Exploring the predictive capabilities of convolutional neural networks on neural responses recorded in mice brains

Marcus Berget

May 2023

ABSTRACT: This research paper explores the predictive capabilities of Convolutional Neural Networks (CNNs) on neural responses recorded in mice brains. The study utilizes the Allen Brain Observatory dataset, which provides two-photon calcium imaging data from visual areas in the mouse's primary visual cortex. The dataset contains recorded neural activity in response to 119 grayscale images presented to the mice. The research aims to train a CNN, specifically the VGG19 architecture, on the neural recordings to predict the patterns of neural activity associated with visual stimuli. The analysis focuses on evaluating the accuracy of the network's predictions and its potential for understanding neural representations and brain activity. The results indicate challenges in training the network, suggesting potential limitations in the data quality, complexity of neural responses, preprocessing techniques, or network architecture. Further investigations are recommended to address these limitations and enhance the network's learning capabilities

1 Introduction

In recent years, advancements in neuroscience and artificial intelligence (AI) have converged, opening up new possibilities for utilizing computational models to gain a better understanding of the dynamics of biological neural systems.

Inspired by biological visual sensory processing [2], CNNs have proven to be highly effective in image recognition tasks. By emulating the hierarchical organization of the visual cortex, CNNs extract and learn meaningful features from input stimuli, enabling accurate classification and prediction. The ability of CNNs to mimic visual processing in biological systems has led researchers to question whether these computational models can extend beyond image recognition and provide insights into the function of biological systems.

The use of CNNs to predict how mice brains respond shows great potential for future research. By training a CNN on datasets containing recorded neural activity, we can evaluate its effectiveness in predicting neural responses. If this approach proves successful, it could provide a valuable tool for studying neural representations and deciphering brain activity. Ultimately, this can enhance our understanding of how biological systems process and interpret sensory information.

This research paper aims to investigate the predictive capabilities of CNNs on neural responses recorded in mice brains. We will utilize the pretrained CNN VGG19, along with a fully connected network, to train on a comprehensive dataset of neural recordings. Our analysis will focus on examining the network's accuracy in predicting neural activity patterns associated with visual stimuli. If our approach proves successful, it could pave the way for generating optimal visual stimuli that elicit the most exciting responses from neurons.

2 Theory

Allen Brain Observatory

The project relies on data obtained from the Allen Brain Observatory, which offers a vast collection of searchable data from numerous two-photon calcium imaging sessions conducted across various visual areas and depths within the visual cortex [1]. Two-photon calcium imaging is a technique widely employed in neuroscience to visualize and monitor the activity of individual neurons in living tissue. It is a type of fluorescence microscopy that enables researchers to observe changes in calcium ion concentrations within neurons, serving as an indirect measure of neuronal activity (see fig 2).

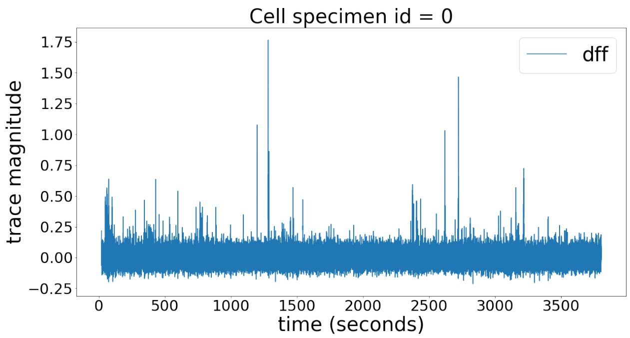

Figure 1: Sample of the fluorescence trace of a single neuron across the experiment.

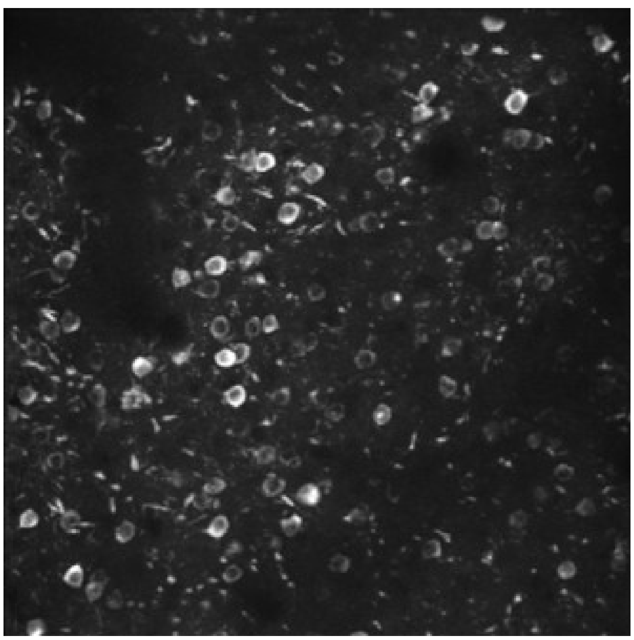

Figure 2: Sample image from a calcium imaging experiment [1]

In the experiment used for the project, the mouse was exposed to natural images such as the one in fig 3 during the experiment while calcium imaging was done of the primary visual cortex. The fluorescence traces of 212 distinct were recorded during the exposure to the images.

Figure 3: One of the 119 images used in the experiment

Deconvolution of calcium imaging data

As seen in Fig 1, there is a lot of noise in the calcium imaging data. This noise, stemming from various sources such as detector imperfections, movement artifacts, or background fluorescence, can obscure the underlying neural activity and make it challenging to extract accurate information from the raw fluorescence traces. To address this issue, a computational technique called deconvolution is often employed in the analysis of calcium imaging data. Deconvolution algorithms aim to unravel the true neural activity by separating the effects of the indicator's response dynamics, noise, and other artifacts from the observed fluorescence signals.

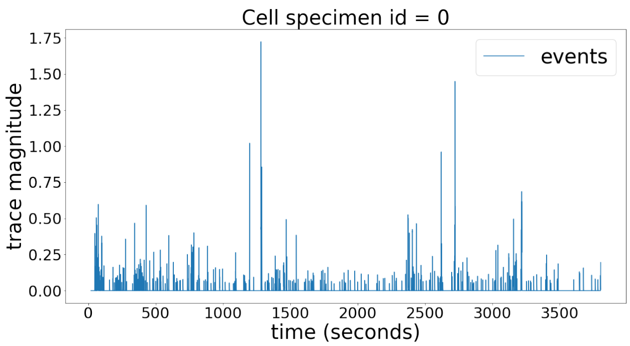

Figure 4: Sample of the deconvolved traces of a single neuron across the experiment.

Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a type of deep learning model designed specifically for processing images. CNNs are comprised of convolutional layers which acts like filters for extracting meaningful features from the images sent through as input. The convolutional operations allow the network to capture spatial dependencies and detect patterns in the images. The network learns the key features by using the famous backpropogation algorithm. In addition to convolutional layers, CNNs often include other layers such as pooling layers. Pooling layers make the feature maps smaller, which helps to simplify calculations and focus on the most important details. After the convolutional layers it's common to include fully connected layers which integrate the extracted features and make the final predictions.

VGG19

Transfer learning is a technique in machine learning where a pre-trained model, such as VGG-19, is utilized as a starting point for solving a new task. VGG-19, a CNN, has been trained on the ImageNet dataset for large-scale image classification. It processes images as inputs in order to classify the images from a pool of 1000 image classes.

The architecture of VGG-19 is first comprised of a set of convolutional layers, followed by a set of fully connected layers. The input images undergo spatial convolution using a set convolutional layers, each followed by a rectifying nonlinearity (ReLU). Additionally, the network includes five pooling layers, responsible for downsampling the feature maps by a factor of two. This downsampling is achieved by selecting the local maximum value from four neighboring pixels. The convolutional layers are grouped into five sets, named conv1 to conv5, consisting these groups are 64, 128, 256, 512, and 512, respectively [3].

3 Methods

Preprocessing

In the experiment conducted to gather the data, a mouse was exposed to 119 grayscale images, presented 50 times each in a random order. Calcium imaging was employed to record the baseline-corrected fluorescence traces (dF/F) of 212 distinct neurons located in the mouse's primary visual cortex. The raw data utilized consists of dF/F traces recorded over 114,111 time steps for each neuron. Each image exposure lasted for 250 milliseconds, which approximately corresponds to 7 time steps within the data.

dF/F traces

To make the data suitable for analysis, the relevant traces corresponding to the times when the images were shown were extracted and stored in a new array. This array had a shape of (212 neurons, 5950 total image presentations, 7 time steps per image). Then, the mean of the traces was calculated for each of the 7 time steps, resulting in a final array with a shape of (212 neurons, 5950 image presentations).

Events

Similarly, the procedure employed for analyzing the dF/F traces was also applied to the events data. However, instead of calculating the mean of the 7 timesteps during which the images were presented, the maximum value among those 7 timesteps was selected. This decision was made to simplify the analysis process, as using the maximum values provided a straightforward representation of the events data.

Image set

Initially, the natural image dataset undergoes preprocessing, resizing all images to a fixed size of 48x48 pixels. Although attempts were made to use higher resolution images, the computational demands exceeded the capacity of the laptop running the network due to limited GPU resources. Additionally, the neural responses were standardized before being used in the network.

Network Architectures

In this project, a network was built using the convolutional layers of the VGG19 architecture. Two different approaches were explored: using all the convolutional layers and using only the first five layers. The decision to focus on the first five layers was based on a study that found this configuration to be effective in predicting how neurons respond to new images that were not part of the training data [3]. After the convolutional layers, a fully connected part was added. The width and depth of this part were adjusted and fine-tuned to find the best setup for achieving optimal results.

Configurations

To evaluate the learning parameters, an initial setup of the fully connected layers was created. The dimensions of the first, second, third, and last layers were (36864, 4096), (4096, 1000), (1000, 250), and (250, 212) respectively. The activation functions ReLU and tanh were tested for all layers except the last one. The network hyperparameters were then sequentially fine-tuned in the following way:

- **Initialization:** The weights of the fully connected layers were initialized using Kaimin/He, normalized Xavier, and no initialization.
- **Learning rate**: 20 values in the range $\eta \in [1\text{e-}5, 1]$ were used.
- **Momentum**: 10 values in the range $\alpha \in [0, 1]$ were used.
- **L2 regularization** 10 values in the range $\lambda \in [1\text{e-}5, 1]$ were used.

Considering all possible combinations of these parameters was computationally impractical. Therefore, the training process proceeded as follows: the network was trained with different learning rates using all the mentioned initializations. However, no significant differences were observed among the learning rates as none resulted in a loss converging to an acceptable value. Here, an acceptable value was defined as one that is lower than the standard deviation of the standardized data, which is 1. Thus, a learning rate of 1e-3 was chosen for further training runs.

Next, different values within the specified ranges were tested for momentum and regularization. Similar to the learning rate, no substantial effects were observed for different choices. As a result, a momentum value of 0.9 and a regularization value of 1e-3 were used for subsequent training runs.

Furthermore, the depth and width of the fully connected layers were adjusted. Various configurations were tested, including wider and deeper setups, as well as narrower and shallower alternatives. This exploration aimed to identify the optimal configuration for the network architecture.

A similar process was undertaken for the events; however, due to the deconvolution of the dF/F traces being achieved only towards the end of the project, fewer configurations were explored to set up the network for this data.

4 Results

It was difficult to tune the parameters because as seen in the sample configurations in figs 5, 6, the network doesn't seem to learn. This is also indicated by fig 7 in which we can see that the model has zero predictive power, by fluctuating around zero. There wasn't any other configurations that improved the learning, and the loss profile was the same for all runs, except for a few experienceing exploding/vanishing gradients.

Figure 5: The training loss of an attempt to train the network on the deconvolved dF/F traces (events). The hyperparameters used for this configuration were $\eta = 1\text{e-}5$, $\alpha = 0.9$, $\lambda = 0$, batch size = 64, tanh activation functions.

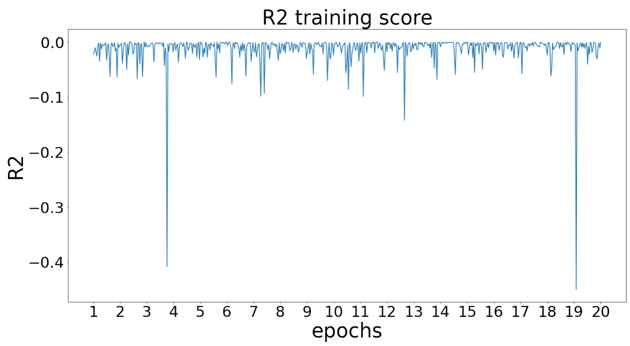

Figure 6: The R2 training score of an attempt to train the network on the deconvolved dF/F traces (events). The hyperparameters used for this configuration were η = 1e-3, $\alpha = 0.9$, $\lambda = 0$, batch size = 16, ReLU activation functions.

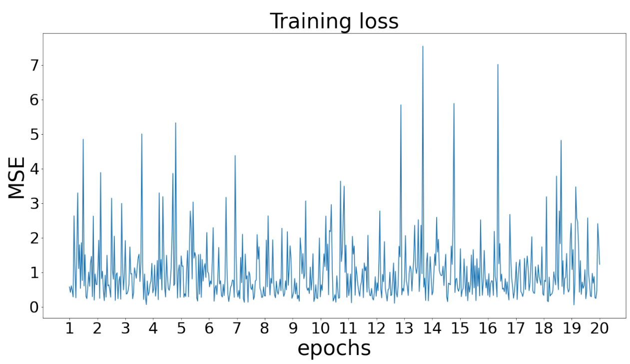

Figure 7: The training loss of an attempt to train the network on the deconvolved dF/F traces (events). The hyperparameters used for this configuration were η = 1e-3, $\alpha = 0.9$, $\lambda = 0$, batch size = 16, ReLU activation functions.

5 Discussion

The observed results raise an intriguing question regarding the learning capacity of the network utilized in this study. Exploring the reasons behind the network's inability to learn is crucial for future investigations and potential improvements.

One possibility is that the quality of the data may hinder the training of any machine learning algorithm to accurately predict the responses. Assessing the data quality can be difficult and time consuming, but is worth pursuing if given the time. However, it's worth noting that given the extensive utilization of the Allen data as a comprehensive and well-organized source for 2-photon calcium imaging, commonly employed in in silico studies, it is unlikely that poor data quality is the sole reason for the network's lack of learning.

Alternatively, it is plausible that the neurons recorded during the experiment may not respond directly to the presented images and could have other functionalities within the primary visual cortex. Additionally, it is worth considering the mouse's attention level during the experiment, as it could influence the neural responses. There are other experiments containing additonal data such as pupil size, running speed, etc. that could influence the neural responses. This is an exiting avenue to explore and could potentially lead to great learning improvements of the network.

Another factor to consider is the potential loss of information during the pre-processing of the dF/F traces. By only utilizing the timesteps corresponding to image presentation, the analysis may fail to capture the influence of prior neural activity on the current timesteps. This limitation could hinder the network's ability to learn complex temporal dependencies in the data. The potential temporal dependies also begs the question whether a recurrent neural network could better predict the neural responses.

To assess the feasibility of the network's learning capabilities, one approach is to substitute the fully connected layer with a linear regression layer. If this simplified model converges to a reasonable score, it would suggest that any challenges lie within the network architecture rather than the inherent limitations of the data itself. Exploring alternative model architectures and optimization techniques could be such as this could be useful for future investigations.

Another a potential limitation in the approach used in this project, involves reducing the size of the images. Downsizing the images may result in a loss of crucial information required for learning meaningful features. However, a previous study following a similar procedure successfully trained a network using images of dimensions 64×36 pixels [4]. This suggests that downsizing the images might still be feasible for achieving the desired outcomes.

6 Conclusion

The attempt to train the network described in this study did not yield successful results. Despite trying various configurations, the network failed to learn and accurately predict the neural responses recorded in the mouse's primary visual cortex. However, there are several promising avenues to explore in order to potentially achieve successful training of a CNN for this task.

One possible approach is to test the network with different experiment data. By using other datasets, it may be possible to explore other potential dependencies, such as pupil size and location, and running speed. Another avenue worth considering is the implementation of a RNN. Furthermore, training the network on higher-resolution images, rather than downsized ones, might provide additional information and details that could aid in predicting neural responses more accurately.

References

[1] `https://observatory.brain-map.org/visualcoding/`

[2] Grace W. Lindsay, 2019 Convolutional Neural Networks as a Model of the Visual System: Past, Present, and Future

[3] Santiago A. Cadena et. al., 2019 Deep convolutional models improve predictions of macaque V1 responses to natural images

[4] Edgar Y. Walker et. al., 2019 Inception loops discover what excites neurons most using deep predictive models