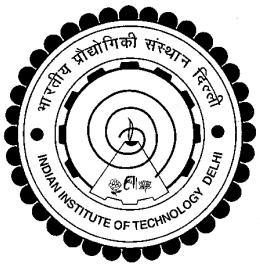

Dept of Physics, IIT Delhi

Course Code : PYD411
Academic Year : 2022 - 2023, Semester - I

Physics at the Large Hadron Collider: A Data Analytic Approach

Krish Vijayan (2019PH10634)
Ishaan Watts (2019PH10629)

Adviser: Prof. Abhishek Muralidhar Iyer

Abstract: The emergence of modern Machine Learning techniques such as Neural Networks Auto-encoders [1] can be widely used in Particle Physics [2]. This has commonly been known as Multivariate Analysis in High Energy Physics as well. One application of these methods is in the Large Hadron Collider, to classify certain events as anomalies using parameters such as the transverse momentum, pseudo-rapidity and invariant mass.

Signature of student 1: .....

Signature of student 2: .....

Signature of the adviser: .....

1 Introduction

The Large Hadron Collider at CERN, which is currently the world's largest and most powerful particle accelerator, has the potential to unlock interactions beyond the Standard Model (SM) of physics [3]. Several attempts have been made to use the data obtained from LHC to find gaps in the existing theories. This challenge of anomaly detection is quite famous [4, 5] and has a lot of scope for analysis.
The main aim of the project is to study the complex working of particle accelerators and utilize the huge amount of data produced (LHC's proton bunches cross paths over 10 million times per second, and each possible collision is a new event) to classify any deviations from the Standard Model. The discovery of an expected signal, such as the Higgs Boson, is associated with a p-value. Based on features such as the energy and momenta of various types of particles, the trained model should be able to assess whether a given event is an anomaly with a high level of confidence.

2 Theoretical Model

The constructed data-sets for the Neural Networks and Auto-encoder implemented can be described separately. The theoretical model can be divided into the following sub-parts:

2.1 Processes Involved

A typical process which contributes to the data-sets used in this project is the collision of a quark (q) and an anti-quark ($\bar{q}$) to become a photon (γ) which further decays into an electron ($e-$) and a positron ($e+$) in the final state.

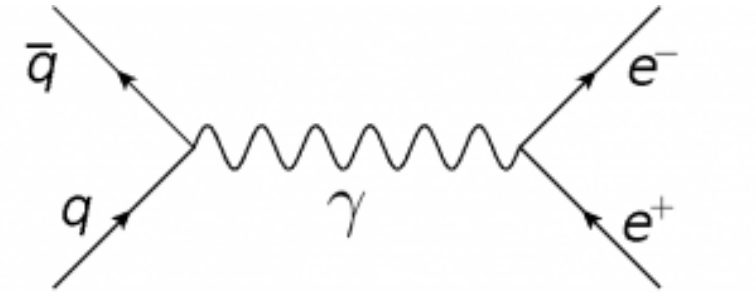

Figure 1: The Feynman Diagram of the process mentioned above.

The final state of this process is a set of four-vectors from which different features can be constructed, which include:

$$p_T = \sqrt{p_x^2 + p_y^2}, \tag{1}$$

$$\eta = -ln(tan(\frac{\theta}{2})). \tag{2}$$

Here p_T is the *transverse momentum*, η is the *pseudo-rapidity* and θ is the angle of deflection measured from the z direction.

In order to reject one data set in favour of the other, the difference in pseudorapidities of the final state particles is considered to be a distinguishing feature:

$$\Delta\eta = \eta_1 - \eta_1' \tag{3}$$

Another important metric which is relevant to the classification is the *Signal Discovery Significance* (Z) [6], given by:

$$Z = \sqrt{\sum_{i=1}^{N}(2(s_i + b_i)log(1 + \frac{s_i}{b_i}) - 2s_i)} \tag{4}$$

(summed over all the bins, s_i and b_i are the expected numbers for signal and background events in the i^{th} bin)

Using the mass and momentum, the *invariant mass* simplifies the HEP involved, since it is independent of the Lorentz frame. It is defined as:

$$m = \frac{\sqrt{E^2 - p^2c^2}}{c^2} \tag{5}$$

For multiple measurements of the process, one can determine whether a *Z-Boson* was formed by looking at the values for the invariant mass through statistical methods.

2.2 Generation of Data

The data-sets used for each of the 3 tasks are:

1. A data-set containing a *background* and a *signal* (bump) with $46,612$ and $4,601$ values of $\Delta\eta$ respectively is used to calculate the signal discovery significance.

2. The data-set used to train the Neural Network models to separate 2 distributions contains $122,854$ (Set-A) and $133,619$ (Set-B) values of $\Delta\eta$.

3. For anomaly detection with Auto-encoders, there are 5 training samples (each value represents the invariant mass) with the following details:

Data-set	**Number of Values**
Signal with peak at 0.5TeV	3982
Signal with peak at 1TeV	4409
Signal with peak at 2TeV	4600
Signal with peak at 3TeV	4602
Background Distribution	46611

Table 1: Invariant Mass Measurement Data-sets

For this task, more training data was also generated artificially by adding +-5% noise to the background bin values, such that new distributions follow a similar trend to the actual data, making the model robust to small disturbances. Also, this data was normalized to scale up the values to avoid any precision errors in the program, while keeping the prior distribution the same.

2.3 Neural Network Architecture

An *Artificial Neural Network (ANN)* [7], with one input layer, one hidden layer and an output layer is constructed. This system of interconnected nodes with weights and biases aims to decrease a particular loss function in order to get maximum accuracy. The prediction is done based on 2 classes - whether the particle belongs to the Set-A distribution or the Set-B distribution. The loss function for the network is *Binary Cross-Entropy (BCE) loss*, given by:

$$L(y,p) = -\sum_{i=1}^{m} \left(y_i \log(p_i) + (1-y_i)\log(1-p_i)\right) \tag{6}$$

where $y_i \in \{0,1\}$ is the actual class the i^{th} example belongs to and p_i is the probability with which the model predicts it to belong to the 1^{st} class. This loss function is summed over all the m samples to get the total loss, which is minimized over 500 epochs.

The architecture of the fully-connected neural network can be illustrated as follows, where the number of units in the hidden layer can vary (in this case it has 10 units):

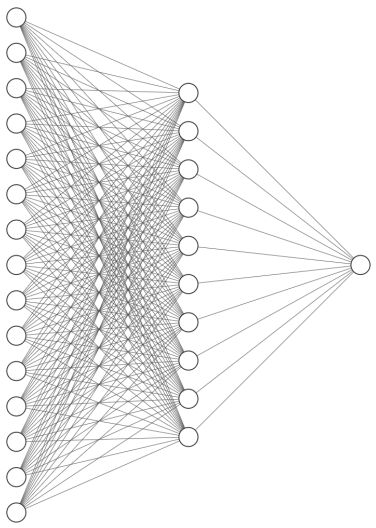

Figure 2: Neural Network Architecture: Input Layer, Hidden Layer and the Output Layer. The activation functions are implicit

The activation functions used between these layers are a Rectified Linear Unit (ReLU) and Sigmoid respectively. Mathematically, these functions are given by:

$$\sigma(z) = \frac{1}{1+e^{-z}}, \tag{7}$$

$$ReLU(z) = max(0,z). \tag{8}$$

The choices for the loss optimizer are the *Adam Optimizer*, *RMSProp* and *Stochastic Gradient Descent (SGD)* [**article2**], which are the most common choices for training.

The best values for the hyper-parameters (such as the number of hidden units, learning rate and choice of optimizer) are found using a grid-search framework *Optuna*, discussed in more detail in the results.

2.4 Auto-encoders

Auto-encoders [8] are a type of self-supervised neural networks in which the input and output are identical. They compress the input into a code with fewer dimensions and then reconstruct the output based on this representation. Also known as the latent-space representation, the code is a concise "summary" or "compression" of the input.

Encoder, code, and decoder are the three components that make up an auto-encoder. The encoder compresses the input and generates the code, while the decoder reconstructs the input using only the code. (Figure 3):

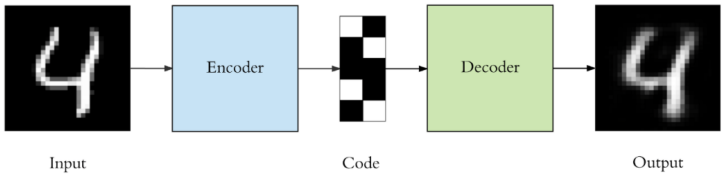

Figure 3: Auto-encoder Architecture

To construct an auto-encoder, we require three components: an encoding method, a decoding method, and a loss function to compare the encoded output to the desired output. The model is trained using the following architecture (Figure 4):

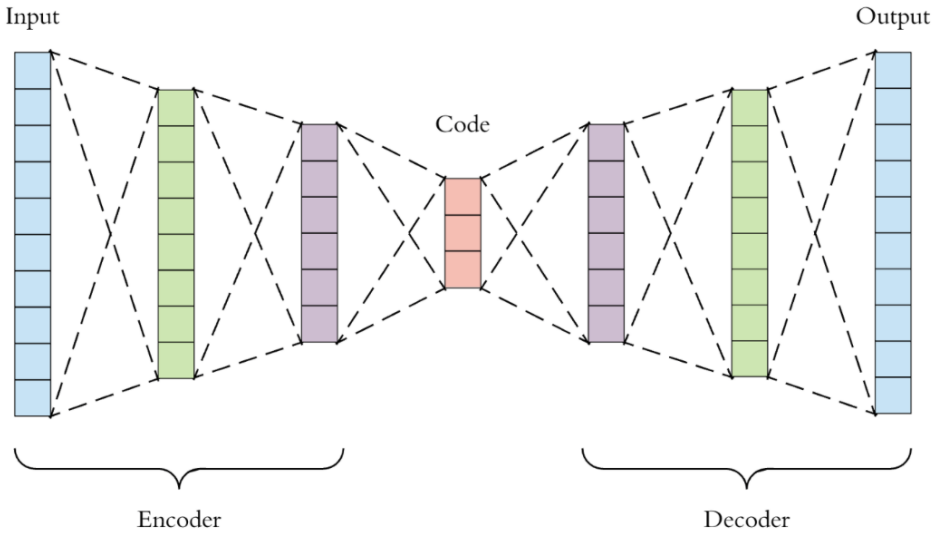

Figure 4: Layers of the Auto-encoder

The input and output layers, have 50 units (which correspond to number of bins in the distribution histogram), whereas the 'green' hidden layers have 32 units. The 'violet' hidden layers have 16 units and the Code has 8 units.
Each layer also has a ReLU activation function (Eq.7). The training takes place over 200 epochs, with a learning rate of 10^{-3} and Adam Optimizer. The loss function used is *Mean Squared Error (MSE)* given by:

$$L(y) = -\frac{1}{2m}\sum_{i=1}^{m}\sum_{j=1}^{50}(x_{ij} - \hat{x}_{ij})^2 \tag{9}$$

where x_{ij} is the j^{th} bin value of original input and $\hat{x}_{ij}$ is the j^{th} bin value of reconstructed output averaged over m training examples.

3 Results and discussion

3.1 Signal Discovery Ratio

The values from the first data-set (Data 1) are plotted in a histogram, which results in the following curve (Figure 5). The value of Z for the given distribution is calculated to be **2183.79**.

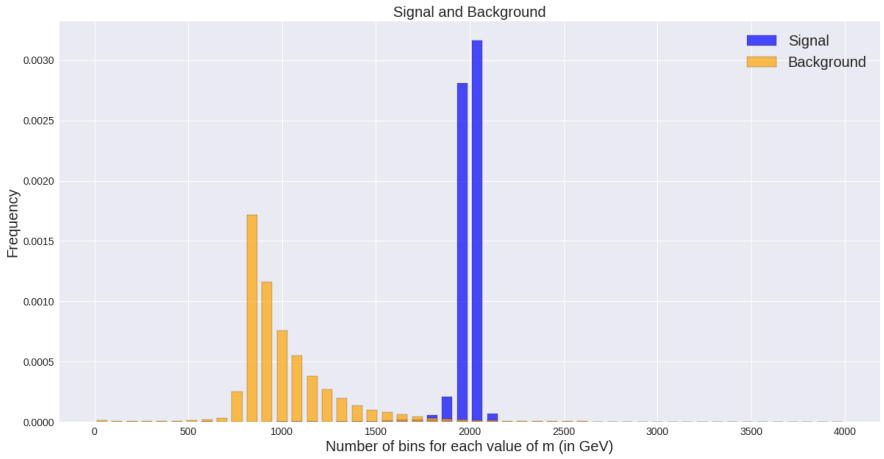

Figure 5: Histogram of the range of values of $\Delta\eta$ vs the frequency, after normalization

3.2 Neural Networks

Subsequently, values from the classification (Data 2) are grouped into 50 equal buckets based on their frequency, which yields the following plot of the data-set in order to get a clear visual representation of the distributions:

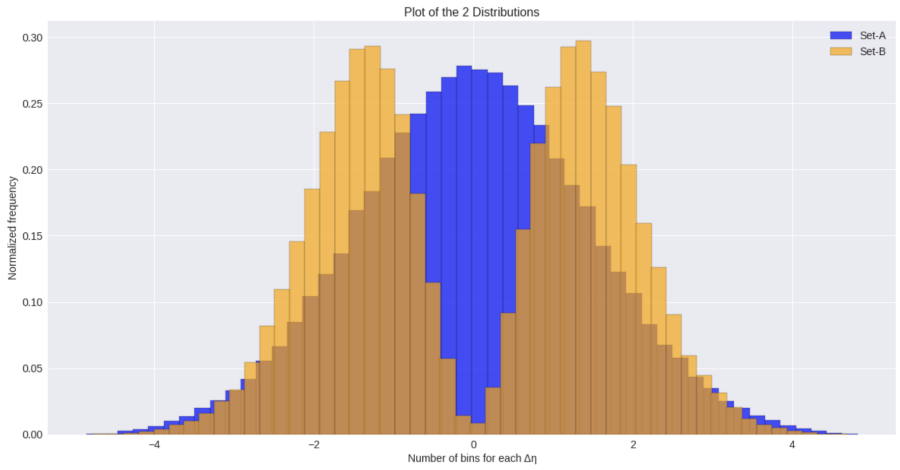

Figure 6: Histogram of the range of values of $\Delta\eta$ vs the frequency, after normalization

Tuning the hyper-parameters using *Optuna*, multiple trials (with 30 different values) were executed with a TPE Sampler (which uses a Gaussian Mixture Model) and the

accuracy for each trial was noted. Some of the notable results (1 for each optimizer) are:

Accuracy	Learning Rate	Number of Hidden Units	Optimizer
0.9697	0.0359	112	Adam
0.8834	0.0242	119	RMSprop
0.6589	0.1282	76	SGD

Table 2: Results of the hyper-parameter search, sorted in decreasing order of accuracy

The model with 112 hidden units, Adam optimizer and a learning rate of 0.036 performs the best on the training data, having an accuracy of **96.83% on the test data** as well. It is also observed that the overall accuracy increases as the number of units increase, along with the added advantage of the Adam Optimizer. On the other hand, SGD performs poorly.

For this model, the BCE loss varies with 500 iterations as shown in Figure 7. An ROC curve was also plotted for this model's predictions, which yields the following graph (Figure 8). The area under the curve is found to be **0.9973**.

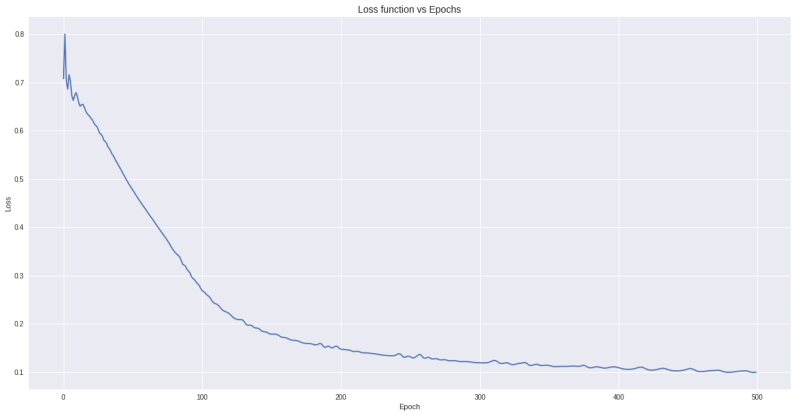

Figure 7: Loss function vs Epochs for the ANN

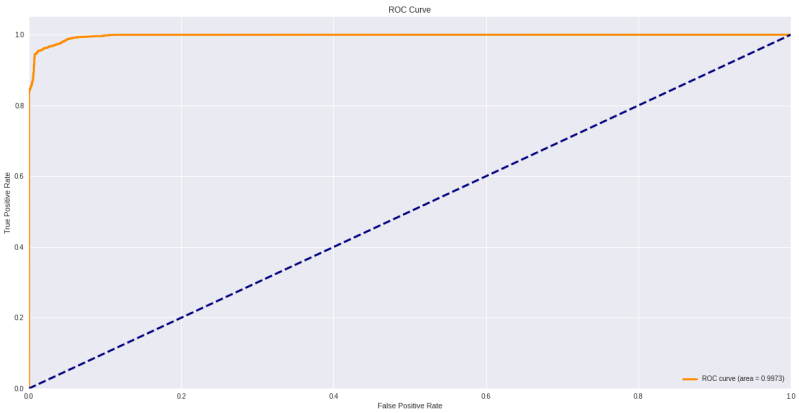

Figure 8: Plot of the *True Positive Rate (TPR)* vs the *False Positive Rate (FPR)*

3.3 Auto-encoders

For each of the signals in Table 1, the normalised histograms were plotted by grouping the values into 50 bins against the background, and the following plot was obtained:

Figure 9: Signals with peaks at 0.5,1,2,3 TeV with background

The model is trained on the input background and the new distributions generated by adding random noise to the background. The reconstruction loss of the model is also plotted with the number of iterations, which is a smooth decreasing curve indicating successful minimization of the MSE error:

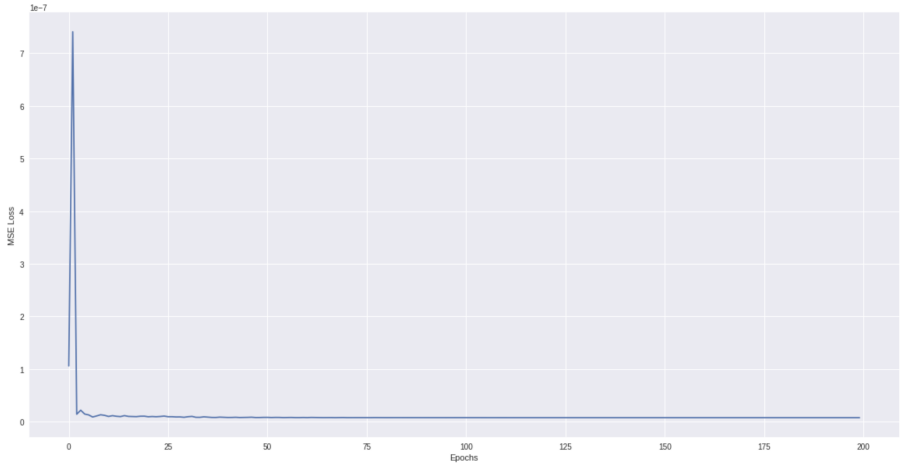

Figure 10: Loss function vs Epochs for the Auto-encoder

The auto-encoder learnt the reconstruction for the given data, without exactly learning the exact values of the input (which would just be an identity function). The reconstruction MSE loss for the original background is **7.13807408e-09**. We can see the reconstructed output in the following Figure 11:

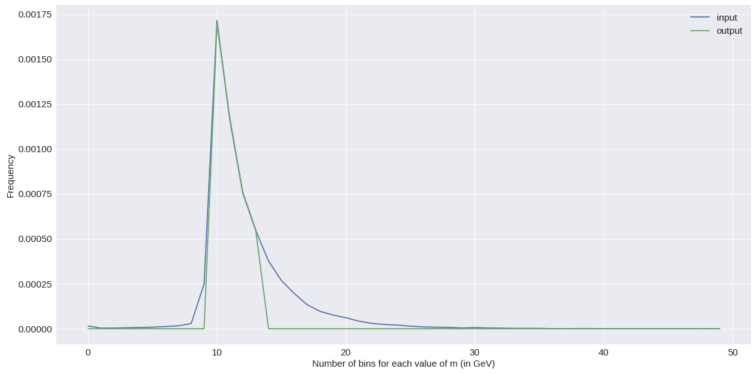

Figure 11: Background (Input) and its Reconstruction (Output)

Next a threshold value is calculated on the losses of the training examples by taking the mean of losses and adding the standard deviation. Calculated value is **7.138074077332806e-09**.
Now, we reconstruct the signal distributions by feeding them as input to the model. Since the model has only learned to reconstruct the background, it has high losses on the signal distributions. As these values are higher than the calculated threshold, the model classifies them as anomalies.

Input	Loss
0.5TeV	8.183e-07
1TeV	4.273e-07
2TeV	4.596e-07
3TeV	4.472e-07
Background	7.138e-09

Table 3: Loss for various inputs

From the Table 3 we can observe that the reconstruction losses for signals are greater than the reconstruction loss of background by a magnitude of 100. We have also plotted the reconstructions for the signal in Figure 12:

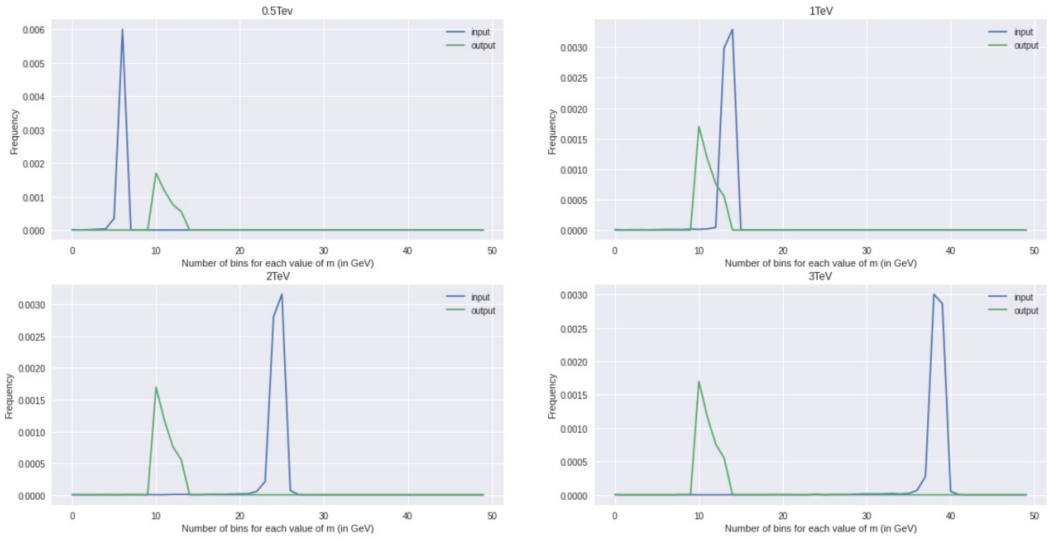

Figure 12: Signals (Input) and their reconstruction (Output)

4 Conclusions

1. A high value of Z indicates that for a sufficiently large background we can differentiate between signal and background curves.

2. We are able to distinguish between distributions with high accuracy in a Supervised Learning task using machine learning models.

3. Using unsupervised learning methods, it is feasible to identify anomalies in the data, allowing us to identify if a *z-boson* was formed in the process by learning the distributions.

   All the programs can be found in the PYD411 Google Colab Notebook (linked).

Acknowledgements

Prof. Abhishek Muralidhar Iyer deserves special thanks for introducing us to the problem description and guiding us through the procedure. We would also like to thank Mr. Shirsh Mall, an MSc. student working with Prof. Iyer, for sharing his expertise on the project.

References

[1] Marco Farina, Yuichiro Nakai, and David Shih. "Searching for new physics with deep autoencoders". In: *Physical Review D* 101.7 (Apr. 2020). DOI: 10.1103/physrevd.101.075021. URL: https://doi.org/10.1103%2Fphysrevd.101.075021.

[2] Matthew Feickert and Benjamin Nachman. *A Living Review of Machine Learning for Particle Physics*. 2021. DOI: 10.48550/ARXIV.2102.02770. URL: https://arxiv.org/abs/2102.02770.

[3] Tom W. B. Kibble. *The Standard Model of Particle Physics*. 2014. DOI: 10.48550/ARXIV.1412.4094. URL: https://arxiv.org/abs/1412.4094.

[4] Gregor Kasieczka et al. "The LHC Olympics 2020 a community challenge for anomaly detection in high energy physics". In: *Reports on Progress in Physics* 84.12 (Dec. 2021), p. 124201. DOI: 10.1088/1361-6633/ac36b9. URL: https://doi.org/10.1088%2F1361-6633%2Fac36b9.

[5] Thea Aarrestad et al. "The Dark Machines Anomaly Score Challenge: Benchmark Data and Model Independent Event Classification for the Large Hadron Collider". In: *SciPost Physics* 12.1 (Jan. 2022). DOI: 10.21468/scipostphys.12.1.043. URL: https://doi.org/10.21468%2Fscipostphys.12.1.043.

[6] F. Conventi et al. *A new test for non-universality at proton colliders*. 2021. DOI: 10.48550/ARXIV.2101.06088. URL: https://arxiv.org/abs/2101.06088.

[7] Enzo Grossi and Massimo Buscema. "Introduction to artificial neural networks". In: *European journal of gastroenterology hepatology* 19 (Jan. 2008), pp. 1046–54. DOI: 10.1097/MEG.0b013e3282f198a0.

[8] *Applied Deep Learning - Part 3: Autoencoders*. https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798.

2019PH10634_PYD411_ER.pdf

ORIGINALITY REPORT

10%	9%	5%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to uwe Student Paper	2%
2	archives.univ-biskra.dz Internet Source	2%
3	Submitted to IIT Delhi Student Paper	1%
4	ebin.pub Internet Source	1%
5	Submitted to University of Sheffield Student Paper	1%
6	arxiv.org Internet Source	1%
7	Bruno Cavalcante de Souza Sanches. "An application specific signal processor for gaseous detector systems in high energy physics experiment.", Universidade de Sao Paulo, Agencia USP de Gestao da Informacao Academica (AGUIA), 2021 Publication	1%

8	www.dtic.mil Internet Source	1 %
9	www.frontiersin.org Internet Source	1 %
10	authors.library.caltech.edu Internet Source	1 %
11	pubmed.ncbi.nlm.nih.gov Internet Source	1 %

Exclude quotes	On	Exclude matches	< 10 words
Exclude bibliography	On