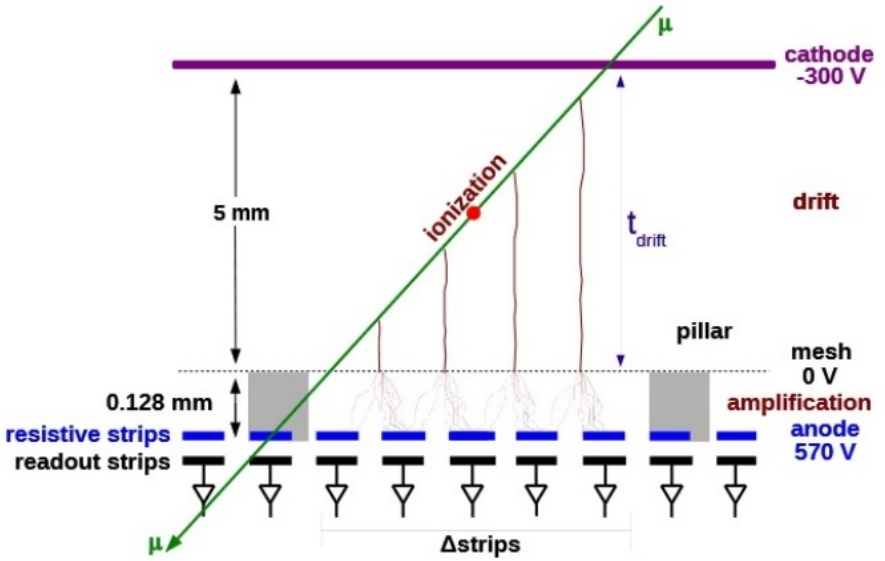
Particle track analysis with ML



Schematic of a resistive strip micromegas detector (taken from [Lösel, 2017]).

Idea : train a neural network to reconstruct from a signal

Simplified simulation

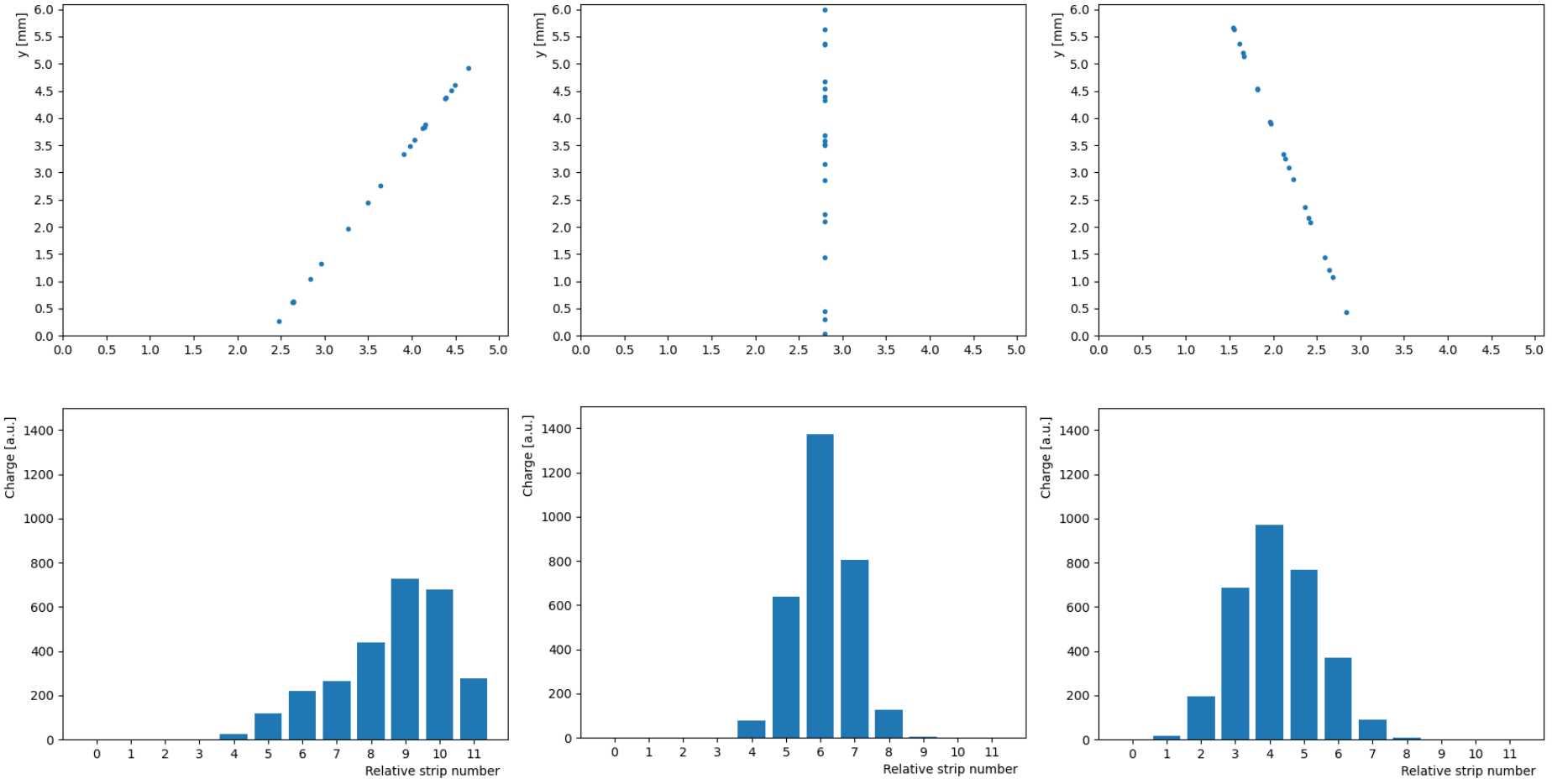
20 random ionisations along a line with angles between 60 and 120° to the detector.

Each ionisation produces a charge distribution on the detector in Gaussian form.

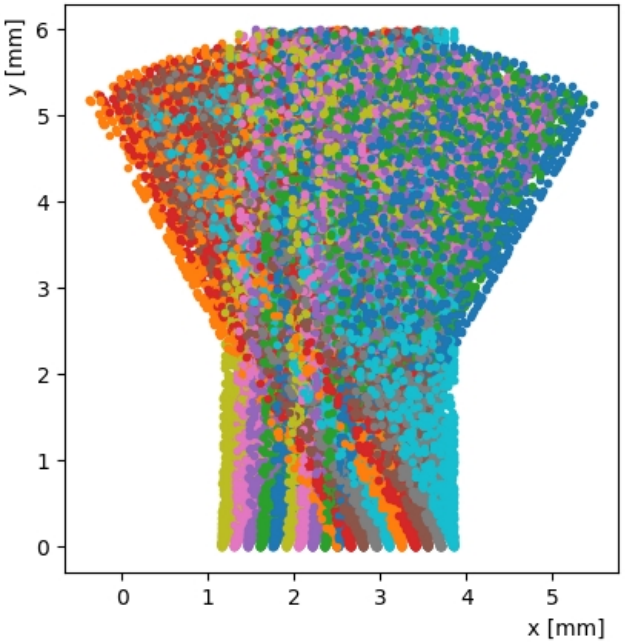
The charge distribution is integrated over each strip of width 0,425mm. Total of 12 strips.

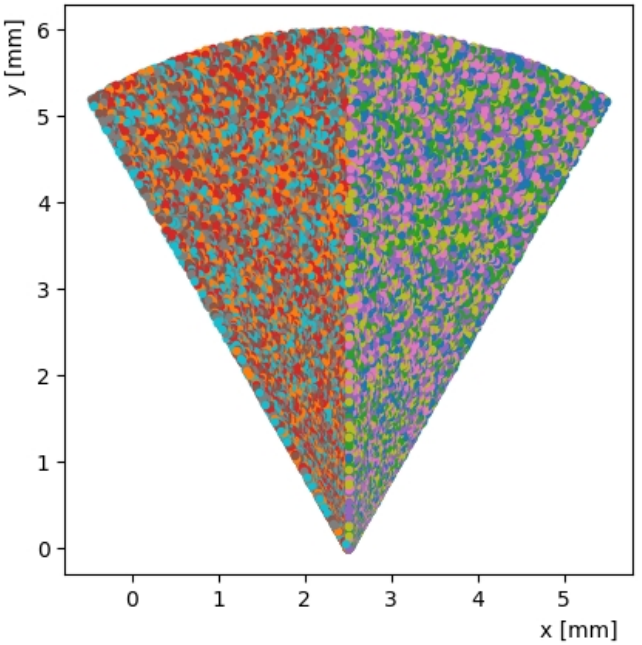
3000 events with given position and angle.

Simplified simulation



Generated data





Training set of 3000 entries + validation set

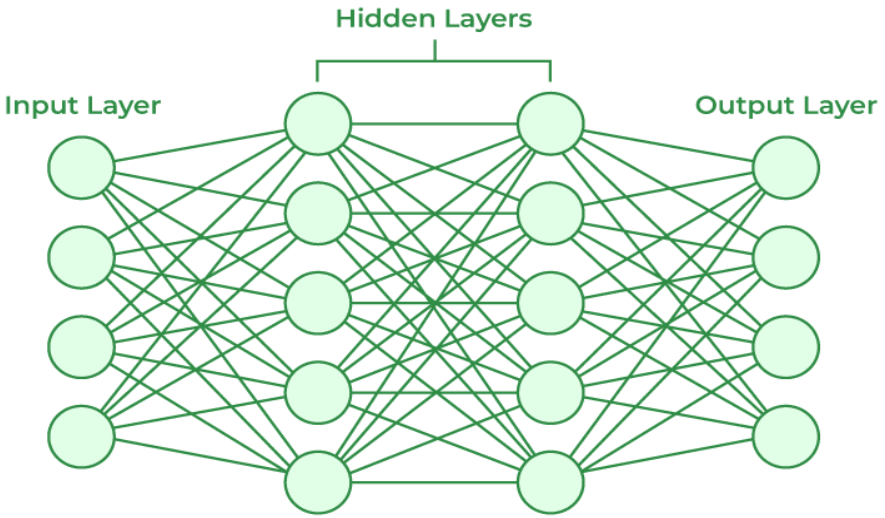
Data for position training

Position 1 to 4mm , Angle 60 to 120°

Data for angle training

Fixed position , Angle 60 to 120°

Neural network



Input layer 12 neurons / Output layer 1 neuron

3 hidden layers with 450 neurons

Regression problem → MSE loss

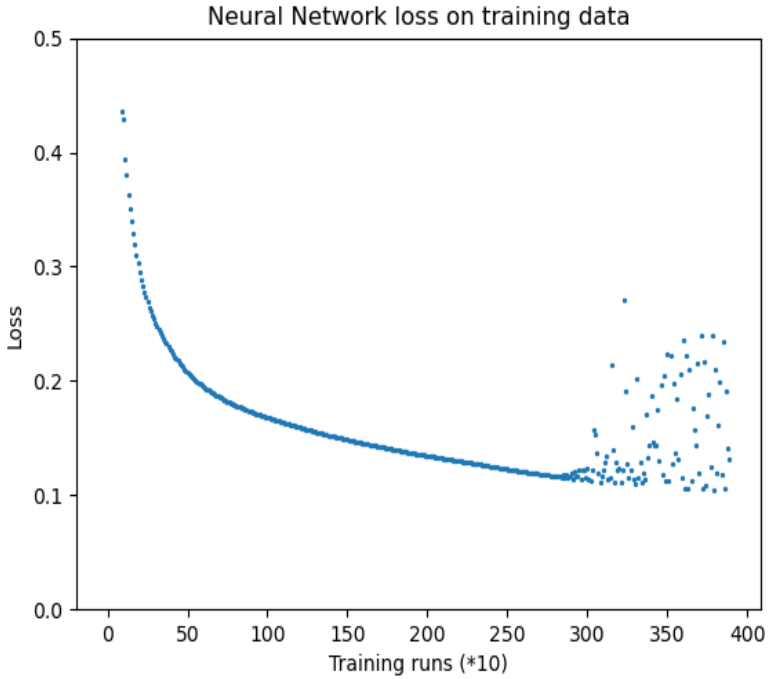
Learning rate = 1e-4, Optimizer = Adam

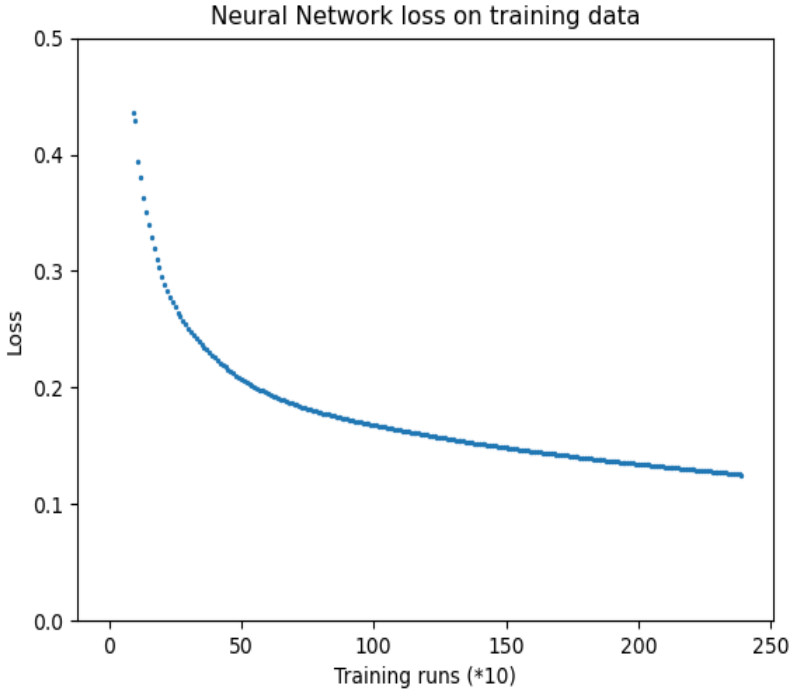
Two NN : one for angle and one for position.

Input : charges on the 12 strips for an event

Output : position or angle

Overtraining



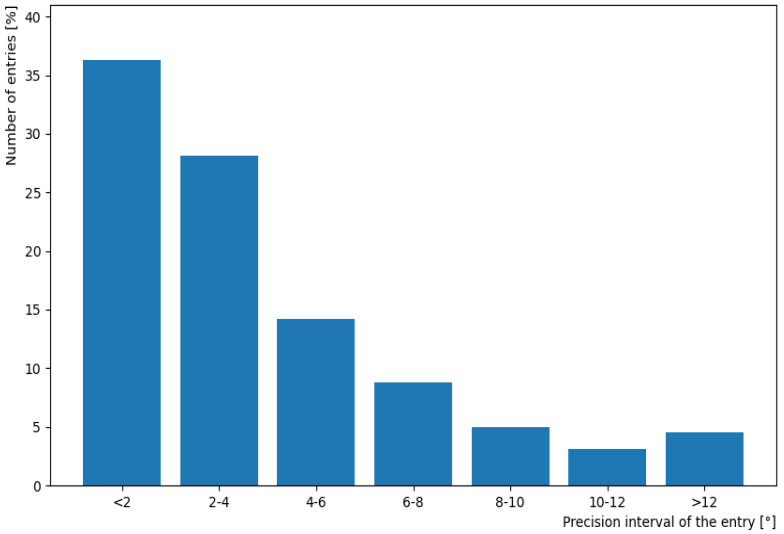


Problem on the left : overtraining leads to neural network loosing precision on a data to accomodate to another.

This also leads to a worse precision that can be seen with the validation data.

Results

Angle reconstruction

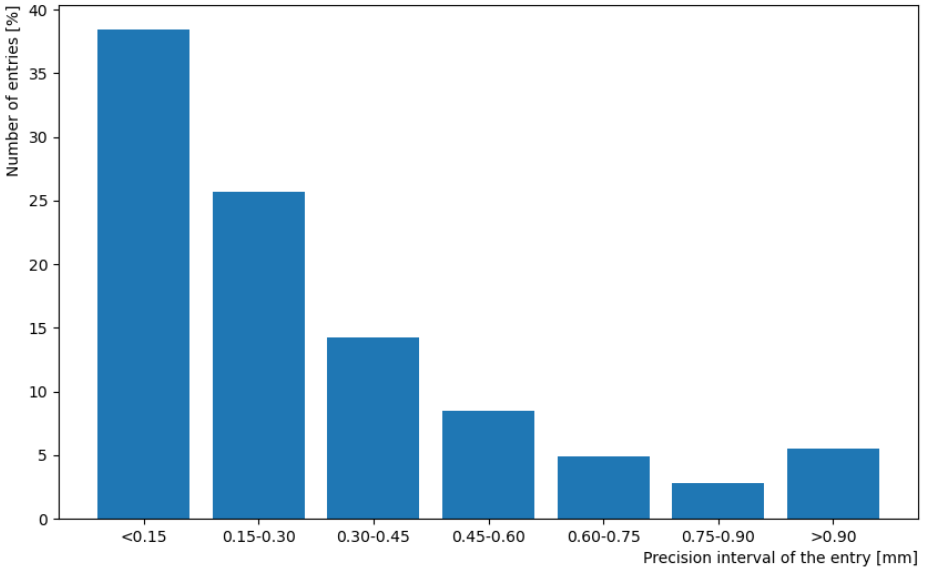


Average precision : 4°

sin(4°)*6mm ~ 0.400mm

!! Worst precision : 20°

Position reconstruction



Average precision 0.250mm

!! Worst precision : 3mm

Problem

How do we get higher accuracy and narrower distribution ?

Hyperparameter tuning

More data // data augmentation

Searching where NN fails to learn

Separate data over multiple NN ?