Electron Identification Using Neural Networks

Paweł Drabczyk, AGH University of Science and Technology, Cracow

Supervisors: Cyril Becot, Sarah Heim and Stefan Richter

4th September 2019

Abstract

Identification of electrons is extremely important for many analyses. In current method 19 quantities describing track quality, shower-shape and track to cluster matching are used to distinguish between real electrons and background, but the correlations between them are neglected. This report describes a way to take these relationships in account using neural networks in order to improve electron performance in ATLAS detector.

Contents

1 Introduction

Many final states of proton-proton collisions at the Large Hadron Collider contain electrons and positrons[1]. These particles are very important part of ATLAS experiment's Standard Model, Higgs boson and beyond the Standard Model research. Therefore identifying electrons coming from decays of heavier particles with high efficiency and good background rejection is crucial for a successful physics programme.

The method used for electron identification in previous LHC runs is described in several papers [1, 2]. Current method uses probability density functions of 19 shower-shape, track quality or track-cluster matching variables to calculate the likelihood of particle to be an electron. The procedure is described more precisely in section 2, the most important matter is that current LH method neglects all correlations between the variables.

Machine learning is one of the ways to look for these relationships. With provided MC samples for signal (electrons) and background (hadronic jets) deep neural networks[2] can be trained to identify electrons. The same 19 variables as in LH method are not used separately, but as 19-dimensional input space. NN's architecture, optimization and data pre-processing is explained in section 4. Result are shown in section 5.

2 Theory of Electron Identification

The electron identification have to be preceded by reconstruction procedure. General purpose of reconstruction is to create pairs of tracks and clusters that are loosely matched. It is described more precisely in references [1, 2].

In current electron identification method electron likelihood L is calculated as a product of n probability density functions (P) for signal S and background B:

$$L_{S(B)}(x) = \prod_{i=1}^{n} P_{S(B),i}(x_i) \tag{1}$$

x is the vector or variables specified in figure 1. In general different variables can have discriminating power between electrons and light-flavour jets, photon conversions or electrons from semileptonic decays of hadrons containing heavy-flavour quarks. As formula 1 shows the correlation between ID quantities are not used.

Equation 1 is used to calculate discriminant d_L for each electron candidate:

$$d_L = \frac{L_S}{L_S + L_B} \tag{2}$$

[1] Further in this report term "electron" means both electrons and positrons
[2] Further abbreviated to NN

Putting a cut on higher values of d_L allow to increase background rejection $\frac{1}{\epsilon_B}$, but for the price of lower signal efficiency ϵ_S. Dependent on needs of analysis it is possible to chose "tight" (the most restrictive) , "medium", "loose" or "veryloose" operating point.

Discriminating variables depend on the kinematics, therefore identification is done bins of the electron candidate's transverse momentum and absolute value of pseudorapidity.

Figure 1: Description of electron identification variables. Rejects columns tells whether the quantity has important role in discrimiating electrons and light-flavour jets (LF), photon conversions (γ) or electrons originating from semileptonic decay of hadrons containing heavy-flavour quarks (HF). Usage column gives information whether in LH method the quantity was used to calculate electron LH (LH) or it was used as fixed selection criterion (C). 3x3, 3x5, 3x7 and 7x7 refer to areas of $\Delta\eta \times \Delta\phi$ in units of 0.025×0.025 [1]

Type	Description	Name	Rejects			Usage
			LF	γ	HF	
Hadronic leakage	Ratio of E_T in the first layer of the hadronic calorimeter to E_T of the EM cluster (used over the range $\lvert\eta\rvert < 0.8$ or $\lvert\eta\rvert > 1.37$)	R_{hadl}	x	x		LH
	Ratio of E_T in the hadronic calorimeter to E_T of the EM cluster (used over the range $0.8 < \lvert\eta\rvert < 1.37$)	R_{had}	x	x		LH
Third layer of EM calorimeter	Ratio of the energy in the third layer to the total energy in the EM calorimeter. This variable is only used for $E_T < 80$ GeV, due to inefficiencies at high E_T, and is also removed from the LH for $\lvert\eta\rvert > 2.37$, where it is poorly modelled by the simulation.	f_3	x			LH
Second layer of EM calorimeter	Lateral shower width, $\sqrt{(\Sigma E_i \eta_i^2)/(\Sigma E_i) - ((\Sigma E_i \eta_i)/(\Sigma E_i))^2}$, where E_i is the energy and η_i is the pseudorapidity of cell i and the sum is calculated within a window of 3×5 cells	$w_{\eta 2}$	x	x		LH
	Ratio of the energy in 3×3 cells over the energy in 3×7 cells centred at the electron cluster position	R_ϕ	x	x		LH
	Ratio of the energy in 3×7 cells over the energy in 7×7 cells centred at the electron cluster position	R_η	x	x	x	LH
First layer of EM calorimeter	Shower width, $\sqrt{(\Sigma E_i (i - i_{\max})^2)/(\Sigma E_i)}$, where i runs over all strips in a window of $\Delta\eta \times \Delta\phi \approx 0.0625 \times 0.2$, corresponding typically to 20 strips in η, and $i_{\max}$ is the index of the highest-energy strip, used for $E_T > 150$ GeV only	w_{stot}	x	x	x	C
	Ratio of the energy difference between the maximum energy deposit and the energy deposit in a secondary maximum in the cluster to the sum of these energies	E_{ratio}	x	x		LH
	Ratio of the energy in the first layer to the total energy in the EM calorimeter	f_1	x			LH
Track conditions	Number of hits in the innermost pixel layer	n_{Blayer}		x		C
	Number of hits in the pixel detector	n_{Pixel}		x		C
	Total number of hits in the pixel and SCT detectors	n_{Si}		x		C
	Transverse impact parameter relative to the beam-line	d_0		x	x	LH
	Significance of transverse impact parameter defined as the ratio of d_0 to its uncertainty	$\lvert d_0/\sigma(d_0)\rvert$		x	x	LH
	Momentum lost by the track between the perigee and the last measurement point divided by the momentum at perigee	$\Delta p/p$	x			LH
TRT	Likelihood probability based on transition radiation in the TRT	eProbabilityHT	x			LH
Track–cluster matching	$\Delta\eta$ between the cluster position in the first layer and the extrapolated track	$\Delta\eta_1$	x	x		LH
	$\Delta\phi$ between the cluster position in the second layer of the EM calorimeter and the momentum-rescaled track, extrapolated from the perigee, times the charge q	$\Delta\phi_{\text{res}}$	x	x		LH
	Ratio of the cluster energy to the track momentum, used for $E_T > 150$ GeV only	E/p	x	x		C

As it was mentioned before formulas 1 and 2 neglect correlations between ID quantities. Neural Network with the same input as LH method can be used to look for these re-

lationships, output of NN can be treated exactly the same as d_L discriminant from LH method.

3 Data Samples

NN should be trained on electrons with wide range of transverse momentum and pseudorapidity, therefore signal MC sample was coming from $Z \to e^+e^-$ decay. Distribution of invariant mass of 2 electron candidates with highest transverse momentum in the signal event is shown on the figure 2. Red line shows Z boson mass, most of events have invariant mass equal to it.

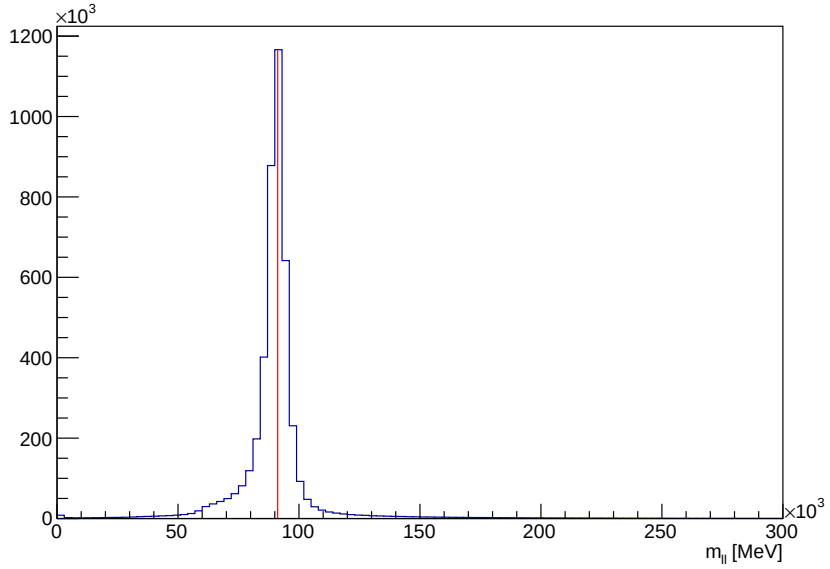


Figure 2: Distribution of invariant mass of 2 electrons with highest transverse momentum in the signal events. The red line indicate Z boson mass.

Sample with events containing 2 hadronic jets with filter over jet momentum was considered as a background. Similar to the signal, the distribution of invariant mass of 2 electrons with highest transverse momentum in the event is shown on the figure 3. One of possible checks for electron contamination of background sample is to look for peak around Z boson mass. On this plot we can see that the result of the check is negative.

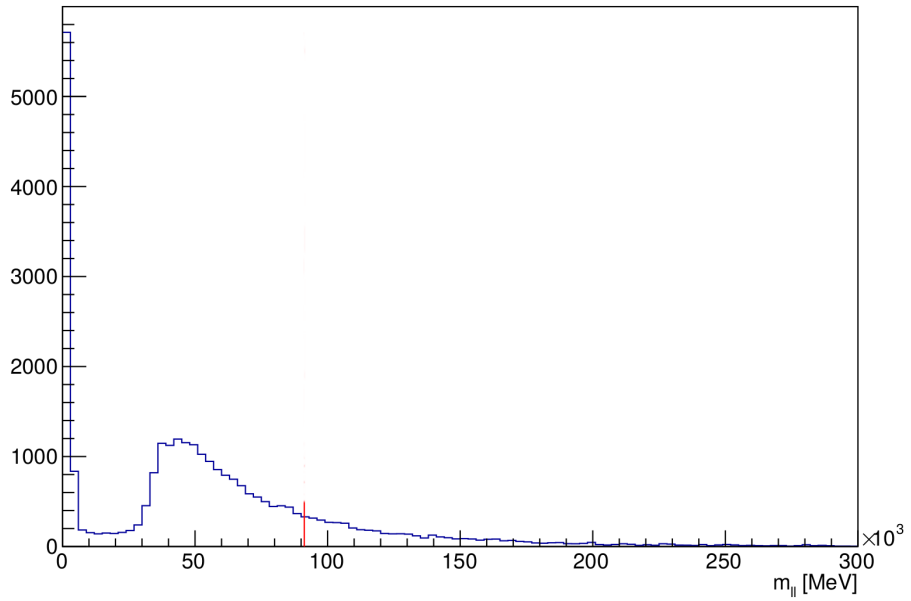


Figure 3: Distribution of invariant mass of 2 electrons with highest transverse momentum in the background events.

Figures 4 and 5 show differences in distributions of transverse momentum and pseudorapidity of signal and background samples.

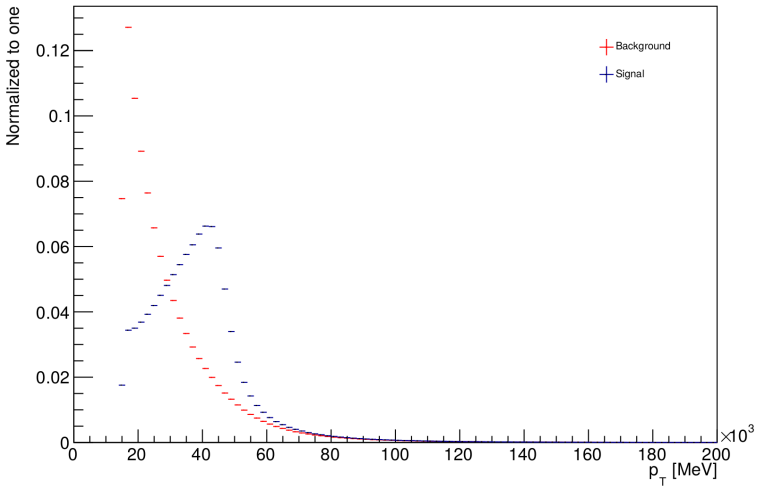


Figure 4: Comparison of transverse momentum distribution for signal and background sample.

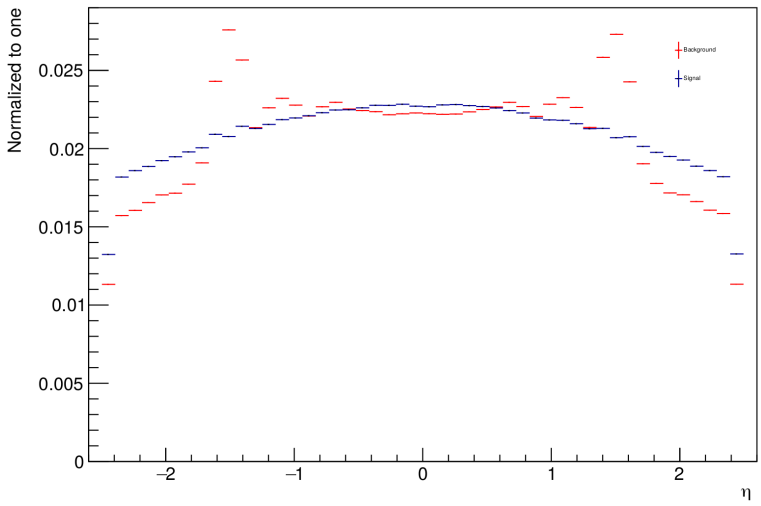


Figure 5: Comparison of transverse pseudorapidity for signal and background sample.

Below there are plots of quantities describing shower-shape in calorimeter (E_{ratio}, figure 6), track quality (d_0, figure 7) and track-calo matching ($\Delta\phi_{res}$, figure 8) for signal and background samples. The plots are normalized to one to allow see the differences in distributions' shapes.

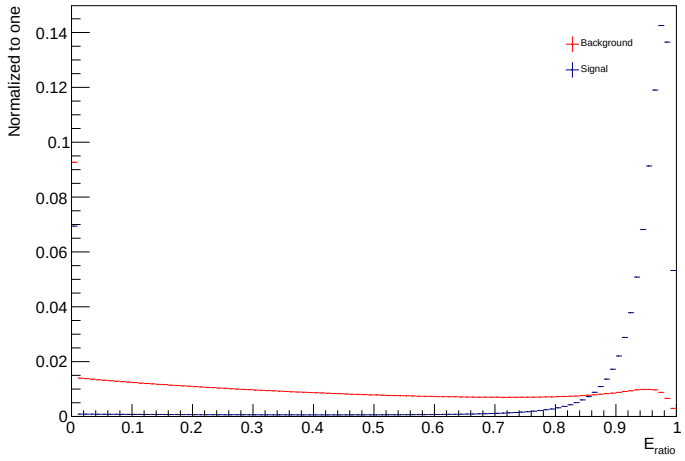


Figure 6: Ratio of the energy difference between the maximum energy deposit and the energy deposit in a secondary maximum in the cluster to the sum of these energies for signal and background samples.

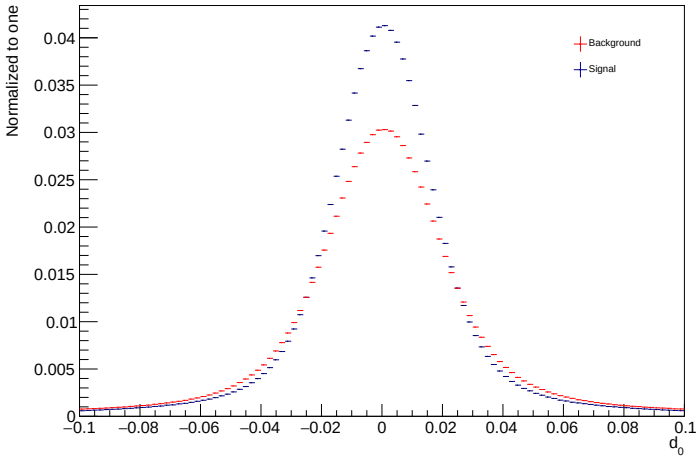


Figure 7: Transverse impact parameter relative to the beam-line for signal and background samples.

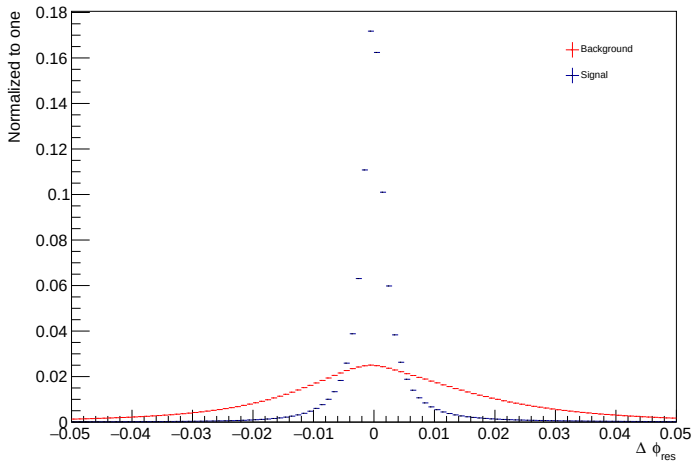


Figure 8: $\Delta\phi$ between the cluster position in the second layer of the EM calorimeter and the momentum-rescaled track, extrapolated from the perigee, times the charge q for signal and background samples.

4 Input Preprocessing and Neural Network Architecture

4.1 Data Sample Balance

4.1.1 Undersampling

Training of NN is optimal if they have equal amount of signal and background events in training set. If the data samples do not have the same size they can be balanced using various techniques, one of them is undersampling. It means choosing random events from bigger class to have at the end same amount of signal and background (figure 9).

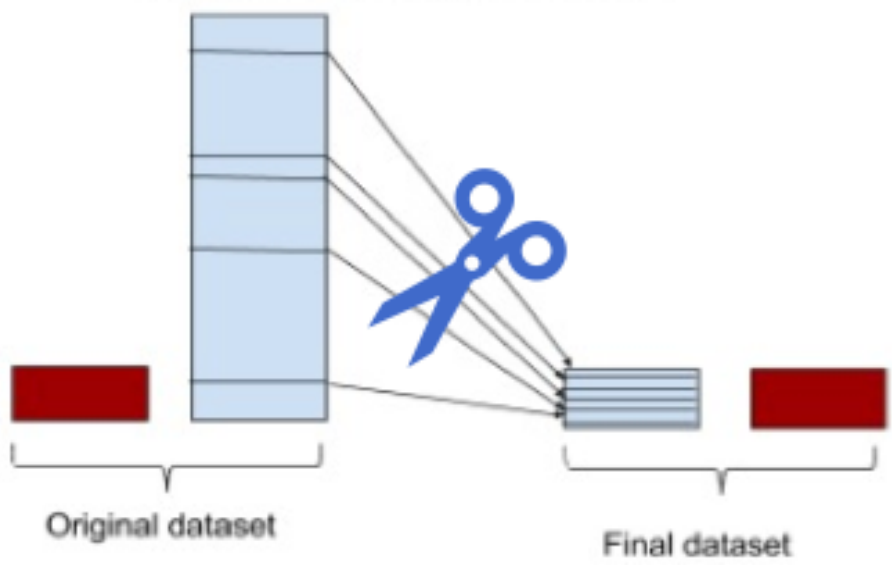


Figure 9: Visualization of undersampling technique.

4.1.2 $\mathbf{p_T}$ distribution normalization

Figure 4 shows differences in transverse momentum of signal and background sample. Because the electron identification is supposed to be applicable in many different analyses with different background kinematics, we do not want the neural network to use the background p_T spectrum in our training setup to help it identify electrons. Therefore we try to eradicate any discriminating information stemming from the p_T spectra of signal and background by reweighting the spectra to look the same, it was done in three variants.

- Training weights were calculated using formula 3 in small p_T bins (1 GeV width) to reach same reweighted p_T distribution of background as the signal sample (weights calculated with this formula were used just for background, for signal just event weight was used). Both sums iterate over all electron candidates in 1 GeV p_T bin, $\sum_i \omega_i^{sig}$ means sum of event weights for signal, $\sum_i \omega_i^{bkg}$ is calculated in a similar way.

$$f_{minibin} = \frac{\sum_i \omega_i^{sig}}{\sum_i \omega_i^{bkg}} \tag{3}$$

- Weights were applied both for signal and background in 1 GeV bin with formulas 4 and 5 to make p_T distribution flat.

$$f_{sig} = \frac{1}{\sum_i \omega_i^{sig}} \tag{4}$$

$$f_{bkg} = \frac{1}{\sum_i \omega_i^{bkg}} \tag{5}$$

- Control version, only event weights were applied, p_T distributions are the same as on figure 3.

4.2 Architecture

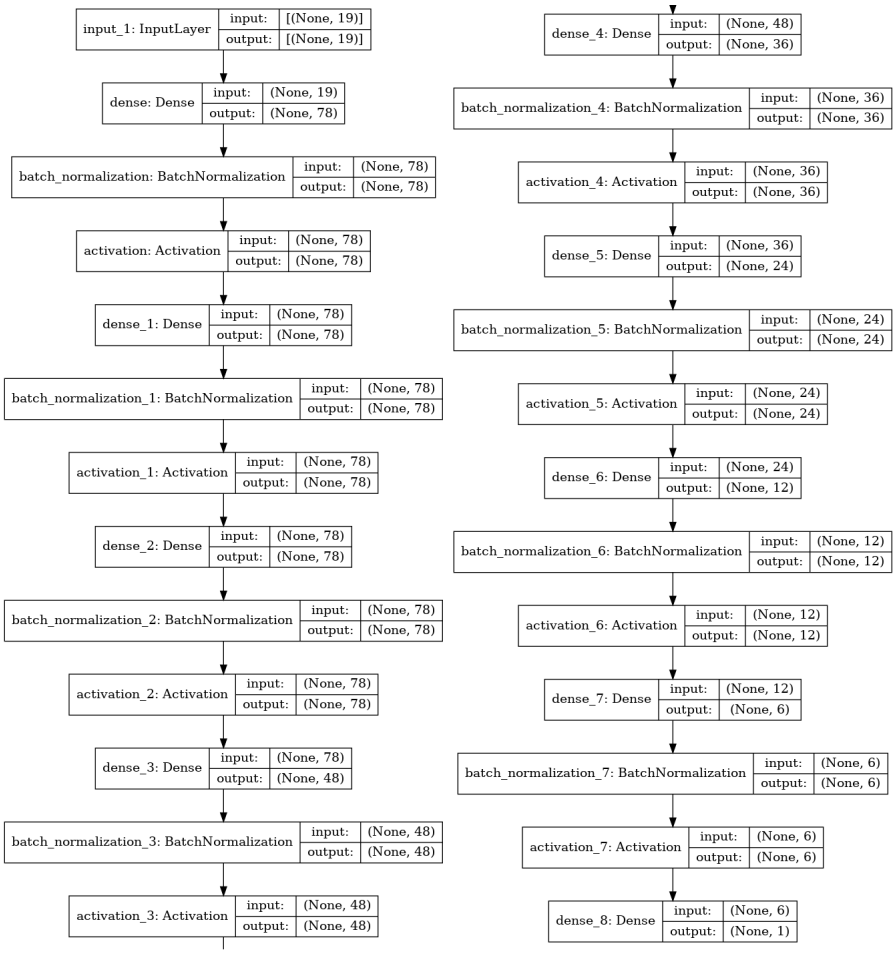


Figure 10: Schema of the model.

Machine learning was implemented using Keras library with tensorflow library backended. As it was mentioned in section 2 input for NN was the same as for LH method, additionally variables were transformed using formula 6 in order to reach distribution with mean equal to 0 and standard deviation equal to 1 for each quantity (Keras function "StandardScaler" was used to do it). z is variable value after transformation, x is a value before, u is mean value of variable and s is its standard deviation

$$z = \frac{x - u}{s} \tag{6}$$

Different model sizes were tested (4 to 10 layers with various number of neurons), the best final version had 10 layers (8 hidden layers, number of neurons $19 \to 78 \to 78 \to 78 \to 48 \to 36 \to 24 \to 12 \to 6 \to 1$), what gives 21433 trainable parameters.

Loss function used in a model was binary cross entropy. All layers were using ReLU activation function, except of output layer which was using sigmoid activation function (the reason is that we want the neural network output value to lie within a bound interval, namely [0, 1], the ReLU function gives a half-open interval $[0, \infty)$). The output should reflect the likelihood of the particle to be an electron. There were 70 of training epochs, this number was found experimentally as big enough to converge the NN.

5 Results

In each case neural network was trained and tested in η bin from -0.7 to 0.7 and in p_T bin from 20 GeV to 60 GeV.

5.1 Same $\mathrm{p_T}$ distribution for signal and background

The formula for reweighting transverse momentum to reach same distribution for signal and background was described in section 4.1.2. Reweighted distributions are on figure 11.

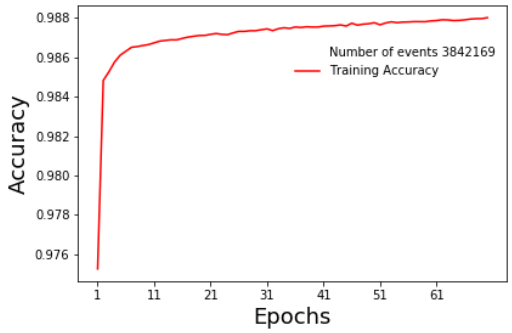
(a) Signal

(b) Background

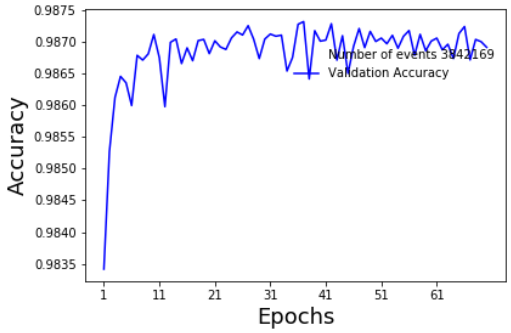
Figure 11: Distribution of transverse momentum reweighted with training weights.

Data sample was splitted into 3 parts: training, validation and testing in order to keep opportunity to check network performance. Control plots for NN training can be seen at figure 12. Accuracy is defined as number of well identified objects divided by number of all objects (in this case identified means with score higher than 0.5 (for signal events) or lower than 0.5 (for background objects)).

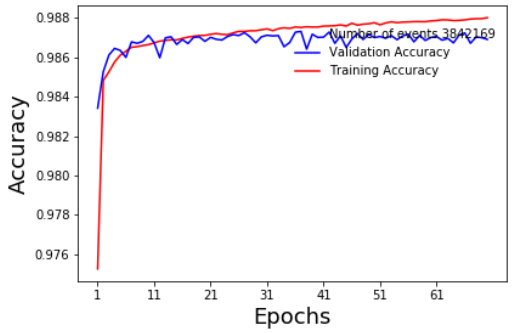
Control plots are used to look for pathological behaviors in network performance. For example if for higher epochs validation accuracy is decreasing, validation loss function is increasing and training accuracy is very high, overtraining might have occurred. In this case control plots look good.



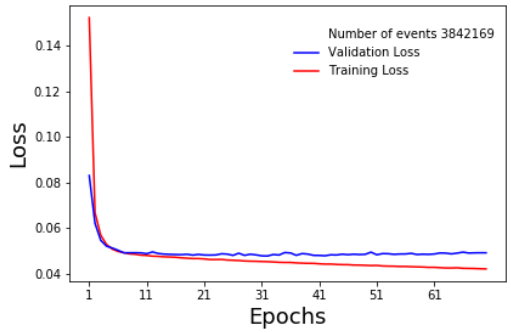
(a) Accuracy for training sample.



(b) Accuracy for validation sample.



(c) Accuracy for training and validation samples.



(d) Loss function for training and validation samples.

Figure 12: Distribution of transverse momentum reweighted with training weights.

Output of the neural network for testing sample can be seen of figure 13. For signal events score is much closer to 1 and for background events to 0.

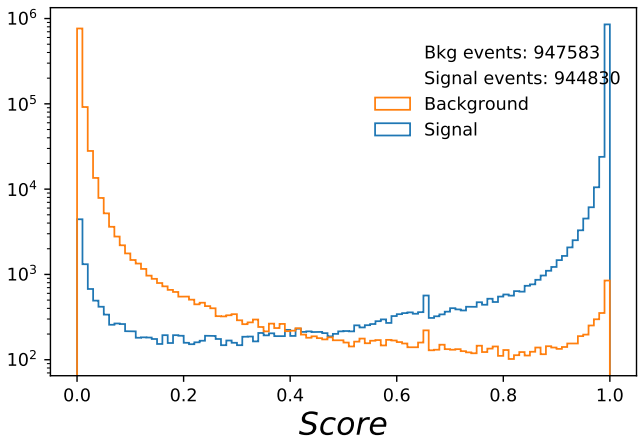


Figure 13: Score distribution for signal and background events.

Of course there are still some background events with score close to 1, there is even a peak for background around score equal to 1, It can be understood as a fraction of jets extremely similar to electrons, which is not distinguishable for the model. Analogical situation happens for electrons with score close to 0.

To identify electron it is necessary to put arbitral cut on score, above which we recognize event as an electron. Value of the cut is based on specific needs of analysis. A receiver operating characteristic[3] curve is on figure 14, it shows signal efficiency ϵ_{sig} versus background rejection $\frac{1}{\epsilon_{bkg}}$, different points correspond to different score thresholds. Signal (background) efficiency $\epsilon_{sig(bkg)}$ is defined as number of signal (background) events with score higher than threshold divided by number of all signal (background) events. In case of background it is more convenient to use background rejection $\frac{1}{\epsilon_{bkg}}$. On the plot there are also approximate result for LH method corresponding to "veryloose", "loose", "medium" and "tight" working points. For all of them except of "veryloose" (currently it is not standard working point) neural network is performing better than LH method.

[3] Further abbreviated to ROC

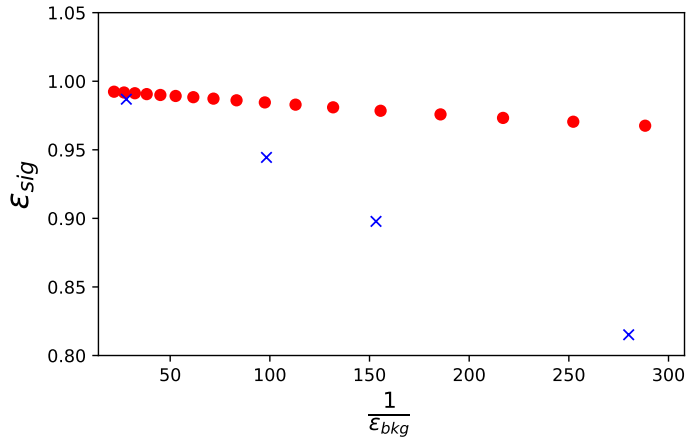
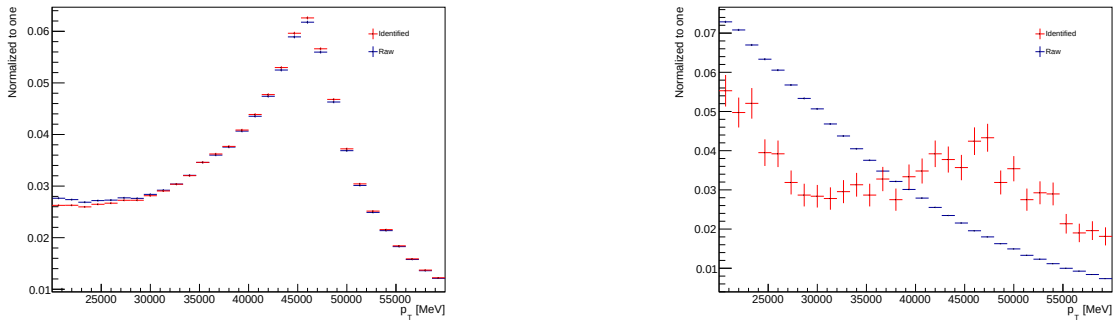


Figure 14: Signal efficiency versus background rejection for different score thresholds (red dots) compared to approximate LH method results (blue crosses)

5.1.1 Transverse momentum dependence

Identification model performs different in different p_T region. Transverse momentum distributions before and after the identification can be seen on the plot 15. The plots are weighted with event weights, they have only statistical errors and they are normalized. Normalization allows to see better the differences in shape (if red curve is higher than blue, more events are passing identification than when it is opposite). Score threshold was chosen to have background rejection between "medium" and "tight" working point.

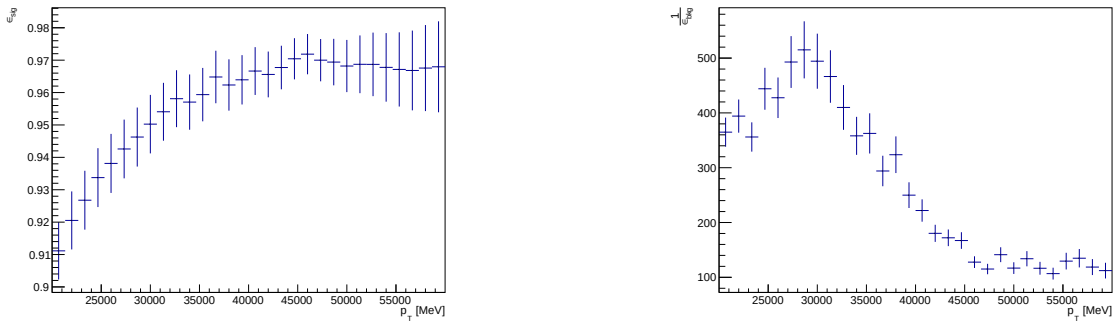


(a) Signal

(b) Background

Figure 15: p_T distribution before and after the identification.

Interpretation of plot 15 is easier using plot 16, which is created by dividing number of identified electrons by number of all electron candidates in each p_T bin (in case of background also raised to power -1).



(a) Signal efficiency

(b) Background rejection

Figure 16: Signal efficiency and background rejection in function of p_T

For low p_T values signal efficiency is low but background rejection is higher, for higher values it is opposite.

5.1.2 Pile-up dependence

Normalized distribution of number of vertexes before and after the identification can be seen on the plot 17. Neural network performance is independent on pile-up, because shape of distribution is almost unchanged. This is also shown on the plot 18, it contains signal efficiency which is calculated by dividing number of identified electrons by number of all electron candidates in each $nvtx$ bin.

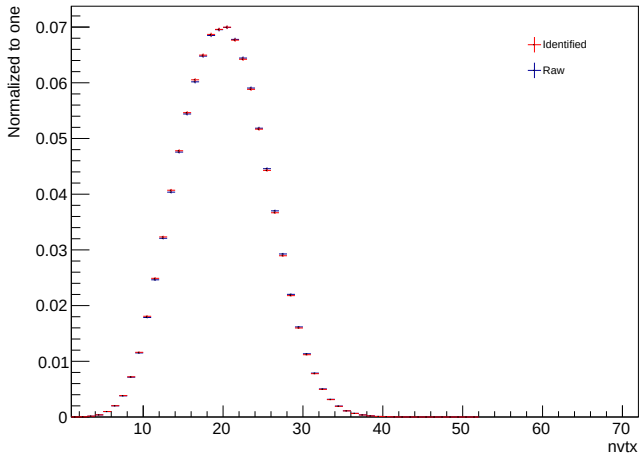


Figure 17: Number of vertexes before and after the selection

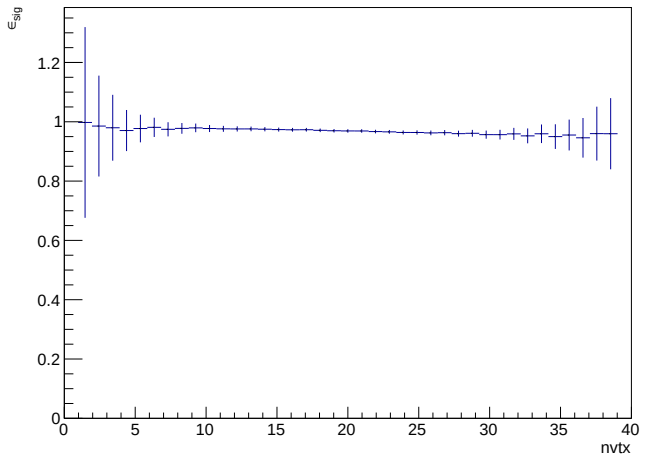


Figure 18: Signal efficiency in function of number of primary vertexes

5.2 No additional training weights

In case of using only event weights the results are very similar to results in section 5.1. On plot 19 ROC curve can be seen, it is almost the same as on plot 14.

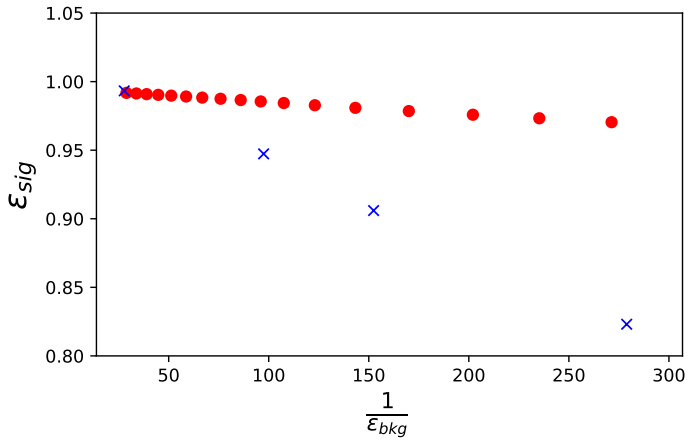
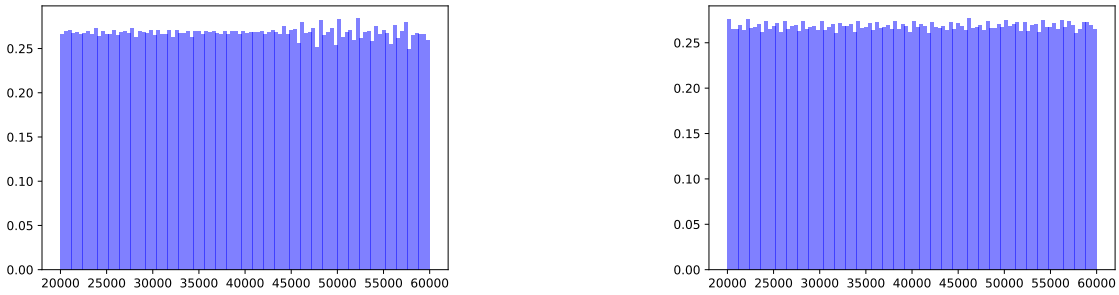


Figure 19: Signal efficiency versus background rejection for different score thresholds (red dots) compared to approximate LH method results (blue crosses)

5.3 Flat $\mathrm{p_T}$ distribution

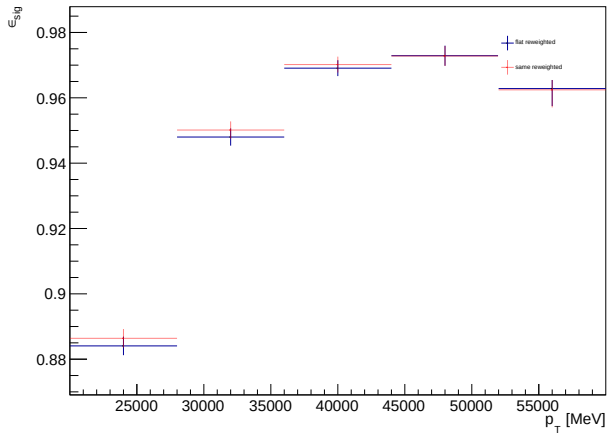


(a) Signal

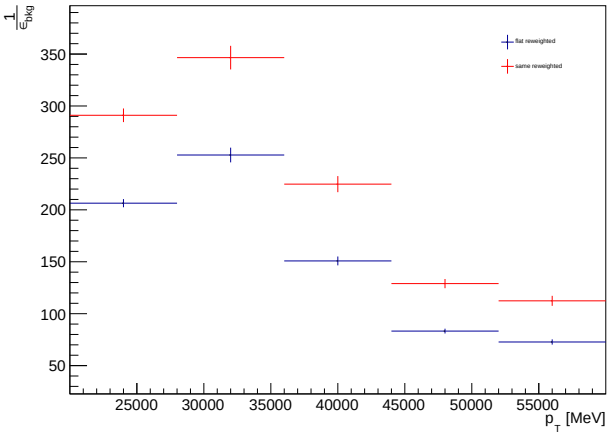
(b) Background

Figure 20: Distribution of transverse momentum reweighted with training weights.

In case of reweighting p_T to reach flat distribution (figure 20) the performance of the NN is significantly worse (21) than in case of "same" p_T distribution . With similar signal efficiency, background rejection is much worse (different score thresholds for both distributions were used to reach similar signal efficiency).



(a) Signal efficiency



(b) Background rejection

Figure 21: Comparison of signal efficiency and background rejection between flat p_T reweighting and "same" p_T reweighting.

6 Summary and Outlook

- Neural networks are powerful tools with huge potential in electron identification. They might give better results than current LH method, however their performance have to be evaluated carefully.
- Model performance is dependent on transverse momentum of the electron, but it is independent on pile-up level.
- The idea is to use NN with the same architecture, but trained on different p_T and η bins.

- There are plans to test it on real data in the future.
- Adding additional input variables might bring better results (e.g. p_T or raw information about cells in cluster).

References

[1] ATLAS Collaboration, Electron reconstruction and identification in the ATLAS experiment using the 2015 and 2016 LHC proton–proton collision data at $\sqrt{s}$ = 13 TeV
arXiv: 1902:04655

[2] ATLAS Collaboration, Electron performance measurements with the ATLAS detector using the 2010 LHC proton-proton collision data
arXiv: 1110:3174