Residual neural networks for neutrino event classification

Miroslav Kubu

Faculty of Nuclear Sciences and Physical Engineering
Czech Technical University in Prague

June 23 2019

Outline

- Introduction to the NOvA Experiment
- Residual Neural Networks
- Methods for steady learning
- Neutrino event classification
- Summary

The NOvA Experiment

Mapping of $\nu_\mu \to \nu_e$ oscillation

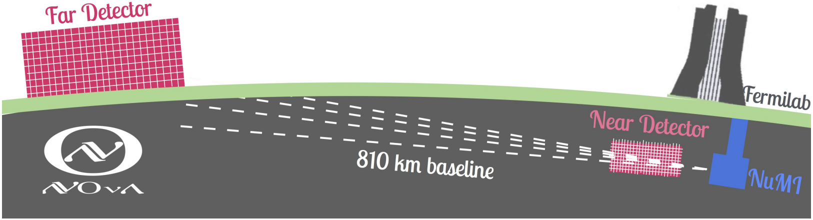


Figure 1: NOvA scheme [1].

The NOvA Experiment

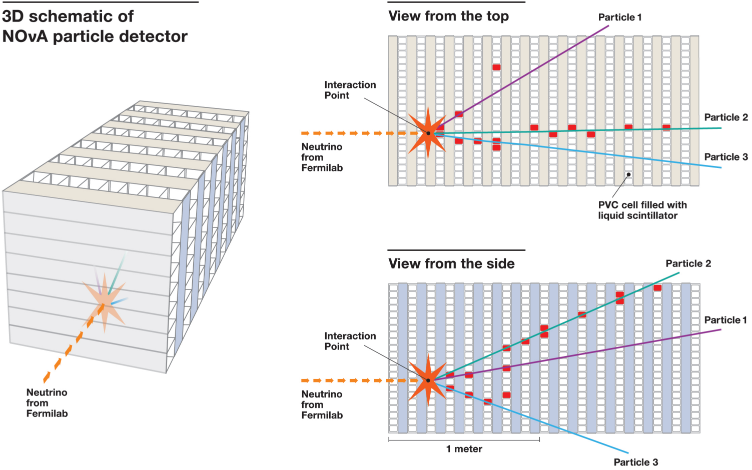


Figure 2: NOvA detector scheme [2].

Interaction	Decay products
ν_μ CC	muon + hadron
ν_e CC	electron + hadron
ν_τ CC	tau + hadron
ν NC	unknown neutrino + hadron

Table 1: Neutrino interaction in NOvA detectors [2].

How to classify the pixelmaps? Let's try CNN.

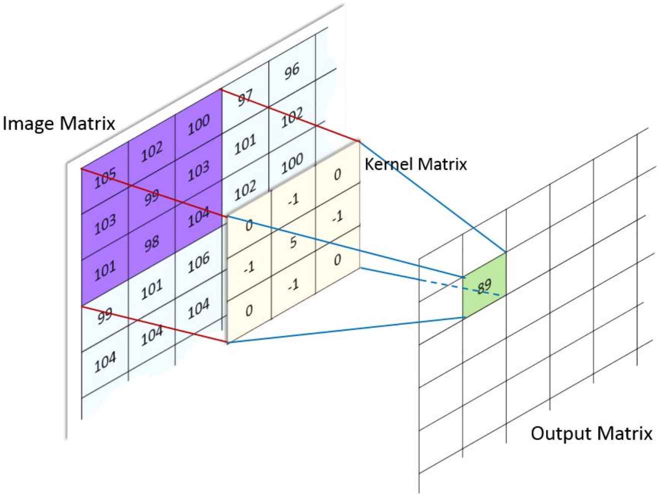


Figure 3: Matrix convolution in CNN [5].

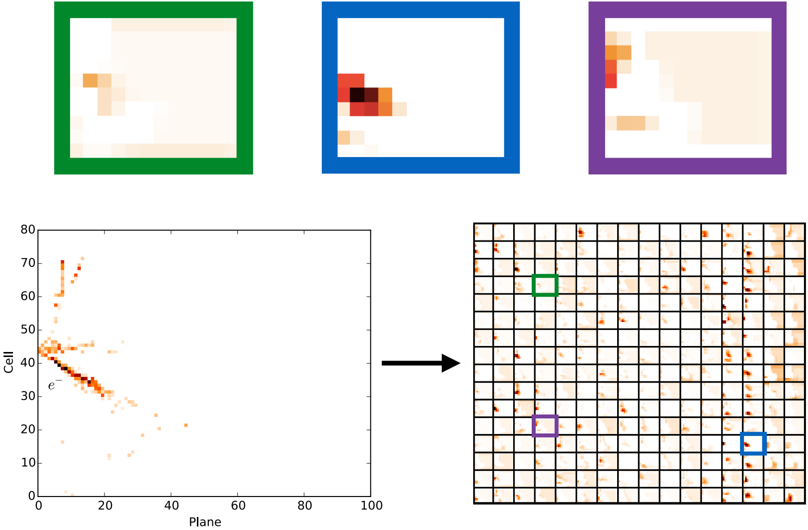
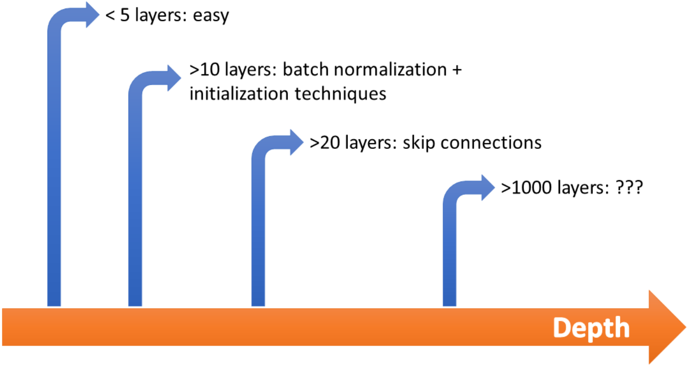


Figure 4: Interaction pixelmap with learned features [2].

Back to deep learning



Weight initialization

How to set initial weights?

Let $W \in \mathbb{R}$ be a random variable.

- Random initialization

$$W \sim \mathcal{N}(0, \sigma^2), \text{ where } \sigma^2 > 0, \tag{1}$$

$$W \sim U(-a, a), \text{where } a > 0. \tag{2}$$

- He initialization

$$W^j \sim \mathcal{N}\left(0, \sqrt{\frac{2}{n_j}}\right). \tag{3}$$

- Xavier initialization

$$W^j \sim U\left(-\frac{\sqrt{6}}{\sqrt{n_j + n_{j+1}}}, \frac{\sqrt{6}}{\sqrt{n_j + n_{j+1}}}\right). \tag{4}$$

Batch normalization

Assume $\vec{x}_i = \left(x_i^1, x_i^2, \ldots, x_i^d\right)$. Then, we normalize $\vec{x}_i$ for each training sample $i \in \{1, \ldots, n\}$ in terms of dimension.

$$\hat{x}_i^k = \frac{x_i^k - \hat{\mu}^k}{\hat{\sigma}^k}, \tag{5}$$

where

$$\hat{\mu}^k = \frac{1}{n}\sum_{i=1}^{n} x_i^k, \tag{6}$$

$$\hat{\sigma}^k = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i^k - \hat{\mu}^k\right)^2}. \tag{7}$$

Add trainable parameters $\gamma^k, \beta^k \in \mathbb{R}$:

$$y_i^k = \gamma^k \hat{x}_i^k + \beta^k. \tag{8}$$

Residual Neural Networks

Problems connected to deep neural networks training (DNN)

- Vanishing gradient problem
- Degradation problem

Residual Neural Networks (ResNet) concept

- input $x \in \mathbb{R}^d$
- desired mapping $\mathcal{H}(x) : \mathbb{R}^d \to \mathbb{R}^q$
- residual mapping $\mathcal{F}(x) = \mathcal{H}(x) - x$

Residual Neural Networks

Residual learning $\mathcal{H}(x) = \mathcal{F}(x) + x$

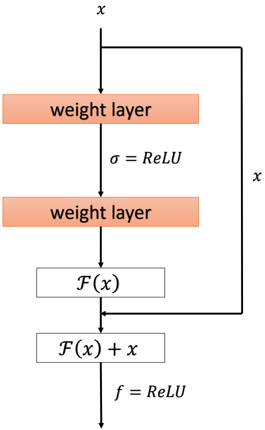


Figure 5: Residual block with shortcut connections.

Residual Neural Networks

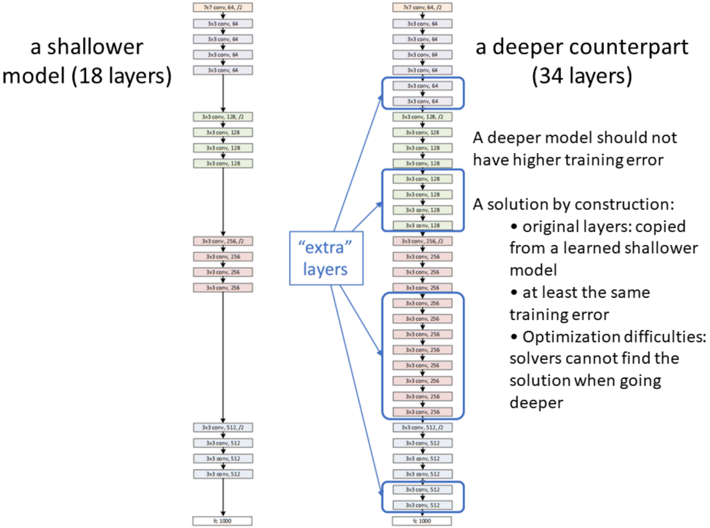


Figure 6: Kaiming He accuracy saturation experiment [4].

Residual Neural Networks

Residual learning $\mathcal{H}(x) = \mathcal{F}(x) + x$

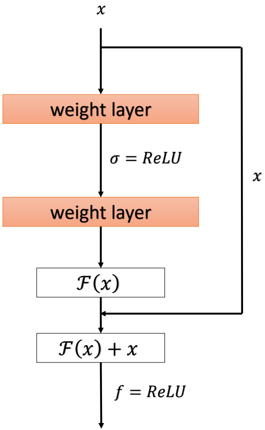


Figure 7: Residual block with shortcut connections.

Residual Neural Networks

Let F be a mapping within a residual block.

$$x_{l+1} = x_l + \mathcal{F}(x_l), \tag{9}$$

thus

$$x_L = x_l + \sum_{i=l}^{L-1} \mathcal{F}(x_l) \tag{10}$$

Let E be an error function, then

$$\frac{\partial E}{\partial x_l} = \frac{\partial E}{\partial x_L}\frac{\partial x_L}{\partial x_l} = \frac{\partial E}{\partial x_L}\left(1 + \frac{\partial}{\partial x_l}\sum_{i=l}^{L-1} \mathcal{F}(x_i)\right) \tag{11}$$

NOvA Data Analysis

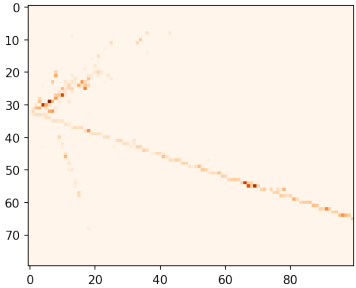
Binary classification task for classes ν_e and ν_μ

Dataset

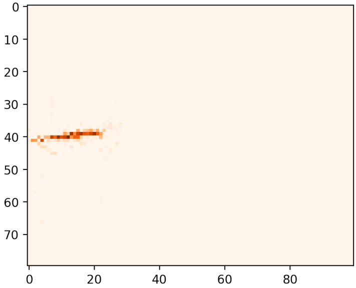
- 29 235 observations for two views (13 901 ν_e and 15 334 ν_μ)
- $2 \times 80 \times 100$ dimension, pixel intensity $I \in \{0, \ldots, 255\}$
- data distribution
    - 80 % training
    - 10 % validation
    - 10 % testing

Interaction	Decay product	Trajectory
ν_μ CC	muon + hadron	Long trajectory with low $\mathrm{d}E/\mathrm{d}x$ of the muon.
ν_e CC	electron + hadron	Electron shower with large variance and fast energy loss.

Table 2: Neutrino interactions with corresponding trajectories in the NOvA detectors [2].



(a) ν_μ CC interaction



(b) ν_e CC interaction

Model implementation

Models

- Basic CNN - illustrative model
- ResNet 1 - ResNet based on CNN
- ResNet 2 - deep ResNet model

Training algorithm

- Mini-batch GD with batches of 32 observations
- Weight initialization - Xavier
- Termination - early stopping with patience - 10 epochs

Implementation

- Keras, Tensorflow backend

Basic CNN

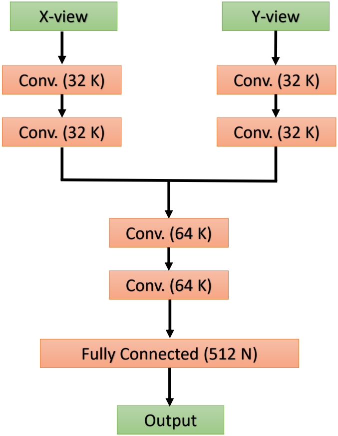


Figure 9: Basic CNN scheme.

ResNet 1

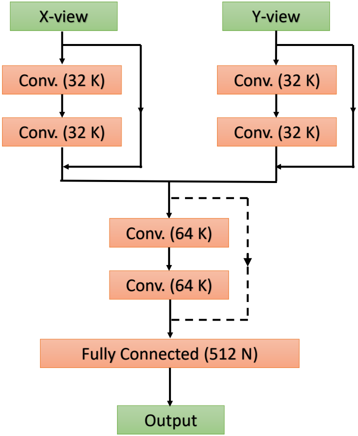


Figure 10: ResNet 1 scheme.

ResNet 1

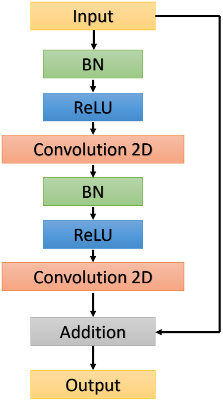


Figure 11: Residual block scheme.

ResNet 2

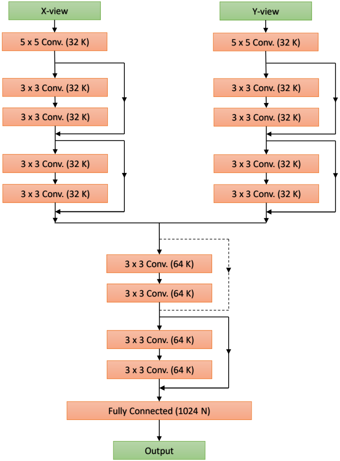


Figure 12: ResNet 2 scheme.

Basic CNN

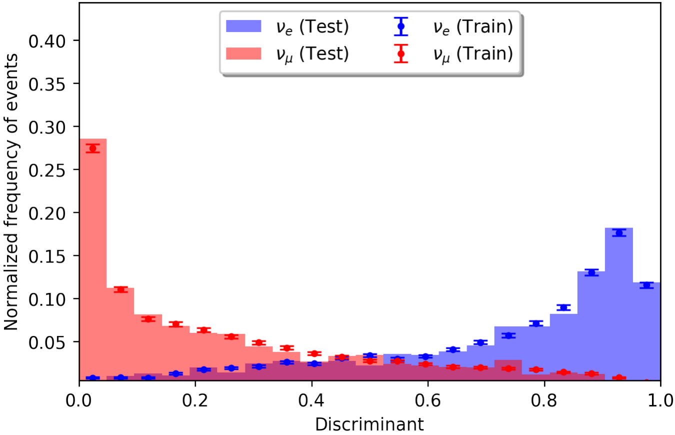


Figure 13: Control plot of basic CNN.

ResNet 1

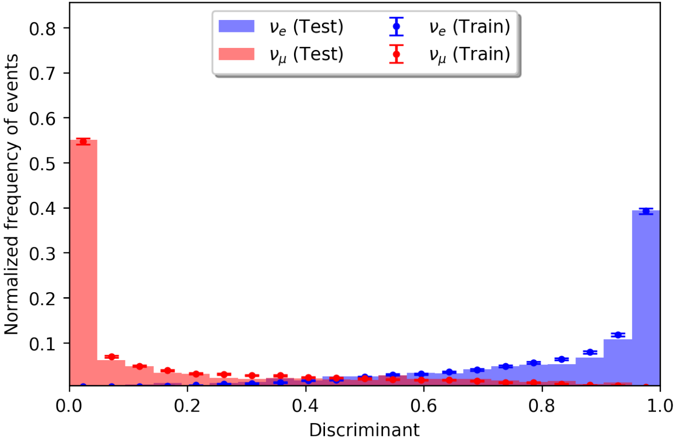


Figure 14: Control plot of ResNet1.

ResNet 2

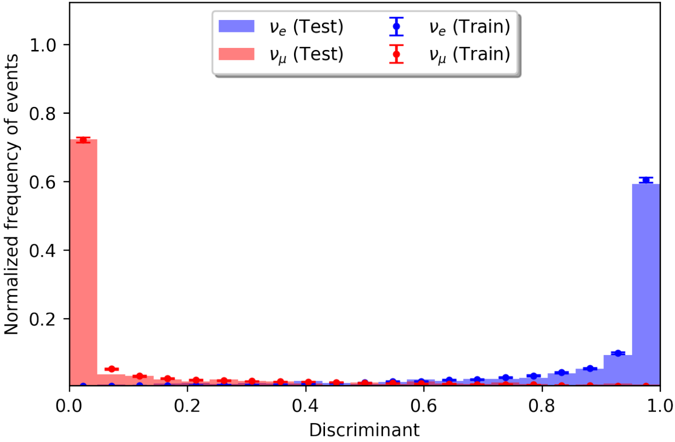
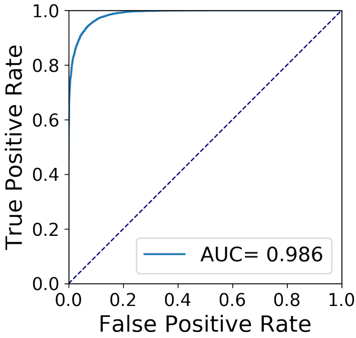


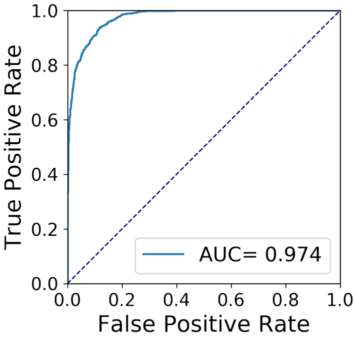
Figure 15: Control plot of ResNet2.

Results

Architecture	AUC (train)	AUC (test)
Basic CNN	0.901	0.905
ResNet1	0.962	0.941
ResNet2	0.986	0.974



(a) ResNet2 ROC on training set



(b) ResNet2 ROC on testing set

Binary classification metrics

Specificity

$$\varrho_B = \frac{TN}{TN + FP} \tag{12}$$

Sensitivity

$$\varepsilon_S = \frac{TP}{TP + FN} \tag{13}$$

Precision

$$PPV = \frac{TP}{TP + FP} \tag{14}$$

F1 score

$$F1 = 2 \cdot \frac{precision \cdot sensitivity}{precision + sensitivity} \tag{15}$$

ResNet 2

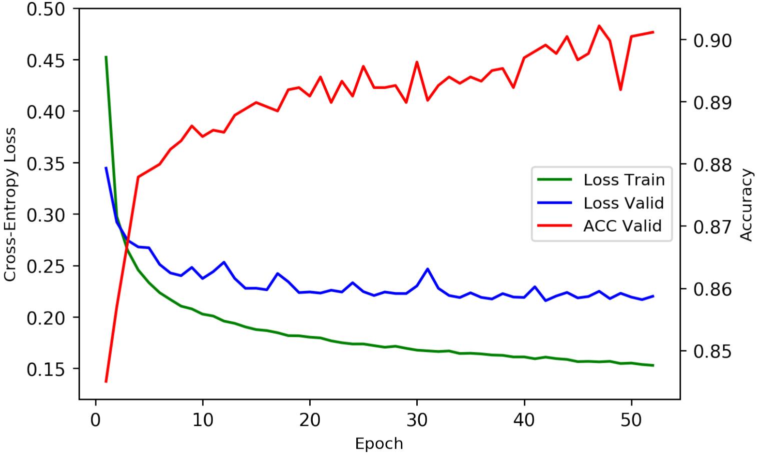


Figure 17: ResNet2 training.

ResNet 2

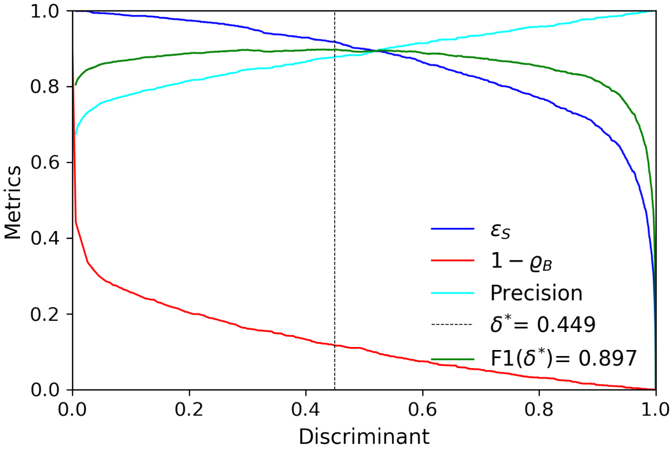
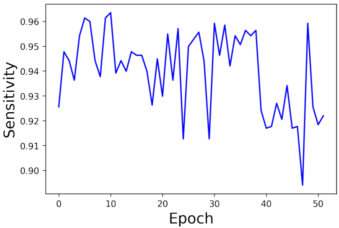
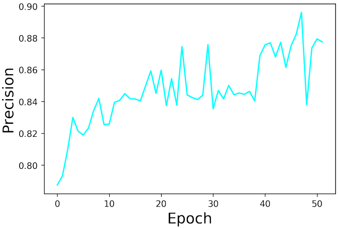


Figure 18: Selected metrics with optimal threshold value for F1 score maximization.

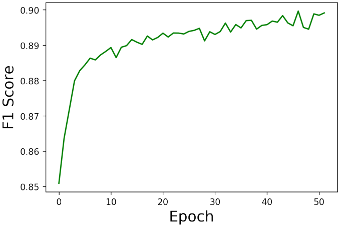
ResNet 2



(a) Sensitivity



(b) Precision



(c) F1 score

Summary

- Training of DNN is problematic.
- It is desired to use batch normalization and suitable weight initialization for DNN.
- It is desired to use skip connections and residual learning for even deeper DNN.
- We successfully implemented ResNet models in Tensorflow and used them for ν_μ, ν_e binary classification task on NOvA data.

Literature and sources

- PSIHAS, F. -RADOVIC, A. Assorted CVN Plots For Blessing. [online - 29.03.2019], 2018. `https://nusoft.fnal.gov/nova/blessedplots/`.
- AURISANO, A. et al. A Convolutional Neural Network Neutrino Event Classifier. *Journal of Instrumentation* 11 P09001, September 2016.
- HE, K. and SUN, J. Convolutional neural networks at constrained time cost. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- HE, K. - ZHANG, X. - REN, S. - SUN, J. Deep Residual Learning for Image Recognition *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. December 2016, ISBN: 978-1-4673-8851-1.
- KHOEE, A. Convolution in Image Processing. [online - 12.6. 2019] `http://arsham.me/`.