Neuroscientific Prediction Model of Mouse Brain Activity Patterns

Jinxi Zhang *

Department of Statistics, University of California, Davis, United States

* Corresponding Author Email: jokzhang@ucdavis.edu

Abstract. This study presents a neuroscientific predictive model, developed using a subset of data collected by Steinmetz et al. (2019), which records neural activity in mice during a visual discrimination task. Leveraging advanced data analysis techniques, including logistic regression, Principal Component Analysis (PCA), and k-mean clustering, the model aims to predict trial outcomes- success or failure - based on the patterns of neural activity and the visual stimuli presented to the mice. This predictive model is designed to address the challenges of interpreting complex neural datasets. It emphasizes the relationship between neural responses and behavioral outcomes. The logistic regression framework was chosen for its proficiency in binary classification, while PCA and k-means clustering facilitated dimensionality reduction and data segmentation, respectively. These methodologies enhance the manageability of this neural data. Upon analysis, the model displayed moderate predictive accuracy, indicating areas for further improvement such as random forests or support vector machines, which might offer improved performance in handling the dataset's intricacies.

Keywords: Neuroscientific prediction, mouse brain, activity patterns.

1. Introduction

The workings of the brain have long fascinated scientists and laypersons alike. In the quest to unravel these mysteries, neuroscientific research has continually evolved, adapting methodologies to decode complex neural networks and cognitive processes. Central to this exploration is the use of animal models, particularly mice, due to their genetic and physiological similarities to humans and the relative ease of genetic manipulation. For decades, mice have been particularly instrumental in neuroscience, offering a window into the mechanisms of the brain. Their neuroanatomy shares fundamental structures with the human brain, making them an ideal model for studying neurological diseases, brain development, and neural functioning. Recent advances in neural recording technologies, such as the Neuropixels probes used by Steinmetz et al, have enabled the capture of vast amounts of data from the mouse brain [1]. These technologies provide insights into how neuronal activity underpins behavior and decision-making processes.

This project builds upon the foundational work of Steinmetz et al, which conducted comprehensive neural recordings in mice engaged in a visual discrimination task [1]. The study captured data from thousands of neurons across multiple brain regions. Our objective is to develop a neuroscientific predictive model that aims to predict the outcome of each trial in these mouse experiments, categorized as either success or failure, based on the intricate patterns of neural activity and the specific visual stimuli presented. Besides building a predictive model, the project interprets and sheds light on the relationship between neural activity and behavioral responses in mice.

2. Variables & Methods

2.1. Model Overview

In this study, we employed a logistic regression model that focused on classifying trial outcomes (feedback types) based on visual stimuli randomly presented to the four mice. The dataset used for this project was sourced from a subset of data collected by Steinmetz et al [1]. As Maalouf in [2] states, "Logistic regression (LR) continues to be one of the most widely used methods in data mining

in general and binary data classification in particular" (p.281). Furthermore, logistic regression is well-suited for delivering probability estimates and can be effectively adapted for multi-class classification scenarios. It is compatible with a wide range of unconstrained optimization methods, enhancing its applicability in diverse analytical contexts [2]. Therefore, we chose logistic regression as the primary modeling technique, as our project involves binary classification tasks that predict the binary outcomes of trial feedback (success or failure).

2.2. Variable Selection

In this sub-section, we detail the descriptions of the key variables used in our analysis. The dataset comprises six variables: 'feedback_type', 'contrast_left', 'contrast_right', 'time', 'neuron_spikes', and 'brain_area'. 'feedback_type' is a binary variable representing the outcome of each trial, coded as '1' for success and '-1' for failure. This variable is essential as the primary dependent in our predictive model, providing a crucial understanding of the relationship between neural activity and the behavioral responses of the mice. 'contrast_left' indicates the contrast level of the visual stimulus presented on the left screen, while 'contrast-right' measures the contrast level on the right screen. Including both left and right contrast levels enables the model to assess the impact of asymmetrical visual stimulation on neural activity and trial outcomes. The 'time' variable represents the centers of the time bins for spike data, segmenting neural activity data into distinct time intervals. However, this variable will not be used in our analysis. Incorporating time would significantly increase the dataset's dimensionality, as each time point adds dimension. Furthermore, it demands methodologies capable of handling time series data, which involve more complex algorithms such as time series analysis, recurrent neural networks, or dynamic time warping. 'neuron_spikes' also denoted as 'spks' indicates the number of spikes of neurons in the visual cortex within defined time bins. This variable is critical as it provides a direct measure of the firing patterns of neurons, fundamental to understanding neural responses to stimuli. Analyzing these spike trains allows us to decode neural firing patterns associated with different stimuli and feedback types, offering insights into the neural basis of sensory processing and decision-making. Lastly, 'brain_area' identifies the specific brain area where each neuron is located. Including this variable enables for a more comprehensive analysis of neural data by correlating neural activity with specific brain regions.

2.3. Data Preprocessing

In this project, several data preprocessing methods were employed to optimize the performance of the predictive model. These methods were specifically chosen to address the challenges inherent in this dataset, such as high dimensionality and variability across sessions.

The first preprocessing step involved transforming skewed predictors and then centering and scaling them before performing Principal Component Analysis (PCA). This approach ensures that PCA focuses on the underlying relationships in the data rather than being influenced by distributional differences and the original measurement scales [3]. As such, continuous variables like 'contrast_left', 'contrast_right', 'spks_mean', and 'spks_standard_deviation' were standardized. The standardization process involved subtracting the mean and dividing by the standard deviation for each variable. This transformation allows for comparison and analysis of variables with different scales (e.g. one variable in the range of 0 to 1 and another in the range of 100 to 1000), which is crucial for techniques sensitive variable scaling, like PCA and k-means clustering used in this project.

Principal Component Analysis was the second method implemented. PCA serves as a primary tool for reducing dimensionality, crucial for handling large datasets by reducing them to fewer dimensions without significant information loss [4]. PCA also has wide application in fields like neuroscience for spike-triggered covariance analysis as stated in [4], which fits perfectly in this project. The spike data alone is high-dimensional since it includes numerous neurons across various time bins in 18 sessions with each session containing several hundreds of trials, and each trial consists of multiple variables. So, PCA was a necessary preliminary step. It reduced the number of variables to a smaller set of principal components that still capture most of the variance in the data. PCA improved model

performance, mitigated noise and overfitting, and enhanced interpretability by focusing on key structures and patterns in the data. In addition, PCA allowed the project to be more interpretable by focusing on the main structures and patterns, particularly helping in identifying shared patterns that are more robust to session-to-session variations as stated in [4].

The third preprocessing method data was K-means Clustering. This well-known clustering algorithm partitions a collection of objects into clusters based on similarity, as determined by a specific clustering algorithm. [5]. K-means clustering was chosen for its suitability for datasets with numerical variables ('contrast_left', 'contrast_right', 'spks_mean', etc.) and its ability to minimize variance within clusters, aligning with the quantitative nature of our data. It was applied after standardizing the dataset and conducting PCA. "The good choice of k is an essential and important feature for building meaningful homogenous and separate clusters when applying the k-means clustering" [5]. Determining an optimal number of clusters, a crucial aspect of k-means was achieved using the Elbow method. This involved plotting the within-cluster sum of squares against various numbers of clusters to identify the 'elbow' point. Each cluster's centroid can be understood as the mean of the points in that cluster, providing a straightforward interpretation of what each cluster represents in the context of neural activities and stimuli responses.

2.4. Model Specifications

In this section, the model specifications will be addressed and justified. For K-means clustering, the selected optimal number of clusters is k=6. This is based on the Elbow method, in which, the optimal number was chosen at the point where the reduction in variance within clusters began to diminish, indicating a balance between cluster compactness and the number of clusters. For Principal Component Analysis, the number of Components obtained is 5. We retained the first few principal components that cumulatively explained approximately 95% of the variance in the data. This choice was made to ensure that we captured most of the data's variability. For the logistic regression, the model classification threshold was set at 0.6 for predicting the binary outcome. 0.6 was chosen after examining the distribution of predicted probabilities. A threshold of 0.6 provided the best balance between sensitivity (true positive rate) and specificity (true negative rate) in our preliminary tests, thus optimizing the model's performance in correctly classifying trial outcomes.

2.5. Computational Challenges

In the process of conducting method is this analysis, several computational challenges were encountered. The first obstacle encountered was the complexity of the data structure. As mentioned above, the dataset involves multiple sessions and countless trials, so selecting relevant variables to construct a sub-dataset is necessary to reduce the noise and complexity of the model. In a general sense, the model performs better with a carefully curated set of features.

2.6. Scaling Data

In this section, results will be shown and interpreted accordingly. As shown in table 1, "contrast_left" (Mean = 0.34186184) This value indicates that, on average, the contrast level of the left stimulus across all trials and sessions was approximately 0.34 before standardization, suggesting that a moderate level of the left visual stimulus was presented to the mice on average. Similarly, "contrast_right" has an average value for the right stimulus contrast suggesting a slightly lower but still moderate level of visual stimulation on the right hemisphere, which, was approximately 0.32. For "spks_mean", it indicates that, on average, the spike count across all neurons and trials is 0.034. "spks_standard_deviation" shows that the average standard deviation of spike counts prior scaling is 0.19. "total_neuron_count", which is approximately 910, is the average count of neurons recorded per session. This number provides an idea of the scale of neural data being analyzed, also providing a sense of whether the sample size is large enough.

Table 1. Centering: Selected Variables means: Mean of each variable before scaling

scaled: center		
contrast_left	contrast_right	spks_mean
0.34186184	0.32414879	0.03426265
spks_standard_deviation		total_neuron_count
0.19093487		909.7970872

Alike the table above, the one below is the scaled scale (standard deviation of each variable before scaling) The interpretation to each of these variables will be similar to the previous table. Hence, interpretation will not be given, but only the results. The standard deviations for each variable ("contrast_left", "contrast_right", "spks_mean", "spks_standard deviation", "total_neuron_count") are 0.39405298, 0.38526040, 0.01232839, 0.03493018, and 294.34913111, respectively. These variables are important because they give an idea of how much variation there is in each variable. For instance, the high standard deviation in "total_neuron_count" suggests substantial variability in the number of neurons recorded across different sessions, as shown in table 2.

Table 2. Scaling: Std Dev: Standard deviation of each variable before scaling

scaled: scale		
contrast_left	contrast_right	spks_mean
0.39405298	0.3852604	0.01232839
spks_standard_deviation		total_neuron_count
0.03493018		294.3491311

2.7. Employing Principal Component Analysis

As mentioned, PCA is particularly useful in the context of this project because it transforms the large set of variables into a smaller one while still containing most of the useful information in the large set.

PC1: "spks_mean" (0.65093175) and "spks_standard_deviation" (0.65711551) have high positive loadings. This indicates that PC1 is strongly influenced by the overall level and variability of neural activity (spike counts). "total_neuron_count" (-0.32309643) has a moderate negative loading, suggesting that as the neuron count increases, the values on PC1 tend to decrease.

PC2: "contrast_left" (-0.855256344) shows a strong negative loading, meaning PC2 captures a lot of the variance associated with the left contrast. "contrast_right" (0.510572686) has a significant positive loading, indicating that higher right contrast values contribute positively to PC2.

PC3: "contrast_right" (-0.69819450) has a strong negative loading, indicating that this component captures aspects of the right contrast. "total_neuron_count" (-0.58745887) also contributes negatively to this component, suggesting a relationship between the neuron count and right contrast in this dimension.

PC4: "contrast_right" (0.4607780) and "contrast_left" (0.3199984) both contribute positively, but "total_neuron_count" (-0.7414218) has a strong negative influence. This suggests that PC4 represents a complex interplay between contrast levels and neuron count.

PC5: "spks_mean" (0.701762711) and "spks_standard_deviation" (-0.711809225) show strong but opposite loadings. This indicates that PC5 captures the contrast in variability and average of spike counts, emphasizing trials where these two measures diverge.

2.8. Employing K-means Clustering

Figure 1 represents the results of the Elbow Method. It involves plotting the within-cluster sum of squares (WCSS) against the number of clusters (K).

X-axis (Number of Clusters): This axis represents different numbers of potential clusters into which the dataset could be partitioned. It ranges from 2 to 15 clusters.

Y-axis (within-group Sum of Squares): This axis shows the within-cluster sum of squares for each possible number of clusters. WCSS is a measure of the compactness of the clusters and is calculated as the sum of the squared distances between each point and the centroid of its assigned cluster. Lower values indicate that points are closer to their centroids, suggesting better clustering.

The Elbow Point: The "elbow" is the point in the graph where the rate of decrease in WCSS sharply changes. This point represents a diminishing return in the reduction of WCSS with the addition of more clusters. Therefore, in the graph, the elbow appears to occur around 6 clusters.

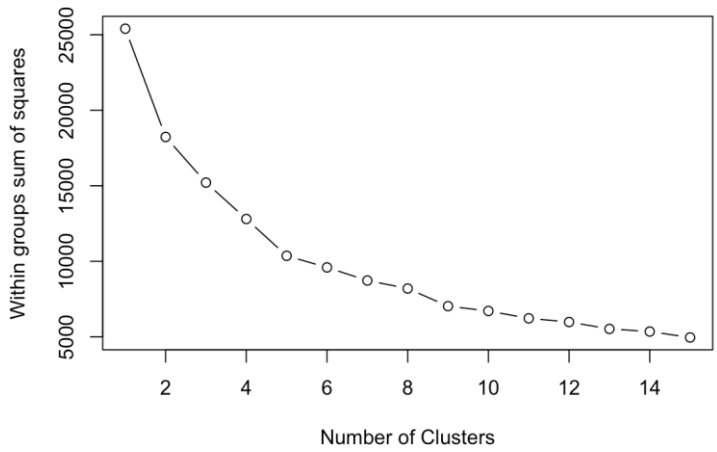

Figure 1. Elbow Method: the optimal number of clusters: 6

There are six distinct clusters (labeled 1 through 6), each represented by a different color. These clusters were determined by k-means clustering applied to the PCA-reduced data, as shown in figure 2.

The spread along the PC1 and PC2 axes suggests that PC1 explains more variance than PC2, as is typical with PCA.

The clusters are visually separable along the PC axes, although there is some overlap, especially between clusters represented by green and blue. This might suggest that while the k-means clustering has found distinct groupings within the data, the separation is not absolute, and there might be shared characteristics between these clusters.

The density of the points within each cluster varies, with some clusters appearing more densely packed than others.

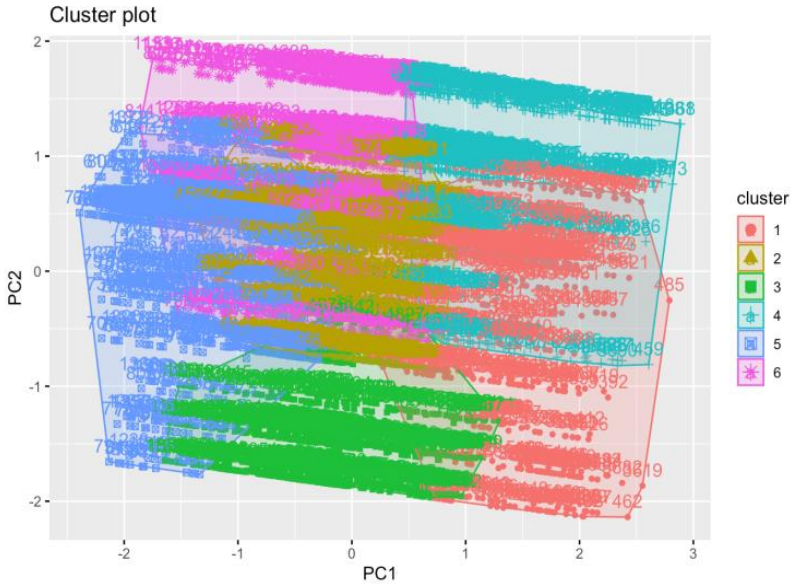

Figure 2. Cluster plot from PCA results

3. Model Training and Prediction Results

In the model training and prediction section, the seed is first set to ensure that the data can be reproduced. Then, the sample. Split () method is used to divide the sub-data frame into a training set and a test set. Moving on to the next step, I chose to use glm () to fit a logistic regression model to the training data. Following this, predict () is used to anticipate the feedback type for the test data based on the model. Finally, a confusion matrix and misclassification error rate are generated to display the performance of the model and the proportion of observations that are incorrectly classified or predicted.

3.1. Model Parameter Estimates and Standard Errors

The model parameter estimates and standard errors are shown in table 3.

For intercept (Estimate: 0.01475682, Std. Error: 0.22263564, P-value: 0.9471565):

The intercept is not significantly different from zero, suggesting that when all other variables are at their mean value, the log odds of the outcome being a success are close to zero. The high p-value indicates that this estimate is not statistically significant as well.

In contrast Right (Estimate: -0.06410709, Std. Error: 0.03899858, P-value: 0.1002937):

A negative coefficient suggests that as "contrast_right" increases, the log odds of a successful outcome decrease, although this relationship is not statistically significant (p-value > 0.05).

For spks mean (Estimate: -9.53399159, Std. Error: 14.79537426, P-value: 0.5193614):

The large negative coefficient coupled with a high standard error and non-significant p-value indicates uncertainty in this variable's influence on the model.

For spks standard deviation (Estimate: 2.42159690, Std. Error: 2.27846248, P-value: 0.2879308):

A positive coefficient, but with a high p-value, suggests an uncertain relationship between spike count variability and the outcome.

Table 3. The model parameter estimates and standard errors

	Estimate	Std. Error	t value	Pr (> | t |)
(Intercept)	0.01475682	0.22263564	0.06628239	0.9471565
contrast_right	-0.06410709	0.03899858	-1.64383138	0.1002937
spks_mean	-9.53399159	14.79537426	-0.64439003	0.5193614
spks_Std. Error	2.4215969	2.27846248	1.06282062	0.2879308
spks_mean: spks_Std. Error	41. 16911521	37.92136747	1.08564427	0.2777052

3.2. Test Sets by splitting 25% of the original data into test sets

About 75% of the data is split into training sets and 25% into test sets, as shown in table 4.

Table 4. Misclassification

Misclassification Error Rate: 0.341		
	Predicted	
Actual	-1	1
-1	21	347
1	86	816

3.3. Test Sets from two distinct new sessions

Given two distinct test sets, the model gives a Misclassification Error Rate of 0.285, indicating that approximately 28.5% of predictions were incorrect, which is moderate in predictive accuracy.

The Confusion Matrix provides a more detailed analysis of the model's performance. The model failed to correctly predict any true negatives (-1), as indicated by 0 in the true negative cell. There is a high number of false negatives (55), suggesting the model is less effective at predicting failures.

Conversely, it performed relatively better in predicting successes, with 143 true positives. Only 2 false positives were observed, as shown in table 5.

Table 5. Test Sets from two distinct new sessions

Misclassification Error Rate: 0.285		
	Predicted	
Actual	-1	1
-1	0	55
1	2	143

4. Conclusion

This study successfully develops a neuroscientific predictive model using data from Steinmetz et al (2019), demonstrating the efficacy of logistic regression, PCA, and k-means clustering in interpreting complex neural data from mice. The model can achieve a misclassification rate of approximately 28.5%, indicating moderate accuracy, which is surprising. Yet, it also suggests that there is room for improvement as 28.5% is considered high in a predictive model. There are plenty of factors that could potentially impact the model such as the complexity of the data, the chosen features, or the inherent variability in neural responses. Our model accurately predicts trial outcomes based on neural activity patterns, offering insights into the neural mechanisms underlying decision-making processes. These findings highlight the potential of integrating computational techniques with neuroscientific research, paving the way for future studies in brain function and its implications in neurological disorders. This work underscores the transformative impact of data science in advancing our understanding of the brain.

References

[1] Steinmetz, N. A., Zatka-Haas, P., Carandini, M., Harris, K. D. (2019). Distributed coding of choice, action and engagement across the mouse brain. Nature, 576 (7786), 266 - 273.

[2] Maalouf, M. (2011). Logistic regression in data analysis: an overview. International Journal of Data Analysis Techniques and Strategies, 3 (3), 281 - 299.

[3] Kuhn, M., Johnson, K. (2013). Applied predictive modeling, New York: Springer, 26 (13).

[4] Hasan, B. M. S., Abdulazeez, A. M. (2021). A review of principal component analysis algorithm for dimensionality reduction. Journal of Soft Computing and Data Mining, 2 (1), 20 - 30.

[5] Mehar, A. M., Matawie, K., Maeder, A. (2013). Determining an optimal value of K in K-means clustering. In 2013 IEEE International Conference on Bioinformatics and Biomedicine.