Volume 11 Issue 4 (2022)

The Analysis of the Accuracy of Support Vector Regression Classification of a Neural Dataset Originating in the Motor Cortex of a Macaque Monkey

Misha Nivota[1] and Omar Tawakol[2#]

[1]Eastside Preparatory School, USA
[2]The University of Chicago, USA
[#]Advisor

ABSTRACT

Neural decoding is a constantly adapting field of applied machine learning, using many different machine learning algorithms to analyze neural data. One such analysis of neural data is the interpretation and analysis of data originating the motoro cortex and the prediction of stimuli given to an organism using the motor response. In looking at a specific dataset, containing data from the motor cortex of a macaque monkey, where its neural response to directional stimuli was measured. Two different neural decoding algorithms were previously used to analyze this dataset, yet the highest accuracy they yielded was below 90%. There is a need for machine-learning based neural decoding algorithms to decode movement-relating neural data with higher accuracy. Support vector regression (SVR), a linear regression based model of the machine learning algorithm support vector machines, was chosen for analysis. In this study, we aim to evaluate the predictive accuracy of SVR using a dataset obtained from a macaque monkey. The model predicted the directional stimuli with an accuracy of 82.47% percent.

Introduction

Neural decoding is a method of neural mapping that allows scientists to predict external stimuli using the neural activity produced in the brain in response to a given stimuli. Utilizing machine learning, algorithms are able to predict external stimuli by learning from labeled data (neural activity data with the specific stimulus). Once they have properly learned the features of neural activity that can be used in stimulus identification, algorithms can then be applied to neural activity data without given stimuli and can predict the stimuli given from the corresponding neural activity (Glaser).

Neural decoding can be used to improve assistive technology for those with with paralysis and can be utilized to diagnose neural diseases. It also helps us understand more about the brain, identify functions of specific neurons and neuron clusters. Neural decoding can also be integrated with brain-computer interfaces and other neural prosthesis to improve their functionality and the quality of patients' lives.

There are many neural decoding algorithms, each approaching the analysis of neural data in different ways and each having their own advantages and disadvantages. This study aims to evaluate another neural decoding algorithm, known as support vector regression.

In this project, data from a previous study was used, in which the cortical neural data from a macaque monkey was analyzed. The goal of the previous study was to predict the directional stimuli presented to the monkey based on neural data from the motor cortex from the monkey's response to the stimulus. Various methods could be used, including population vectors and maximum likelihood estimation. However, we believed that another model could be created that would decode the macaque data with a higher accuracy. By using a different algorithm, the accuracy could be improved. Support Vector Regression was chosen as the algorithm that would decode the macaque data with the best accuracy. The goal for this study was to create a new decoding model for the macaque monkey data using Support Vector Regression to improve upon previous models and yield higher accuracies.

Methods

The dataset in this study from the motor cortex of a macaque monkey was previously analyzed with other algorithms. The goal of this study was to utilize another neural networking algorithm to increase the accuracy, and, after some research on neural decoding algorithms, Support Vector Regression was chosen to analyze the data.

Support Vector Regression is a regression-based model of Support Vector Machines (Goel). The aim of Support Vector Machines is to define a multidimensional hyperplane that separates the different classifications or labels groups of data in all dimensions (I know python (Director)). If multiple classifications are necessary, multiple hyperplanes will be used. Hyperplanes are created by equally dividing the space between extreme points, the data points of separate classifications that least fit into their separate classifications. These points define the margins of the separation space and the hyperplane dissects that multidimensional space in half. The distance between the extreme points and the hyperplane is known as the best margin. But Support Vector Machines can only be used with data buckets, therefore, in order to analyze linear data, Support Vector Regression is used instead. The regression algorithm draws a line of best fit using the Support Vector hyperplane in order to predict future values and classify linear data. In Support Vector Regression the best margin is known as epsilon-deviation and the hyperplane is defined by: $y = < w, xi > + b$.

Support Vector Regression was selected because of providing relatively high accuracy in previous studies that used motor cortex data. This algorithm is characterized by its insensitivity to the displacements caused by outlying data points, can be used for any number of classifiers, and requires less computation than other regression models. It is also relatively easy to implement.

This study utilized an existed system of neural networking models to create our SVR model. A GitHub repository, created by Kording Labs, included several different neural networking models, including SVR, and evaluated their efficacies (Benjamin and Glaser). This repository also contained instructions on how to use their pre-created models and examples of their code. These examples were used to produce a working SVR model with Kording Lab's code in this study. The data Kording Labs used was in .pickle format, while the used in this study was in .mat format. Therefore, the raw data had to be preprocessed prior to using it in this study.

The structure of this study's data was a series of continuous timestamps of individual neurons marking when they fired action potentials, known as spike times, along with a series of continuous start times for each trial. The output of this data was a series of directions, which were the stimuli from each trial. The data from Kording Labs was designed as a matrix of time intervals and neurons, where each cell included the number of times each neuron fired within a given time bin. This study's data had to be organized in a similar fashion to be inputted into Kording Lab's SVR model (Benjamin and Glaser). First, each neuron's data had to be separated by trial. Using the go times in this study's .mat data, which contained the timestamps that each trial started. It was determined whether a specific spike count for a specific neuron occurred during a specific trial. If so, that spike was recorded and the total spikes during each trial for a specific neuron was outputted and placed into a matrix. The final matrix for the study's data was a matrix of trials by neurons, with 158 trials and 143 neurons, and each cell was a count of how many times each neuron fired during each trial. This data preprocessing can be accessed using the GitHub repository and is in the file labeled "Data Pre-Proc". Once the pre-processing was complete, the new data was entered into the SVR model. There were a few small errors when inputting the data into the model, such as the data being saved as list items instead of arrays. However, after a few easy fixes, the model ran with the motor cortex activity data.

Results

After conducting a series of tests with the model, the following results were obtained. The SVR model predicted the directional stimuli given to a macaque monkey using its neural activity with an r^2 of 0.82468871, or an accuracy of 82.47%.

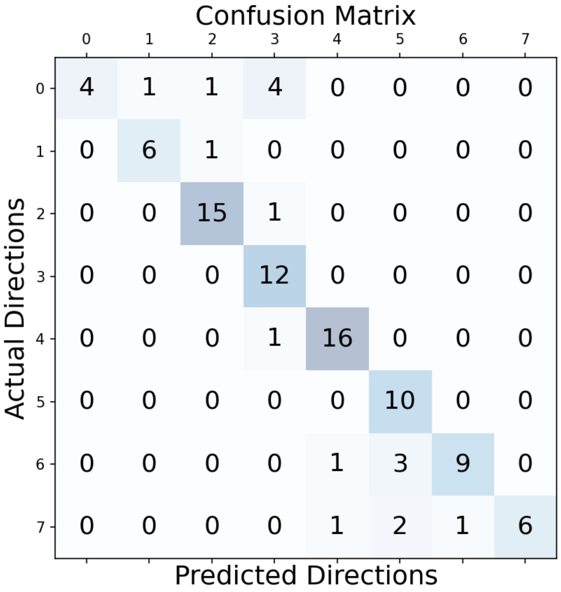

Figure 1. SVR Confusion Matrix. The numbers on each axis represent specific directions, starting with an upwards direction for number 0 and moving clockwise to direction 7, with a new direction every 45 degrees.

Discussion

Looking at the results, it is clear that the model used did not decode the macaque monkey's neural data with a higher accuracy than previous studies. This model only yielded an accuracy of 82.47% while maximum likelihood estimation (MLE) yielded an accuracy of 86.25%. This r^2 of 0.8247 was not originally achieved and considerable tuning had to be completed within the model to increase its accuracy.

The initial SVR model had an r^2 of 0.12. This was the basic model implemented from Kording Labs and due to the poor accuracy, changes were made to the original code to improve the model. A process of trial and error was used to improve accuracy, in which many different tuning methods were attempted and tested. In the initial model, trials before and after the current data points were used in each iteration of decoding, but that process was removed in the final model, because it was used for continuous data and the monkey data was intermittent. Another feature of the original model removed data for any neuron with less than 100 spikes, but the data from the monkey was recorded over a shorter period of time and significant portions of neurons had less than 100 spikes, so that limitation was decreased to less than 1 spike. The last adjustment was made in the split of data for training, testing, and validation. Kording Labs' dataset was split so that 50% was used for training, 15% for validation, and 15% for testing. The last 20% of their data was deemed unusable due to issues with how the data was recorded. After different attempts, it was concluded that the best splits for the macaque monkey data was 60% used for both training and validation and the last 40% was used for testing.

Conclusion

To reiterate, there was a 3.78% difference in the accuracies of the SVR and MLE models. While our research suggested that support vector regression was an algorithm that could accurately predict stimuli from neural data originating in the motor cortex, the classifications of those datasets may have been more distinctive. Support vector regression splits the classifications of data based on different groupings of datapoints in specific classifications with similar characteristics, similar to KMeans Clustering. If characteristics are shared in points in different classifications, support vector regression will not accurately classify datapoints. And when it comes to neural activity, some neurons will have little to no change in activity due to the specific stimuli studied, simply because their functions are not related to said stimuli. It is difficult to tell the function of individual neurons without collecting data, therefore data from unnecessary neurons may have been collected and could have swayed the model's predictions due to their changes in activity unrelated to the manipulated variable studied. SVR models also have no way to clarify their predictions by factoring in probability, because they differentiate points simply by defining them as above or below a defined hyperplane (Pedamkar).

As seen in Figure 1, the model made incorrect predictions when analyzing the upwards direction, labeled 0 on the matrix. The SVR model only predicted half of the upwards directions correctly, incorrectly classifying the other half of the data as the third direction. The model also inaccuractly predicted some of the data points where the correct classification was the 6th direction. Instead, the model concluded that the data belonged in the 5th directional classification. These inconsistencies could be due issues in data for the 0th and 6th directions. Issues in the data could be due to a number of reasons, including not recording enough data from neurons tuned towards the 0th and 6th directions, which would mean that there was less overall data for the 0th and 6th directions. With less data, the SVR model would have less to analyze and predict those directions less accurately.

While the SVR model did not predict the directional stimuli with a higher or equal accuracy to the maximum likelihood estimation model, its accuracy is still relatively good. An r^2 of 0.8247 is a high number and shows that SVR is reliable in predicting the directional stimuli, even though it might not be the most accurate. The dataset used was also only contained 22,594 datapoints and the template for this SVR model used a dataset of over 500,000 datapoints. Since our model's accuracy grew as we used more data for training, there is evidence to suggest a larger dataset would have yielded in more accurate predictions. There could have been problems with the template used itself. While research was conducted to find a suitable working model of SVR, it is possible that certain parts of the template code were incompatible with the dataset and types of classification this model needed to use. However, as stated above, a 4% difference in accuracy between support vector regression and maximum likelihood estimation is minor, and this study did demonstrate support vector regression's efficacy in the neural decoding of data originating in the motor cortex.

Limitations

This study has potential limitations. First and foremost, there are the limitations discussed in the Conclusion. These include limitations with the small data size, the model template used, and a lack of data from neurons tuned in specific directions. Beyond these limitations mentioned, there was not unlimited time to work on the model itself and it is possible there are more changes that could be made to improve accuracy. Since we built off an existing model, there is a chance that a support vector algorithm created from scratch, specifically for this data, might yield a higher accuracy as well.

Acknowledgements

We thank the department of neuroscience at the University of Chicago for the dataset of neural data from a macaque monkey. We thank Mr. Sreeram Nivarthi for providing his services when producing the support vector regression model used in this study.

References

Benjamin, A., & Glaser, J. (n.d.). *GitHub—KordingLab/Neural_Decoding: A python package that includes many methods for decoding neural activity*. Retrieved July 9, 2022, from https://github.com/KordingLab/Neural_Decoding

Glaser, J. I., Benjamin, A. S., Chowdhury, R. H., Perich, M. G., Miller, L. E., & Kording, K. P. (2020). Machine Learning for Neural Decoding. *ENeuro*, *7*(4). https://doi.org/10.1523/ENEURO.0506-19.2020

Goel, A. (2020, December 20). Support vector machine in Machine Learning. *GeeksforGeeks*. https://www.geeksforgeeks.org/support-vector-machine-in-machine-learning/

I know python (Director). (2020, April 25). *Machine Learning With Python Video 17: Support Vector Regression (SVR)*. https://www.youtube.com/watch?v=-EjQWqHMsog

Pedamkar, P. (2020, February 6). Support Vector Regression | Learn the Working and Advantages of SVR. *EDUCBA*. https://www.educba.com/support-vector-regression/

Support Vector Regression—The Click Reader. (n.d.). Retrieved June 13, 2022, from https://www.theclickreader.com/support-vector-regression/
https://doi.org/10.47611/jsr.v8i2.775