Introduction & Motivation

We seek to process signal pulses from a detector used in our physics lab $A(t)$ and identify the start time $t(0)$ and energy $E(0)$ of the pulse. Currently, our routines for registering a pulse are not accurate, and often classify detector noise as a pulse. This issue is significant, because the separation of signal and noise is critical to the success of the experiment, and is easily generalized to any other detector of this type. Furthermore, the problem of processing data to find pulses, and characterizing them, has the potential for a wider range of uses. We implement liner regression, fully connected neural networks (FCNNs), convolutional neural networks (CNNs), and kernelized principle component analysis (KPCA) with FCNN to predict $t(0)$ and found the most success with standard PCA + FCNN.

Features

The input features were traces with two channels: $A(i) \in \mathbb{R}^{2048 \times 2}$. For the linear regression and FCNN we flattened the trace to a 4096 dimensional vector. For the CNN we kept the shape of the trace. For PCA + FCNN model we flattened the traces and used PCA to find the first 1024 principle of the components (PCs) using 20% of the training set – which explain 89.83% of the variance. We decided to try PCA because we plotted the correlation matrix of the traces and noticed the all the points of the trace are positively correlated. For the Kernel PCA, we used a radial basis kernel, and 1024 PCs. We tried this too because we thought the relationship between the projected features could be non-linear.

Discussion

From this project we were able to develop a methodology to construct an effective tool that we might use as part of our physics experiment to determine the start-time of pulses measured by our detector. After constructing our dataset from our Monte Carlo simulations, we trained liner regression, FCNN, CNN, a standard PCA fed into FCNN, and KPCA with a radial basis kernel fed into a FCNN to predict $t(0)$ and had the best Test set MAE with the standard PCA + FCNN method. Our goal was to get to a MAE around 1 or 2, however the lowest we ever got on training was 4. This is likely due to the fact that the pulses are so noisy – which is why we chose this challenging problem in the first place. An important insight from this project was that more complex models don’t always produce better results, as can be seen comparing the CNN and KPCA+FCNN with the PCA+FCNN. Another lesson we learned was that producing the dataset and preparing it for training can be the most time intensive step. Finally, while we didn’t accomplish exactly what we set out to do we are content with our results and will continue improving on them.

Models

We represent $A(i) \in \mathbb{R}^{2048 \times 2}$ and $A(0) = Flatten(A(i)) \in \mathbb{R}^{4096}$

In all the models we mininmize the mean squared error (MSE):

$$\frac{1}{m} \sum_{i=1}^{m} (t(i) - t(0))^2,$$

however we are interested in reporting the mean absolute error (MAE):

$$\frac{1}{m} \sum_{i=1}^{m} |t(i) - \hat{t}(i)|.$$

CNN:

```
Input Layer
Conv. width = 4
N Filters = 32
Strides = 1
```

```
Max Pooling
Pool size = 2
Strides = 2
```

```
> Flatten + FCNN + Prediction
```

Linear PCA FCNN: We perform PCA by finding the eigen-basis of the correlation matrix $C$, where $V(k)$ is the k-th principle component:

$$C = \frac{1}{m} \sum_{i=1}^{m} A(i)A(i)^T,$$

$$\hat{A}(k) = V(k)A(i).$$

We then feed the projections, $\hat{A}(k)$, into the FCNN displayed below.

Kernel PCA FCNN: We apply the kernel trick on the projections as follows and feed the result into the same FCNN.

$$\hat{A}(k) = V(k)A(i) = \sum_{j=1}^{m} \phi^T(k) \Phi(i)$$

$$\Phi(i) = \sum_{j=1}^{m} \phi_j K(A(i), A(j))$$

Discussion

From this project we were able to develop a methodology to construct an effective tool that we might use as part of physics experiment to determine the start-time of pulses measured by our detector. After constructing our dataset from our Monte Carlo simulations, we trained liner regression, FCNN, CNN, a standard PCA fed into FCNN, and KPCA with a radial basis kernel fed into a FCNN to predict $t(0)$ and had the best Test set MAE with the standard PCA + FCNN method. Our goal was to get to a MAE around 1 or 2, however the lowest we ever got on training was 4. This is likely due to the fact that the pulses are so noisy – which is why we chose this challenging problem in the first place. An important insight from this project was that more complex models don’t always produce better results, as can be seen comparing the CNN and KPCA+FCNN with the PCA+FCNN. Another lesson we learned was that producing the dataset and preparing it for training can be the most time intensive step. Finally, while we didn’t accomplish exactly what we set out to do we are content with our results and will continue improving on them.

References