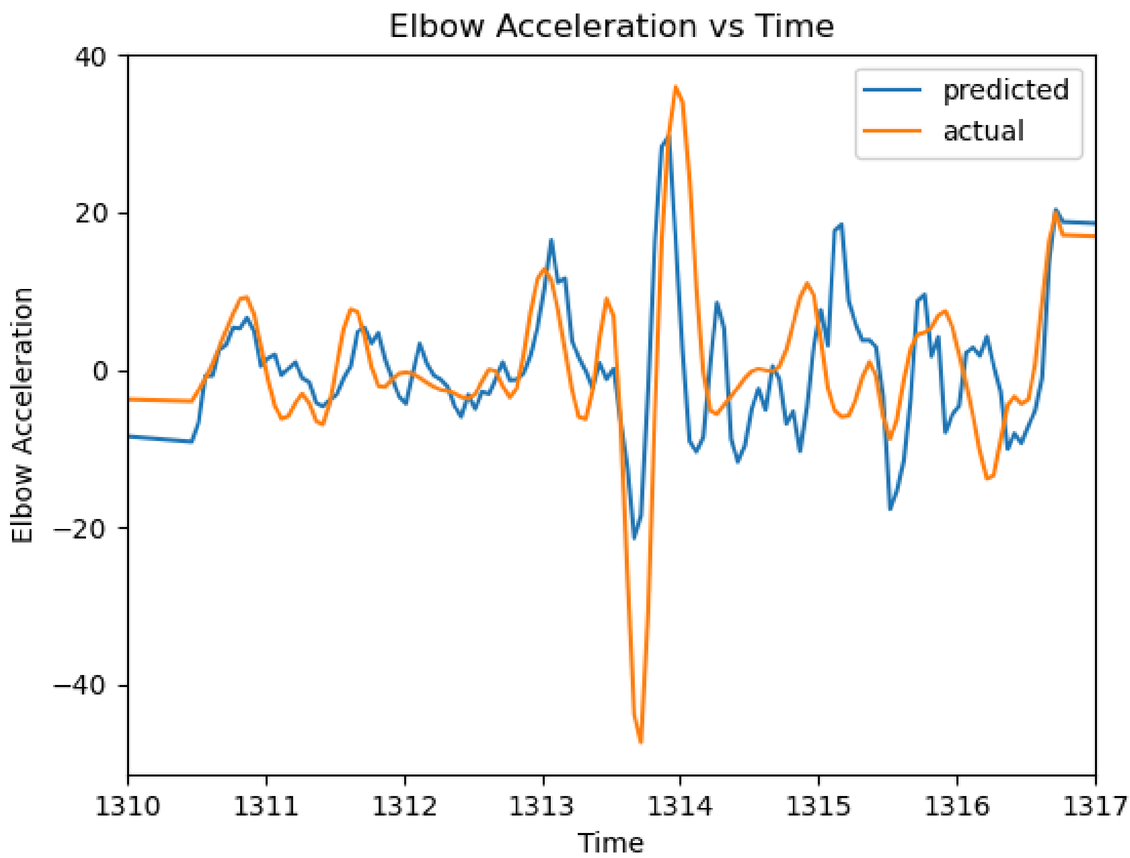
Brandon Michaud

C S 5043 Homework 1

Network Architecture

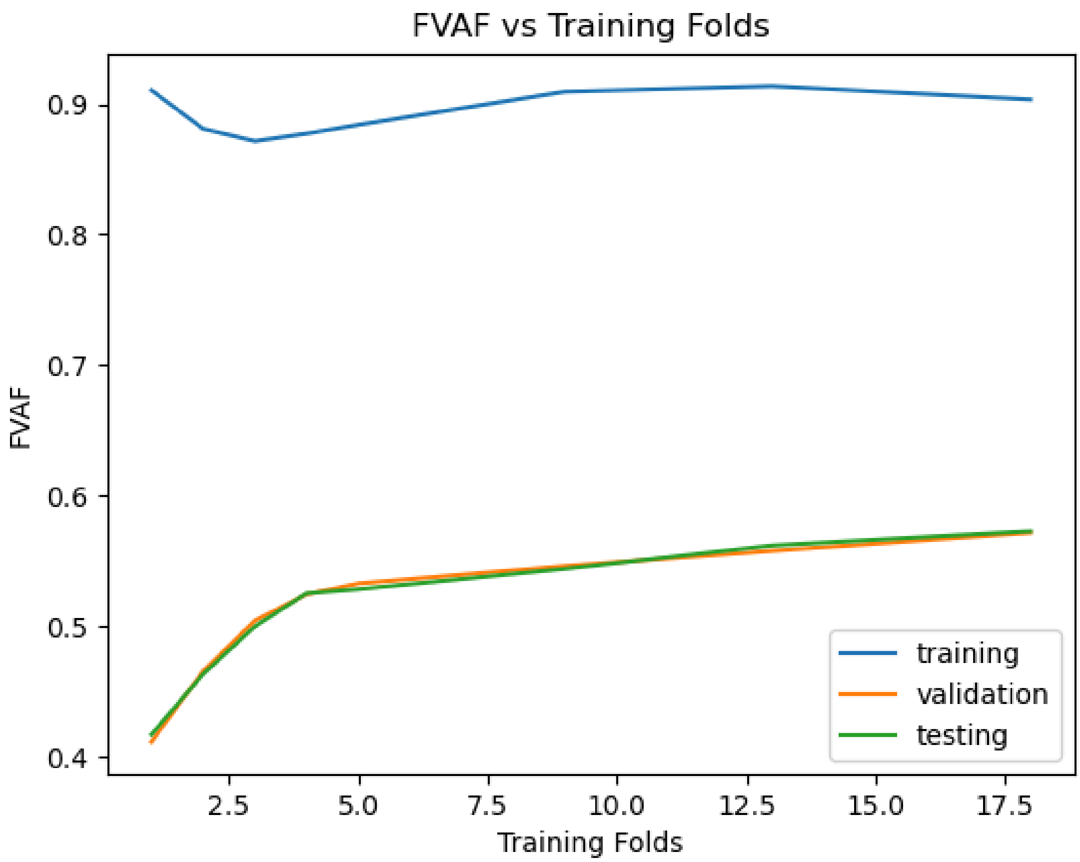
The final architecture I landed on has the input layer, then a hidden layer with 100 neurons, then a hidden layer with 10 neurons, and then the output layer. All layers are fully-connected and sequential. Hidden neurons use an exponential linear unit (ELU) activation function to give the nonlinearity. Output neurons use a linear activation since the range of output is any real number. I used mean squared error for the loss and the Adam optimizer. I found that a learning rate of 0.0001 was optimal for this architecture. The maximum number of epochs I used was 100, but I also implemented early stopping with a minimum change of 0.001 and a patience of 25 epochs. Most experiments stopped around 40 epochs.

Task 1



18 training folds were used to produce this figure. The time period shown is one monkey brain machine interface interval from the test fold (i.e. not showing any discontinuities where the monkey took a break). The predicted elbow acceleration values closely resemble the actual values. The most obvious difference is that the actual elbow acceleration appears very smooth over time whereas the predicted elbow acceleration is more jerky and sharp.

Task 2



The fraction of variance accounted for (FVAF) for the training set is much higher than the FVAF for the validation or testing sets. This is to be expected because the model fits the training set. The maximum FVAF is 1, so the training set performs very well on this metric. For a small number of training folds, the FVAF on the training set is high, but it decreases as the number of folds increases. This is expected because the model becomes more general as more folds are used and thus the performance on all training folds decreases. As more data is processed, this generalization gets better, which is why the training set FVAF rebounds. The validation and testing curves are very similar, indicating that our validation set is a good representation of our testing set. For both the validation and testing set, the FVAF is low for a low number of training folds and increases as the number of training folds increases. This is expected because more training data allows the model to generalize better on the validation and testing sets. All FVAF values for all training set sizes for all sets are considered useful by the definition given in the specification.