**July 2018 Interim Report – Brendon Matusch**

**Summary**

This project is an effort to optimize discrimination between alpha particles and neutrons, using machine learning, based on various data collected during the PICO-60 experiment. Since I started work on this project in late June, I have developed several deep learning techniques for this task, and compared their effectiveness to the neural network applied in the PICO-60 paper, as well as the Acoustic Parameter (AP) frequency analysis function, which is the main tool used for this task in the PICO-60 project.

**Key Results**

I have tested many different techniques and configurations. On this page, I have highlighted four of the most successful, starting very similar to the neural network analysis in the PICO-60 paper and progressively making use of lower-level data representations to achieve higher performance. The class-wise standard deviation is used as a performance metric; this is defined on page 2.

Based on this metric, **performance was improved 49% relative to the PICO-60 neural network, and 45% relative to AP.** In addition, **performance was improved 40% relative to AP on a data set without any fiducial, pressure, or audio wall cuts.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Neural network architecture** | **Fiducial/pressure/audio wall cuts applied?** | **Input data format** | **Performance compared to PICO-60 NN** | **Performance compared to AP** |
| 2-layer perceptron with 50% dropout regularization | Yes | Position-corrected 8-band Fourier transform (piezo\_E\_PosCor) | 46% lower class-wise standard deviation | 19% lower class-wise standard deviation |
| 3-layer perceptron with 25% dropout regularization | Yes | Non-corrected 8-band Fourier transform (piezo\_E) with positional input (X, Y, Z) | 46% lower class-wise standard deviation | 26% lower class-wise standard deviation |
| 3-layer perceptron with no regularization | Yes | Non-corrected custom 20-band Fourier transform with positional input (X, Y, Z) | 49% lower class-wise standard deviation | 45% lower class-wise standard deviation |
| 3-layer perceptron with no regularization | No (data set is much larger and has greater acoustic variation) | Full resolution custom Fourier transform (50,001 bands) with positional input (X, Y, Z) | Not applicable (network does not meaningfully function on this data set) | 40% lower class-wise standard deviation |

**Performance Analysis**

The performance of neural networks relative to AP and the network used in the PICO-60 paper was done using the class-wise standard deviation. This is a function defined as *C* below, where *N* and *A* are the sets of outputs of the binary discriminator in question (AP or a neural network), corresponding to the sets of neutrons and alpha particles respectively.

The first step is to calculate the standard deviation *S* of the union of the two sets. This gives an indication of the scale of the overall distribution. When *N* and *A* are divided by *S*, they are normalized so that the standard deviation of their union is equal to 1.

While the neural network’s outputs are bounded in the range of 0 to 1 with a sigmoid activation, AP has a significantly wider range. Normalization of the union prevents this from creating a bias where AP would produce a higher standard deviation with a similarly proportioned error.

The second step is to calculate the mean of the standard deviations of the normalized sets of neutrons and alpha particles individually. This is an indication of how tightly clustered or widely dispersed the discriminator’s predictions are. Very consistent predictions of *x* for neutrons and *y* for alphas, with minimal variance off of those specific values, will produce a low class-wise standard deviation.

This performance metric is used rather than a simple accuracy score because it takes in to account not only whether the discriminator is correct or incorrect, but how confident it is. The intent is for a decisive discriminator, which produces a wide separation between the two classes, to be preferred over one that produces a nebulous cloud of outputs with a seemingly arbitrary decision boundary.