Modelling Neuron Encoding of Visual Stimuli in the Anteromedial Visual Area

Diego Cerretti Beatrice Citterio Giovanni De Muri
Mattia Martino Sandro Mikautadze

Bocconi University - BAI - Academic Year 2023-2024

1 Introduction

This work aims to uncover the fundamental principles underlying how neurons encode and process sensory inputs. In particular, our goal is to **develop mathematical models that best predict individual neurons' spike counts in response to visual stimuli in the visual cortex**. This task is crucial for identifying the key variables influencing the response to specific visual features and for unraveling the complexities of neuronal responses in a concise framework. We will use the Allen dataset, which includes measurements from a mouse brain, and focus on neurons from the **anteromedial visual area** (VISam). VISam is a small area of the brain, so it can result in a high chance of finding specialized neurons. Previous studies in the field (Marshel et. al, 2012) showed that VISam is highly direction-selective and has a high response to low spatial frequency inputs. Moreover, VISam is the most represented region in the dataset, with 135 measured units. We will study **static gratings** inputs, which are visual stimuli characterized by three features: orientation, spatial frequency, and phase.

So, in mathematical terms, **given orientation (x_1), spatial frequency (x_2), and phase (x_3), we want to find a function $f(x_1, x_2, x_3)$ to predict the spike count for each neuron**. Notice that here we do not aim at discovering the function f from data, but rather at finding the best fit to the data, from a set of candidate models we will define. The neurons we will use here will be a subset of VISam, selected according to the highest variance and the highest range in the spikes, with a check for modulation, and we will infer the function f with a **multi-layer perceptron** and **linear regression models** (ref. Section 2).

2 Methods

2.1 Preliminary Analysis

First, we conduct a preliminary data exploration to investigate the spiking activity of neurons across VISam. This can be useful since it can allow us to find general behaviors of the whole area. We study the variability of the spiking activity, fixing a numerical variable at a time. By plotting all possible combinations of phase and orientation, we see that the mean neural response as a function of the frequency seems to have high values for small frequencies and then tends to decrease with the increase in frequency (left Fig. 1). This suggests that most neurons in the area respond differently to spatial frequency, and mostly to lower values. Furthermore, by fixing different values of frequency and phase, the average spike per unit as a function of the orientation seems to decrease for median values of orientation (50 to 100 degrees) and tends to increase for flat angles, i.e. 0 and 150 degrees (center Fig. 1). On the other hand, phase seems to have no impact in shaping the neural response in the selected area (right Fig. 1). These aspects will be later investigated by doing the regressions (ref Section 3).

2.2 Neuron Selection

As said in Section 1, we have 135 neurons in VISam. Running regressions on those neurons is not effective, since not all of them will show significant responses to the the stimulus. Thus to find responsive neurons, we defined some methods to choose the most representative units to focus on for the regression models. The intuition is to select neurons that have the highest variance and those that have the highest range in spiking activity, as this will give us the most representative neurons in the area. In particular, we chose the neurons with either variance or range above the 80th percentile. To achieve this, for each neuron, we computed the standard deviation of its spike counts and then selected those above the 80th percentile. The same approach is used to compute the range and then select the neurons of interest.

These selection methods already shrink the set of neurons to a dozen highly responsive ones, but to be sure the spikes are due to the inputs and not to the inherent stochasticity of the response, we tried to account for modularity. In particular, we say that the response of a unit is modulated by a variable if we can find a statistically significant difference in the

spike count as the variable changes. Our approach to identify modulated neurons was the following: for each unit, we took the best horizontal line fitting the data (that is given by the mean of the spike count) and we computed the average MSE. We expect that if the error is small, then the data is well-fitted by a horizontal line, thus there is no modularity. Instead, if the error is large, then we can expect to observe some sort of modularity (Fig. 2). Out of the units we had chosen before, we applied the check for modularity and picked the 5 units with the highest error.

2.3 Regression Methods

Once we select the neurons, we perform the regression on each of these, using the static gratings variables as inputs. First, we train for 50 epochs an MLP with 7 hidden layers of size 64-128-256-512-1024-256-64, RMSE loss, Adam optimization with weight decay 0.025, and leaky ReLU activations with slope 0.2. The second approach involves using linear regression, using either linear, quadratic, sinusoidal, or combined (i.e. linear, quadratic, and sinusoidal) input features. We will minimize the MSE. To do model selection, we apply the step-down method, with a significance level of 0.1. This allows us to get a model that is simple, resource-efficient, and interpretable. We investigate the performance with the adjusted R^2 and test loss. We expect MLP to perform better than linear regressions, since with MLPs we're letting the network learn the best possible nonlinear function with no constraints. We also expect higher-order features to outperform linear inputs in linear regression. Overall, due to the stochasticity of spike counts, we expect the errors to be high.

3 Results

3.1 Regression - Highest Range Selection

Surprisingly, we see that MLPs perform poorly on all selected neurons. By plotting the graphs of the resulting models we infer that the MLP is not able to fully capture the nonlinearity of the data (Fig. 3). As for the linear models, in general, they perform better than the MLPs. We notice that in almost all the linear models the phase is rarely included, which suggests low statistical relevance on the outcome. Instead, there is a strong dependence on spatial frequency and orientation.

Looking at each linear regression model in detail, we see that simple linear features do not capture the true correlations (Fig. 4). Quadratic models seem to fit the data well (Fig. 5), while sinusoidal and combined terms yield a too-oscillatory behavior (Fig. 6,7). Notice that confidence intervals are very large, indicating the low certainty of the correctness of the model, possibly due to the high variability of the data.

Overall, for every neuron, by increasing the complexity of the model, the test loss tends to decrease while the adjusted R^2 tends to increase (Fig. 8a). The best results, in this view, are obtained by combined feature models. However, the graphical interpretations suggest that they are not the best fit, and we might prefer the quadratic models instead.

3.2 Regression - Highest Variance Selection

As before, we apply the same methods to each neuron. Again, the MLP doesn't work very well, while for linear models the phase seldom appears in the final versions of the models. Moreover, we still observe the oscillatory behavior mentioned before when sinusoidal functions are included in the inputs. The only difference between this analysis and the analysis on highest-range neurons is that here the model with sinusoidal terms seems to perform worse than the one with quadratic features, even in terms of loss and adjusted R^2 (Fig. 8b). For this reason, and also observing the resulting plots of the models, quadratic functions seem to fit the data best, giving a more generic result.

4 Discussion

From the results, we can conclude that **the models that seem to be the most effective have a quadratic dependence on the inputs**. Moreover, phase is not statistically significant, unlike spatial frequency and orientation, which are good predictors of spike counts. So, **we have found evidence that responsive neurons in VISam tend to react with respect to orientation and low-spatial-frequency visual inputs**. This is exactly what we expected and goes in accordance to Marshel et. al, 2012.

Even though we obtained meaningful results, our methods suffer from some limitations. To improve the performance of our models, we could try different ways of selecting neurons and check whether they are modulated or not. Our method, while simple and interpretable, may not be very adequate in some situations. Further improvements of the models include regularizing to penalize large models (e.g. LASSO), using ensemble methods (e.g. XGBoost), adding constraints to the regression (e.g. positivity of spike counts), or adding lower frequency terms to prevent too-fast oscillations in the sinusoidal models, as previously mentioned. Also, a possible improvement could be obtained by changing the loss function. Indeed, here we have only tried to minimize the MSE, but using different choices of loss (e.g L1 loss) would allow to weight data points differently. This could help in dealing with the stochasticity of the neural response. Finally, increasing the number of measurements per neuron and the total number of neurons could definitely improve the accuracy of the models.

A Supporting Images

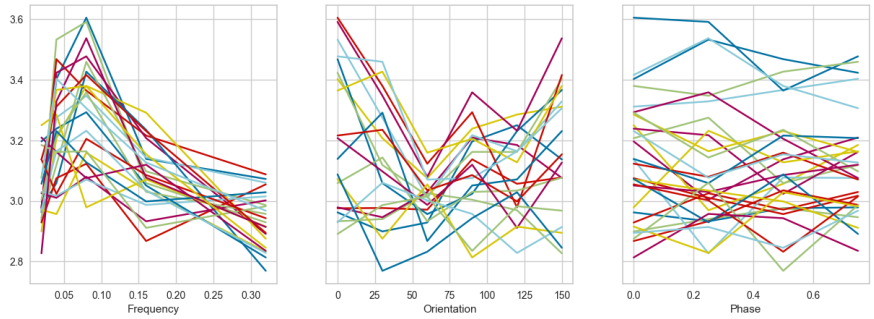


Figure 1: Average spikes per unit, fixing two parameters at a time.

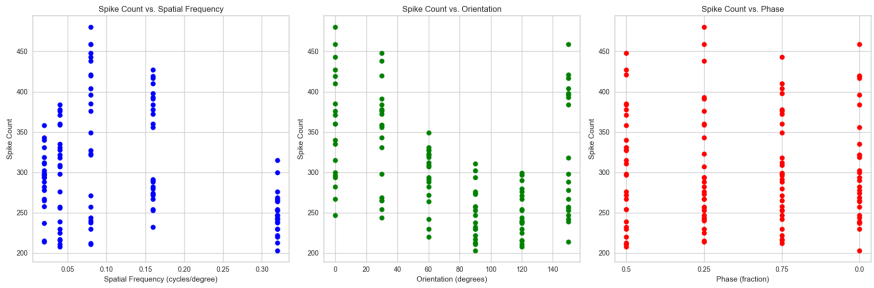


Figure 2: Example of a modulated neuron (951093283) with our criterion.

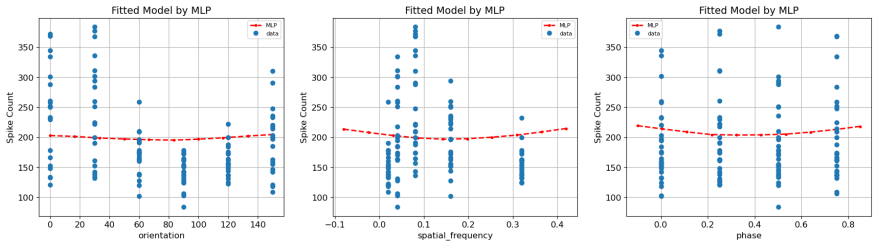


Figure 3: MLP output (neuron 951092437, selected by highest range). Train/test loss: 74.54/63.14.

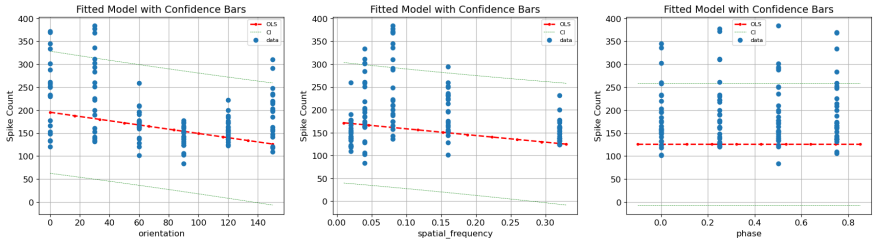


Figure 4: First-order regression output (same neuron as above). Equation is $y = 189.41 - 23.61orient - 15.94freq$. Adjusted R^2: 0.152, train/test loss: 69.974/58.346

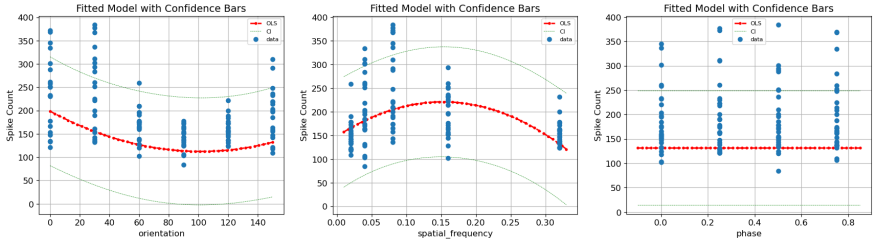


Figure 5: Quadratic regression output (same neuron as above). Equation is $y = 189.07 - 86.86orient + 104.79freq + 66.83orient^2 - 122.02freq^2$. Adjusted R^2: 0.363, train/test loss: 69.974/48.821

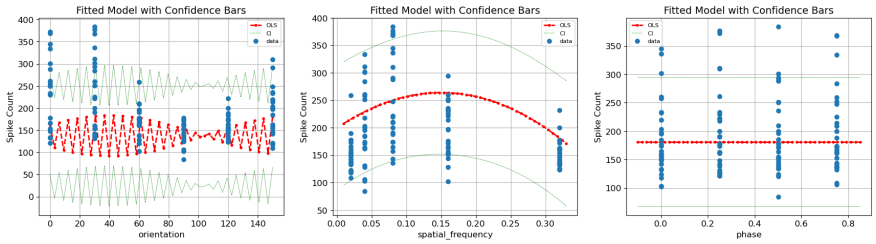


Figure 6: Sinusoidal regression output (same neuron as above). Equation is $y = 192.30 - 27.26\sin(orient) + 91.98\sin(freq) + 14.49\cos(orient) + 108.72\cos(freq)$. Adjusted R^2: 0.387, train/test loss: 69.974/54.102

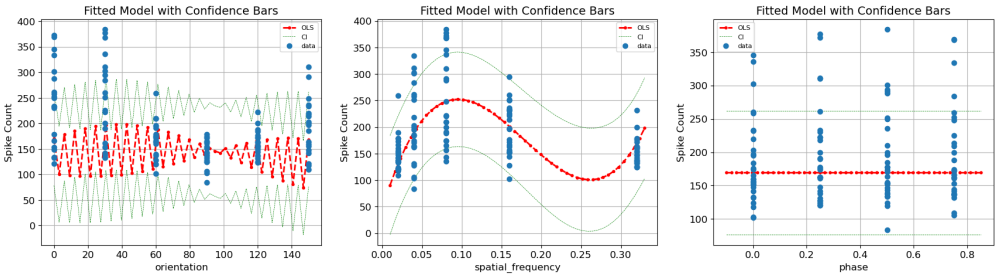
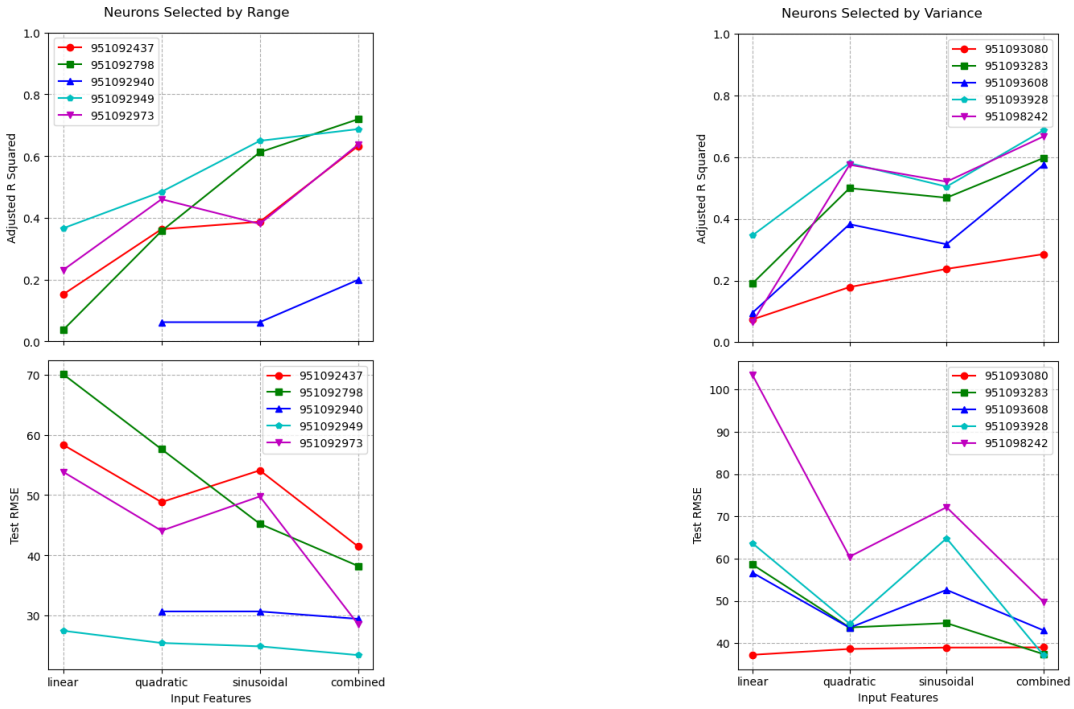


Figure 7: Combined regression output (same neuron as above). Equation is $y = 189.57 + 44407.51freq - 30.94orient^2 + 17.31(orient * freq) - 1377.48freq^2 - 26.22\sin(orient) - 43090.47\sin(freq) + 21.86\cos(orient)$. Adjusted R^2: 0.632, train/test loss: 69.974/41.440



(a) Neurons selected by range

(b) Neurons selected by variance

Figure 8: Adjusted R^2 and RMSE test loss as a function of the input complexity of the linear regressions.