Contents

Chapter 1

Automated Data Quality Assessment as a Novelty Detection Problem

---

The certification process of the physics data obtained by the CMS Experiment as usable for physics analysis is the final step in the CMS DQM procedure as described in Section **??**. Current certification conducted by human experts is labor intensive and time consuming task. The ever increasing physics data volume as well as detector complexity calls for ways to automate this monitoring step.

Present decisions are based on histograms integrated on acquisition run basis (see Chapter **??**). One acquisition run could be a relatively long interval. The work of pin-pointing the exact times affected by anomalous behavior can require further investigation and use of non-event data. Besides, in those cases statistics are often too limited for human assessment. As a consequence, the certification flag can be inaccurate when transient problems throughout a run are overlooked. Similarly, useful data are discarded from runs with malfunctions. A certification protocol based on shorter interval, using luminosity sections (LSs, see Chapter **??**), is more desirable. LS-based quality labels are already in place, obtained via application that monitors the powering and voltages delivered to the various sub-detectors (Rapsevicius et al., 2011). However CMS Collaboration is looking for means to improve LS-based certification.

The detector data is high dimensional which naturally points toward solutions based on deep learning algorithms. Anomalies caused by detector malfunctions or not optimal software reconstruction are difficult to enumerate a priori and occur rarely. Consequently, use of supervised anomaly detection methods such as binary classification neural networks is problematic as positive (anomalous) class may be misrepresented in the training set. Furthermore, the characteristics of *good* data are evolving with LHC or CMS configuration. In this novelty detection context we base our prototype on a semi-supervised approach which uses deep autoencoders, trained on the data acquired during 2016 LHC campaign.

A key advantage of this approach is that the reason for flagging a sample is easily explainable, which can be further used to ascribe the origin of the problems in the data to a specific sub-detector or physics objects.

This chapter introduces a new tool for the CMS DC process which targets challenges listed above and automates the protocol. In summary the main aspects of this work are:

- detecting different types of anomalies with high sensitivity and specificity affecting the CMS detector using only data certified as globally good;
- assessing the (*mis*)behavior based on shorter interval than what is realistically feasible with current protocol;
- achieving stable performance over time compared to fully supervised models as nature of the data evolves;
- allowing for fine grained interpretation of the classification results, which can be further used to ascribe the origin of the problems; *and*
- providing additional tool in CMS DQM toolbox that minimizes the risk of human mistakes and speeds up the certification procedure.

1.1 The Dataset and Preprocessing

The dataset contains all LSs data recorded from June to October 2016 resulting in 163684 samples. Nearly all of the reconstructed particle collections (e.g. photons, muons, etc.) are included. This way all CMS sub-detectors are well represented. This accounts to total of 401 physics variables (e.g. transverse momentum, energy, cluster multiplicity, particle direction) of the measurement of particular physics objects. We relay on the quality labels (good or bad) based on the manual work done by human detector experts.

All the distributions come from a dataset in AOD format (Della Negra et al., 2005). AOD format provides data for physics analysis in a convenient, compact format. It contains a copy of all the high-level physics objects, plus information sufficient to support typical analysis actions. We choose it as the best trade-off between the level of reconstruction (number of features needed to describe each LSs) and the amount of information stored in those features. Past research (Borisyak et al., 2017) utilized miniAOD dataset that has less features and consequently less information to learn from.

Naturally, features have different ranges and distributions. During preprocessing the data are standardized by subtracting the mean and scaling the features to unit variance independently on each feature. This way we expect to improve the training speed.

1.1.1 Lumisection Representation

To aim for higher time granularity of the classification results all the data are divided into shorter time quanta when compared to current CMS DC process, i.e. LSs instead of acquisition runs.

As explained earlier, human experts make decisions regarding the data quality based on histograms. Simple tests e.g. mean included in a given range, are often utilized for this task. When a sub-detector exposes an abnormal behavior e.g. becomes unresponsive, it is reflected in measured or reconstructed properties. In case of an anomaly, the histograms should show a considerable deviation from the nominal shape. To mimic the logic of current procedure we decided to represent each sample as a 2807 dimensional vector that accumulates five quantiles, mean and standard deviation for all 401 variable distributions.

1.1.2 Different Event Topologies

In the CMS Experiment, the physics data is stored in different primary datasets (PDs). PDs are subsets of the event stream acquired by the CMS experiment grouped to satisfy constraints on the physics content, data processing and handling. Currently the DC process uses number of PDs tailored for the physics objective i.e. SingleMuon PD for *muons* or EGamma PD for *electrons*. For our primary study we have decided to use a dataset tailored for *jet* analysis which represents all CMS subdetectors since it contains every physics object and particle constituent needed for data quality certification. The proposed strategy has to be generic enough to be applicable for different PDs and in the future it is critical that the performance for all PDs is measured.

Ultimately, one could use 21 independently trained classifiers, each specializing in classification of each PD, similarly to the architecture proposed in (Azzolini et al., 2017). The global status of the detector could then be derived as a logical `AND` of the intermediate results coming from each of the models.

1.2 Methods and Experimental Design

Fortunately for the CMS collaboration the data produced by the experiment is infrequently corrupted. Anomalies account for roughly 2% of the dataset which is a small set of examples of failures. Moreover, emerging, unprecedented failures are difficult to anticipate. This makes supervised methods vulnerable to incomplete or inadequate representations of potential failures. A semi-supervised anomaly detection approach with model learning only negative class distribution was implemented to account for this challenge. During operation, model aims at identifying unobserved

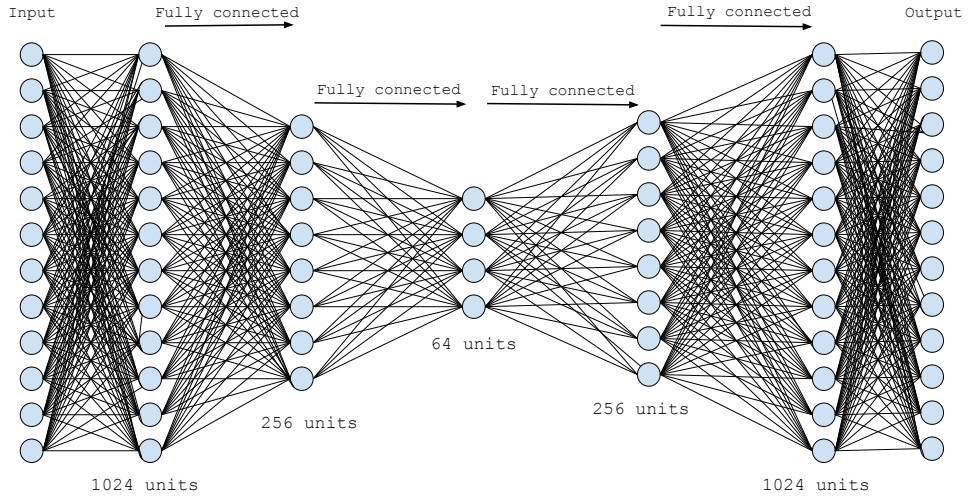

FIGURE 1.1: Proposed base architecture. The hyper parameters were chosen using grid search.

patterns in newly recorded observations. In this manner we intend to retain the full potential to catch all the future and unseen detector failure modes.

To this purpose we exploit autoencoders which, when trained on negative class, will yield sub-optimal representations for novel samples and consequently decoder outputs. Discrepancy between input and the output indicates that a sample is likely generated by a different process, hence should be flagged as problematic.

We use base architecture shown in Figure 1.1 and propose different regularization techniques. Our sparse autoencoder (Ranzato et al., 2006) has additional $L1$ kernel regularization (10^{-5}) on all the hidden nodes. This constraint penalizes the output of the hidden unit kernels and forces them to be close to zero. The exact penalty term was established using random search. For contractive autoencoder (Rifai et al., 2011), additional regularization improves model robustness against small variations in the training examples. Lastly, a variational autoencoder (Rezende et al., 2014) was tested (see Section **??**). Because our values are not scaled to predefined range, we use parametric rectified linear unit (He et al., 2015) as activation function in output layer. Hidden units are also using this type of activation. We train the network with Keras (Chollet et al., 2015) and TensorFlow (Abadi et al., 2016) using the Adam optimizer (Kingma & Ba, 2014) (with a learning rate of 0.0001, $\beta_1 = 0.7$, $\beta_2 = 0.9$) and early stopping mechanism monitoring validation dataset with patience set to 32 epochs. The network is instructed to minimize mean squared error between input X and the output $\hat{X}$ vector:

$$\epsilon = \frac{1}{n}\sum_{i=0}^{n}(x_i - \hat{x}_i)^2$$

Once deployed, the algorithm will evaluate samples one-by-one in the order of recording by the apparatus. To simulate this production scenario, we split dataset into training (60%), validation (20%) and testing (20%) set after sorting all samples

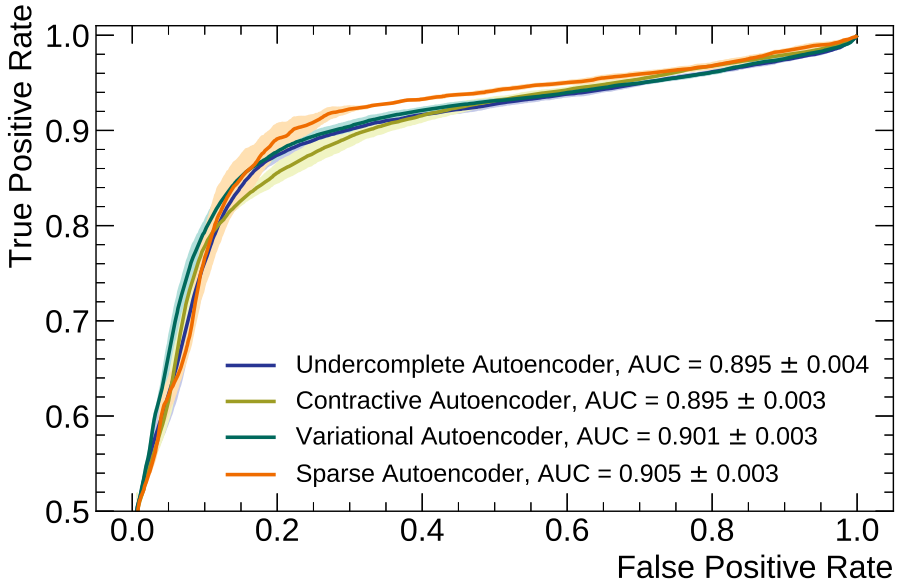

FIGURE 1.2: ROC and AUC of different autoencoder models using different regularization techniques.

chronologically. This way we account on periods of similar detector response. As LHC and CMS configuration evolves gently with time, random splitting could lead to unintended data snooping where model is tested on LSs nearly identical to ones in the training set. At early stages of this study it was noticed that the contamination in training set harms the performance of the algorithm. Thus all the positive samples are removed from training and validation sets. Test set is extended by those previously removed anomalous samples. Including more positive examples in the test is a better approach, as the set has limited amount of them. This helps qualify performance of various methods given that bad LSs should always be qualified as bad.

The final decision function is computed using mean squared error of the worst 100 reconstructed features (TOP100) to mirror human decision process:

$$\text{TOP100} = \frac{1}{100}\sum_{i=1}^{100} sorted(x_i - \hat{x}_i)^2.$$

The difference between reference and recorded distributions is dominated by noise. Hence, experts pay attention only to significant deviations.

1.3 Experimental Results and Discussion

Final ROC curves for models and their corresponding AUC are reported in Figure 1.2. All models show good performance, especially sparse autoencoder.

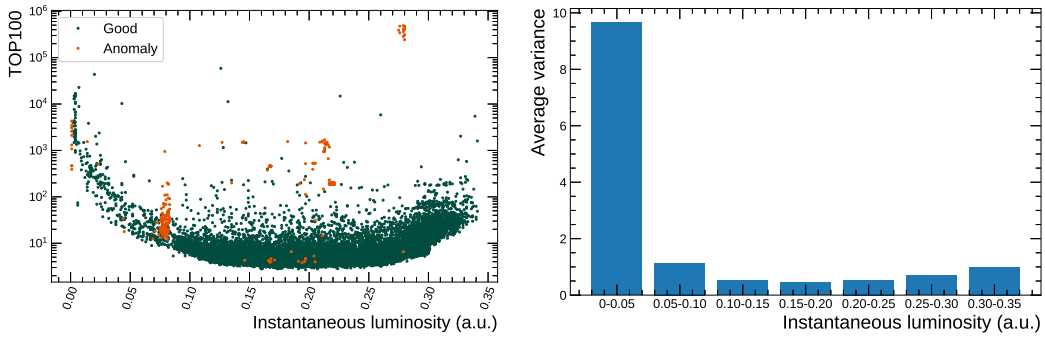

FIGURE 1.3: Anomaly score w.r.t. instantaneous luminosity (left) of sparse autoencoder. The waviness is caused by the average variance of feature values in different instantaneous luminosity ranges (right).

Figure 1.3 shows the error yield for each sample in the held out test set as a function of LHC instantaneous luminosity (*intensity* of the LHC beams). The error is visibly higher in low and high luminosity regions as the average data spread for features in those regions is higher when compared to others. This results in model being unable to capture full data variability. We hypothesized that this dependence was also caused by smaller amount of samples coming from those regions. Sample weights were used in order to penalize error in those regions more, but no performance improvement was noticed. Adding additional autoencoder input carrying value of instantaneous luminosity has neither changed the performance.

In order to improve classification results, we investigated means for systematic feature selection. Same features may harm the overall performance as they are close to being constant-valued and thus useless, or when it is impossible to reconstruct them properly. Features with minimal variability were removed. Subsequently Pearson correlation coefficient between training and reconstructed training data was calculated for each feature. When below certain threshold feature was removed and model was retrained using new, smaller bag of features. Figure 1.4 shows performance of a new sparse autoencoder trained with list of pre-selected features given the above criteria. While it is visible that this pruning process hurts overall model performance it could be beneficial when one needs to maximize true positive rate under low false positive rate constraints.

1.3.1 Comparison with Supervised Anomaly Detection

It is worth noting that it was expected that the performance of the ML algorithms can change dramatically over the course of learning and rapidly improve as more data is evaluated and labeled by the experts, and thus available for training. The performance is expected the have some intrinsic limit, especially in periods when novel failures emerge. In this evolving conditions context the CMS collaboration is looking for tools guaranteeing stable performance over time even at the cost of slightly lower performance.

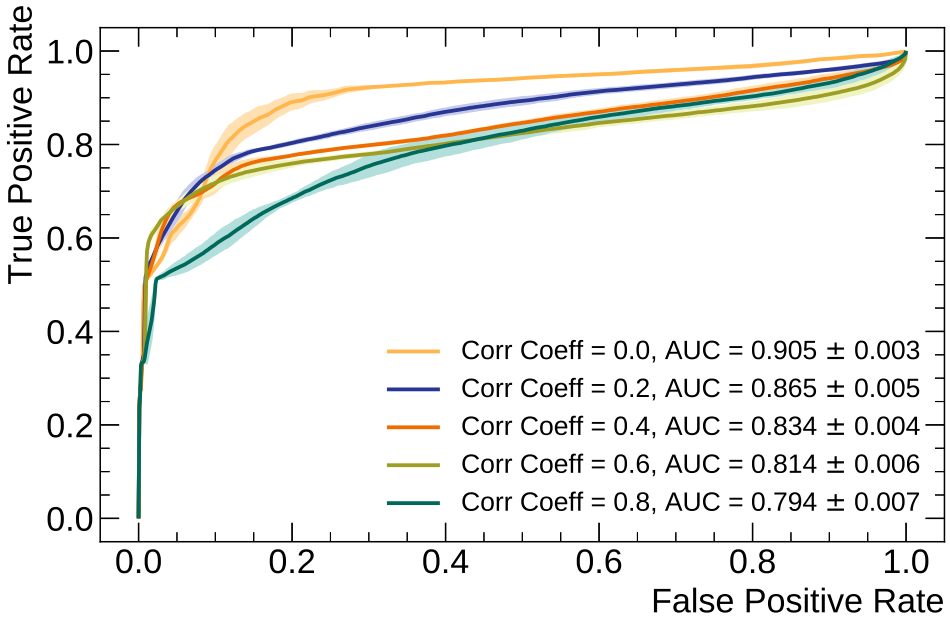

FIGURE 1.4: ROC curves for models using different input feature set. Feature selection does not improve overall ROC AUC, but changes the shape of the curve.

We evaluated performance of XGBoost (a supervised method), Isolation Forest and our sparse autoencoder as a function of time. Every 20% of chronologically sorted dataset, which constitutes approximately one month of data taking, all models were retrained using all available past data. Figure 1.5 shows performance evolution for each model calculated since last retraining time. The visible performance drops around 0.3, 0.4 and 0.65 are caused by appearance of novel problems. The autoencoder performance is less affected by those events than the performance of XGBoost. Nevertheless, fully supervised approach may still be a powerful extension to the current protocol as its performance is frequently superior.

1.4 Understanding Classification Results

Using the granularity of the MSE, the autoencoder reconstruction can be examined for each feature in a sample. The misbehaving variables whose contribution to the overall error is high can be singled out. This method provides additional way to interpret the results, in a simple human-consumable form. Figure 1.6 shows a visualization for such investigation with grouped features (according to different physical meaning). Using human expert knowledge those plots can map output of reconstruction to specific detector failures.

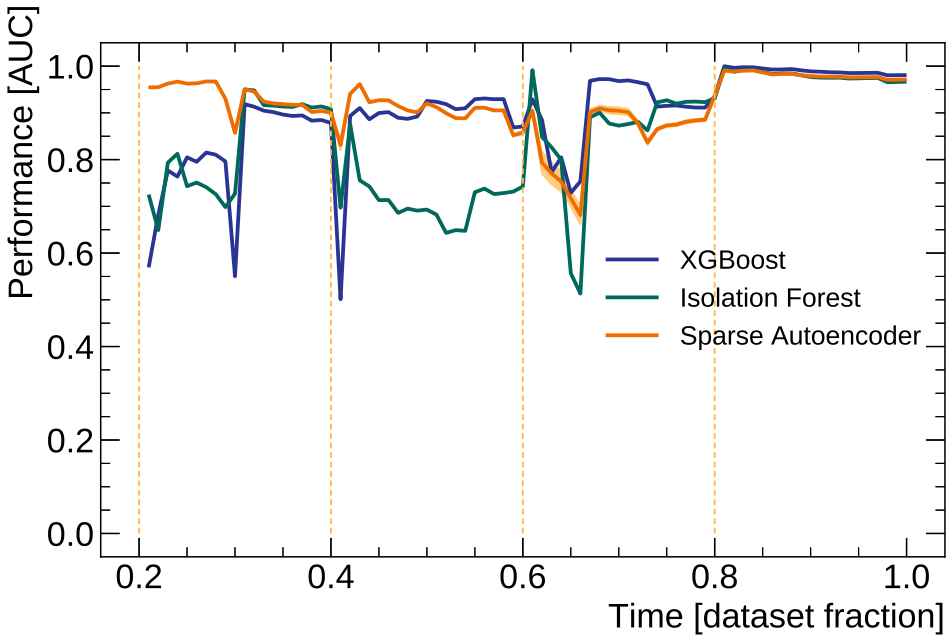

FIGURE 1.5: Performance of different strategies as a function of time. After each 0.2 of the dataset, each method is retrained on all past data points. AUC is reported since last retraining.

1.5 Conclusions and Practical Considerations

This work demonstrated that a semi-supervised anomaly detection using deep autoencoder based strategy can successfully produce certification flags. This comes without sufficient amount of anomalous examples and based on shorter interval then what is offered by current protocol. Finally, the algorithm has additional level of interpretability, specifying the root of the problem.

This approach was successfully qualified on CMS data collected during the 2016 LHC run. It paves the way to automating data quality assessment process with competitive specificity and sensitivity, when compared against the outcome of the manual certification by experts. However, it is not necessary to classify all the data without any human intervention. Instead the system can call for verification for questionable cases still limiting human labor required in reinforcement learning setup.

DC experts are validating the approach on more recently collected data from 2018 LHC run and coming from different PDs. Efforts to integrate the tool in the existing protocol as an assistance for human experts were undertaken. Yet, solving the luminosity dependence of the reconstruction remains a priority.

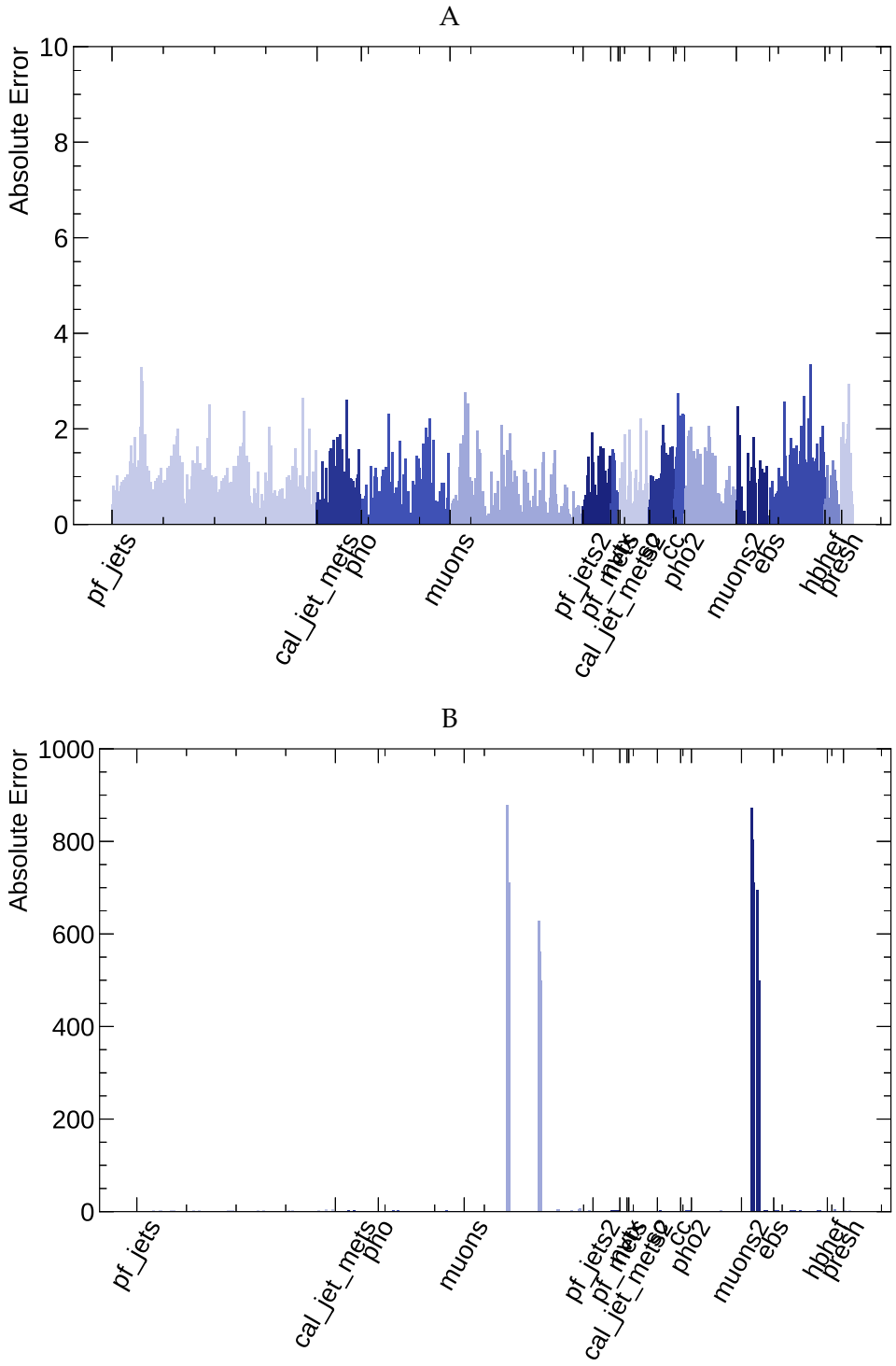

FIGURE 1.6: Reconstruction error of each feature for two samples. Different colors represent features linked to different physics objects. For a negative sample (A) we can expect similar amplitude across all objects with small absolute scale. Anomalous samples (B) have clearly visible peaks for problematic features (muons).

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... others (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Azzolini, V., Borisyak, M., Cerminara, G., Derkach, D., Franzoni, G., De Guio, F., ... others (2017). Deep learning for inferring cause of data anomalies. *arXiv preprint arXiv:1711.07051*.

Borisyak, M., Ratnikov, F., Derkach, D., & Ustyuzhanin, A. (2017). Towards automation of data quality system for CERN CMS experiment. In *Iop conf. ser j phys confer ser* (Vol. 898, p. 092041). doi: 10.1088/1742-6596/898/9/092041

Chollet, F., et al. (2015). *Keras.*

Della Negra, M., Foà, L., Hervé, A., & Petrilli, A. (2005). *Technical design report* (Tech. Rep. No. CERN/LHCC-2005-023). CMS computing.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings to iccv* (pp. 1026–1034). doi: 10.1109/ICCV.2015.123

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Ranzato, M., Poultney, C., Chopra, S., & LeCun, Y. (2006). Efficient learning of sparse representations with an energy-based model. In *Proceedings of the 19th international conference on neural information processing systems* (pp. 1137–1144).

Rapsevicius, V., et al. (2011). CMS Run Registry: Data certification bookkeeping and publication system. In *Journal of physics: Conference series* (Vol. 331, p. 042038).

Rezende, D. J., Mohamed, S., & Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st international conference on international conference on machine learning - volume 32* (pp. II-1278–II-1286).

Rifai, S., Vincent, P., Muller, X., Glorot, X., & Bengio, Y. (2011). Contractive auto-encoders: Explicit invariance during feature extraction. In *Proceedings of the 28th international conference on machine learning (ICML-11)* (pp. 833–840).