Project in group 4:

Particle image classification by machine learning

Presenter : Hikari Hirata
Motoko Fujiwara

Members : Tsutomu Fukuda
Seiso Fukumura
Amon Furuichi
Takuro Hasegawa
Hsiang-Yu Huang

Motivation

Reasons to choose Challenge-A;

- (We are beginners of machine learning(ML), and it looks like a little easier than others...)

- Hirata (experimentalist) wants to learn ML of classification, and consider to use it for signal selection of physics analysis.

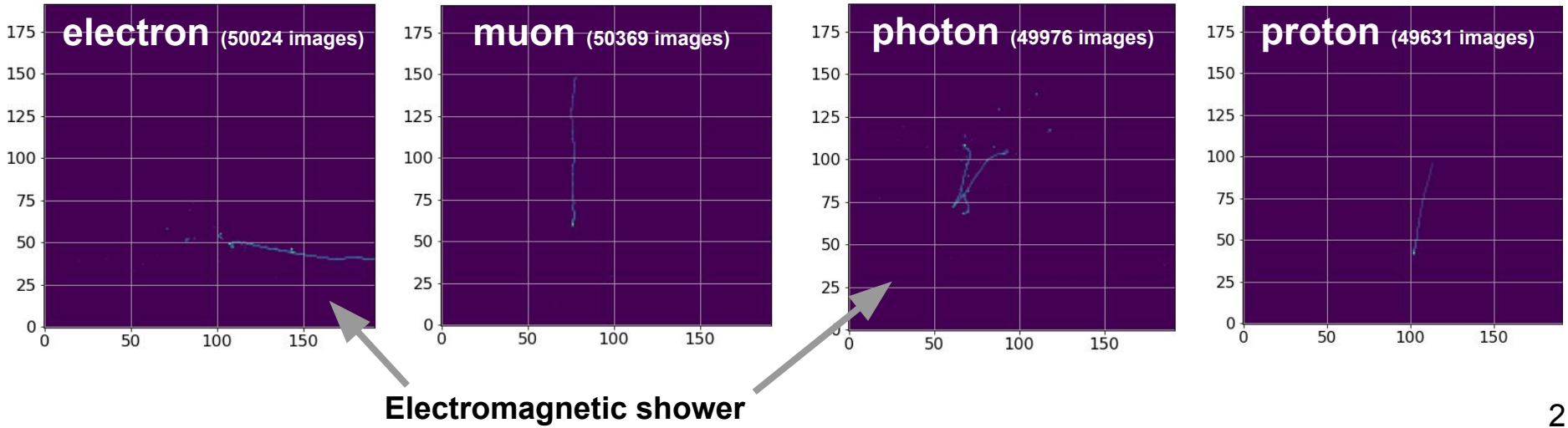
- Fujiwara (theorist) was wondering how ML identifies particle species.
  (ML is a totally “black box” for me....) ”What’s happen in that box?”

Data set

Input Data

400000 2D images: simulated particle trajectories in the liquid argon medium

→ train: 200000 image (50 %)
test: 200000 image (50 %)



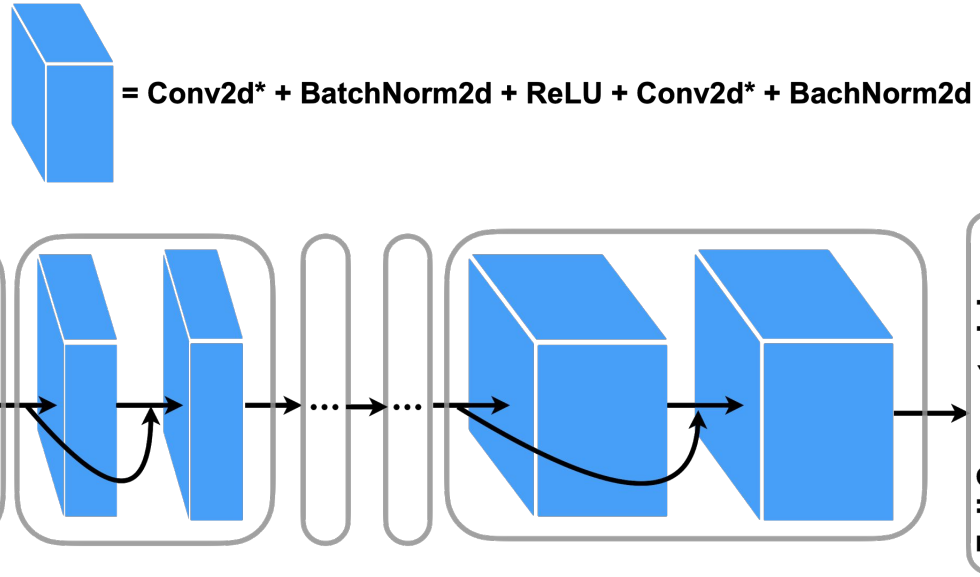
Our Model

CNN: `ResNet`
(Consists of convolution layers with shortcut connections)

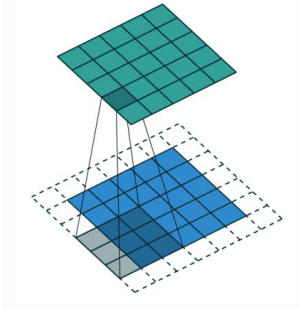
Append input information of a layer to output of the layer.
→ It can solve vanishing gradient
(We need your explanation!)

For implementation, use ResNet(10,1,16,[2,2,2,2,2]) in IntroNeuralNetwork.resnet.py
(i.e. The same network in Challenge-A-ExampleTrain.ipynb)

Concrete structure:



*Stride = 1, Padding = 1



Should we set 4?
(We realized last night...)

Evaluating machine

$$\text{Accuracy} = \frac{\text{TP} + \text{TF}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

Training

- 30000 iterations
- Mean: 0.9618
- STD: 0.0237

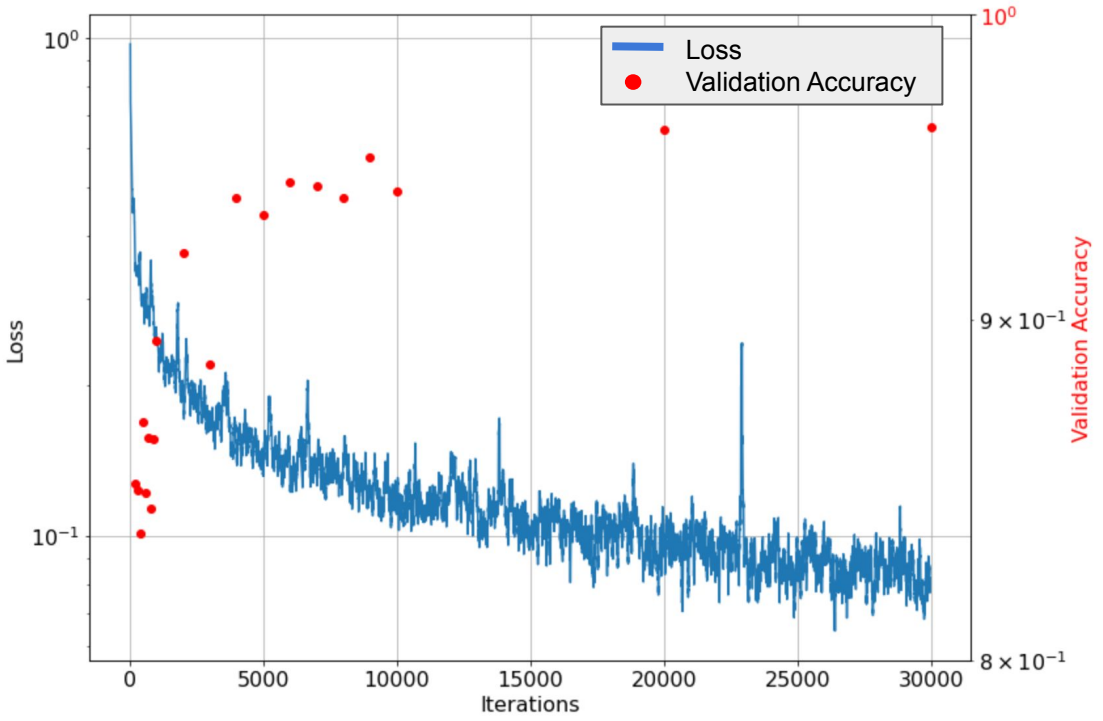
(Final values)
II
Best values!

Validation Accuracy

No "Overfitting" during 30000 iterations

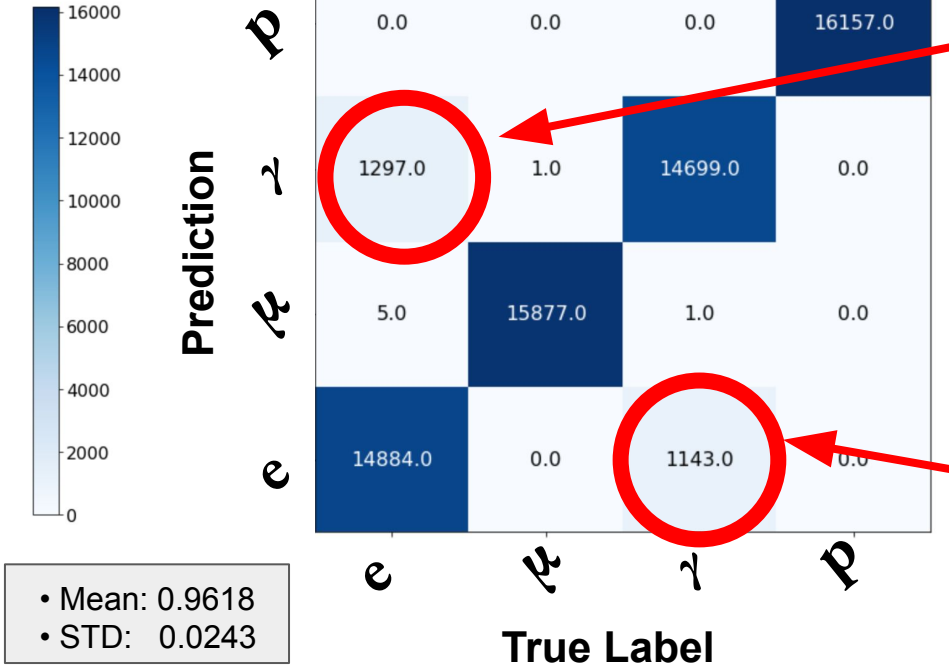
Accuracy increases monotonously over time

→ We stop learning at 30000 iterations
(due to the time limitation)

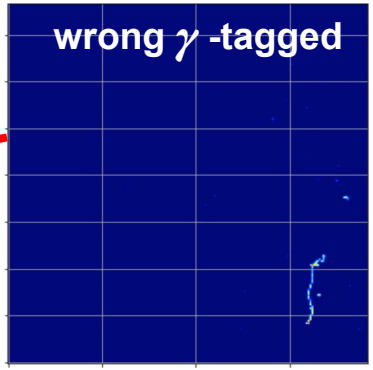


Discussions

Our question: When & Why our machine make mistakes?



- Mean: 0.9618
- STD: 0.0243



We cannot attribute the mistake to being location of images
(trans. inv. in CNN)

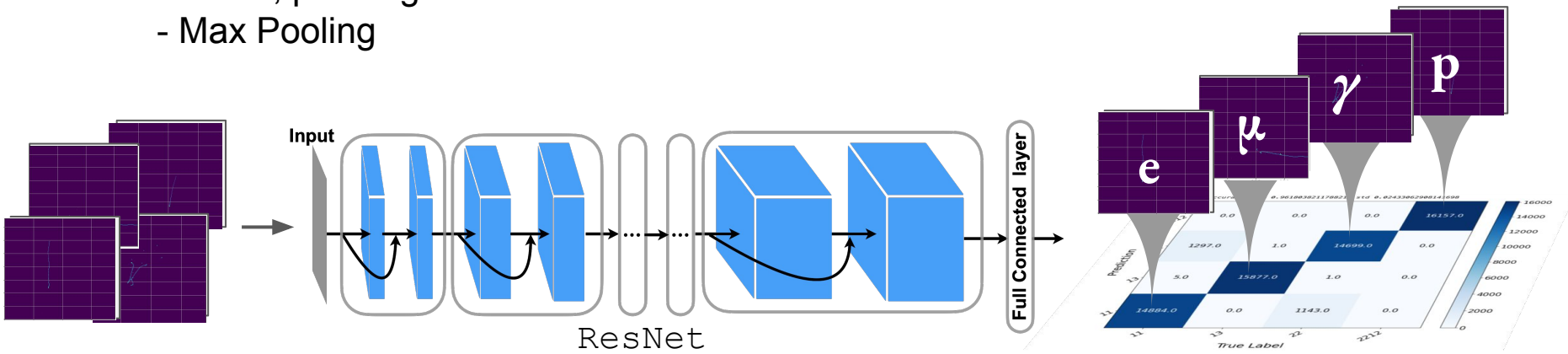
Both { e & γ } has elemag shower structures

Summary

We performed the image classification of the particle trajectories with `ResNet`

- No overfitting during 30000 steps learning
  → How many steps for overfitting?

- ML also make mistakes in { electron vs photon } classification (so as humans…!)
  → How to choose hyperparameter to decrease the loss?
  - Stride, padding value
  - Max Pooling



Thank you!

Kobayashi-Maskawa
(2008 Nobel Prize)

ES building in Nagoya U.

http://web-honbu.jimu.nagoya-u.ac.jp/fmd/02construction/tatemono/11_es/index.html

Backup

Accuracy Evaluations

[ref. tutorial slide in Day1 (by Terao-san)]

Classification performance metrics

How should we compare the classification performance that depends on an interpretation of score (i.e. Q)? This may be application specific, but a standard procedure exists with useful jargons :)

	Label P=1	Label P=0
Prediction Q=1	True Positive (TP)	False Positive (FP)
Prediction Q=0	False Negative (FN)	True Negative (TN)

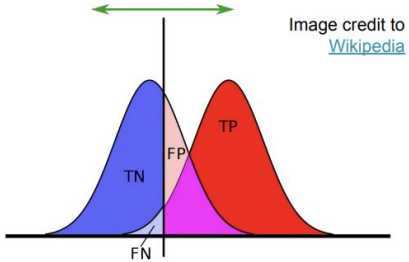


Image credit to Wikipedia

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

$$\text{False Positive Rate (FPR)} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

$$\text{True Positive Rate (TPR)} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

We can create a curve of FPR v.s. TPR by varying the score threshold. This is **Receiver Operating Characteristic** (**ROC**) curve. The area under this curve may be used as a performance metric.