Stochastic and Physical Monitoring Systems
SPMS 2018
June 18 - June 22, 2018
Dobřichovice, Czech Republic

Deep Learning in High Energy Physics

Miroslav Kubů, Petr Bouř

Department of Mathematics, FNSPE, CTU in Prague, Czech Republic

Email: `kubumiro@fjfi.cvut.cz, petr.bour@fjfi.cvut.cz`

Abstract. Data analysis in high energy physics (HEP) includes solving complex classification tasks. That is why specific machine learning approaches such as artificial neural networks (ANN) [1] are often utilized today. We present our ANN implementations for Higgs boson occurrence events separation in Monte Carlo simulated data. Our results demonstrate the benefits of deep learning approaches in HEP data analysis and show the great performance for classification of the particle decays.

Key words: Binary classification; Deep neural networks; Higgs boson; High energy physics; Machine learning.

1 Introduction

The primal insight to the laws of physics and the structure of matter comes from modern particle accelerator experiments that are studying colliding particles and their decays. Signal separation and statistical homogeneity testing are important steps in analyzing these events, see [2], [3], [4] and [5]. In this paper, we aim to provide comparison of different ANN architectures for binary classification of particle collision events. The choice of the architecture is crucial for the ANN classifier's performance. Although any separation rule in supervised learning can be theoretically approximated by a shallow network consisting of a single hidden layer [6], recent successes in deep learning are connected to deep neural networks (DNN), which use multiple hidden layers. Empirical experience shows that even with the same number of neurons, implementation of the more hidden layers leads to better performance of the classifier.

2 HIGGS Dataset

We demonstrate the use of DNN in HEP on the binary classification task of Higgs boson occurrence separation [7]. For three implementations of different ANN architectures, we show that the shallow networks may fail to capture the proper separation boundary in contrast to the deep networks. The HIGGS dataset consists of 11 million observations produced by Monte Carlo simulation. The dataset was nearly balanced, with 53% of the Higgs boson events. Because of the computational costs, we used a training set of only 2.6 million observations. The hyperparameters of the networks were tuned on a set of 100,000 validation observations. Each event consists of 28 observed parameters and the indication of Higgs boson occurrence.

3 Methods

For a detailed description of the terms used below, see [8]. The hidden units in all of the three implementations use a *tanh* activation function, while the output layer is made of a *sigmoid* function. The weight updates $\Delta\boldsymbol{\theta}^{(k)} \in \mathbb{R}^m, m, k \in \mathbb{N}$ for training inputs $\boldsymbol{x}_i \in \mathbb{R}^d, d \in \mathbb{N}$ and outputs $y_i \in \mathbb{R}$

$$\Delta\boldsymbol{\theta}^{(k)} = \boldsymbol{v}^{(k)} = \alpha\boldsymbol{v}^{(k-1)} - \epsilon_{k-1}\nabla_{\boldsymbol{\theta}}\left(\frac{1}{n}\sum_{i=1}^{n} L\left(f(\boldsymbol{x}_i;\boldsymbol{\theta}^{(k-1)}), y_i\right)\right), \tag{1}$$

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} + \Delta\boldsymbol{\theta}^{(k)}, \tag{2}$$

are computed using stochastic gradient descent for a binary cross entropy lost function L with an initial learning rate $\epsilon_0 = 0.01$ and a weight decay $\Delta\epsilon = 10^{-5}$ for each epoch. In addition, we used classic momentum method [9] with a momentum coefficient $\alpha = 0.9$. The forward propagation was performed on mini batches consisting of 100 training examples. The training was terminated when the loss function stopped decreasing in the 10 consecutive epochs.

The classifiers were tested on a set of 500,000 test observations. The performance of classifiers was assessed by means of a receiver operating characteristic (ROC) [10]. Better performing models indicate the higher value of AUC (area under the ROC curve). In addition to AUC, we also use a few other metrics such as signal efficiency ε_S, background efficiency ε_B and significance Z, see [8]. The evaluation of these metrics is illustrated in Figure 1.

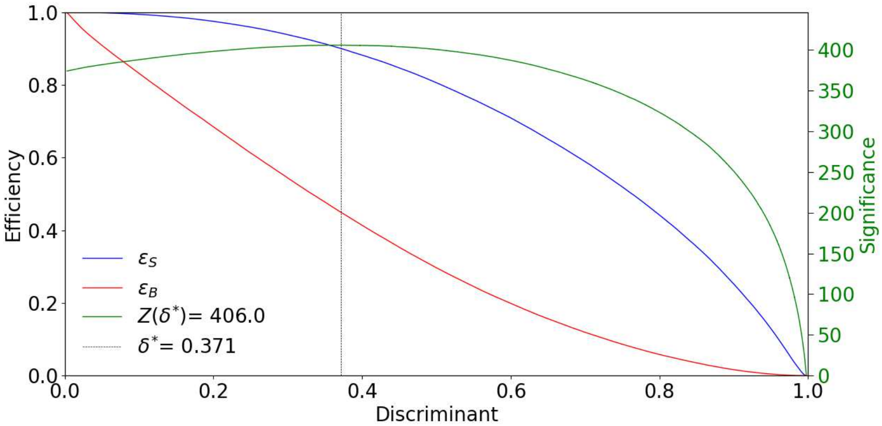

Figure 1: Evaluation metrics for the DNN classifier.

4 Classifiers

We aim to show the difference in performance between the single neuron and the neural network consisting of one or multiple hidden layers. Thus, we firstly implemented a simple perceptron classifier consisting of a single neuron. Afterwards, we tried to improve the performance of the classifier by implementing one hidden layer of 650 units. In order to reduce overfitting, we used dropout regularization [11] through which we are stochastically dropping the neurons in the hidden layer with 40% probability. Finally, we implemented DNN with 3 hidden layers consisting of 400, 600 and 400 units. Similarly to the previous implementation, we tried to avoid overfitting by using dropout regularization. A detailed description of the methods used above is available in [8].

5 Results

As Table 1 shows, the perceptron performs significantly worse than the two other implementations. Although the addition of one hidden layer caused notable increase of AUC compared to the perceptron, performance of the single hidden layer classifier is still much lower than the DNN. As Figure 3 indicates, dropout regularization managed to prevent the overfitting problem even in the DNN implementation. Thus, we may observe the overall benefits of DNN in comparison to the shallow networks. The difference between AUC for the test and the training data is depicted in Figure 2.

Architecture	AUC (training)	AUC (test)
Perceptron	0.666	0.668
ANN (650N)	0.766	0.767
DNN (400N-600N-400N)	0.850	0.842

Table 1: Classification performance of different ANN architectures.

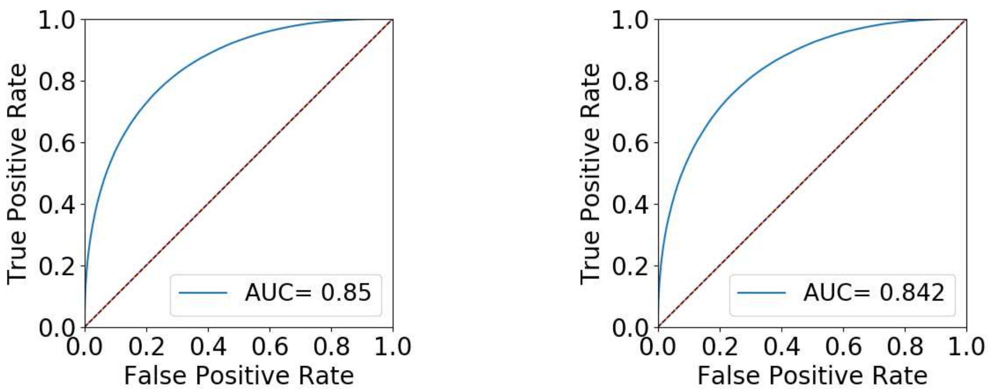

Figure 2: ROC curve for the DNN classifier (training on the left, test on the right).

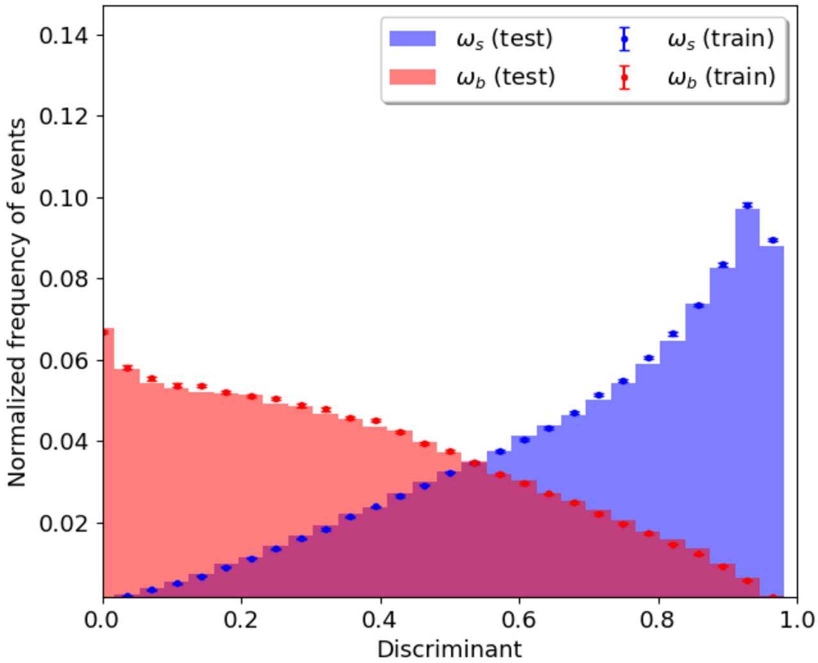

Figure 3: Control plots for the DNN classifier. ω_S denotes the class of signal observations (Higgs boson occurrence events). ω_B denotes the class of backgrounds observations.

6 Conclusion

We compared the three different ANN architectures for a binary classification task for data originating from HEP. With the architecture consisting of three hidden layers, we were able to reach 0.842 AUC for the test sample. As our analysis shows, DNN perform significantly better in complex classification tasks compared to the shallow networks.

Acknowledgements

This work was supported by the grants LTT18001 (MYES), LM2015068 (MYES), SGS18/188/OHK4/3T/14 (CTU) and Fermilab-CZ CZ.02.1.01/0.0/0.0/16_013/0001787 (MYES).

References

[1] C. M. Bishop. *Pattern Recognition and Machine learning*. Springer, 2006.

[2] M. A. Acero, P. Adamson, L. Aliaga, P. Bouř, F. Jediný, V. Kůs, J. Smolík, et al. (NOvA Collaboration). New constraints on oscillation parameters from v(e) appearance and v(mu) disappearance in the NOvA experiment. *Physical Review D*, **98 (3)**, 032012, 2018.

[3] M. Štěpánek, J. Franc, V. Kůs. Model based clustering method as a new multivariate technique in high energy physics. *Journal of Physics: Conference Series*, **490**, 012225, 2014.

[4] M. Štěpánek, J. Franc, V. Kůs. Modification of Gaussian mixture models for data classification in high energy physics. *Journal of Physics: Conference Series*, **574**, 012150, 2015.

[5] J. Trusina, J. Franc, V. Kůs. Statistical homogeneity tests applied to large data sets from high energy physics experiments. *Journal of Physics: Conference Series*, **936(1)**, 012046, 2017.

[6] S. Haykin. *Neural networks, A comprehensive foundation.* Pearson Education, 2001.

[7] P. Baldi, P. Sadowski, D. Whiteson. Searching for Exotic Particles in High Energy Physics with Deep Learning. *Nature Communications*, **5** 2014.

[8] M. Kubů. *Convolutional Neural Networks in High Energy Physics.* Bachelor thesis, FNSPE CTU in Prague, 2018.

[9] B. T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, **4(5)**, 1–17, 1964.

[10] J. Davis, M. Goadrich. *The relationship between precision-recall and roc curves.* Technical report n. 1551, University of Wisconsin Madison, January 2006.

[11] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, **15**, 1929–1958, 2014.