ECE 189 - Fall 2021

Daniel Smith
University of California, Los Angeles

2 December 2021

ABSTRACT

The Goal of this project is to build a brain-machine interface decoder to decode the velocities of a trial monkey's hand. By implementing the optimal linear estimator algorithm in MATLAB, we are able to get a rough idea of the velocities of the monkey's hand based on neural activity.

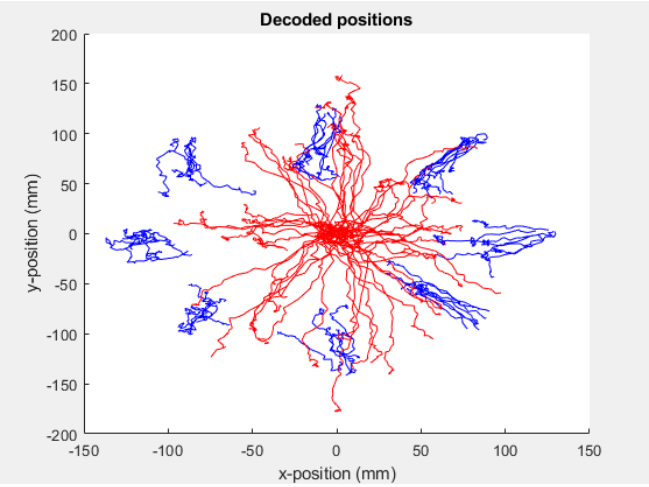

Figure 1. Vanilla Decoder

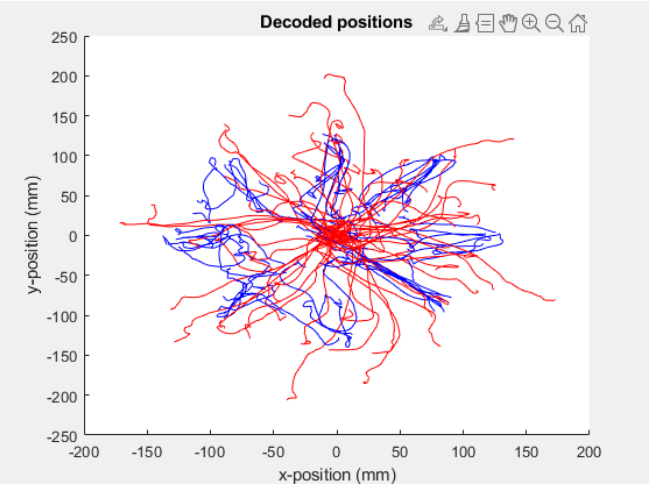

Figure 2. Low Pass Filter Decoder

1 INTRODUCTION

The linear estimator algorithm works by optimizing the parameters θ, where θ are the coefficients of the function that is determined by minimizing the Mean Squared Error. The Mean Square Error is the Loss function we use as it penalizes points further away from our function and gives us only positive values. The objective of this project is to augment the data in order to obtain a lower Mean Squared Error than the vanilla decoders Mean Square Error.

2 AUGMENTING FEATURES

We implement 3 different augments: Low Pass Filter, High Pass Filter, Difference in Neural Data. The vanilla decoder's Mean Squared Error is 3.4926e+03 while the plot is shown at Figure. 1.

2.1 Low Pass Filter

A Low Pass Filter will filter out low frequencies in our neural data. When the Low pass Filter is implemented, the Mean Square Error Value we obtain is 2.1392e+03. This value is lower than the vanilla decoder MSE, which indicates that the a low pass filter does indeed improve performance. When we look at the plot (see Figure. 2), we see that it does a much better job compared to the vanilla decoder of matching the center-out and center-back reaches.

2.2 High Pass Filter

A High Pass Filter will filter out high frequencies in our neural data. When the High Pass Filter is implemented, the Mean Square Error is 2.1683e+03. This value is still lower than the vanilla decoder MSE, which indicates it still has a better performance. However it is still marginally worse than the Low Pass Filter, in terms of MSE. The plot (Figure. 3) shows that both the Low Pass Filter and High Pass Filter have similar plots. Regardless, both Filters will still work better than no Filter at all.

2.3 Difference of Neural Data

Using the difference of neural data will give us a MSE of 9.0438e+03 While this error is certainly high and does not really track the velocity of the monkey's hand it does give some insight as to where the buttons were located according to (Figure. 4). So something like this would not be too useful for us but may be beneficial to still keep and create, if we needed access to the button locations for some reason.

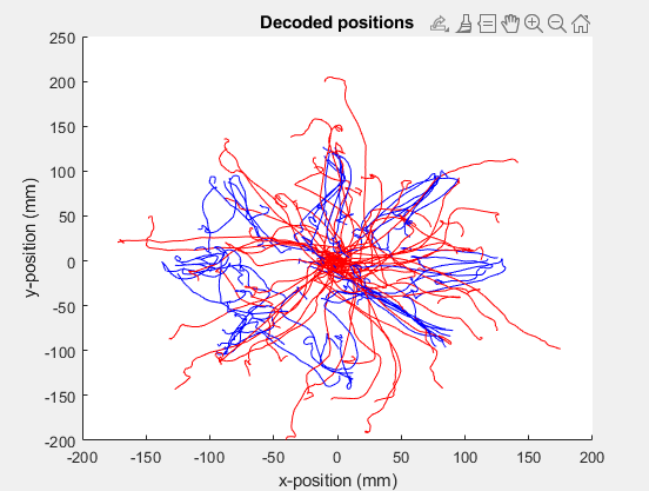

Figure 3. High Pass Filter Decoder

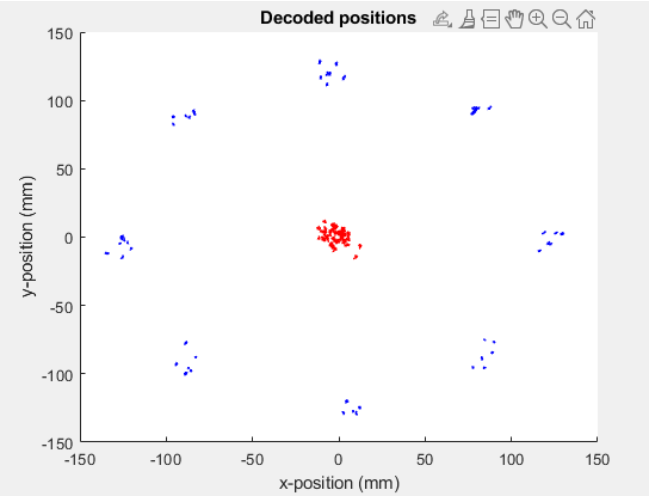

Figure 4. Difference of Neural Data

3 ADDITIONAL WORK

There are several different ways to augment the data. The method I chose was to use a different Kernel to mimic a High Pass Filter. If we use the kernel: [-1, -1, 4 , -1, -1], we can filter out some of the lower frequencies in the data. While the Mean Squared Error is fairly high at 9.0620e+03, it is interesting to note that this function has a similar MSE as the Difference in Neural Data as well as somewhat similar plots. As seen in Figure. 4 and Figure. 5 both seem to have the center-out in the middle at (0,0) while also having the center-back at where the buttons are.

4 CONCLUSIONS

Overall using any Filter is better than using None. This is because neural data is often plagued with noise, due to the complexity of the domain and the tools used to measure them. While the implementations were fairly basic, we still managed to obtain fairly decent results. It is still possible to use more advance machine learning algorithms, such as a MLP or Deep Neural Network to obtain more accurate results.

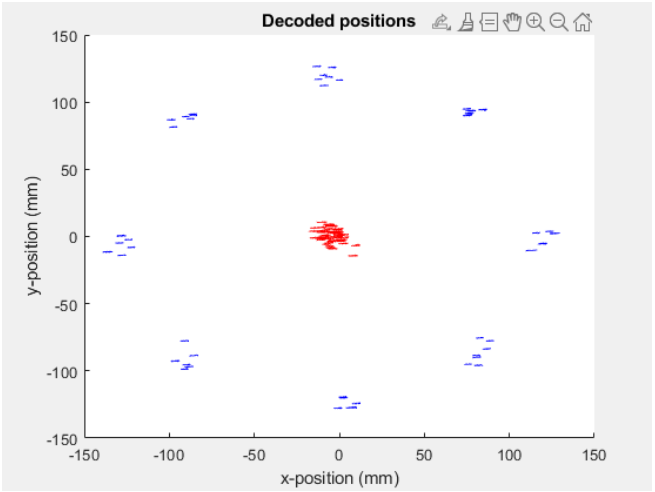

Figure 5. High Pass Filter Decoder

ACKNOWLEDGEMENTS

The data used in this project is property of Prof. Krishna V. Shenoy and Prof. J.C. Kao.

This paper has been typeset from a TEX/LATEX file prepared by the author.