LIP-STUDENTS-24-7

Searching for new physics phenomena in ATLAS data using machine learning

Carolina Sobral[1,a], João Pedro Barros[1,b], Gabriel Lopes[1,c] and Filipe Marinho[1,d]

[1] *LIP and University of Minho, Braga, Portugal*

Project supervisor: Nuno Filipe Castro

November 1, 2024

Abstract. The ATLAS detector is one of the major detectors at LHC (Large Hadron Collider), where one can study and search for new physics phenomena beyond the Standard Model of particle physics (SM).The ATLAS Open Data project provides a series of datasets and analysis codes that can be used by the public to learn about high-energy physics or even to carry out research. A data analysis of new physics events contains simulation data, the signal, as well as data from phenomena already described by the standard model (SM) also simulated, the background of the analysis. We created a neural network to distinguish between signal and background, and used supervised machine learning to train the neural network to distinguish between these two classes of events.

KEYWORDS: ATLAS Open Data, ZPrime, Background, Signal, Neural Network

1 Introduction

The aim of this work is to explore the potential of machine learning tools in analyzing data related to new physics phenomena, specifically beyond the Standard Model. It is also given some basic notions in the dataset analysis, as it also constituted an important part of this work.

The project followed a structured approach, beginning with an examination of plots from an ATLAS open data analysis of the beyond-the-Standard-Model process, Z' $\rightarrow$ $t\bar{t}$, obtained in a virtual environment. The analysis data was converted into comma-separated values (CSV) files for processing in a Python notebook. Once the data was in this format, code was developed to recreate the same plots observed in the virtual machine.

To build a neural network, it is important to first understand which variables exhibit strong correlations. The data in the CSV's files were then separated in three equal groups: one for training, one for validation and the other for testing in equal proportions. By applying supervised machine learning techniques, a neural network was developed that effectively distinguishes between the signal and background.

2 Theoretical Background

2.1 The Standard Model of particle physics (SM)

The SM is the model that describes all known fundamental particles, as well as three of the four fundamental forces, the electromagnetic, weak (together make the electroweak force), and strong forces. Particles are typically classified by their spin: fermions have half-integer spin, while bosons have integer spin. Every particle is either a boson or a fermion. In this model, fermions are separated into two different classes depending on which different forces they can interact with other particles. Quarks are particles that can interact via electromagnetic and strong forces, whereas leptons can also interact by electromagnetic and weak forces. Bosons are the particles responsible for carrying the forces. The photon is the mediator of the electromagnetic force, the gluon is the mediator of the strong force and the Z and W bosons are responsible for the weak force. The Higgs boson does not mediate any force, but couples with masses, giving every elementary massive particle its rest mass.

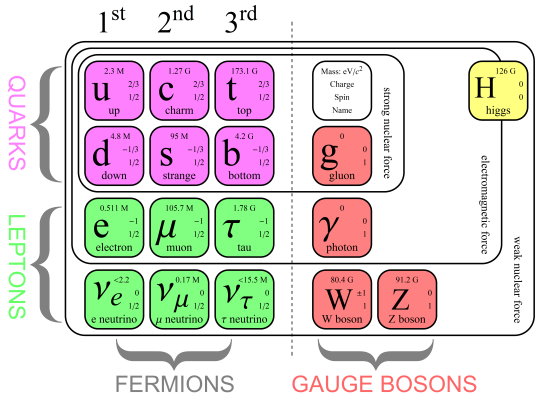


Figure 1: Standard model of paticle physics [1]

2.2 Physics Beyond the Standard Model (BSM)

Even though the SM can describe with great accuracy the interactions of fundamental particles, there are still some loose ends to the full picture of the theory. These are some of the physical phenomena that the SM does not explain:

- The matter anti-matter asymmetry in the universe;
- The neutrino mass: the neutrino's mass can be put by hand in the SM, but we still do not know the correct mechanism that originates it;
- Dark matter and dark energy: The SM only describes 5% of our universe (all the matter and anti-matter), but it does not provide any fundamental particle to explain dark matter.

[a] e-mail: carolinajardim222@gmail.com
[b] e-mail: joaopedromartins.barros@gmail.com
[c] e-mail: pg54378@alunos.uminho.pt
[d] e-mail: filipecerqueiramarinho@gmail.com

To try to explain these shortcuts of the SM there is the need to look for physics beyond the Standard Model (BSM).

2.3 The Large Hadron Collider (LHC) and the ATLAS Experiment

The LHC, the world's largest particle accelerator located at CERN, accelerates particles (typically protons or heavy ions) in a 27-kilometer tunnel, enabling high-energy collisions. ATLAS, one of four major detectors at the LHC, was pivotal in the discovery of the Higgs boson in 2012 and facilitates the study of fundamental forces and the search for new physics beyond the Standard Model, such as dark matter and supersymmetry.

The ATLAS detector[2], 46 meters long and 25 meters in diameter, has several layers for detecting particles and measuring their properties, including:

- Inner Detector: it tracks the path of the charged particles to determine their trajectories and momenta.
- Electromagnetic and Hadronic Calorimeters: which measure the energy deposited by a particle.
- Muon Spectrometer: where the muons that pass through most of the detector are tracked.
- Magnetic fields: they bend the paths of charged particles to help measure their momentum.
- Trigger System: it filters the data, selecting the events that could be interesting to analyze.

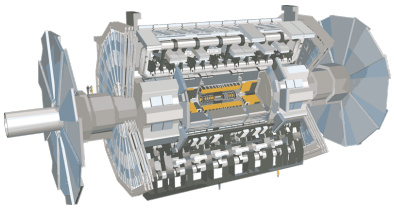


Figure 2: ATLAS detector [3]

In ATLAS, a right-handed coordinate system is used for tracking particles, with the x-axis pointing toward the LHC center and the z-axis aligned along the tunnel. Isolated electrons typically originate from Z or W bosons, or τ leptons, while non-isolated electrons are often produced by hadron decays.

The isolation is determined using the pseudorapidity η and the azimuthal angle ϕ, which is measured around the beam axis. The ATLAS detector is designed to be symmetrical in ϕ.

In proton-proton collisions, the exact momentum along the beamline is unknown because the interacting partons (quarks and gluons) carry an unknown fraction of the proton's momentum. However, the total momentum perpendicular to the beamline is zero before the collision. As a result, the transverse momentum p_T, the momentum component perpendicular to the beamline, becomes the most important variable in such collisions.

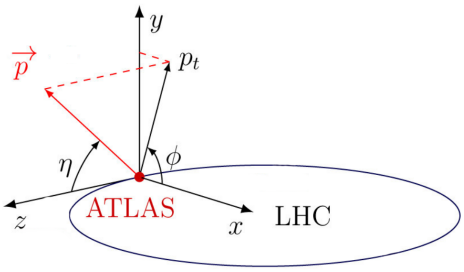


Figure 3: ATLAS detector coordinates [4]

Both the LHC and ATLAS have advanced human understanding of the universe's most fundamental building blocks. The Higgs boson discovery was a monumental achievement, but there are many unanswered questions that ATLAS and the LHC continue to investigate.

2.4 Machine Learning [5]

Machine learning is a branch of artificial intelligence (AI) that enables computers to learn from data and improve their performance on specific tasks. It involves developing models that recognize patterns and make predictions or decisions based on new data.

2.4.1 Supervised Learning with Neural Network

One of the most powerful and commonly used machine learning models is the neural network. Inspired by the structure of the human brain, neural networks consist of interconnected layers of nodes, or "neurons." Each neuron receives inputs, processes them, and passes the results to the next layer. Through a process called "training," the network adjusts the connections between neurons based on the error of its predictions, using algorithms such as backpropagation.

Since this study involves two types of data (signal and background), we will employ a machine learning approach known as supervised learning. Supervised learning is a method where a model is trained on a labeled dataset, meaning the data includes both input features and their corresponding correct outputs, or "labels." The model learns to associate the input data with the labels by identifying patterns within the dataset. The primary objective of supervised learning is to enable the model to generalize from the training data, allowing it to make accurate predictions when faced with new, unseen data.

3 Data and Methods

3.1 ATLAS Open Data

The ATLAS Open Data platform [6], provided by CERN, offers comprehensive datasets and tools for analyzing particle physics experiments. It is accessible to the public, enabling the study of high-energy physics (HEP).

In this work, we will focus on the 13 TeV dataset, collected by the ATLAS detector at the LHC in 2016, corresponding to an integrated luminosity of 10 fb^{-1}. The dataset includes proton-proton (pp) collision data, along with Monte Carlo (MC) simulated samples modeling both Standard Model (SM) and beyond Standard Model (BSM) processes. These simulations are used to predict signal and background distributions.

The SM simulations cover top-quark pair production, single-top production, W+jets and Z+jets, diboson production (WW, WZ, ZZ), and Higgs production. Additionally, BSM processes such as heavy Z' and SUSY production are simulated. Both the collision data and MC samples undergo the same reconstruction algorithms and analysis processes, with a loose preselection applied to streamline processing.

3.2 C++ Data Processing

As previously mentioned, this study is conducted using ATLAS Open Data. To facilitate the analysis of the ZPrime, we utilized VirtualBox to run the ATLAS virtual machine. The codebase is written in C++ and ROOT, a programming language developed by CERN to assist scientists in their research. The framework [7]is divided into two main components:

- The "Analysis" directory: it performs the particular object selection and stores the output histograms;
- the "Plotting" directory: it makes the final Data / Prediction plots.

Within the Analysis component, a comprehensive code applies multiple cuts to the data obtained from ATLAS.

Our objective is to export the information from these plots into a different format, such as CSV, in order to process it further in Python, where we aim to develop a neural network to study the dataset.

3.3 Z' Boson Decay to Top Quarks

Various BSM physics models predict the existence of a gauge boson with similar couplings to the standard Z-boson, but with higher mass, which is called the Z'-boson. The existence of this kind of boson could help to explain the existence of Dark Matter. There are many possible ways for the decay of the Z'-boson, but in this paper, we focus on the Z' $\rightarrow$ $t\bar{t}$ decay.

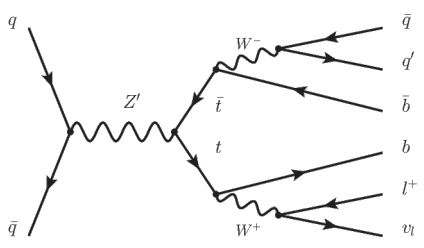


Figure 4: Z' $\rightarrow$ $t\bar{t}$ decay [8]

3.3.1 ZPrime boosted analysis and selection cuts

Any ATLAS analysis starts with objects reconstruction and identification. The dataset from open data used for this analysis is called '1largeRjet1lep', which means that it contains events with at least one large-R jet and a lepton. One can look at the decay represented in Fig. 5 and see that this description exactly represents this decay chain. A large R-jet a group of smaller quarks jet that, do to high momentum of the particle that decays into this quarks, can be treated as a single jet of a higher final radius and cross-section. The fact that the Z' has a mass in the order of the TeV is a reason why we can get the production of this jets.

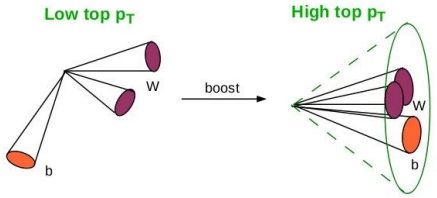


Figure 5: Representation of a large-R jet [9].

An analysis requires the reconstruction of objects from the data collected from the detector. This section contains an example of how to reconstruct masses with the ATLAS dataset. First of all, for a certain analysis it is needed to define the selection cuts. This cuts work as a filter of a number of variables in order to obtain a certain final result, for example, a selection of a certain value for missing transverse energy (E_T^{miss}) and transverse momenta (p_T). For this analysis, the selection cuts were the following [10]:

- Single-electron or single-muon trigger satisfied;
- Exactly one lepton (electron or muon) with p_T > 30GeV;
- E_T^{miss} larger than 20 GeV and $E_T^{miss} + M_T^W > 60$ GeV;
- At least one small-R jet close to the lepton, i.e. with ΔR(small-R jet, lepton) < 2.0;
- Exactly one large-R jet, passing the simplified top-tagging requirements: a mass larger than 100 GeV and N-subjettiness ratio $\tau_{32} < 0.75$;
- This top-tagged large-R jet must be well separated from the lepton, $\Delta\phi$(large-R jet, lepton) > 1.0, and from the small-R jet associated with the lepton, ΔR(large-R jet, small-R jet) > 1.5;
- At least one b-tagged (MV2c10 @ 70 % WP) small-R jet that fulfils the following requirements: it is either inside the top-tagged large-R jet, ΔR(large-R jet, b-tagged jet)< 1.0, or it is the small-R jet associated with the lepton.

3.3.2 Mass reconstruction in ZPrime analysis

Starting in the lower decay branch from the Z' decay (Fig. 5) in can reconstruct the W-boson mass in the following way [11]:

$$(\mathbf{P}_W)^2 = M_W^2 c^2 \quad (1)$$

$$M_W^2 c^2 = (\mathbf{P}_l + \mathbf{P}_\nu)^2 \quad (2)$$

$$\implies M_W^2 c^2 = M_l^2 c^2 + M_\nu^2 c^2 + 2\mathbf{P}_l \cdot \mathbf{P}_\nu \quad (3)$$

Note that in this procedure it was ignored any mass contribution from the neutrino. Adding this do the small-R jet 4-momenta, it is possible to reconstruct the top quark mass. For the upper decay branch the situation is a little different, because it is just a large-R jet, the is no lepton to deal with. In this case, it is just needed to select the large-R jet accordingly with the selection cuts mention previously we can get its 4-momenta, as the squared of that quantity is the anti-top mass. Adding the masses of both top and anti-top quarks it is obtained the Z' mass, which distribution is shown in Fig. 6a.

3.4 Python Data Processing

Using the CSV files containing all the essential information, we proceeded to replicate the C++ code [12] in Python. Additionally, we recreated the various graphics that were generated by the virtual machine.

First, the datasets for signal (ZPrime) and background processes were loaded from CSV files. Background samples were consolidated, and event weights were calculated using cross-sections, integrated luminosity, and sum of weights. We then normalized the data by applying weight adjustments based on luminosity and cross-section of the samples, crucial for comparing different datasets on equal footing. Event yields and their corresponding uncertainties were calculated for signal (ZPrime) and background categories, including diboson, single top, top, and V+jets processes.

Sample	Events	Entries
Diboson	22.34 ± 1.0	3322
ZPrime	426.62 ± 5.33	8375
Single Top	610.78 ± 7.07	9470
Top	13600.48 ± 37.09	154765
V + jets	682.68 ± 5.94	73739

Table 1: Event Yields and Entries by Sample

With the calculated yields, all necessary components are available to recreate the Open Data graphs generated in ROOT. As shown in Fig. 6, both graphs are validating the implemented python framework.

4 Machine Learning Approach

4.1 Data Preprocessing and Feature Selection

To prepare the data for training the neural network, we begin by organizing the dataset, with an initial total of 141 columns. Our first step in feature selection involved filtering the dataset by retaining only numerical columns and removing any columns where all values were zero. Additionally, we excluded features with fewer than 1.500 non-zero values, reducing the dataset to 108 columns.

Given that this is still a high-dimensional dataset, further reduction was necessary. To resolve this, we performed a correlation analysis by calculating the correlation matrix for all 108 features, (see Appendix 7.1). This allowed us to understand the relationships between variables.

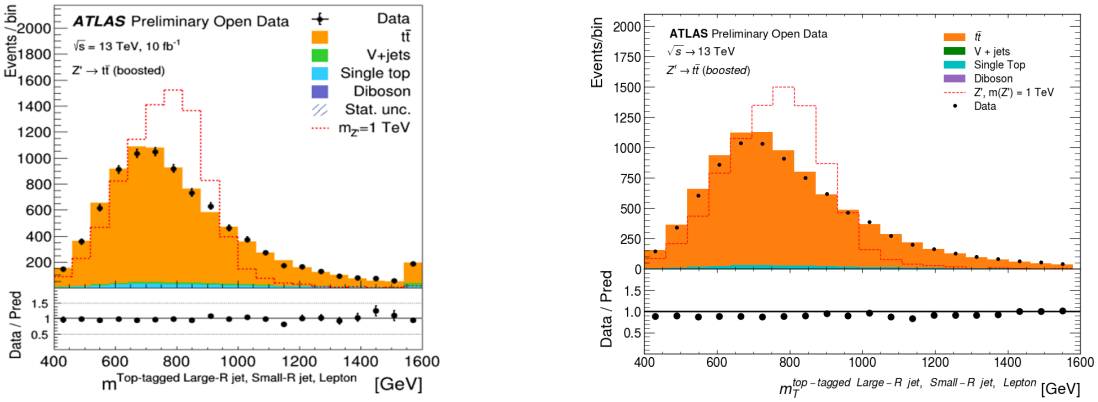
To focus on the most relevant relations, we applied a mask, like a threshold, to the correlation matrix, filtering it to display only correlations above 0.4 (see Appendix 7.2). This enabled us to identify features with stronger interdependencies.

From our correlation analysis, and from the previously study of the variables, we identified the most most significant variables to include in the neural network. We know that TTbar_M and met_et are key features on out study, so we did an analysis on their correlations with other variables.

Based on this correlation study and our previous filtering steps, we determined that the most strong correlations and relevance to the physical processes under investigation, for the neural network would be the following 32 (see Fig. 7 :

- TTbar_pt, TTbar_E, TTbar_M
- top_pt, top_E
- met_et
- largeRjet_n, goodbjet_n
- lep_pt, lep_E
- jet_pt0, jet_E0, is_good_after0
- jet_pt1, jet_E1, jet_pt2, jet_E2, jet_pt3, jet_E3
- largeRjet_pt0, largeRjet_E0, largeRjet_m0, is_top0
- largeRjet_pt1, largeRjet_E1, largeRjet_tau321, largeRjet_m1, is_top1
- largeRjet_pt2, largeRjet_E2, largeRjet_m2
- goodjet_afterdRlep_n

Additionally, we generated graphs for all 32 features, differentiating between signal and background (see Appendix 7.3). The goal was to examine the disparities between signal and background, highlighting the features with the most significant distinctions. Noteworthy examples of such features include "top_pt" and the "largeRjet".



(a) (b)

Figure 6: Comparison of ZPrime mass graphs: (a) Graph generated in the ATLAS Virtual Machine using C++ and ROOT; (b) Reconstruction of the graph in ROOT using Python. Graph a) presents an small pike in the end 1600 GeV which represents the rest of the data that appears above that mass. This is not the case for the graph b) because this was not taken into account for its generation.

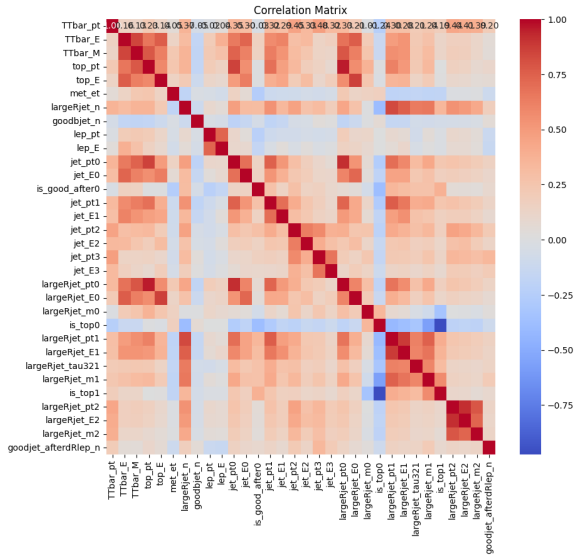


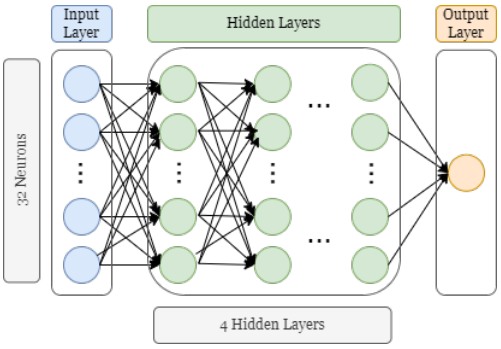
Figure 7: Correlation Matrix for the chosen 32 features

4.2 Data Splitting and Weight Normalization

To train and evaluate our neural network, it is important to divide the dataset into distinct subsets: training, validation, and testing. This approach allows the model to learn from one segment of the data while being evaluated on unseen data, thus providing an accurate assessment of its performance.

The dataset is then divided into three approximately equal parts. Subsequently, we compute the normalized weights for each subset based on the distribution of event labels. For each unique label in the dataset, we carry out weight normalization to ensure that the weights sum to one within each split. This normalization is essential for reducing bias during training, especially when dealing with imbalanced classes.

4.3 Model Training



We are now ready to construct our neural network. We defined a neural network architecture comprising three distinct components:

1. **Input Layer:** The input layer consists of 32 neurons, each representing one feature from the dataset. This layer serves as the entry point, taking in preprocessed data and forwarding it through the network, and by assigning each feature to a distinct neuron, we preserve all relevant information for learning.

2. **Hidden Layers:** we create four fully connected hidden layers, each with 32 neurons. These hidden layers are equipped with ReLU (Rectified Linear Unit) activation functions, which help in handling complex patterns by introducing non-linearity. This also mitigates the vanishing gradient problem, making the training process more efficient.

3. **Output Layer:** The final layer has a single neuron that outputs a value between 0 and 1, thanks to the Sigmoid activation function applied here. This is suitable for binary classification tasks, where the goal is to distinguish between signal and background events. The output represents the probability that the input data belongs to the signal class.

To prepare the input dataset for our neural network, we first shuffle all the elements. After that, we separate the signal (label) from the background. The network will only be provided with the background data, while the signal will serve as the true label for comparison during training. We will utilize binary cross-entropy (BCE) as the loss function, with added weighting to account for the imbalance between the signal and the background. The weights applied will be 0.2 for the background and 5.0 for the signal, reflecting their respective importance in the loss calculation. For optimization, we will employ the Adam optimizer, with a learning rate of 0.001, betas set to (0.9, 0.999), and a weight decay of 0.005.

$$\text{BCE Loss} = -w_n\{-\frac{1}{N}\sum_{i=1}^{N}[y_i \log(p_i) + (1-y_i)\log(1-p_i)]\} \tag{4}$$

The model training process is carried out over 300 epochs. During each epoch, the model is set to training mode, and the optimizer's gradient values are reset to zero to ensure fresh gradient updates. The model then makes predictions based on the input data, and the loss is calculated using the specified loss function and the the labels. Afterward, backpropagation is performed, with the loss gradients being computed and propagated through the network. The optimizer then updates the model's parameters accordingly.

5 Results and Discussion

5.1 Results for the Training Model

After training our neural network for 300 iterations, we observed a consistent decrease in the training loss. The training process achieved a final loss value of approximately 0.19, indicating that the model successfully learned from the data over the course of the epochs.

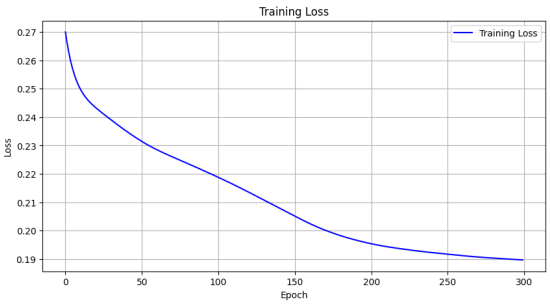


Figure 8: The Training Loss

Next, we evaluated the model's performance by generating the ROC (Receiver Operating Characteristic) curve, which is a graphical plot that illustrates the diagnostic ability of a binary classifier by plotting the true positive rate (sensitivity) against the false positive rate at various threshold settings. The AUC (Area Under Curve) is a single scalar value representing the overall ability of the model to distinguish between the signal and background, with a value of 1 indicating perfect separation and 0.5 indicating no discriminative power. In this case, our model achieved an AUC of 0.82, demonstrating a reasonable level of discrimination between the signal and background (see Fig. 9).

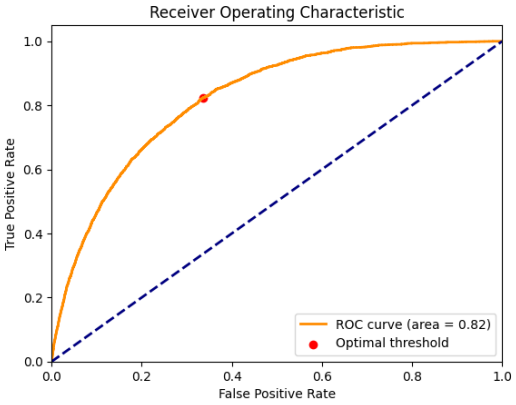


Figure 9: The ROC (Receiver Operating Characteristic) curve and AUC (Area Under Curve)

The optimal threshold for classifying between the two was identified as shown on the ROC curve, achieving the following performance metrics:

- **Accuracy:** 0.834
- **Precision:** 0.118
- **Recall:** 0.590
- **F1 Score:** 0.197

The confusion matrix, shown below, further illustrates the model's classification performance:

$$\begin{bmatrix} 67746 & 12616 \\ 1172 & 1690 \end{bmatrix}$$

This indicates that 67,746 background events were correctly classified, while 16,306 signal events were misclassified.

These results indicate that the neural network model effectively distinguished between signal and background in the dataset. The ROC curve, showed that the model has a strong ability to differentiate between the two classes, though there is still room for improvement, particularly in increasing the precision and F1 score. The precision of 11.81% indicates that while the model has a high recall (59.05%) for identifying the signal, many of the positive classifications were false positives. The relatively low F1 score (0.197) reflects this trade-off between precision and recall, emphasizing the need to fine-tune the model further to achieve a more balanced classification performance.

5.2 Results for the Validation Model

The evaluation of our trained neural network was performed on the validation dataset to assess its performance. The model's accuracy on the validation set was 71%. This

indicates that the model has learned meaningful patterns in the data, but further improvements can be made. We also generated a validation loss graph to analyze the model's progress throughout the training process (see Fig. 10).

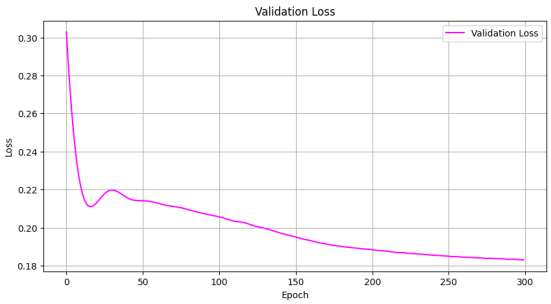


Figure 10: The Validation Loss

To better understand the model's ability to distinguish between signal and background events, the predictions were categorized into these two classes.

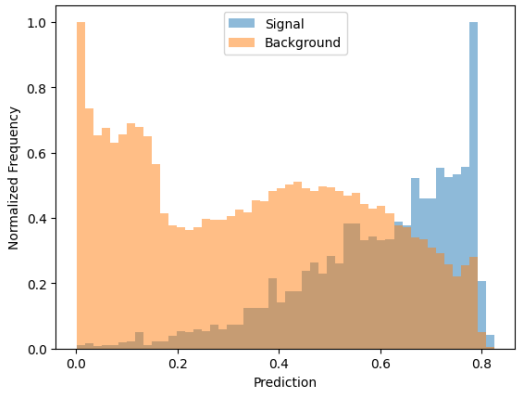


Figure 11: Normalized Distribution of Model Predictions for Signal and Background Events

The x-axis represents the predicted probability that an event is classified as a signal, ranging from 0 to 1. Predictions closer to 0 suggest the event is more likely to be classified as background, while values approaching 1 indicate a higher likelihood of being classified as signal. The y-axis shows the normalized frequency of predictions within each class.

The blue histogram illustrates the distribution of the model's predictions for signal events, with most predictions clustered between 0.7 and 0.9. This suggests the model accurately classifies many signal events with high confidence, demonstrating that it has effectively learned to identify key features of the signal class. On the other hand, the orange histogram represents background events, which are largely predicted with probabilities near 0. This indicates strong confidence in the model's ability to correctly identify background events as non-signal, or noise.

6 Conclusion

In this study, we successfully developed a machine learning model capable of distinguishing between signal and background events in ATLAS data, with a reasonable degree of accuracy. The results indicate that the model performs well, particularly in cases where the predictions are at the extremes (closer to 0 for background and closer to 1 for signal). However, some overlap in the mid-range of predictions (see Fig. 11) suggests that there are areas where classification could be further improved.

The training and validation loss curves provide insights into the model's learning process, showing a decrease in both losses over time, which suggests the model is effectively minimizing error during training. An unusual observation is that the validation loss remains consistently lower than the training loss, contrary to the typical pattern where training loss is expected to be lower due to the model's exposure to the training data. This unexpected behavior call for further investigation to identify its cause.

While this model shows the ability to separate signal from background, improvements are needed through optimization of the neural network architecture and advanced hyperparameter tuning. Further adjustments, such as trying different neural network designs, fine-tuning hyperparameters, or applying better feature selection and regularization methods, could help reduce errors and improve accuracy.

It is also important to note that while the current model can distinguish between signal and background, more work is required to use this approach to search for new physics, such as detecting a hypothetical Z' boson. Future research should focus on improving the model's precision and performance to better explore these new physics possibilities.

Acknowledgements

We would like to begin by expressing our gratitude to Professor Nuno Castro for the incredible opportunity provided to us. Through this experience, we have gained valuable knowledge in both particle physics and machine learning — subjects that will be essential for our futures. We are truly thankful for the chance to learn from you.

We also extend our appreciation to the entire LIP Minho team for their guidance throughout the internship, particularly Maura Barros, whose support and assistance during our summer internship were invaluable.

We acknowledge the work of the ATLAS Collaboration to record or simulate, reconstruct, and distribute the Open Data used in this paper.

References

[1] T. Saleem, Ph.D. thesis, SOLEIL synchrotron (2019)

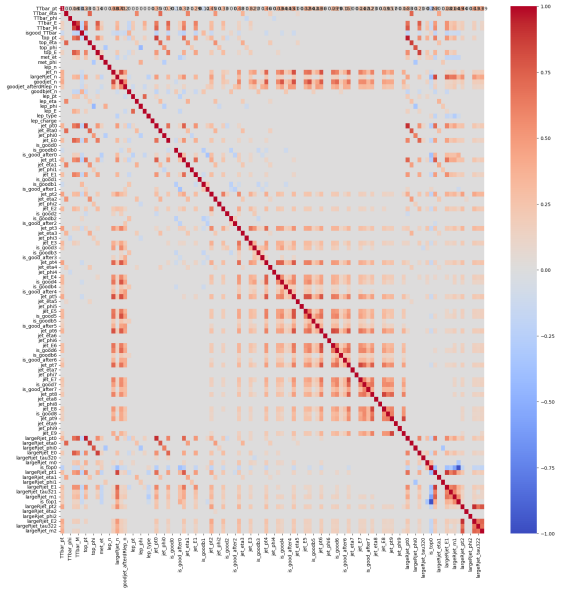
[2] ATLAS Collaboration, Journal of Instrumentation (2008)

[3] ATLAS Colaboration, https://atlas.cern/Discover/Detector, [Online]

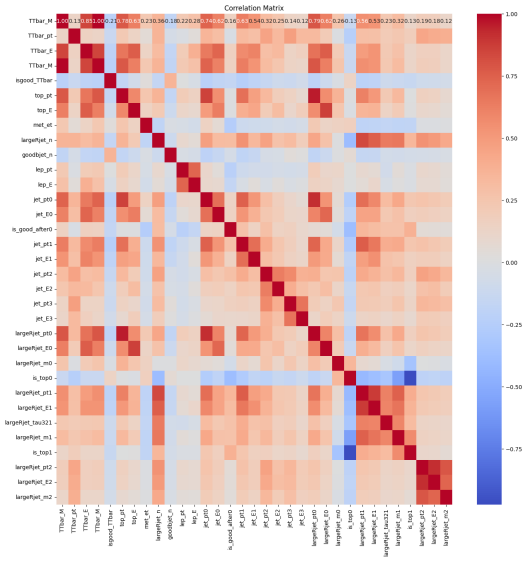
[4] G. Strong, Machine Learning: Science and Technology **1** (2020)
[5] A. Burkov, *The hundred-page machine learning book* (2019)
[6] ATLAS Collaboration, CERN Open Data Portal (2020)
[7] F. Ould-Saada, Epj Web of Conferences (2020)
[8] J.D. (https://physics.stackexchange.com/users/327166/jdelaney), *Why do z or h bosons can decay to a pair of quarks $t^{+}t$ with mass larger than boson in the inertial reference frame of boson?*, Physics Stack Exchange, `https://physics.stackexchange.com/q/692475`
[9] ATLAS Colaboration, https://cds.cern.ch/record/1744224/files/ (2014), [Online]
[10] ATLAS Collaboration, CERN Document Server (2020)
[11] D. Griffiths, *Introduction to Elementary Particles*, 2nd edn. (WILEY-VCH, 2008)
[12] ATLAS Colaboration, https://github.com/atlas-outreach-data-tools/atlas-outreach-cpp-framework-13tev (2020), [Online]

7 Appendix

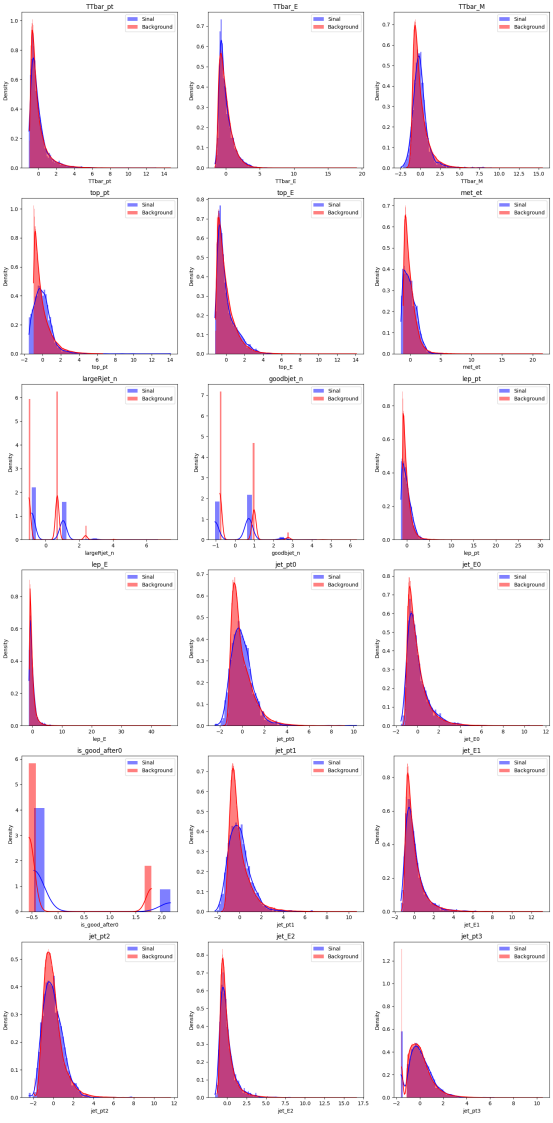
7.1 Correlation Matrix for the 108 features

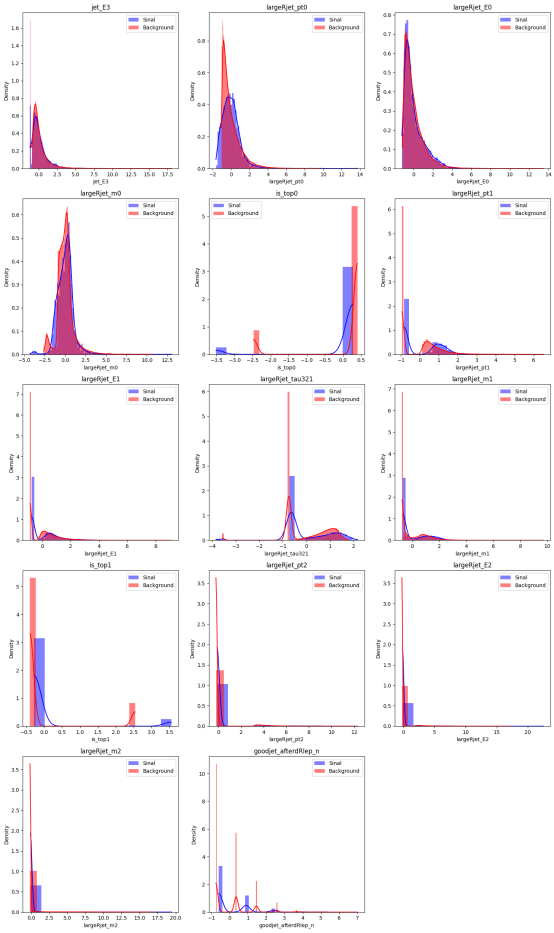


7.2 Correlation Matrix with a 0.4 mask



7.3 Signal vs Background Graphs Comparison




jet_E3
largeRjet_pt0
largeRjet_E0
largeRjet_m0
is_top0
largeRjet_pt1
largeRjet_E1
largeRjet_tau321
largeRjet_m1
is_top1
largeRjet_pt2
largeRjet_E2
largeRjet_m2
goodjet_afterdRlep_n
Sinal
Background
Density