An analysis of air shower data using Neural Networks.

By Miro Joensuu

Contents

The Experiments

- Maket-Ani
- KASCADE
- Airshowers

Neural Networks

- Basic architecture
- Learning
- Feed forward network

My work so far

- Optimization of Network
- Preliminary training results
- Outlook

The Experiments

Maket-Ani

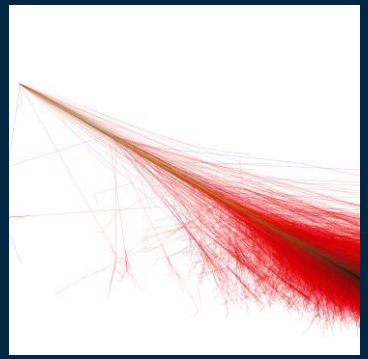
- Decommissioned cosmic ray experiment in Armenia
- Open and reusable data on KCDC datacentre
- No available simulation data
- Small number of features

KASCADE

- Cosmic ray experiment in Karlsruhe, Germany
- Available simulation data to train network
- More measured features

Airshowers

Shower Simulation using CORSIKA → Detector simulation using CRES



Data reconstruction using KRETA

Experiment

Airshowers

Input (true shower information)

- Energy
- Particle mass
- X, Y
- Zenith, Azimuth
- N.o. different particles at observation level

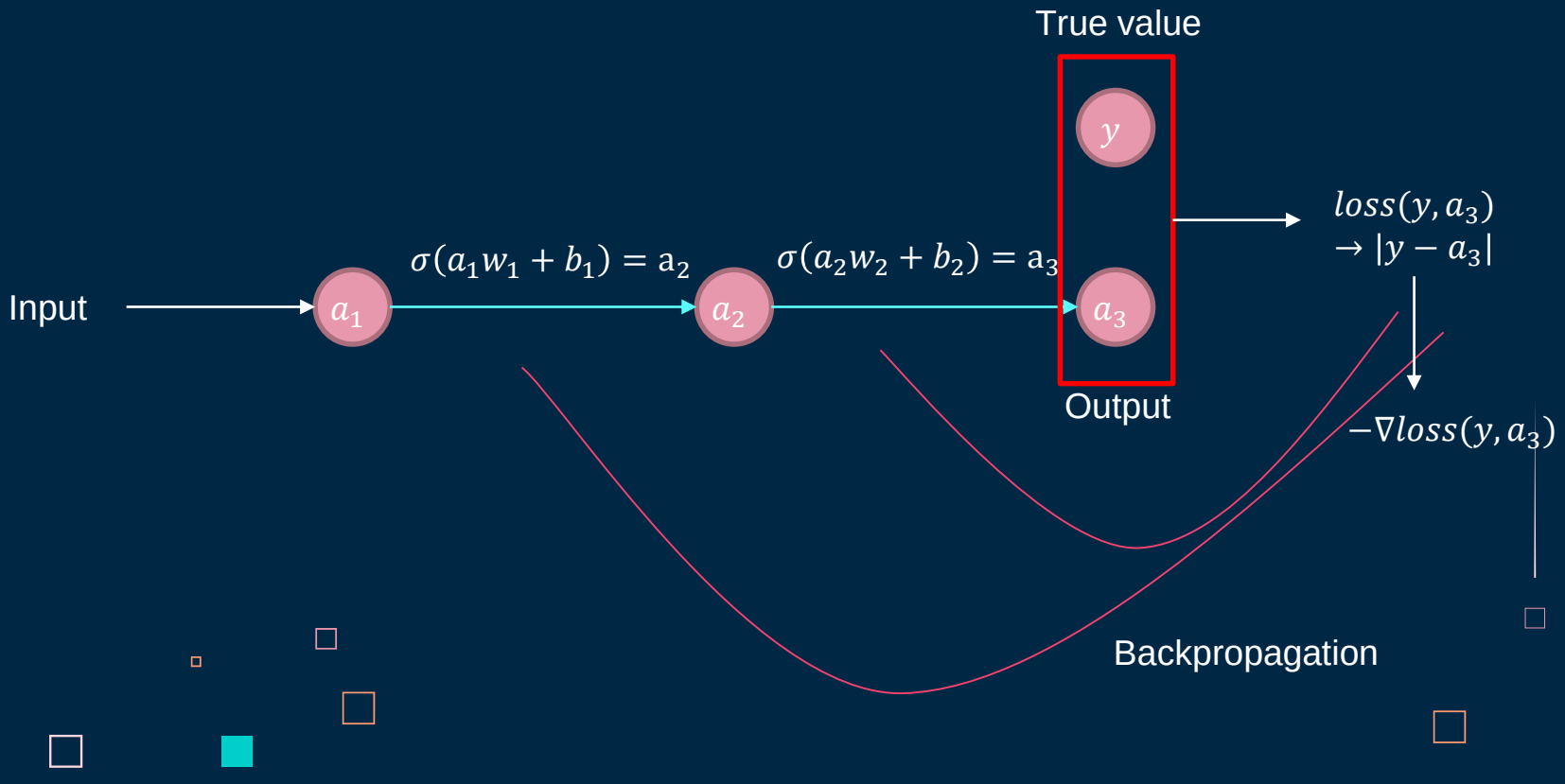
Shower Simulation → Data reconstruction

Output (reconstructed information)

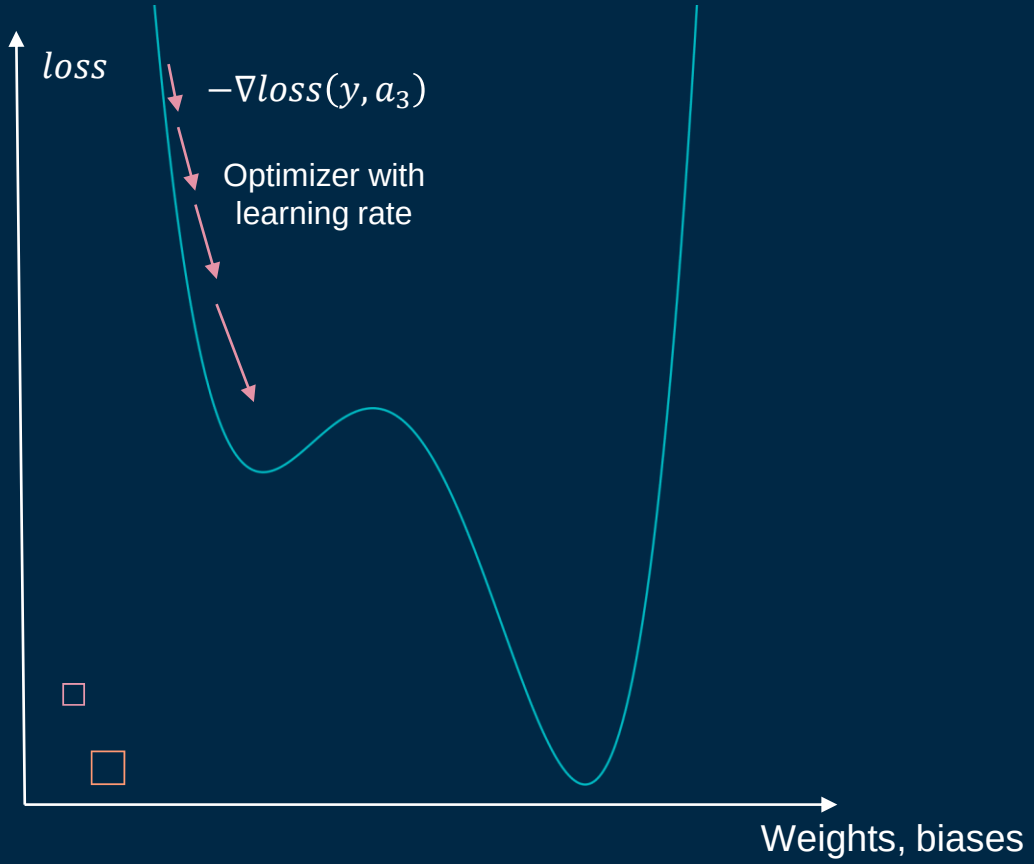
- Energy
- X, Y
- Zenith, Azimuth
- N.o. electrons, muons
- Age
- …

Neural Networks

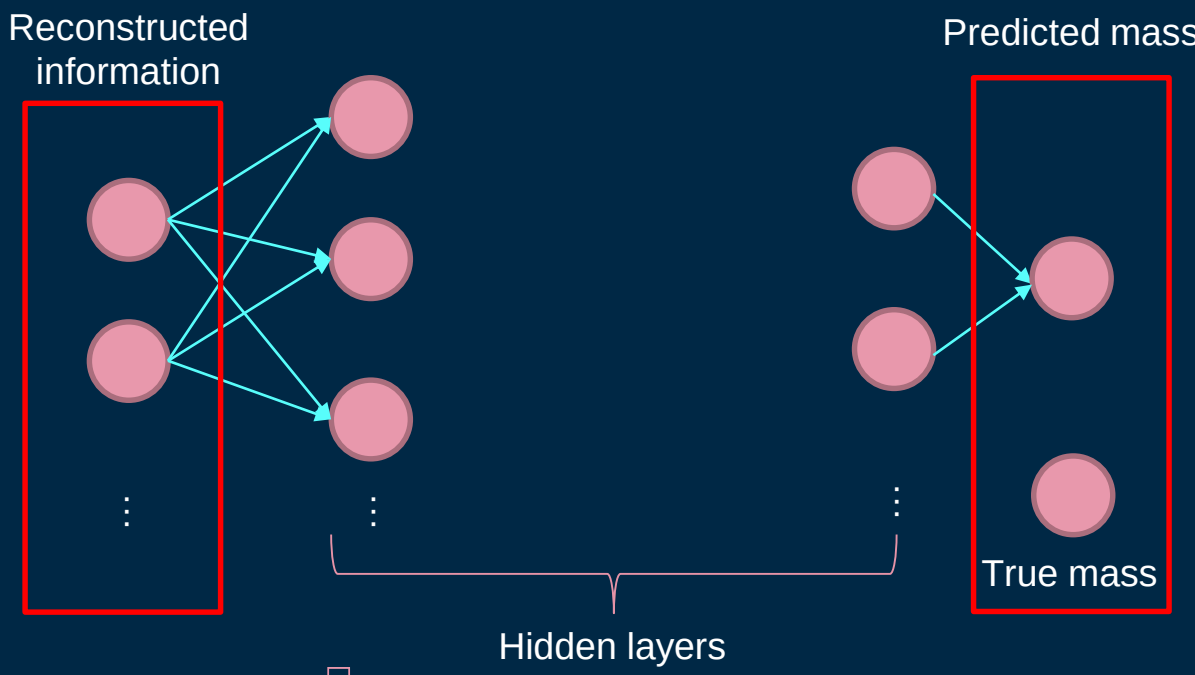
Basic architecture



Learning


loss
$-\nabla loss(y, a_3)$
Optimizer with learning rate
Weights, biases

Feed forward network



→ Complex Tool for fitting predicted mass to reconstructed information

My work

Main ideas

- Optimize network architecture
- Distinguish between proton and iron
- Quantify accuracy of network

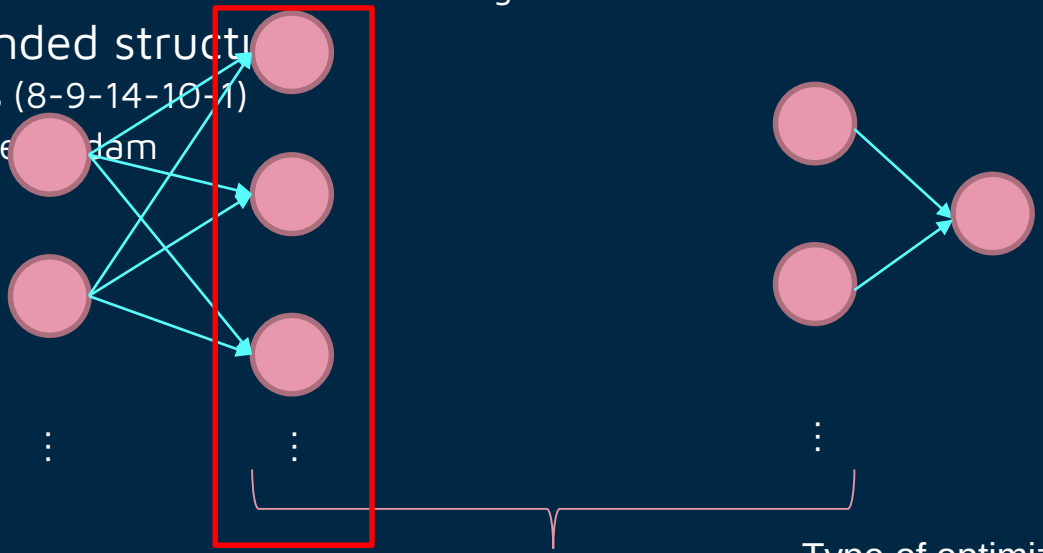
Network architecture

- Desired output → $\ln(M_{H,He,C,Si,Fe})$
  - Distance between possible outputs → 1
- Loss function → SmoothL1Loss
  - Increases punishment after ± 0.5
- Network with 2 hidden layers
  - 8 – 10 – 5 – 1
- Optimizer → Adam
  - Adaptive learning rate

kaggle™

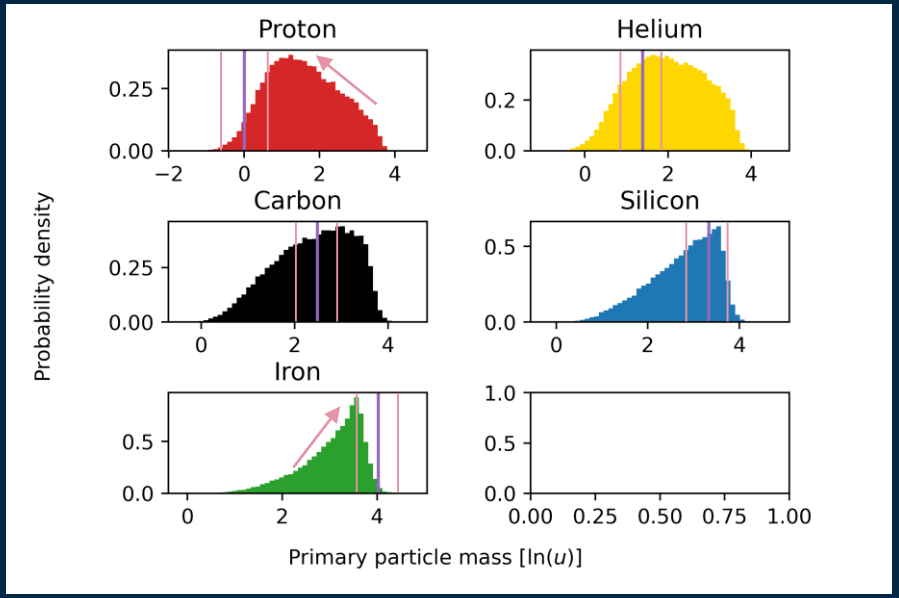
Optimization of architecture

- Hyperparameter optimization
  - Where does the loss function converge fastest
- Recommended structu
  - 3 layers (8-9-14-10-1)
  - Optimize dam
- Not tested



Results

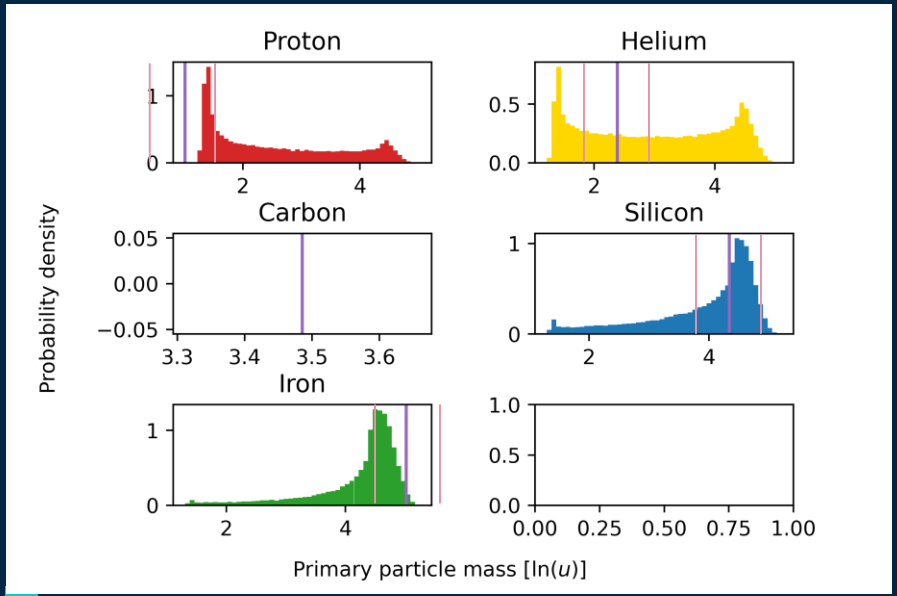
- Quantify using element plots
  - Probability to predict a certain mass for each element
- So far best model using all elements



33% accuracy
1.35 nats Kullback-Leibler divergence between Proton and Iron

Removing carbon from training set

- Clearer distinction between Proton and Iron
- Worse prediction for Helium



39% accuracy
13.99 nats Kullback-Leibler divergence between Proton and Iron

Outlook

- Use network on experimental data & compare with literature
- Implement network optimized by optuna
- Implement support vector regression
- See how well network performs on reduced data set (Maket-Ani)

References

Thank you to Paras Koundal from the KIT for tutoring & to Wladyslaw Trzaska for the support at CERN

https://pytorch.org
https://optuna.org
https://www.kaggle.com
https://scikit-learn.org/stable/index.html
https://kcdc.ikp.kit.edu/static/pdf/kcdc_mainpage/Maket-Ani-Manual.pdf
https://kcdc.ikp.kit.edu/static/pdf/kcdc_mainpage/kcdc-Simulation-Manual.pdf
https://machinelearningmastery.com/divergence-between-probability-distributions/
https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/
https://www.iap.kit.edu/kascade/english/index.php