Deep learning for 40 MHz scouting with Level-1 trigger muons for CMS at LHC run-3

July-September 2020

AUTHOR(S):
Maria Popa**, Babeș Bolyai University**

SUPERVISOR(S):
Thomas James
Emilio Meschi

Special thanks to Ema Puljak

CERN openlab

PROJECT SPECIFICATION

CMS will include a new paradigm for the Level 1 Trigger for LHC run 3. It will for the first time enable the reading out of trigger objects at the full collision rate (40 MHz), in order to perform studies and take measurements not possible within the constraints of the 100 KHz Level 1 accept rate.

One such set of trigger objects are the Global Muon Trigger objects. The Global Muon Trigger accumulates muon candidates from barrel, endcap, and overlap trigger regions, and selects eight based on their quality and transverse momentum to be sent to the Global Trigger.

A deep learning machine inference solution has been proposed to manipulate these trigger objects such that they are more usable in offline or semi-offline analysis, rather than simply near the triggering thresholds. By using the offline-reconstructed objects as a target for the training of an artificial neural network, this project aims at providing muon parameters adequate for standard handling in a physics analysis. The machine inference is expected to be carried out in real time using data-processing PCIe boards provided by Micron.

ABSTRACT

The muon track finder of the CMS experiment at the Large Hadron Collider uses custom FPGA-based processors to identify muons and measure their momentum for a fast Level-1 trigger selection. A 40 MHz scouting system at CMS will provide fast statistics for detector diagnostics, alternative luminosity measurements, and new analysis possibilities.

Deep learning is a subfield of machine learning algorithms that uses multiple hidden neural layers to extract relevant features from raw inputs. Previous studies have demonstrated the potential of deep learning in many areas of particle physics. The purpose of this study is to analyse the performance of deep learning algorithms to recalibrate the muon track parameters (transverse momentum, η and φ) for the best resolution. Deep learning regression models are compared against simple linear fits. The performance of these models is evaluated and compared.

TABLE OF CONTENTS

1. INTRODUCTION

The Large Hadron Collider (LHC) [1] is the largest (27 km) and most powerful particle accelerator ever built. It accelerates protons to nearly the speed of light and then collides them at four locations around its ring, producing new particles that move outwards from the collision point.

One of the four collision points hosts the CMS (Compact Muon Solenoid) detector [2]. CMS is a general all-purpose particle detector. The possibility to detect the Standard Model (SM) Higgs boson played a crucial role in its conceptual [3]. It consists of multiple layers of detectors and material (including the 3.8 T solenoidal magnet) that exploit the different properties of particles to measure the energy and momentum of each one. Muons are charged particles similar to electrons, but are about 200 times heavier. One can obtain a particle's trajectory by tracking its position through the multiple layers of detectors. Figure 1 depicts a muon's curved trajectory in four muon detector stations. In total there are 1400 muon chambers, including drift tube detectors, cathode strip chambers, and resistive plate chambers [4]. A sectional view of the CMS detector can be seen in Figure 2.

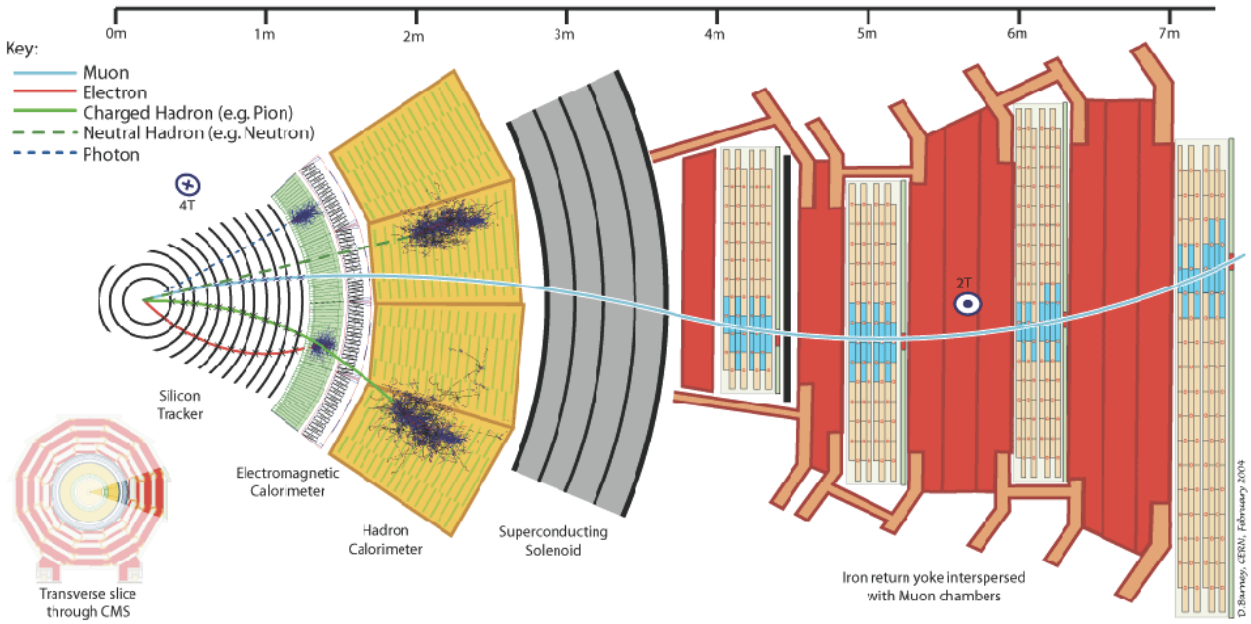

Figure 1. A transverse slice of the CMS detector barrel. The trajectory of an example muon is shown. [5]

The LHC delivers proton-proton collisions to CMS at a 40 MHz bunch crossing rate. Each bunch crossing (event) generates a huge amount of data in the detectors. As a result of bandwidth limitations, it is therefore only possible to read full event information out of the detector at 100 KHz. In fact, only around 1000 full events per second can be permanently stored. CMS runs a two-tier trigger system in order to select the potentially interesting events for read out and analysis.

The Level-1 (L1) trigger [6], located off-detector, consists of custom electronics boards based on field programmable gate arrays (FPGAs), and performs partial reconstruction on a small subset of the event data, selecting events at a maximum rate of 100 kHz to be triggered for full read out. The second layer, called the High Level Trigger (HLT), is a farm of processors that analyses the full event information using complex software algorithms, further reducing the event rate to about 1 kHz, which can be stored for offline analysis [7].

The Global Muon Trigger (GMT) [8], part of the L1 trigger, accumulates muons candidates from the barrel, endcap and overlap regions, and selects eight based on their quality and transverse momentum, sending them to the Global Trigger (also part of the L1 trigger) to make the final decision.

L1 scouting is a new paradigm for data collection at CMS. L1 scouting consists of capturing, reducing, and analysing trigger-level information from the various L1 trigger processors, and storing only relevant high-level information about physics objects.

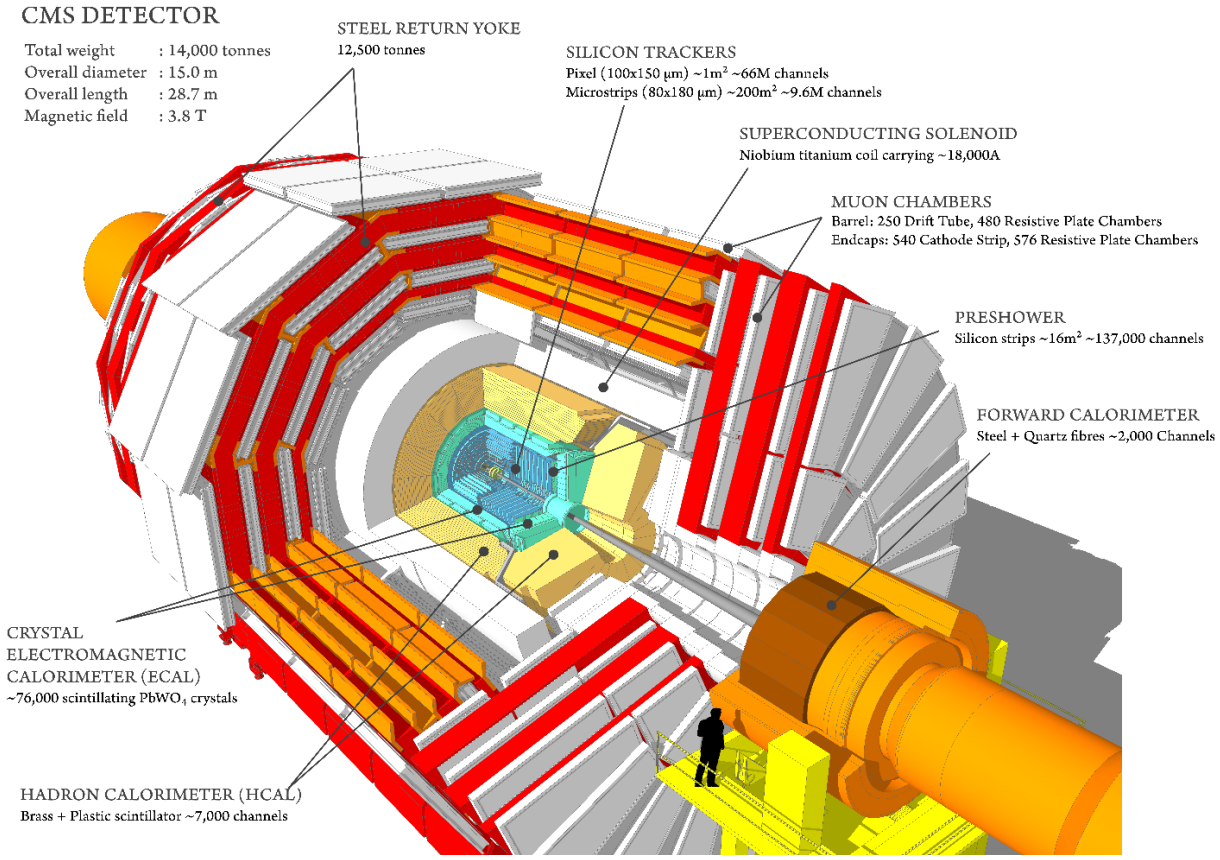

Figure 2. Sectional View of the CMS detector [8]

Machine Learning (ML) is a method of data analysis that automates analytical model building. ML is the process of teaching a computer system how to make accurate predictions when fed data. Neural networks (NN) are a biologically inspired programming paradigm that enable a computer to learn from observational data. The main components of a neural network are: inputs, outputs, and hidden layers that perform nonlinear transformations on the inputs. Deep Learning (DL), a sub-category of ML, is a powerful set of techniques for training multilayered neural networks.

The purpose of this study is to analyse the performance of DL algorithms in recalibrating the GMT muon parameters for the best resolution, using matching offline fully reconstructed muon tracks as the target. In particular, the p_T measurement in the L1 muon trigger is defined so that when selecting muons with a measured p_T greater than or equal to a certain threshold, the trigger condition is 90% efficient for muons with a true p_T grater or equal to the threshold value. The p_T measurement of the L1 trigger is therefore by definition not an estimate of the true p_T and not suitable to be used directly for a physics analysis [7].

Deep learning regression models are compared against simple linear fits and the results are shown in the following sections.

2. DATA ANALYSIS

In the CMS coordinate system (Figure 3), the origin coincides with the nominal collision point, the geometrical centre of the detector. The parameter φ is the azimuthal angle. The pseudorapidity η, is related to the polar angle, θ, and defined by $\eta = -\ln\tan(\frac{\theta}{2})$. The transverse momentum is given by $p_T = p\sin\theta$ [12]. The detector can be divided into three regions of η; the barrel, overlap, and endcap regions, as shown in Figure 4. Different sensor technologies are used in the barrel and endcap regions. The coverage extends up to $|\eta| \leq 2.5$. A different L1 trigger algorithm is used for muon tracking in each of these three regions.

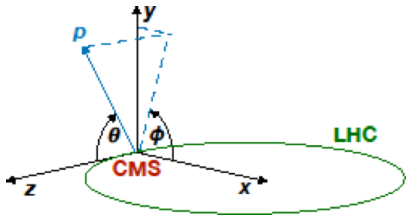

Figure 3. Diagram of the coordinate system used by CMS [9]

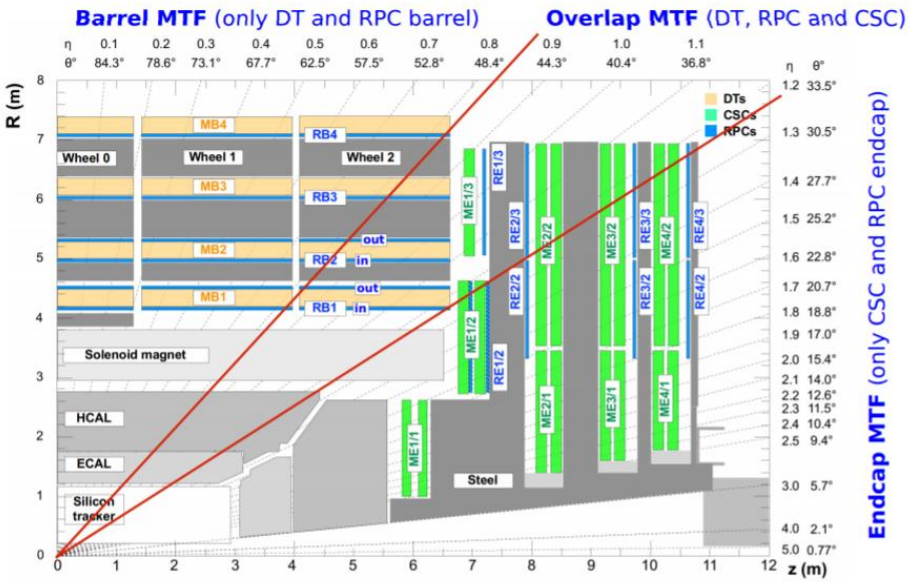

Figure 4. Quarter longitudinal schematic view of the CMS detector [10]

Parameter	Range
φ	[- π, π] radians
φ extrapolated	[- π, π] radians
η	[- 2.45, 2.45]
η extrapolated	[- 2.45, 2.45]
p_T	[0, 255] GeV
charge	[-1,1]
quality	{0, 4, 8, 12}
reconstructed φ	[- π, π] radians
reconstructed η	[- 2.45, 2.45]
reconstructed p_T	[0, 255] GeV
$\Delta \varphi = \varphi - \varphi$ reco	[-0.5, 0.5] radians
$\Delta \eta = \eta - \eta$ reco	[-0.15, 0.15]
$\Delta p_T = p_T - p_T$ reco	[-42.5, 42.5] GeV

Table 1. Parameters in the dataset

An L1 trigger muon object is a 64 bit representation of a muon track [8,14]. A certain number of bits are assigned to different muon parameters, including φ and η (both at the second muon station and after extrapolation back to the vertex), p_T, and charge. These parameters can be used as input to the DL models [7]. Table 1 shows the variables included in the training datasets, which includes the parameters of the matched offline muons.

Dataset	Number of samples
barrel	108 299
overlap	427 165
endcap	5 224 298

Table 2. Dataset sizes

Figure 5 shows the distributions of p_T, η and φ in the barrel dataset.

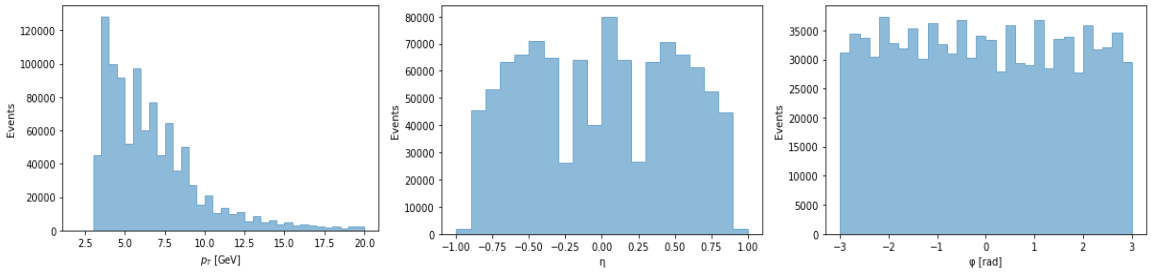

Figure 5. Barrel dataset distributions of p_T, η and φ

Additional restrictions on p_T are made for each dataset, keeping all muons with $2.5 < p_T < 45$ GeV. The p_T resolution above 45 GeV is very poor, as these muons do not bend very much in the magnetic field. On the other hand, below 2.5 GeV muons do not consistently reach the muon detectors, being bent too much by the magnetic field. The number of entries in each dataset after applying this preselection are shown in Table 2.

As the FPGA outputs fixed-point integer values for p_T, η, φ we use this integer representation of the data as inputs to the NN.

3. NEURAL NETWORK MODEL

The models are implemented using the Keras framework. The first model used is the baseline model described in [6]. This model is a multi-layered perceptron with:

- An input layer with four inputs, an output layer with three outputs and three hidden layers with 32 nodes per layer.
- The inputs are integer representations of φ, η, p_T and charge.
- The prediction targets are the differences between the L1 and offline reconstructed φ, η and p_T.
- Before training, the target values are standardised to a mean of zero and a standard deviation of one.
- The ReLU function is used for activation in each hidden layer.
- Batch normalization (BN) [13] is used after each activation in the hidden layers.
- The learning rate optimizer is Adadelta [14], with the default parameters. The loss function is logcosh error.

A second, larger model is also evaluated, containing four hidden layers with 128 nodes each.

Eighty percent of the dataset is used to train the model, with the remaining 20 percent reserved for testing.

a. NEURAL NETWORK RESULTS

Figures 6-8 show the results of the baseline NN model, when trained separately on each dataset. In order to evaluate the performance of the NN, the distribution of differences between the GMT outputs and the

offline reconstructed values is compared with the distribution of differences between the NN outputs and the offline reconstructed values. It is observed that each of the models show an improvement in parameter resolution when compared with the raw GMT values.

i. BARREL

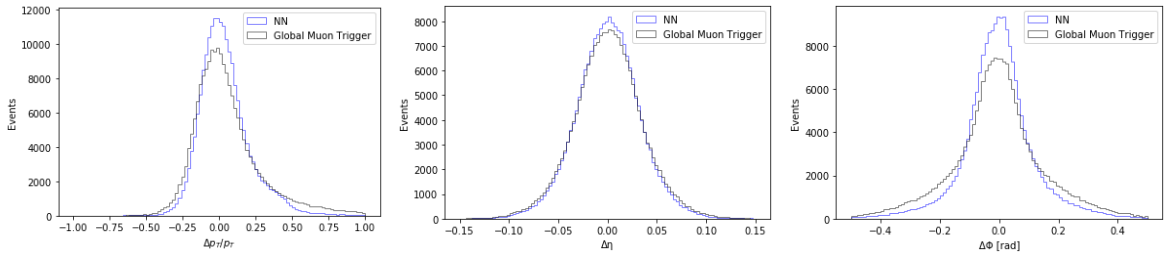

Figure 6. The difference between the GMT and the offline reconstructed values, compared to the difference between the NN predictions and the offline reconstructed values, for the barrel dataset.

ii. OVERLAP

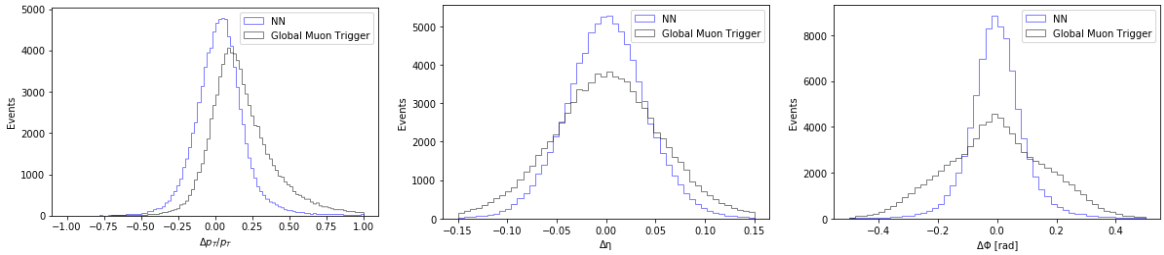

Figure 7. The difference between the GMT and the offline reconstructed values, compared to the difference between the NN predictions and the offline reconstructed values, for the overlap dataset.

iii. ENDCAP

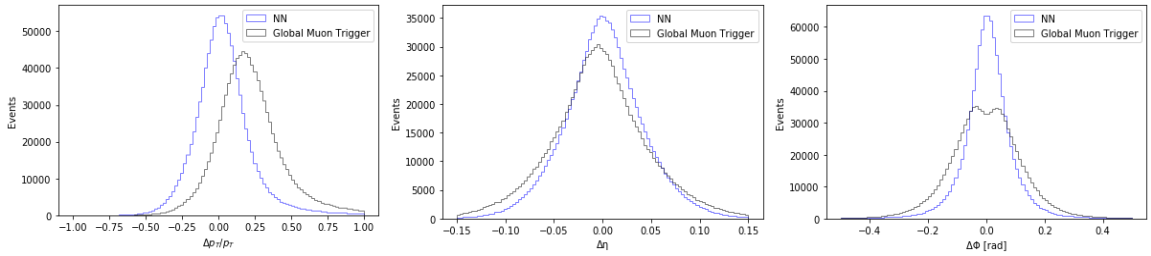

Figure 8. The difference between the GMT and offline reconstructed values, compared to the difference between the NN predictions and the offline reconstructed values, for the endcap dataset.

The performance of the deep learning model trained on the dataset which combines all three regions was also evaluated. As the three datasets are of different sizes, with the endcap dataset containing the majority of the data, it is necessary to weight the training loss by the ratio of N samples in individual dataset to N samples in the combined dataset. This forces the network to learn from each dataset equally. Figure 9 shows that this technique improves the overall performance of the model on the combined dataset.

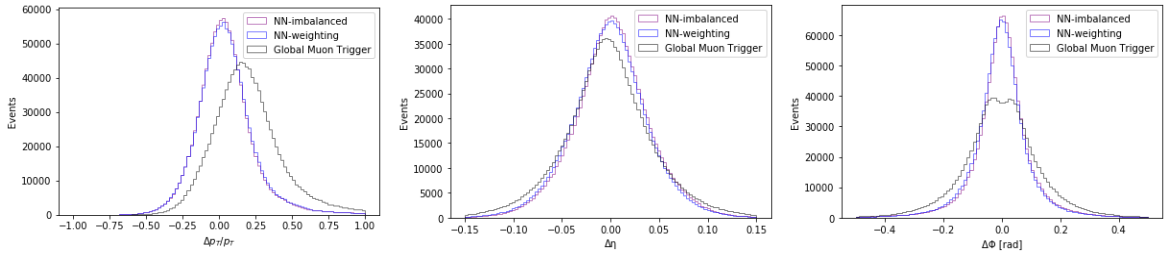

Figure 9. The difference between GMT and offline reconstructed values, compared to the difference between the NN predictions and offline reconstructed values, for the combined dataset (barrel + overlap + endcap), both with and without loss weighting. These plots are dominated by the contribution from the endcap dataset, which contains roughly 90% of the total available data.

4. LINEAR REGRESSION VS NEURAL NETWORK

A linear regression (LR) fit was also applied to the GMT and the corresponding offline reconstructed values in each dataset. We used a one dimensional linear fit for each variable, p_T, φ and η. In order to evaluate the performance of the regressor, the distribution of differences between the GMT outputs and the offline reconstructed values is compared with the distribution of differences between the LR outputs and the offline reconstructed values. For all linear regressions we consider as input the L1 extrapolated values, and as output the offline reconstructed values. In the following subsections we will present the performance of the LR on each dataset and the comparison between the LR and NN models. To evaluate the LR fit, we use the R-squared value, or *linear regression score*. To allow for a straight line fit, φ values very close to $\pm\pi$ were excluded, avoiding the issue of circular wrapping.

a. BARREL DATASET

Table 3 shows the LR score, intercept and coefficient values for p_T, η, and φ. Figure 10 depicts the comparison between the raw GMT, LR, and NN predictions.

We first consider the case of p_T as input and reconstructed p_T as output. An improvement from the raw GMT to the LR estimates is observed.

Parameter	LR score	Intercept	Coefficient
p_T	0.71	2.2	0.60
η	0.95	0.0	0.97
φ	0.95	0.0	0.95

Table 3. LR score, intercept and coefficient score for p_T, η, and φ on the barrel dataset .

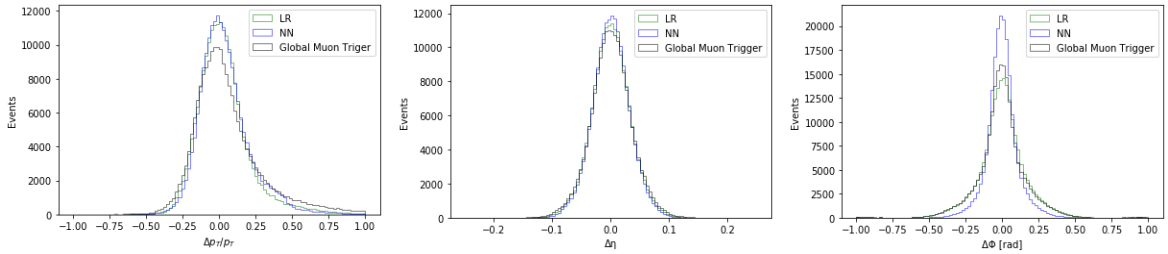

Figure 10. The difference between the NN, LR and raw GMT predictions and the offline reconstructed values, for the barrel dataset.

In the case of η there is no significant difference between LR and NN. The high LR score indicates that the relationship between η and the offline reconstructed η can be well approximated by a straight line. In the case of φ, the NN performed considerably better than the LR.

b. OVERLAP DATASET

The same procedure was applied to the overlap dataset, with the results shown in Table 4 and Figure 11. In this region, the LR does not accurately predict the reconstructed p_T or φ, but performs slightly better on the prediction of η.

Parameter	LR score	Intercept	Coefficient
p_T	0.77	2.2	0.60
η	0.99	0.0	1.0
φ	0.95	0.0	0.97

Table 4. LR score, intercept and coefficient score for p_T, η, φ on overlap dataset

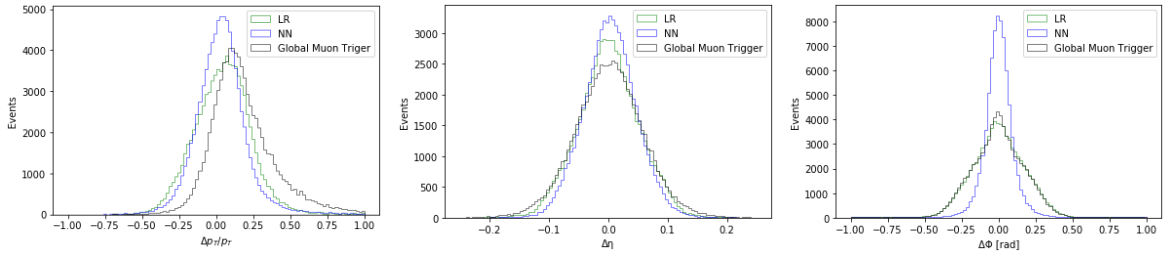

Figure 11. The difference between the NN, LR and raw GMT predictions and the offline reconstructed values, for the overlap dataset.

c. ENDCAP DATASET

Table 5 shows the parameters of the LR fit for the endcap dataset. The performance, when compared to the NN and raw GMT is depicted in Figure 12. The relationship between the p_T and reconstructed p_T is particularly non-linear in this dataset, leading to a low LR score from the fit. The distribution of differences between the LR and the offline reconstructed p_T is narrower than the results for the raw GMT, and has a mean close to zero. On the other hand, the NN distribution is narrower still, with a higher peak. Overall we see a significant difference between the performance of the NN and the LR, even in the case of η.

Parameter	LR score	Intercept	Coefficient
p_T	0.66	1.2	0.53
η	0.99	0.0	0.98
φ	0.91	0.0	0.95

Table 5. LR score, intercept and coefficient score for p_T, η, and φ on the overlap dataset .

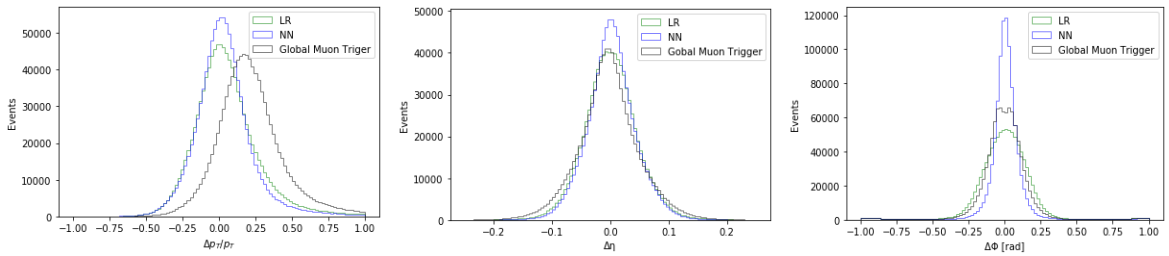

Figure 12. The difference between the NN, LR and raw GMT predictions and the offline reconstructed values, for the endcap dataset.

In conclusion, the simple linear regression does not match the performance of the neural network solution, and a more complex fit is clearly required to properly re-calibrate the data.

5. NEW NEURAL NETWORK MODEL

The baseline model was also compared to an improved deep learning model with four hidden layers and 128 nodes per layer. This comparison was made on the combined dataset, with the reweighing as described in Section 3a. As expected, the improved neural network produces a narrower distribution of differences when compared to the baseline model (see Figure 13), but the improvement is very small. The increased complexity and computational expense of the larger model largely outweighs the minimal performance improvement observed.

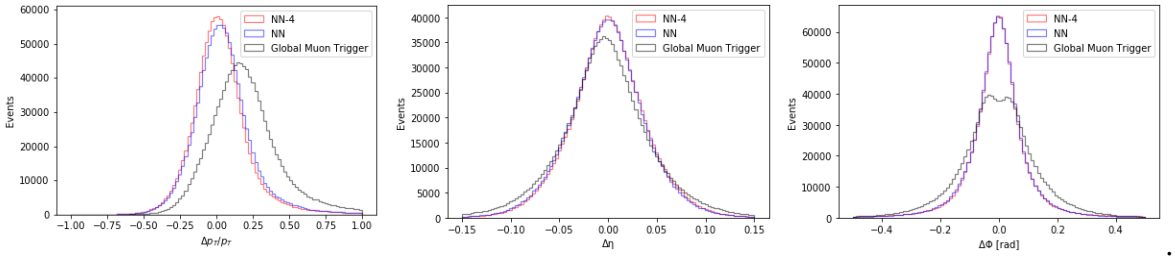

Figure 13. The difference between GMT and offline reconstructed values of, compared to the difference between the NN predictions and reconstructed values and NN-4 (the new neural network with 4 layers) predictions and reconstructed values.

6. SUMMARY

Deep learning models show huge potential for future use within the CMS L1 scouting system. The baseline neural network is able to recalibrate the GMT muon parameters to be significantly closer to the offline reconstructed muon parameters. The performance of the neural network was also compared to a simple linear regression approach. The linear regression shows some improvements in comparison with the raw GMT values, but performs much worse than the neural network. A more complex NN model was also tested, and shows better results in comparison with the baseline model, however, the increased complexity and computational expense of the larger model largely outweigh the minimal performance improvement observed. Future work includes using data from a the Kalman filter [15] track finder that will replace the L1 barrel muon track finder algorithm in 2021, running the models on the FPGA boards, and checking the performance of the deep learning model for a semi-offline analysis.

References

[1] L. Evans, P. Bryant et al., *LHC machine*, August 2008, JINST 3 S08001, doi:10.1088/1748-0221/3/08/S08001

[2] CMS Collaboration, *The CMS experiment at the CERN LHC*, 2008 JINST 3 S08004, doi:10.1088/1748-0221/3/08/S08004

[3] CMS Collaboration, *Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC*, B716 (2012) 30–61, doi:10.1016/j.physletb.2012.08.021

[4] The CMS Collaboration, *Muon detectors*, 2011-11-23, http://cms.web.cern.ch/news/muon-detectors

[5] D. Barney, *CMS slice*, CMS-OUTREACH-2018-017, https://cds.cern.ch/record/2628641

[6] V. Khachatryan et al. [CMS], *The CMS trigger system*, JINST 12, P01020 (2017), doi:10.1088/1748-0221/12/01/P01020

[7] D. Golubovic, T. James, E. Meschi, *40 MHz Scouting with deep learning in CMS*, CTD 2020, doi:10.5281/zenodo.4034400

[8] J. Fulcher, J. Lingemann, D. Rabady, T. Reis, *The new global muon trigger of the CMS experiment*, RT 2016, doi:10.1109/TNS.2017.2663442

[9] T. James, *A hardware track-trigger for CMS at the high luminosity LHC*, PhD Thesis, CMS-TS-2018-025; CERN-THESIS-2018-241

[10] W. Zabolothny, A. Byszuk, *Algorithm and implementation of muon trigger and data transmission system for barrel-endcap overlap region of the CMS detector*, March 2016, doi:10.1088/1748-0221/11/03/C03004

[11] *Detector Design*, 2014, http://cms.web.cern.ch/news/cms-detector-design

[12] The CMS Trigger and Data Acquisition Group, *The CMS high level trigger*, November 2005, https://arxiv.org/ftp/hep-ex/papers/0512/0512077.pdf

[13] S. Ioffe and C. Szegedy, *Batch normalization: accelerating deep network training by reducing internal covariate shift*, March 2015, arXiv:1502.03167

[14] M. Zeiler, *ADADELTA : An adaptive learning rate method*, December 2012, arXiv:1212.5701

[15] M. Bachtis, C. Foundas, P. Katsoulis, *Upgrade of the CMS barrel muon track finder for HL-LHC featuring a Kalman Filter algorithm and an ATCA host processor with ultrascale+ FPGAs,* TWEPP2018, doi:10.22323/1.343.0139