2 Inference of anomaly sources

In most real-world applications, detection of an anomaly is followed by an investigation into the causes of the anomaly. In complex experiments, detectors typically consist of several subdetectors [1,15], each reading its own set of values, and it is useful to infer the subset of subdetectors affected by a particular anomaly [100].

Additionally, the following assumption can be safely made: measured values can be split into groups such that an anomaly affecting a subset of these groups does not affect values from the other groups. We refer to such groups as channels. Typically, each subdetector corresponds to its channel since an anomaly in a subdetector does not interfere with the operations of the others.

In the corresponding contribution [100], we consider the problem of inferring a subset of channels affected by a given anomaly. Additionally, we assume that channel-level labels, i.e., an indicator if an anomaly is affecting a particular channel, are not available, only global labels are given, i.e., an indicator of the presence of an anomaly in at least one unspecified channel.

For each individual channel we introduce a neural network and combine outputs of these network with the following activation function:

$$\varphi(x) = \exp\left(\sum_{j=1}^{n} f^j(x^j) - n\right); \tag{7}$$

where: $x^j \in \mathcal{X}^j$ — a feature vector corresponding to the j-th channel, $f^j : \mathcal{X}^j \mapsto [0,1]$ — network associated with the j-th channel.

Then the joint network is trained to minimize cross-entropy loss of φ with respect to global labels.

Theorem 2 *Given that fraction of anomalous samples is less than* $1/2$*, number of channels* $n \geq 4$ *and for each channel* $j \in \{1, \ldots, n\}$*:*

$$\operatorname{supp} P(x^j \mid A^j) \;\cap\; \operatorname{supp} P(x^j \mid \bar{A}^j) = \varnothing;$$

where A^j *and* $\bar{A}^j$ *denote presence and absence of anomalies affecting channel*

j, then a solution $\{g^j : \mathcal{X} \mapsto [0,1]\}_{i=1}^n$ *that minimizes cross-entropy loss:*

$$\mathcal{L}[\varphi] = -\frac{1}{N}\sum_{i}^{N}\left[y_i \log \varphi(x_i) + (1-y_i)\log(1-\varphi(x_i))\right]; \tag{8}$$

$$\varphi(x) = \exp\left(\sum_{j=1}^{n} f^j(x^j) - n\right); \tag{9}$$

decomposes anomalies into affected channels, i.e., for each channel j:

$$g^j(x^j) = \begin{cases} 1, & \textit{if } \bar{A}^j; \\ 0, & \textit{otherwise.} \end{cases} \tag{10}$$

Reference to a proof of the theorem can be found in the corresponding contribution [100].

Informally, Theorem 2 states that under particular assumptions, a trained network recovers subsystem-level labels. The intuition behind the proof is the following. In the case of normal samples, the loss is minimized when all neural networks output 1, similarly, in case of an anomaly, when networks corresponding to channels affected by the anomaly output 0. The most notable case, however, is when a network is presented with an anomaly that does not affect its channel: since channels are independent, such a case is indistinguishable from a normal one based on features from the channel. However, the activation function is constructed in such way that, under theorem's assumptions, the penalty for predicting 1 in such case is always offset by the gains from correct predictions in normal cases, thus, forcing the network to always predict 1 when no anomaly is present in its channel, even if the anomaly is present elsewhere.

The proposed method was evaluated on a manually labeled CERN CMS data set; the main quality metrics are presented in Figure 9. First, as expected, outputs of the networks are concentrated around 0 and 1, only in rare cases in between. Second, all networks correctly identify normal samples with high accuracy. Additionally, for a significant portion of anomalous samples, all networks report anomaly; however, it should be noted that predicting 1 for an anomalous sample does not necessarily indicate an error. In order to further assess the performance of the method, outputs of each network were valuated against labels for individual subdetectors (Figure 10): outputs of each network are highly predictive of anomalies in subdetectors related to its channel, at the

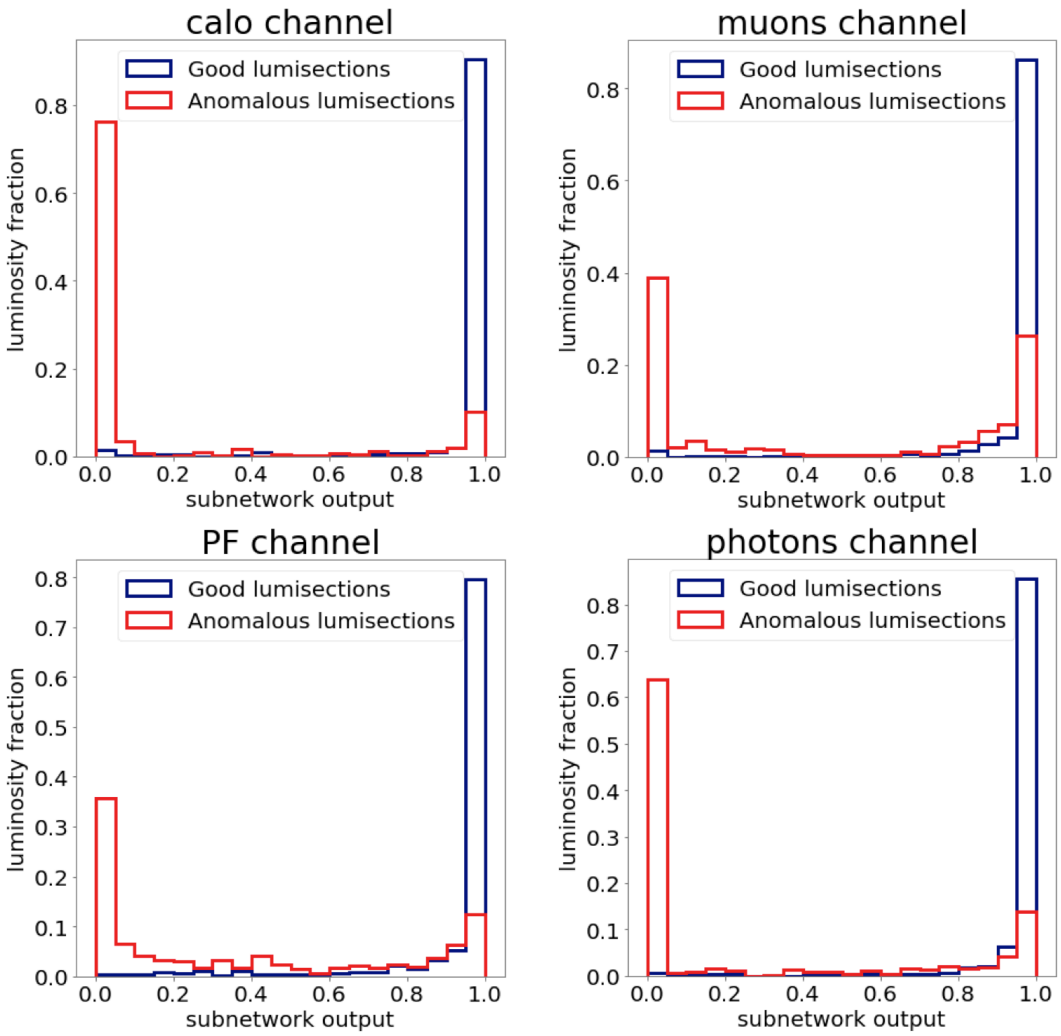


Figure 9: Results of the proposed method on CERN CMS data: data samples are referred to as lumisections, luminosity of a lumisection indicates importance of the latter.

same time, much less predictive or independent from anomalies in subdetectors unrelated to the corresponding channel. For instance, anomalies in ”muons” subdetector are strongly correlated with outputs of the network that corresponds to the ”muon” channel, and the correlation is significantly lower for other channels. Note that positive performance against subdetectors unrelated to a channel is expected as some anomalies affect multiple channels simultaneously. In general, outputs of the network are in agreement with the domain expertise.

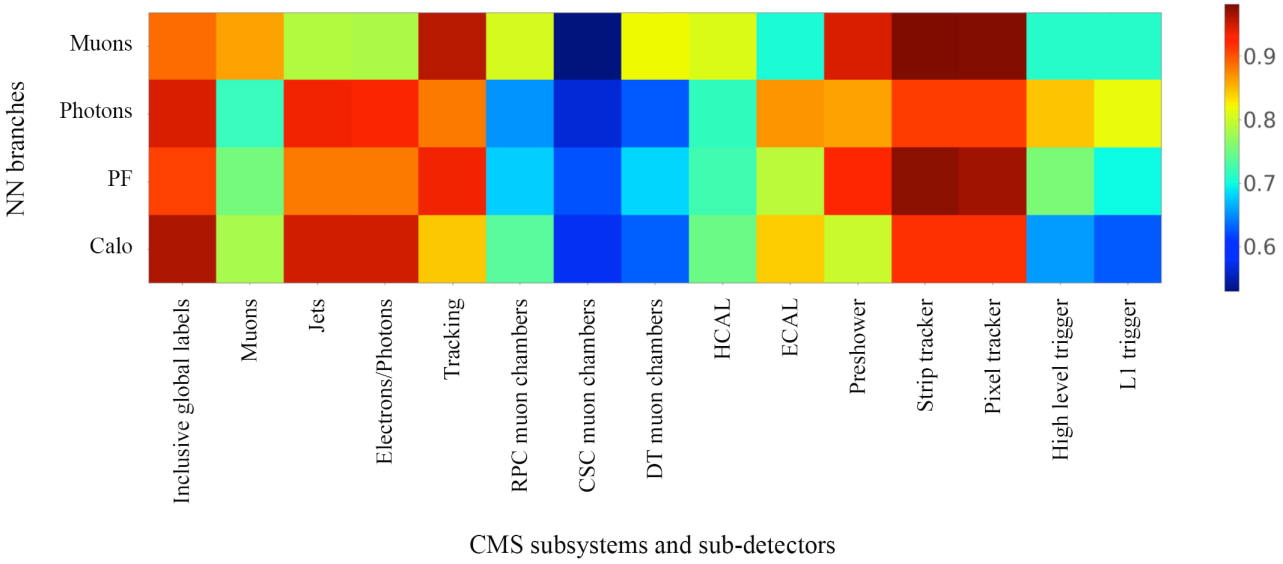


Figure 10: Results of the proposed method on CERN CMS data: rows represent data channels as defined above, columns — physical detector subsystems, colors indicate ROC AUC of the predictions for the corresponding channel against manual labels for the corresponding subsystem. Note that some subsystems might depend on data from multiple channels, other subsystems, or external data.

2.3 Manual labeling assistance

Methods proposed above rely on manually labeled data for training, and, due to high dimensionality, they typically require large training samples. In natural sciences, manual data labeling is often a non-trivial and labor-consuming task. For example, in the CERN LHCb experiment, an expert needs to check several dozens of histograms and plots before making a decision regarding a single sample [35].

To address this issue, we consider an active learning system that aims at assisting experts by making automated decisions when predictions are certain [77]. In cases when predictions are uncertain, the system delegates decision to the expert, the sample is appended to the training data set, and the underlying classifier is retrained. A prediction is certain if the score of the underlying classifier is higher than threshold τ_L ('certainly normal') or lower than τ_P ('certainly anomalous'), where the thresholds are determined by employing cross-validation and comparison against external constraints on acceptable pollution rate P_0 (fraction of anomalous samples among classified as certainly normal) and loss

Algorithm 3: Active learning system for manual labeling assistance.

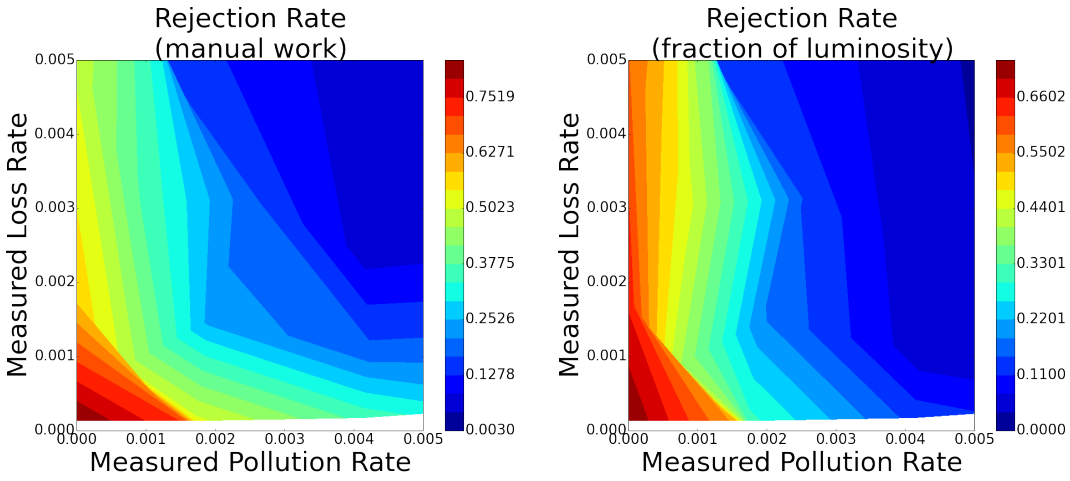
Input: $L_0 \in \mathbb{R}, P_0 \in \mathbb{R}$ — constraints on loss and pollution rates

$\tau_L, \tau_P \leftarrow 0, 1$;
classifier $\leftarrow (x \mapsto 1/2)$;
$X, Y \leftarrow \varnothing, \varnothing$;
for $i = 1, \ldots, N$ **do**
    $x_i \leftarrow$ new sample;
    $p_i \leftarrow$ classifier(x_i);
    if $p_i > \tau_L$ **then**
        automatically label x_i as normal sample;
    else if $p_i < \tau_P$ **then**
        automatically label x_i as anomalious sample;
    else
        $y_i \leftarrow$ request expert label;
        $X, Y \leftarrow (X, x_i), (Y, y_i)$;
        compute predictions P on X with k-fold cross-validation;
        `// thresholds for acceptable loss and pollution rates`
        $\tau_L \leftarrow \max\{\tau \mid \hat{L}_\tau(P, Y) \leq L_0\}$;
        $\tau_P \leftarrow \min\{\tau \mid \hat{P}_\tau(P, Y) \leq P_0\}$;
        retrain classifier;
    end
end

rate L_0 (fraction of normal samples classified as certainly anomalous) [78]. The procedure is outlined in Algorithm 3.

The system was evaluated on open CERN CMS data; results are shown in Figure 11: Notice, that under the most severe constraints (pollution and loss rates less than 10^{-4}) the system is able to save at least 20% of manual labor, and even a mild relaxation of the constraints (pollution and loss rates less than 10^{-3}) increases this quantity to more than 50%.

Note that the system learns from manually labeled data and gradually replaces experts; thus, performance improves over time, as demonstrated in Figure 12. On the first iterations, the system requests expert labels for most samples; however, as the size of the training set increases, the system's pre-



(a) Manual work.

(b) Rejected luminosity.

Figure 11: Performance of active learning on open CERN CMS data. Plots shows fraction of manually labeled samples (left) and luminosity (right) as functions of measured loss and pollution rates.

dictions become more reliable, which is reflected in gradually decreasing the number of requests for expert assistance.

In conclusion, this study demonstrates that methods for minimization of data collection can significantly decrease costs associated with manual labeling. In turn, this allows either to reduce costs of training anomaly detection algorithms or to improve the performance of these algorithms by training them on a larger data set.

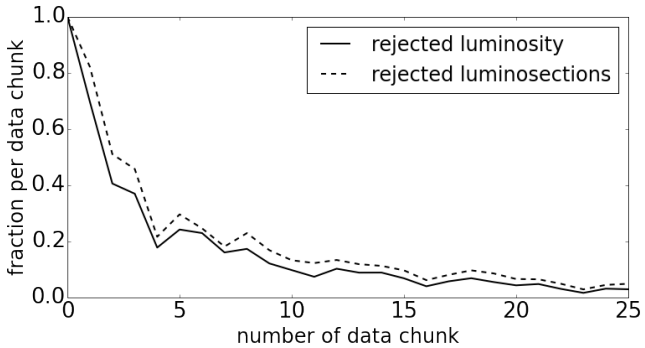


Figure 12: Fractions of manually labeled samples (solid line) and luminosity (dashed line) for each iteration (data chunk).

2.4 Simulation tuning

Anomaly detection methods rely on the assumption that normal training sample is large. Since most of the complex experiments use unique setups and, therefore, operate with unique data, obtaining sufficiently large training data set is a challenging task. This is especially problematic since data quality monitoring operates with raw or minimally-processed data (see, for example, [14]).

One of the primary sources of training data for anomaly detection algorithms is computer simulations. Many natural science experiments employ such simulations; in some areas, such as High Energy Physics, they play an essential part in the experiment. For example, Pythia event generator [101,102] is widely used in CERN experiments and simulates outcomes of proton-proton collisions. Another simulation, GEANT [103], is responsible for simulating detector response and is often used in tandem with event generators. Typically, event generators simulate outcomes of the experiment under nominal settings, as taking into account a wide range of possible anomalies is burdensome. Nevertheless, simulated anomalies can also be taken into account by anomaly detection algorithms introduced above.

Additionally, computer simulations play a vital role in searching for differences between theoretical predictions and observations as they practically represent theoretical models [42].

The major problem that occurs when attempting to train Machine Learning algorithms on simulated data is that most of the simulations contain parameters that are not precisely known for the experiment [56, 57]. A mismatch between simulation parameters and the ground-truth ones might result in degraded performance of anomaly detection methods trained on such data. Such a mismatch is especially problematic since events from poorly tuned simulation might potentially be similar to anomalous behavior or completely negate any attempt at analyzing differences between theoretical predictions and observations [44].

Multiple approaches attempt to circumvent mismatch between simulation and observations by correcting for the differences, including transfer learning [104], training classifier invariant to certain features [105] and using control variables [106]. For the purposes of training anomaly detection models, some methods are not applicable (e.g., control variables [106]); others lack any guarantees crucial for anomaly detection (e.g., transfer learning [104] and piv-

oting [105]).

The most straightforward way to incorporate simulation data is to find values of simulation's parameters such that outputs of the simulation closely match values observed in the real world. This process is usually referred to as fine-tuning. The number of simulation parameters is typically small, e.g., in work [57], authors consider around 20 parameters. Moreover, fine-tuning is often performed on processed data, e.g., standard tune for Pythia event generator is performed on around 400 features [56]. Thus, in practice, fine-tuning requires far less real data samples than training an anomaly detection algorithm on unprocessed or minimally processed data, at the same time offering a source of a priori normal samples that do not require manual labeling.

Recent developments in generative models, namely Generative Adversarial Networks [82], offer a general procedure for fine-tuning simulations. One of the significant obstacles in applying adversarial optimization procedures is the absence of gradient information as simulations often involve sampling random variables and can not be simply differentiated. A recent publication [58] addresses this problem and introduces Adversarial Variational Optimization that combines black-box optimization, namely Variational Optimization, and adversarial learning, allowing to tune non-differentiable generators to match observed data. We refer to any fine-tuning procedure that utilizes adversarial learning as Adversarial Optimization (AO).

Adversarial Optimization aims to minimize a divergence D between the ground-truth distribution P and a distribution Q_θ produced by a simulation with parameters θ:

$$D(P, Q_\theta) \to_\theta \min . \tag{11}$$

AO employs adversarial divergences, i.e., divergences that can be expressed as a minimization problem. For instance, one of the most popular choices, the Jensen-Shannon divergence, can be expresses as:

$$\mathrm{JSD}(P, Q) = \log 2 - \min_{f \in \mathcal{F}} L(f, P, Q); \tag{12}$$

where: $\mathcal{F}$ is the set of all functions $\mathcal{X} \to [0, 1]$ and L is the cross-entropy loss function. In the work [87], we focus on the Jensen-Shannon divergence; however, every result can be applied to most of adversarial divergences employed in practice, e.g., the Wasserstein distance.

However, in complex experiments, simulations tend to be computationally heavy. For instance, the simulation of a single proton collision event in the CERN ATLAS detector takes several minutes on a single core CPU [59]. Due to relatively high dimensionality, at least, high for black-box optimization methods, the amount of samples required for standard AO methods is large, which leads to heavy computational burdens. For example, in our work [87], we consider a simplified version of a real fine-tuning problem with only a single optimized parameter: with $64 \cdot 10^3$ samples solution of Adversarial Bayesian Optimization is only about 10 times closer to the ground-truth than the initial guess. The number of samples required for fine-tuning is expected to be significantly larger in more practical settings with a higher number of parameters.

In the corresponding contribution [87], we propose a novel family of divergences, namely adaptive divergences, that is specifically designed to lower the number of simulation calls. An adaptive divergence is constructed from a family of pseudo-divergences.

Definition 1 *A function* $D : \Pi(\mathcal{X}) \times \Pi(\mathcal{X}) \to \mathbb{R}$ *is a pseudo-divergence, if:*

(P1) $\forall P, Q \in \Pi(\mathcal{X}) : D(P, Q) \geq 0$*;*

(P2) $\forall P, Q \in \Pi(\mathcal{X}) : (P = Q) \Rightarrow D(P, Q) = 0$*;*

where $\Pi(\mathcal{X})$ *— set of all probability distributions on space* $\mathcal{X}$*.*

We also require a producing family of pseudo-divergences to satisfy the following properties.

Definition 2 *A family of pseudo-divergences* $\mathcal{D} = \{D_\alpha : \Pi(\mathcal{X}) \times \Pi(\mathcal{X}) \to \mathbb{R} \mid \alpha \in [0, 1]\}$ *is ordered and complete with respect to Jensen-Shannon divergence if:*

(D0) D_α *is a pseudo-divergence for all* $\alpha \in [0, 1]$*;*

(D1) $\forall P, Q \in \Pi(\mathcal{X}) : \forall 0 \leq \alpha_1 < \alpha_2 \leq 1 : D_{\alpha_1}(P, Q) \leq D_{\alpha_2}(P, Q)$*;*

(D2) $\forall P, Q \in \Pi(\mathcal{X}) : D_1(P, Q) = \mathrm{JSD}(P, Q)$*.*

The following definitions introduce two types of ordered and complete with respect to Jensen-Shannon divergences.

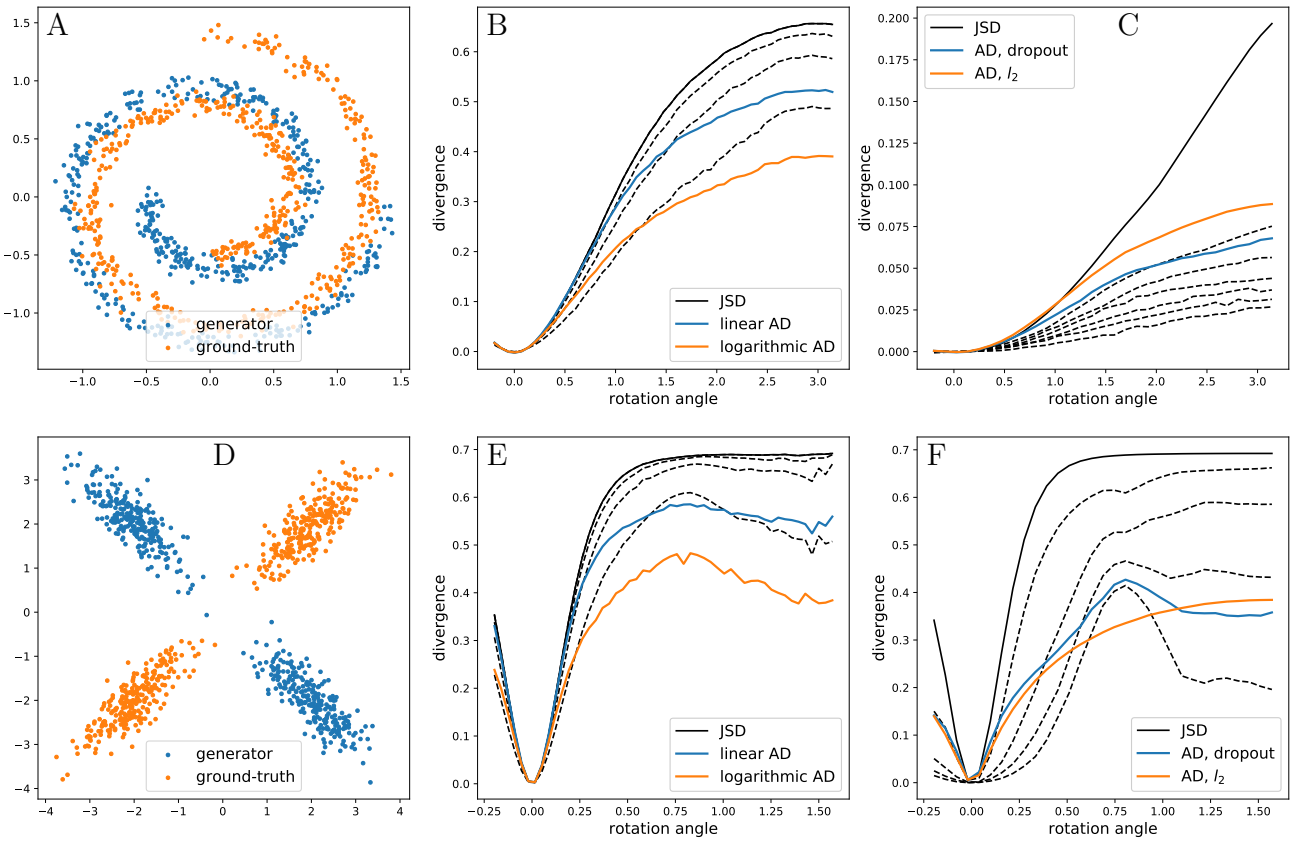


Figure 13: Synthetic examples. (A) and (D): ground-truth distributions and example configurations of generators. Both generators are rotated versions of the corresponding ground-truth distributions. (B) and (E): JSD — Jensen-Shannon divergences estimated by Gradient Boosted Decision Trees; linear AD and logarithmic AD — adaptive divergences based on the same models as JSD with linear and logarithmic capacity functions. (C) and (F): JSD — Jensen-Shannon divergences estimated by fully-connected Neural Networks.

Definition 3 *A model family* $\mathcal{M} = \{M_\alpha \subseteq \mathcal{F} \mid \alpha \in [0,1]\}$ *is complete and nested, if:*

(N0) $(x \mapsto 1/2) \in M_0$*;*

(N1) $M_1 = \mathcal{F}$*;*

(N2) $\forall \alpha, \beta \in [0,1] : (\alpha < \beta) \Rightarrow (M_\alpha \subset M_\beta)$.

Theorem 3 *If a model family* $\mathcal{M} = \{M_\alpha \subseteq \mathcal{F} \mid \alpha \in [0,1]\}$ *is complete and nested, then the family* $\mathcal{D} = \{D_\alpha : \Pi(\mathcal{X}) \times \Pi(\mathcal{X}) \to \mathbb{R} \mid \alpha \in [0,1]\}$*, where:*

$$D_\alpha(P,Q) = \log 2 - \inf_{f \in M_\alpha} L(f,P,Q), \tag{13}$$

is a complete and ordered with respect to Jensen-Shannon divergence family of pseudo-divergences.

Definition 4 *If M is a parameterized model family $M = \{f(\theta, \cdot) : \mathcal{X} \to [0,1] \mid \theta \in \Theta\}$, then a function $R : \Theta \to \mathbb{R}$ is a proper regularizer for the family M if:*

(R1) $\forall \theta \in \Theta : R(\theta) \geq 0$;

(R2) $\exists \theta_0 \in \Theta : \left(f(\theta_0, \cdot) \equiv \frac{1}{2}\right) \wedge (R(\theta_0) = 0)$.

Theorem 4 *If M is a parameterized model family: $M = \{f(\theta, \cdot) \mid \theta \in \Theta\}$ and $M = \mathcal{F}$, $R : \Theta \to \mathbb{R}$ is a proper regularizer for M , and $c : [0,1] \to [0, +\infty)$ is a strictly increasing function such, that $c(0) = 0$, then the family $\mathcal{D} = \{D_\alpha : \Pi(\mathcal{X}) \times \Pi(\mathcal{X}) \to \mathbb{R} \mid \alpha \in [0,1]\}$:*

$$\begin{aligned} D_\alpha(P,Q) &= \log 2 - \min_{\theta \in \Theta_\alpha(P,Q)} L(f(\theta,\cdot),P,Q); \\ \Theta_\alpha(P,Q) &= \operatorname*{Arg\,min}_{\theta \in \Theta} L_\alpha^R(\theta,P,Q); \\ L_\alpha^R(\theta,P,Q) &= L(f(\theta,\cdot),P,Q) + c(1-\alpha)R(\theta); \end{aligned}$$

is a complete and ordered with respect to Jensen-Shannon divergence family of pseudo-divergences.

Definitions 3 and 4 provide a straightforward way to construct families of pseudo-divergences in practice. For example, a complete and nested family of models can be implemented as a sequence of neural networks with a strictly increasing number of units in each layer.

In general, the most relevant cases of ordered and complete with respect to Jensen-Shannon divergence families of pseudo-divergences produced by varying 'capacity' of the underlying classifier f, where 'capacity' might mean the number of units in a neural network or strength of the regularization applied during training. Thus, we refer to parameter α as the capacity of pseudo-divergence D_α with respect to the family $\mathcal{D}$ or simply as the capacity of the pseudo-divergence if the family is clear from the context. The important property of pseudo-divergences defined in such manner is that low-capacity classifiers tend to require a small number of samples to be properly trained; therefore, estimation of pseudo-divergences built from low-capacity classifiers requires fewer

Algorithm 4: A general procedure for computing an adaptive divergence by grid search

Input: $\mathcal{D} = \{D_\alpha \mid \alpha \in [0, 1]\}$ — ordered and complete w.r.t. Jensen-Shannon divergence family of pseudo-divergences;
ε — tolerance;
P, Q — input distributions.

$\alpha \leftarrow 0$;
while $D_\alpha(P, Q) < (1 - \alpha) \log 2$ **do**
  $\alpha \leftarrow \alpha + \varepsilon$;
end
return $D_\alpha(P, Q)$

samples than high-capacity pseudo-divergences. It should be noted that while it is tempting to use low-capacity pseudo-divergences instead of proper divergences, their usage does not guarantee convergence of Adversarial Optimization due to Property (**P2**). At the same time, if a pseudo-divergence $D(P, Q) > 0$ for some P and Q, this automatically means $\mathrm{JSD}(P, Q) \geq D(P, Q) > 0$ and, therefore, Q is not the solution of the fine-tuning problem.

Adaptive divergence exploits the fact, that one can reject some simulation parameters based on computationally cheap pseudo-divergences.

Definition 5 *If a family of pseudo-divergences $\mathcal{D} = \{D_\alpha \mid \alpha \in [0, 1]\}$ is ordered and complete with respect to Jensen-Shannon divergence, then adaptive divergence $\mathrm{AD}_\mathcal{D}$ produced by $\mathcal{D}$ is defined as:*

$$\mathrm{AD}_\mathcal{D}(P, Q) = \inf \{D_\alpha(P, Q) \mid \mathrm{D}_\alpha(P, Q) \geq (1 - \alpha) \log 2\}. \tag{14}$$

The following theorem, combined with the observation that $\mathrm{AD}(P, Q) \geq 0$, states that adaptive divergence is a divergence, therefore, guarantees that AO converges on simulation parameters such that simulation's distribution matches distribution of real data.

Theorem 5 *If $\mathrm{AD}_\mathcal{D}$ is an adaptive divergence produced by an ordered and complete with respect to Jensen-Shannon divergence family of pseudo-divergences $\mathcal{D}$, then for any two distributions P and Q: $\mathrm{JSD}(P, Q) = 0$ if and only if $\mathrm{AD}(P, Q) = 0$.*

Algorithm 5: Boosted adaptive divergence

Input: X_P, X_Q — samples from distributions P and Q;
B — base estimator training algorithm;
N — maximal size of the ensemble;
$c : \mathbb{Z}_+ \to [0, 1]$ — capacity function;
ρ — learning rate;

$F_0 \leftarrow 1/2$;
$i \leftarrow 0$;
$L_0 \leftarrow \log 2$;
for $i = 1, \ldots, N$ **do**
  if $L_i > c(i) \log 2$ **then**
    $F_{i+1} \leftarrow F_i + \rho \cdot B(F_i, X_P, X_Q)$;
    $L_{i+1} \leftarrow L(F_{i+1}, X_P, X_Q)$;
    $i \leftarrow i + 1$;
  else
    return $\log 2 - L_i$;
  end
end
return $\log 2 - L_N$;

A proof can be found in paper [87].

Algorithm 4 outlines a general procedure for computing an adaptive divergence with grid search. Figure 13 demonstrates several variants of adaptive divergence on synthetic examples.

As can be seen from the definition, adaptive divergence 'switches' between pseudo-divergences depending on distribution P and Q: when P and Q are distant, $\mathrm{AD}_{\mathcal{D}}$ tends to select low-capacity pseudo-divergences; when Q approaches P, adaptive divergence employs high-capacity pseudo-divergences, with a proper divergence $D_1 = \mathrm{JSD}$ reserved to 'prove' equality of distributions. This property allows adaptive divergence to sample from the simulation less in cases when the simulation significantly deviates from the ground-truth data without sacrificing the convergence properties of AO.

Additionally, we introduce computationally efficient procedures for estimating adaptive divergence in cases:

Algorithm 6: Adaptive divergence estimation by a regularized neural network

Input: X_P, X_Q — samples from distributions P and Q; $f_\theta : \mathcal{X} \to \mathbb{R}$ — neural network with parameters $\theta \in \Theta$; $R : \Theta \to \mathbb{R}$ — regularization function; c — capacity function; ρ — exponential average coefficient; β — coefficient for R_1 regularization; γ — learning rate of SGD.

$L_{\text{acc}} \leftarrow \log 2$

while *not converged* **do**

$x_P \leftarrow \text{sample}(X_P)$;

$x_Q \leftarrow \text{sample}(X_Q)$;

$\zeta \leftarrow c\left(1 - \frac{L_{\text{acc}}}{\log 2}\right)$;

$g_0 \leftarrow \nabla_\theta \left[L(f_\theta, x_P, x_Q) + \zeta \cdot R(f_\theta)\right]$;

$g_1 \leftarrow \nabla_\theta \|\nabla_\theta f_\theta(x_P)\|^2$;

$L_{\text{acc}} \leftarrow \rho \cdot L_{\text{acc}} + (1 - \rho) \cdot L(f_\theta, x_P, x_Q)$;

$\theta \leftarrow \theta - \gamma\,(g_0 + \beta g_1)$;

end

return $\log 2 - L(f_\theta, X_P, X_Q)$

- families of pseudo-divergences that satisfy Definition 3 with underlying model based on gradient boosting — Algorithm 5;
- families of pseudo-divergences that satisfy Definition 4 with underlying model based on neural networks with dropout regularization or an explicit regularization term — Algorithm 6.

Introduced algorithms were evaluated on one toy example and two realistic fine-tuning problems involving Pythia [101,102] event generator and two black-box optimization algorithms, namely, Bayesian Optimization with Gaussian Processes [107,108] and Adversarial Variational Optimization [58]. Main results are presented in Figures 14 and 15; additional figures can be found in the original paper [87]. As can be seen from the figures, black-box optimization methods with adaptive divergence find solutions about an order of magnitude closer to the ground-truth within the same budget on the number of simulation calls.

In conclusion, adaptive divergence reduces the computational burden as-

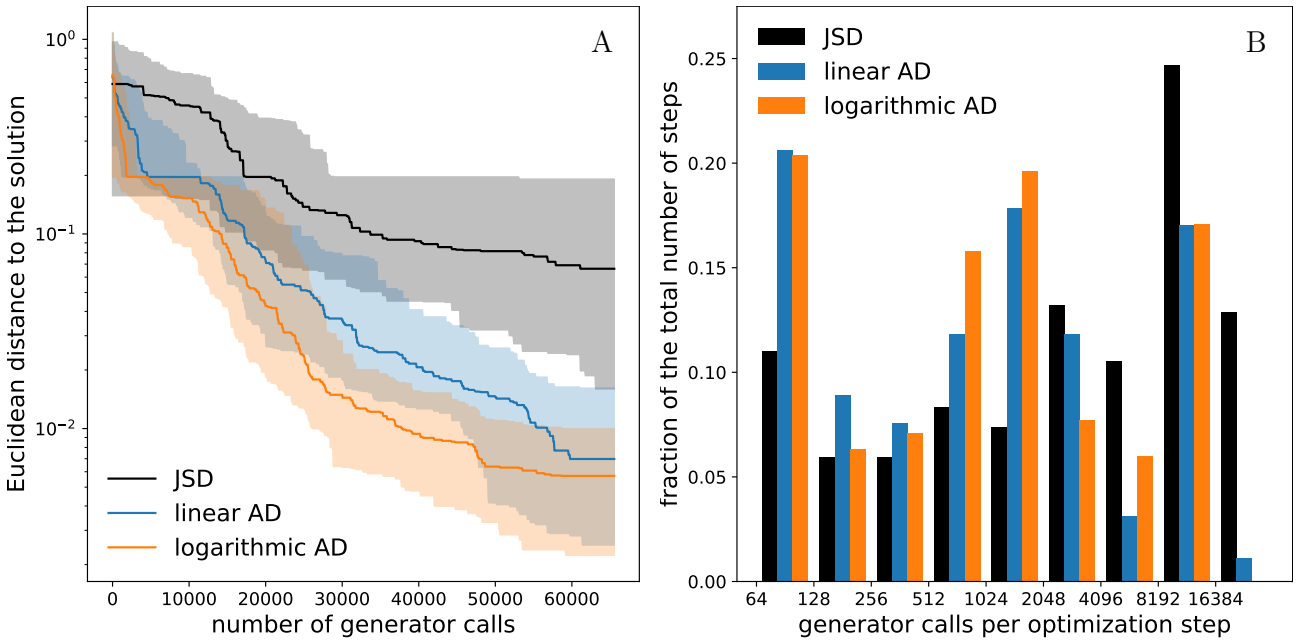


Figure 14: Pythia hyper-parameter tuning, CatBoost. (A) The convergence of Bayesian Optimization on: Jensen-Shannon divergence (marked as JSD), adaptive divergences with a linear capacity function (marked as linear AD), and a logarithmic capacity function (logarithmic AD). Each experiment was repeated 100 times; curves are interpolated, median curves are shown as solid lines, bands indicate 25th and 75th percentiles. (B) Distribution of computational costs per single optimization step measured by the number of calls to the generator requested for divergence estimation; each optimization step requires exactly one divergence estimation; note logarithmic scaling of the x-axis.

sociated with fine-tuning, which, in turn, allows to significantly reduce any mismatch between observations and the simulation. Note, that such mismatch directly translates to a bias of anomaly detection algorithms trained on simulation data and complicates search of subtle disagreements between theory and observations. Therefore, fine-tuning procedures that employ adaptive divergences directly impact the overall performance of data quality monitoring.

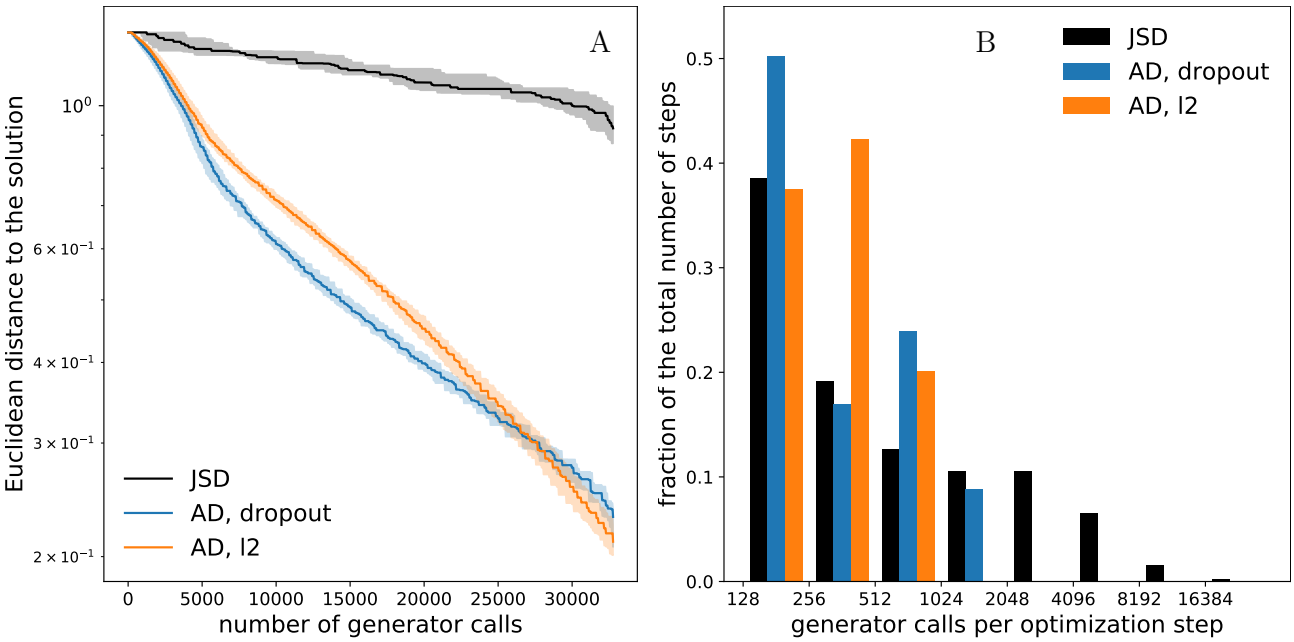


Figure 15: Pythia-alignment, Neural Networks. (A) Convergence of Adversarial Variational Optimization on: adaptive divergence produced by l_2 regularization (AD, l_2), dropout regularization (AD, dropout), and the baseline divergence with constant R_1 regularization (marked as JSD). Each experiment was repeated 20 times; curves are interpolated, median curves are shown by solid lines, bands indicate 25th and 75th percentiles; steps-like patterns are interpolation artifacts. (B) Distribution of computational costs per single optimization step measured by the number of calls to the generator requested for divergence estimation; each optimization step requires exactly one divergence estimation; note logarithmic scaling of the x-axis.

3 Conclusion

Data quality monitoring and anomaly detection methods play an essential role in natural science experiments. Machine Learning approaches to DQM and anomaly detection become increasingly relevant as the complexity of the experiments increases, and theoretical models become increasingly accurate.

In this dissertation, the author focused on the main tasks behind data quality monitoring and the search for discrepancies between theoretical predictions and observations. First, anomaly detection algorithms were considered — the author extended the problem statement of traditional anomaly detection and:

1. introduced a family of novel anomaly detection methods, namely OPE and EOPE classification, capable of taking into account known examples

of anomalies, thus, covering the whole spectrum of problems between one-class and binary classification cases; strictly proved main properties of these methods; demonstrated performance on several common benchmarks, including ones from experimental physics.

Second, to further improve the capabilities of DQM systems, the author:

2. introduced a novel method for identifying channels affected by an anomaly without need for channel-level labels at the training time; strictly proved main properties of the method; evaluated the algorithm on data collected by a large detector, namely the CERN CMS detector.

Third, to collect training data for anomaly detection algorithms and enable search for subtle differences between theory and observations, two potential sources of labeled samples were considered, namely, manual labeling and computer simulations. As a result, the author:

3. evaluated an active-learning-based algorithm on data collected by a large detector, namely the CERN CMS detector; demonstrated significant benefits of the approach in practical DQM settings;

4. introduced a novel family of divergences, namely adaptive divergences, that allows to significantly accelerate fine-tuning of computer simulations; strictly proved the main properties of adaptive divergences; evaluated performance on a realistic fine-tuning problem.

Moreover, all novel methods proposed in this dissertation are applicable outside natural science experiments; notably, OPE and EOPE classification methods, are general-purpose and can be applied for any anomaly detection problems; the method for inferring sources of anomalies relies on assumption non-specific to DQM and can be applied in industrial settings; adaptive divergences are not constrained to fine-tuning procedures and can be employed in any adversarial learning scenario.

References

[1] The CMS experiment at the CERN LHC / The CMS Collaboration, S Chatrchyan, G Hmayakyan et al. // Journal of Instrumentation. — 2008. — aug. — Vol. 3, no. 08. — P. S08004–S08004.

[2] The CMS trigger system / V. Khachatryan, A.M. Sirunyan, A. Tumasyan et al. // Journal of Instrumentation. — 2017. — jan. — Vol. 12, no. 01. — P. P01020–P01020.

[3] The square kilometre array / Peter E Dewdney, Peter J Hall, Richard T Schilizzi, T Joseph LW Lazio // Proceedings of the IEEE. — 2009. — Vol. 97, no. 8. — P. 1482–1496.

[4] Broekema P Chris, van Nieuwpoort Rob V, Bal Henri E. The Square Kilometre Array science data processor. Preliminary compute platform design // Journal of Instrumentation. — 2015. — Vol. 10, no. 07. — P. C07004.

[5] An End-to-End Computing Model for the Square Kilometre Array / R. Jongerius, S. Wijnholds, R. Nijboer, H. Corporaal // Computer. — 2014. — Sep. — Vol. 47, no. 9. — P. 48–54.

[6] Neural networks and cellular automata in experimental high energy physics : Rep. / Paris-11 Univ. ; Executor: B Denby : 1987.

[7] Baldi Pierre, Sadowski Peter, Whiteson Daniel. Searching for exotic particles in high-energy physics with deep learning // Nature communications. — 2014. — Vol. 5, no. 1. — P. 1–9.

[8] Machine learning at the energy and intensity frontiers of particle physics / Alexander Radovic, Mike Williams, David Rousseau et al. // Nature. — 2018. — Vol. 560, no. 7716. — P. 41–48.

[9] Gravity Spy: integrating advanced LIGO detector characterization, machine learning, and citizen science / Michael Zevin, Scott Coughlin, Sara Bahaadini et al. // Classical and Quantum Gravity. — 2017. — Vol. 34, no. 6. — P. 064003.

[10] Modern machine learning methods in HEP / Raphael Friese, Guenter Quast, Roger Wolf, Stefan Wunsch // Verhandlungen der Deutschen Physikalischen Gesellschaft. — 2017.

[11] Carrazza Stefano. Machine learning challenges in theoretical HEP // Journal of Physics: Conference Series / IOP Publishing. — Vol. 1085. — 2018. — P. 022003.

[12] Machine-learning in astronomy / Michael Hobson, Philip Graff, Farhan Feroz, Anthony Lasenby // Proceedings of the International Astronomical Union. — 2014. — Vol. 10, no. S306. — P. 279–287.

[13] LHCb reoptimized detector design and performance: Technical Design Report : Rep. / LHCb-TDR-009 ; Executor: S Cadeddu, P Dalpiaz, Z Guzik et al. : 2003.

[14] Detector monitoring with artificial neural networks at the CMS experiment at the CERN Large Hadron Collider / Adrian Alan Pol, Gianluca Cerminara, Cécile Germain et al. // Computing and Software for Big Science. — 2019. — Vol. 3, no. 1. — P. 3.

[15] Online data monitoring in the LHCb experiment / O Callot, S Cherukuwada, M Frank et al. // Journal of Physics: Conference Series / IOP Publishing. — Vol. 119. — 2008. — P. 022015.

[16] LIGO: the laser interferometer gravitational-wave observatory / BP Abbott, R Abbott, R Adhikari et al. // Reports on Progress in Physics. — 2009. — Vol. 72, no. 7. — P. 076901.

[17] Impact of aerosols and adverse atmospheric conditions on the data quality for spectral analysis of the HESS telescopes / Joachim Hahn, R De los Reyes, Konrad Bernlöhr et al. // Astroparticle Physics. — 2014. — Vol. 54. — P. 25–32.

[18] Mommert Michael. Cloud Identification from All-sky Camera Data with Machine Learning // The Astronomical Journal. — 2020. — mar. — Vol. 159, no. 4. — P. 178.

[19] The OPERA experiment in the CERN to Gran Sasso neutrino beam / R Acquafredda, T Adam, N Agafonova et al. // Journal of Instrumentation. — 2009. — Vol. 4, no. 04. — P. P04018.

[20] Brumfiel Geoff. Particles break light-speed limit. — 2011.

[21] Measurement of the neutrino velocity with the OPERA detector in the CNGS beam / T Adam, N Agafonova, A Aleksandrov et al. // Journal of High Energy Physics. — 2012. — Vol. 2012, no. 10. — P. 93.

[22] CMS data quality monitoring: systems and experiences / Lassi Tuura, A Meyer, I Segoni, G Della Ricca // Journal of Physics: Conference Series. — 2010. — Vol. 219, no. 7. — P. 072020.

[23] Stone Robert, Mukherjee Soma. Environmentally induced nonstationarity in LIGO science run data // Classical and Quantum Gravity. — 2009. — oct. — Vol. 26, no. 20. — P. 204021.

[24] George Daniel, Shen Hongyu, Huerta EA. Classification and unsupervised clustering of LIGO data with Deep Transfer Learning // Physical Review D. — 2018. — Vol. 97, no. 10. — P. 101501.

[25] Li W., Wu G., Du Q. Transferred Deep Learning for Anomaly Detection in Hyperspectral Imagery // IEEE Geoscience and Remote Sensing Letters. — 2017. — Vol. 14, no. 5. — P. 597–601.

[26] Mimicking the human expert: Pattern recognition for an automated assessment of data quality in MR spectroscopic images / Bjoern H. Menze, B. Michael Kelm, Marc-André Weber et al. // Magnetic Resonance in Medicine. — 2008. — Vol. 59, no. 6. — P. 1457–1466.

[27] Identifying, attributing, and overcoming common data quality issues of manned station observations / Stefan Hunziker, Stefanie Gubler, Juan Calle et al. // International Journal of Climatology. — 2017. — Vol. 37, no. 11. — P. 4131–4145.

[28] Stankevicius Mantas, Marcinkevicius Virginijus, Rapsevicius Valdas. Comparison of Supervised Machine Learning Techniques for CERN CMS

Offline Data Certification. // Doctoral Consortium/Forum@ DB&IS. — 2018. — P. 170–176.

[29] Using Artificial Neural Networks for Glitch Identification in Advanced LIGO / Donald Moffa, Kyle Rose, Les Wade et al. // APS Meeting Abstracts. — 2018.

[30] Anomaly detection using Deep Autoencoders for the assessment of the quality of the data acquired by the CMS experiment / Adrian Alan Pol, Virginia Azzolini, Gianluca Cerminara et al. // EPJ Web of Conferences / EDP Sciences. — Vol. 214. — 2019. — P. 06008.

[31] ATLAS online data quality monitoring / C. Cuenca Almenar, A. Corso-Radu, H. Hadavand et al. // 2010 17th IEEE-NPSS Real Time Conference. — 2010. — P. 1–5.

[32] The ALICE online data quality monitoring / Barthelemy von Haller, Adriana Telesca, Sylvain CHAPELAND et al. // 13th International Workshop on Advanced Computing and Analysis Techniques in Physics Research / SISSA Medialab. — Vol. 93. — 2011. — P. 024.

[33] ATLAS offline data quality monitoring / J Adelman, M Baak, N Boelaert et al. // Journal of Physics: Conference Series. — 2010. — apr. — Vol. 219, no. 4. — P. 042018.

[34] Towards automation of data quality system for CERN CMS experiment / M Borisyak, F Ratnikov, D Derkach, A Ustyuzhanin // Journal of Physics: Conference Series. — 2017. — oct. — Vol. 898. — P. 092041.

[35] IOP: LHCb data quality monitoring / M Adinolfi, A Ustyuzhanin, D Derkach et al. // J. Phys.: Conf. Ser. — Vol. 898. — 2017. — P. 092027.

[36] Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC / G. Aad, T. Abajyan, B. Abbott et al. // Physics Letters B. — 2012. — Vol. 716, no. 1. — P. 1 – 29.

[37] Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC / G. Aad, T. Abajyan,

B. Abbott et al. // Physics Letters B. — 2012. — Vol. 716, no. 1. — P. 1 – 29.

[38] GW170814: A Three-Detector Observation of Gravitational Waves from a Binary Black Hole Coalescence / B. P. Abbott, R. Abbott, T. D. Abbott et al. // Phys. Rev. Lett. — 2017. — Oct. — Vol. 119. — P. 141101.

[39] Search for new physics with atoms and molecules / MS Safronova, D Budker, D DeMille et al. // Reviews of Modern Physics. — 2018. — Vol. 90. — P. 025008.

[40] Farina Marco, Nakai Yuichiro, Shih David. Searching for new physics with deep autoencoders // Phys. Rev. D. — 2020. — Apr. — Vol. 101. — P. 075021.

[41] Semi-supervised anomaly detection – towards model-independent searches of new physics / Mikael Kuusela, Tommi Vatanen, Eric Malmi et al. // Journal of Physics: Conference Series. — 2012. — jun. — Vol. 368. — P. 012032.

[42] Andreassen Anders, Nachman Benjamin, Shih David. Simulation assisted likelihood-free anomaly detection // Phys. Rev. D. — 2020. — May. — Vol. 101. — P. 095004.

[43] De Simone Andrea, Jacques Thomas. Guiding new physics searches with unsupervised learning // The European Physical Journal C. — 2019. — Vol. 79. — P. 1–15.

[44] Blance Andrew, Spannowsky Michael, Waite Philip. Adversarially-trained autoencoders for robust unsupervised new physics searches // Journal of High Energy Physics. — 2019. — Vol. 2019. — P. 47.

[45] Novelty detection meets collider physics / Jan Hajer, Ying-Ying Li, Tao Liu, He Wang // Phys. Rev. D. — 2020. — Apr. — Vol. 101. — P. 076015.

[46] Hodge Victoria, Austin Jim. A survey of outlier detection methodologies // Artificial intelligence review. — 2004. — Vol. 22. — P. 85–126.

[47] Chandola Varun, Banerjee Arindam, Kumar Vipin. Anomaly Detection: A Survey // ACM Comput. Surv. — 2009. — Jul. — Vol. 41, no. 3. — Access mode: https://doi.org/10.1145/1541880.1541882.

[48] LHCb VELO Upgrade Technical Design Report : Rep. : CERN-LHCC-2013-021. LHCB-TDR-013 ; Executor: LHCb Collaboration : 2013. — Nov.

[49] (1 + epsilon)-class Classification: an Anomaly Detection Method for Highly Imbalanced or Incomplete Data Sets / Maxim Borisyak, Artem Ryzhikov, Andrey Ustyuzhanin et al. // Journal of Machine Learning Research. — 2020. — Vol. 21, no. 72. — P. 1–22.

[50] Software for the LHCb experiment / Gloria Corti, Marco Cattaneo, Philippe Charpentier et al. // IEEE transactions on nuclear science. — 2006. — Vol. 53, no. 3. — P. 1323–1328.

[51] Petascale High Order Dynamic Rupture Earthquake Simulations on Heterogeneous Supercomputers / A. Heinecke, A. Breuer, S. Rettenberger et al. // SC '14: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. — 2014. — P. 3–14.

[52] TURBULENT MAGNETIC FIELD AMPLIFICATION FROM SPIRAL SASI MODES: IMPLICATIONS FOR CORE-COLLAPSE SUPERNOVAE AND PROTO-NEUTRON STAR MAGNETIZATION / Eirik Endeve, Christian Y. Cardall, Reuben D. Budiardja et al. // The Astrophysical Journal. — 2012. — may. — Vol. 751, no. 1. — P. 26.

[53] Using galaxy formation simulations to optimize LIGO follow-up observations / Elisa Antolini, Ilaria Caiazzo, Romeel Davé, Jeremy S Heyl // Monthly Notices of the Royal Astronomical Society. — 2017. — Vol. 466, no. 2. — P. 2212–2216.

[54] Perdikaris Paris, Grinberg Leopold, Karniadakis George Em. Multiscale modeling and simulation of brain blood flow // Physics of Fluids. — 2016. — Vol. 28, no. 2. — P. 021304.

[55] Agent-based simulation of a financial market / Marco Raberto, Silvano Cincotti, Sergio M. Focardi, Michele Marchesi // Physica A: Statistical Mechanics and its Applications. — 2001. — Vol. 299, no. 1. — P. 319 – 327. — Application of Physics in Economic Modelling.

[56] Skands Peter, Carrazza Stefano, Rojo Juan. Tuning PYTHIA 8.1: the Monash 2013 tune // The European Physical Journal C. — 2014. — Vol. 74, no. 8. — P. 3024.

[57] Ilten P., Williams M., Yang Y. Event generator tuning using Bayesian optimization // Journal of Instrumentation. — 2017. — apr. — Vol. 12, no. 04. — P. P04028.

[58] Louppe Gilles, Hermans Joeri, Cranmer Kyle. Adversarial Variational Optimization of Non-Differentiable Simulators // Proceedings of Machine Learning Research / Ed. by Kamalika Chaudhuri, Masashi Sugiyama. — Vol. 89 of Proceedings of Machine Learning Research. — PMLR, 2019. — 16–18 Apr. — P. 1438–1447.

[59] The ATLAS Collaboration. The ATLAS simulation infrastructure // European Physical Journal C: Particles and Fields. — 2010. — Vol. 70, no. 3. — P. 823–874.

[60] Chalapathy Raghavendra, Menon Aditya Krishna, Chawla Sanjay. Anomaly detection using one-class neural networks // arXiv preprint arXiv:1802.06360. — 2018.

[61] Deep one-class classification / Lukas Ruff, Nico Görnitz, Lucas Deecke et al. // International Conference on Machine Learning. — Stockholm, Sweden, 2018. — P. 4390–4399.

[62] Support vector method for novelty detection / Bernhard Schölkopf, Robert C Williamson, Alex J Smola et al. // Advances in Neural Information Processing Systems. — Denver, United States, 2000. — P. 582–588.

[63] Zhou Chong, Paffenroth Randy C. Anomaly detection with robust deep autoencoders // ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. — Halifax, Canada, 2017. — P. 665–674.

[64] Chalapathy Raghavendra, Menon Aditya Krishna, Chawla Sanjay. Robust, deep and inductive anomaly detection // Joint European Conference on Machine Learning and Knowledge Discovery in Databases. — Skopje, Macedonia, 2017. — P. 36–51.

[65] An Jinwon, Cho Sungzoon. Variational autoencoder based anomaly detection using reconstruction probability // Special Lecture on IE. — 2015. — Vol. 2, no. 1. — P. 1–18.

[66] Learning sparse representation with variational auto-encoder for anomaly detection / Jiayu Sun, Xinzhou Wang, Naixue Xiong, Jie Shao // IEEE Access. — 2018. — Vol. 6. — P. 33353–33361.

[67] Choi Hyunsun, Jang Eric, Alemi Alexander A. Waic, but why? generative ensembles for robust anomaly detection // arXiv preprint arXiv:1810.01392. — 2018.

[68] Tax David MJ, Duin Robert PW. Support vector data description // Machine learning. — 2004. — Vol. 54, no. 1. — P. 45–66.

[69] Boser Bernhard E., Guyon Isabelle M., Vapnik Vladimir N. A Training Algorithm for Optimal Margin Classifiers // Proceedings of the Fifth Annual Workshop on Computational Learning Theory. — COLT '92. — New York, NY, USA : Association for Computing Machinery, 1992. — P. 144–152. — Access mode: https://doi.org/10.1145/130385.130401.

[70] Liu Fei Tony, Ting Kai Ming, Zhou Zhi-Hua. Isolation forest // IEEE International Conference on Data Mining. — Washington, DC, United State, 2008. — P. 413–422.

[71] Baldi Pierre, Sadowski Peter, Whiteson Daniel. Searching for exotic particles in high-energy physics with deep learning // Nature communications. — 2014. — Vol. 5. — P. 4308.

[72] Bekker Jessa, Davis Jesse. Learning From Positive and Unlabeled Data: A Survey // CoRR. — 2018. — Vol. abs/1811.04820. — Access mode: http://arxiv.org/abs/1811.04820.

[73] Elkan Charles, Noto Keith. Learning Classifiers from Only Positive and Unlabeled Data // ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. — KDD '08. — Las Vegas, Nevada, USA, 2008. — P. 213–220.

[74] Northcutt Curtis G., Wu Tailin, Chuang Isaac L. Learning with Confident Examples: rank Pruning for Robust Classification with Noisy Labels // Conference on Uncertainty in Artificial Intelligence, UAI. — Sydney, Australia, 2017.

[75] Pearl Judea. Causal inference in statistics: An overview // Statist. Surv. — 2009. — Vol. 3. — P. 96–146.

[76] Lughofer Edwin. On-line active learning: a new paradigm to improve practical useability of data stream modeling methods // Information Sciences. — 2017. — Vol. 415. — P. 356–376.

[77] RayChaudhuri T., Hamey L. G. C. Minimisation of data collection by active learning // Proceedings of ICNN'95 - International Conference on Neural Networks. — Vol. 3. — 1995. — P. 1338–1341.

[78] Burbidge Robert, Rowland Jem J., King Ross D. Active Learning for Regression Based on Query by Committee // Intelligent Data Engineering and Automated Learning - IDEAL 2007 / Ed. by Hujun Yin, Peter Tino, Emilio Corchado et al. — Berlin, Heidelberg : Springer Berlin Heidelberg, 2007. — P. 209–218.

[79] Lughofer Edwin. Single-pass active learning with conflict and ignorance // Evolving Systems. — 2012. — Vol. 3. — P. 251–271.

[80] Lughofer Edwin. Evolving fuzzy systems-methodologies, advanced concepts and applications. — Springer, 2011. — Vol. 53.

[81] Toni Tina, Stumpf Michael PH. Simulation-based model selection for dynamical systems in systems and population biology // Bioinformatics. — 2009. — Vol. 26, no. 1. — P. 104–110.

[82] Generative adversarial nets / Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza et al. // Advances in neural information processing systems. — 2014. — P. 2672–2680.

[83] Meeds Edward, Leenders Robert, Welling Max. Hamiltonian ABC // Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence. — UAI'15. — Arlington, Virginia, USA : AUAI Press, 2015. — P. 582–591.

[84] Cranmer Kyle, Pavez Juan, Louppe Gilles. Approximating likelihood ratios with calibrated discriminative classifiers // arXiv preprint arXiv:1506.02169. — 2015.

[85] Experiments using machine learning to approximate likelihood ratios for mixture models / K Cranmer, J Pavez, Gilles Louppe, WK Brooks // Journal of Physics Conference Series. — 2016.

[86] Tran Minh-Ngoc, Nott David J, Kohn Robert. Variational Bayes with intractable likelihood // Journal of Computational and Graphical Statistics. — 2017. — Vol. 26, no. 4. — P. 873–882.

[87] Borisyak Maxim, Gaintseva Tatiana, Ustyuzhanin Andrey. Adaptive divergence for rapid adversarial optimization // PeerJ Computer Science. — 2020. — May. — Vol. 6. — P. e274.

[88] Jin Long, Lazarow Justin, Tu Zhuowen. Introspective classification with convolutional nets // Advances in Neural Information Processing Systems. — Long Beach, California, United States, 2017. — P. 823–833.

[89] Progressive growing of gans for improved quality, stability, and variation / Tero Karras, Timo Aila, Samuli Laine, Jaakko Lehtinen // arXiv preprint arXiv:1710.10196. — 2017.

[90] StarGAN: Unified Generative Adversarial Networks for Multi-domain Image-to-Image Translation / Yunjey Choi, Minje Choi, Munyoung Kim et al. // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition / IEEE. — 2018. — P. 8789–8797.

[91] Unrolled Generative Adversarial Networks / Luke Metz, Ben Poole, David Pfau, Jascha Sohl-Dickstein // ICLR. — 2016.

[92] Millman K. J., Aivazis M. Python for Scientists and Engineers // Computing in Science Engineering. — 2011. — Vol. 13, no. 2. — P. 9–12.

[93] van der Walt S., Colbert S. C., Varoquaux G. The NumPy Array: A Structure for Efficient Numerical Computation // Computing in Science Engineering. — 2011. — Vol. 13, no. 2. — P. 22–30.

[94] SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python / Pauli Virtanen, Ralf Gommers, Travis E. Oliphant et al. // Nature Methods. — 2020. — Vol. 17. — P. 261–272.

[95] Scikit-learn: Machine Learning in Python / Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort et al. // Journal of Machine Learning Research. — 2011. — Vol. 12, no. 85. — P. 2825–2830.

[96] Abadi Martín, Agarwal Ashish, Barham Paul et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. — 2015. — Software available from tensorflow.org. Access mode: `https://www.tensorflow.org/`.

[97] PyTorch: An Imperative Style, High-Performance Deep Learning Library / Adam Paszke, Sam Gross, Francisco Massa et al. // Advances in Neural Information Processing Systems 32 / Ed. by H. Wallach, H. Larochelle, A. Beygelzimer et al. — Curran Associates, Inc., 2019. — P. 8026–8037.

[98] Scholkopf Bernhard, Smola Alexander J. Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. — MIT press, 2001.

[99] Kim Taesup, Bengio Yoshua. Deep Directed Generative Models with Energy-Based Probability Estimation // arXiv preprint arXiv:1606.03439. — 2016.

[100] Deep learning for inferring cause of data anomalies / V. Azzolini, M. Borisyak, G. Cerminara et al. // Journal of Physics: Conference Series. — 2018. — sep. — Vol. 1085. — P. 042015.

[101] Sjöstrand Torbjörn, Mrenna Stephen, Skands Peter. PYTHIA 6.4 physics and manual // Journal of High Energy Physics. — 2006. — may. — Vol. 2006, no. 05. — P. 026.

[102] An introduction to PYTHIA 8.2 / Torbjörn Sjöstrand, Stefan Ask, Jesper R Christiansen et al. // Computer physics communications. — 2015. — Vol. 191. — P. 159–177.

[103] Recent developments in Geant4 / J. Allison, K. Amako, J. Apostolakis et al. // Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment. — 2016. — Vol. 835. — P. 186 – 225.

[104] Ganin Yaroslav, Lempitsky Victor. Unsupervised Domain Adaptation by Backpropagation // Proceedings of the 32nd International Conference on Machine Learning / Ed. by Francis Bach, David Blei. — Vol. 37 of Proceedings of Machine Learning Research. — Lille, France : PMLR, 2015. — 07–09 Jul. — P. 1180–1189. — Access mode: `http://proceedings.mlr.press/v37/ganin15.html`.

[105] Louppe Gilles, Kagan Michael, Cranmer Kyle. Learning to pivot with adversarial networks // Advances in neural information processing systems. — 2017. — P. 981–990.

[106] Borisyak M., Kazeev N. Machine Learning on data with sPlot background subtraction // Journal of Instrumentation. — 2019. — aug. — Vol. 14, no. 08. — P. P08020–P08020.

[107] Lizotte Daniel James. Practical bayesian optimization. — University of Alberta, 2008.

[108] Rasmussen Carl Edward. Gaussian processes in machine learning // Summer School on Machine Learning / Springer. — 2003. — P. 63–71.