$\nu_e$ AND $\nu_\mu$ PARTICLE ID WITH MACHINE LEARNING

Stefano Calí, Jonas Eschle, Martina Ferrillo, Jack Jenkins, Kevin Keefe, Edward Millard, Brant Rumberger, Yicong Sui
**Motivation**

Particle Identification (PID) is a crucial challenge in experimental particle physics, ranging from high energy to astro-particle experiments.

**Aim of the Project**

In a neutrino oscillation experiment (\( \nu_\mu \rightarrow \nu_e \)) determine the number of detected \( \nu_e \) interactions on top of 1M of \( \nu_\mu \)s to get a 5 \( \sigma \) significance \( Z \).

\[
Z = \frac{n_s}{\sqrt{n_s + n_b}}
\]
LArTPC - Liquid Argon Time Projection Chamber

Provides 3D tracking and dE/dx measurement:

1. **Charged particle** enters fiducial volume and **ionises** medium

2. Primary ionisation **drifts** in uniform electric field to **readout plane** (XY)

3. Drift component (Z) reconstructed using drift velocity and trigger time
MACHINE LEARNING
“[…] What we want is a machine that can learn from experience”

Alan Turing, 1947
MACHINE LEARNING IN A NUTSHELL (II)

• Input data $X_i$ - features
• Train a model - $f$
• Make predictions $Y$ on new data
The Training Process step by step

- Provide the input \( X_i \) to the neural network
- Calculate how far is the prediction \( \hat{Y} \) from the truth \( Y \) - Loss function \( \mathcal{L} \)
- Determine the gradient of \( \mathcal{L} \) with respect to the parameters \( w_i \)
- Update the parameters \( w_{i+1} \)

\[ w_{i+1} = w_i - \alpha \cdot \nabla_w \mathcal{L} \]

Optimisation algorithm, e.g. Stochastic Gradient Descent evaluated on batches of the input data
THE PROBLEM: IMAGE CLASSIFICATION WITH ML APPROACH
THE PROBLEM: IMAGE CLASSIFICATION WITH ML APPROACH

CONVOLUTIONAL NEURAL NETWORKS (CNNs)

$P(\nu_e)$
CONVOLUTIONAL LAYERS

Image (Visual Observation)  \times  Convolution =  Feature Map
DEEP NEURAL NETWORK OPTIMISATION AND RESULTS
**DNN Architecture Modifications**

- Expand the training sample set by including a second 2D projection of the same track — more **features**

- **Hyper-parameters** tuning, e.g. number of layers, batch size etc.

- **Data augmentation**, *i.e.* artificially increase the training sample size

- **Spatial Transformer Network**
**Loss and Accuracy**

Cross entropy Loss: \( - \sum_{i=1}^{N} Y_i \log(\hat{Y}_i) \)

Accuracy: \( \frac{\text{Nr. Correct predictions}}{\text{Tot. nr. predictions}} \)

![Graph showing training and test loss and accuracy over epochs.](image)
PERFORMANCES AND RESULTS

![Performance Diagram]

SSI 2019 | S. Calí, J. Eschle, M. Ferrillo, J. Jenkins, K. Keefe, E. Millard, B. Rumberger, Y. Sui
# Performances and Results

<table>
<thead>
<tr>
<th>Prediction</th>
<th>True Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>11</td>
<td>13</td>
</tr>
<tr>
<td>12</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
</tr>
<tr>
<td>13</td>
<td>11</td>
</tr>
<tr>
<td>13</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prediction</th>
<th>True Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
</tr>
<tr>
<td>13</td>
<td>11</td>
</tr>
<tr>
<td>13</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
</tr>
</tbody>
</table>

![Graph showing performance and results](image_url)

SSI 2019 | S. Calí, J. Eschle, M. Ferrillo, J. Jenkins, K. Keefe, E. Millard, B. Rumberger, Y. Sui
Training data examples

\[ \gamma \]

\[ \mu^- \]

\[ \pi^+ \]

\[ e^- \]

\[ p \]
PERFORMANCES AND RESULTS

Prediction Accuracy on the test set:

Average: $\approx 88.6\% \text{ w.r.t. (80\%)}$

Electrons: $\approx 92\%$
Assuming: Electron events only contaminated by mis-identification of muon events

Given $N^{\text{tot}}_{\nu_\mu} = 10^6$ events, a $5\sigma$ observation requires:

$$\text{Significance} \quad Z = \frac{n_s}{\sqrt{n_s + n_b}} \geq 5$$

where

$$n_s = N^{\text{tot}}_{\nu_e} \cdot \epsilon_{ee}$$
$$n_b = N^{\text{tot}}_{\nu_\mu} \cdot \epsilon_{\mu e}$$

For $\epsilon_{ee} \approx 0.92$ and $\epsilon_{\mu e} \approx 0.001$, a $5\sigma$ oscillation discovery requires 128 observed $\nu_e$ events

from our DNN
SUMMARY AND CONCLUSIONS

• Used a Deep Neural Network for PID in a LArTPC simulation

• Optimised the DNN by improving the architecture and tuning the hyper-parameters

• Significant <accuracy> improvement: 80% to 88.6%

• Determined the corresponding number of $\nu_e$ events required for a $5\sigma$ discovery in neutrino oscillation experiment: 128
Thank you!
LArTPC - Liquid Argon Time Projection Chamber

- High-mass detector for neutrinos
- Provides 3D tracking and dE/dx measurement
- High-density medium makes amplification unnecessary
- Used by MicroBooNE and to be used as part of the DUNE experiment

Basic concept:
1. Particle enters fiducial volume & ionizes medium
2. Primary ionization drifts in uniform electric field to readout plane
3. Readout plane stores ionization magnitude & position in 2D (xy projection)
4. Drift dimension (z) reconstructed using drift velocity and trigger time
**MACHINE LEARNING ACTIVATION FUNCTIONS**

**Sigmoid**

\[ \sigma(x) = \frac{1}{1+e^{-x}} \]

**Leaky ReLU**

\[ \text{max}(0.1x, x) \]

**tanh**

\[ \tanh(x) \]

**Maxout**

\[ \text{max}(w_1^T x + b_1, w_2^T x + b_2) \]

**ReLU**

\[ \text{max}(0, x) \]

**ELU**

\[ \begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases} \]
CONVOLUTIONAL NEURAL NETWORKS (CNNs)
**Residual Networks**

\[
y = \mathcal{F}(x, \{W_i\}) + W_s x.
\]