PREDICTING SYNAPTIC CONNECTIONS IN DROSOPHILA MELANOGASTER

Joshua Vendrow[⋆] *Alexandra Schtein*[⋆] *Zachary Berger*[⋆]

[⋆]University of California, Los Angeles
Department of Computer Science
404 Westwood Plaza, Los Angeles, CA 90095

ABSTRACT

Wiring logic within the Drosophila visual system has been a popular field of study for decades. Through specific wiring patterns, neurons send accurate signals that allow Drosophila to mature into adulthood. A common theory claims that a neuron's gene expression levels are important for determining wiring patterns during adolescence. In this study, we apply five standard machine learning methods to predict the presence of synaptic connections between neurons from their gene expression levels. We then perform feature selection on this data set, and identify a subset of genes that we suggest could contribute to processes causing synaptic connections.

1. INTRODUCTION

Drosophila Melanogaster, the scientific name for a fruit fly, contains many photoreceptors within its visual system that receive and convert light into neuronal information. These neurons undergo targeting techniques to reach the medulla, a stem-like structure in the fly brain. In Figure 1, we display an image of a neuron targeting to the M10 layer of the medulla. Precise wiring of neurons is necessary for these photo-receptors to correctly process the stimulus information they are receiving. The M1 and M5 layers of the medulla hold an array of neurons [1].

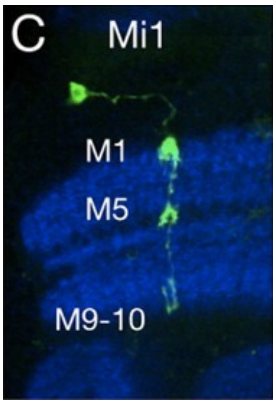

Fig. 1. The Mi1 neuron targeting to the M10 layer of the medulla, passing through the M1 and M5 layers in the Drosophila visual system.

One common hypothesis is that gene expression levels are responsible for specific synaptic connections between neurons [2]. In this study, we aim to use these genes to predict specific synaptic connections. We also apply feature selection methods to identify a subset of genes that lead to the formation of synaptic connections.

Specifically, given a network with weights of synaptic connections between neurons and the gene expression levels of each neuron, we aim to:

1. Predict the presence synaptic connections between neurons given their genetic expression.
2. Identify the genes that contribute most to synaptic connections.

2. DATA

Here we describe the data sets we use in our analysis and describe our pre-processing steps.

2.1. Developmental Transcriptome Data

A transcriptome data set is a collection of neurons with their respective gene expression levels. Our transcriptome data includes the gene expression for 25 unique neuron types that form connections in the M1 and M5 medulla layers. The data set is stored as a matrix, where rows represent neurons and columns represent genes. For each neuron, expression of 4319 genes is shown. The gene expression was sampled at the 48^{th} hour of a Drosophila lifespan.

2.2. Adult Connectome Data

A connectome data set contains the average number of synaptic connections between each presynaptic and postsynaptic neuron. The presynaptic neuron is where the synaptic connection is sent from and the postsynaptic neuron is where the connection is received.

Our connectome data set is sampled from adult Drosophila. It is one weighed network of neuron connections in the Drosophila M1 and M5 layers. Nodes represent neurons and weighted edges represent the average number of synaptic connections between each pair of nodes.

2.3. Pre-Processing

For each pair of neurons we have their underlying genetic expressions and the average number of synaptic connections between them.

We model each data point as a triple (u,v,w), in which u and v represents the pre and post-type neuron respectively and w is the average number of connecting synapses. In running our methods, we model each feature vector as the concatenation of the genetic expression of u and v. See figure 2 for a visualization of the data representation.

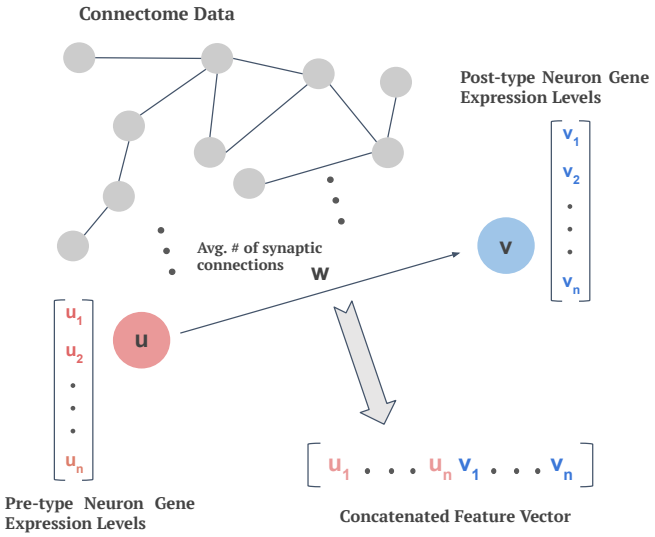

Fig. 2. A visual representation of our data representation procedure used to convert the transcriptome and connectome data into an acceptable data format for our models.

In the regression task, we let w be the label. For the classification tasks, we determined class-type y using a threshold of 0. Namely, if $w = 0$, set y = 0 and $w > 0$, set y = 1.

3. METHODS

Here we describe our experimental setup and the machine learning models that we apply to the data set.

3.1. Experimental Setup

Here we describe our experimental procedure for training and evaluation of our classification models. For a single trial of our training/testing procedure for a given model, we create a random 80/20 training and testing split. First, we select the hyperparemeters by splitting our training set into 5 even folds and performing 5-fold cross-validation for each hyperparemeter. Once the hyperparemeters are chosen, we retrain our model on the full training set and then evaluate on the test set.

For the neural network, after performing cross-validation, we re-split the training data into an 80/20 split for training and validation, train on the train set, and use the validation set to determine at what epoch to terminate. We then use this model as our final model and perform evaluation on the test set.

For each model, we repeat this full process for 10 trials randomizing the train/test split at each step, and average the resulting test accuracies for our final reported test accuracy.

3.2. Models

Here we give a brief description of each model used in our experiments. We use one regression model, which aims to produce a continuous numerical value to approximate the output, and four classification models, which aim to produce a discrete output. In this case the label 0 represents no synaptic connection and 1 represents a non-zero synaptic connection.

3.2.1. Linear Regression

Linear regression aims to compute the optimal affine hyperplane (see, e.g., [3], Section 3.1) and references therein. It does so by minimizing the mean squared error loss

$$L = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2. \tag{1}$$

The R^2 value of a linear regression model is a common metric used to measure the relationship between the features (independent variables) and the labels (dependent variables).

3.2.2. Support Vector Machine

Given a set of data points, a support vector machine finds a separating hyper plane (see, e.g., [3], Section 7) and references therein. It does so by minimizing the number of erroneously classified points, while maximizing the magnitude of the margin separating the classes. Specifically, an SVM solves the optimization task:

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}w^t w + C\sum_{i=1}^{N}\xi_i \\ \text{s.t.} \quad & y_i(w^T x_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \end{aligned} \tag{2}$$

where $\frac{1}{\|w\|}$ is magnitude of the the margin, and w and b represent the separating hyperplane. We perform hyperparemeter selection to experimentally choose C, the misclassification penalty.

3.2.3. Decision Tree

Decision trees are a common machine learning model that form a tree structure to classify a given test data point (see, e.g., [3], Section 1.6) and references therein. At each node, the model splits the input data point along branches based on one of the features. To chose which features to place in this decision tree, the model uses the entropy gain metric, a

measure of in randomness caused by spliting the data along a feature. We perform hyperparemter selection to choose the depth of the decision tree.

3.2.4. K Nearest Neighbors

K Nearest Neighbors (KNN) is a classification model that classifies a test data point by identifying the K closest points in the training set (see, e.g., [3], Section 2.5) and references therein. For these K points the model applies a simple majority vote to choose the label for the test data point. We perform hyper parameter selection to choose K experimentally.

3.2.5. Neural Network

A neural network is a highly influential machine learning model (see, e.g., [3], Section 5.1) and references therein. A neural network passes input through a series of layers parameterized by weight matrices, and updates the weights through a process called back-propagation. Back-propogation is the process of calculating the gradient of each weight matrix in terms of the output of the final layer. In our experiments we use a basic feed-forward neural network in which each layer is modeled by a single matrix multiplication followed by a non-linear activation function. We use 2 hidden layers with 30 nodes per layer and use a RELU activation. During our training procedure, we performed a hyperparameter search over the learning rate.

3.3. Feature Selection Process

In order to determine which genes are most important in the formation of synaptic connections we use feature selection. Generally, feature selection aims to identify the most important features in a data set.

We use a basic feature selection in which each model is run individually on each of the features in the data set. Specifically, for each feature, we extract the column in our data matrix corresponding to that feature then use only that feature to train the model. We then apply the training/testing procedure described in Section 3.1 on this single-column data matrix.

After running this feature selection procedure on each of our models, we generate a ranking $\alpha_m(i)$ for each model m and feature i by sorting the results (test accuracy for classification, R^2 for regression) for each model. For example, $\alpha_m(i) = k$ would suggest feature i is the kth best feature according to model m. Here, $1 \leq m \leq M$ where $M = 4$, representing four of our five models, as we exclude the neural network due to computational constraints. Also, $1 \leq i \leq 8638$, where 8638 is the number of features.

From the $\alpha_m(i)$ values, we calculate a final aggregated rank for each feature by averaging the rankings for each feature from each model. This defines our feature 'importance score'

$$\zeta(i) = \frac{1}{M} \sum_{m=1}^{M} \alpha_m(i)$$

We then get our final ranking for each model by sorting the values of $\zeta(i)$.

4. RESULTS

We first run each model on our full data set to assess the full predictive accuracy of the gene expression data. We then apply feature selection methods as described in Section 3.3 to identify a subset of 'important' features for predicting the formation of synaptic connections.

4.1. General Performance

We first run each of our models on the full data set in order to measure the data set's ability to accurately predict the synaptic connections. In Table 1 we display the results of running each of our models on the full data set (see Section 3.1 for details on the training/testing procedure). For linear regression we list the R^2 value and for the classification models we display the accuracies for the full data set. We also present the sensitivity (accuracy among neuron pairs that truly have a synaptic connection) and the specificity (accuracy among neuron pairs that truly do not have a synaptic connection).

Table 1. Results for running our classification models and linear regression on the data set. We provide accuracies for the full data set, for true examples (having a synapse connection), and false examples (not having a synapse connection).

Model	All	Sensitivity	Specificity
SVM	0.634	0.614	0.654
Decision Tree	0.634	0.6	0.668
KNN	0.632	0.564	0.702
Neural Net	0.673	0.630	0.720
Linear Regression			
R^2	0.337	—	—

4.2. Selecting Top Features

Now, following the procedure described in Section 3.3 we perform a basic feature selection on our data set to identify the genes that contribute most to predicting the presence of synaptic connections. In Table 2 we display the sorted final rankings $\zeta(i)$ for the top ten most important genes in the data set for predicting synaptic connections (See Section 3.3 for the definition of $\zeta(i)$). We note that all 10 of these genes come from gene expression levels for the pre-type neuron (see Section 2.3 for a description of the data representation), suggesting that the gene expression levels of the pre-type neuron are more important than those of the post-type neuron. To investigate this, we also separated the data set into only the gene expression levels of pre-type neurons and post-type neurons

(by extracting only those columns of the data matrix representing gene expression levels of that specific neuron).

In Table 3, we display the results of running our regression and classification models on only the pre-type neuron gene expression levels ('pre-') and only the post-type neuron gene expression levels ('post-'), compared to the accuracy of the full data-set ('all'). We see that, as suggested by the feature selection experiment, the full pre-type neuron expression levels attain a higher classification accuracy than the full post-type neuron gene expression levels by every metric. In fact, for three of the four classification models, the pre-type neuron information outer-performs the full data set, suggesting that the post-type neuron features contains a significant amount of redundant information.

Table 2. Top 10 ten features (genes) in the data set, ranked by the Score metric $\zeta(i)$ that aggregates the rankings of features by each model.

Rank	Gene
1	Stacl
2	Sec61beta
3	HP5
4	CG14757
5	CG9921
6	GluRIA
7	CG4287
8	CG12253
9	ND-75
10	Tob

Table 3. Results for running our classification models and linear regression on full data set, as well as only on the pre-type and post-type neuron information.

Model	All	Pre-	Post-
SVM	0.634	0.653	0.567
Decision Tree	0.634	0.648	0.571
KNN	0.632	0.645	0.580
Neural Net	0.673	0.648	0.572
Linear Regression			
R^2	0.337	0.214	0.094

5. DISCUSSION

Here we discuss the significance of our results and their contributions to efficient experimentation, theories for synaptic specificity, and future research directions.

5.1. Significance of the Data set

Our neural network achieved the highest accuracy of 0.634 on the full data set, and each of our models achieved an accuracy of above 0.6. We also had a significant R^2 value of 0.337. These values are not outstanding, but suggest the data set (which is currently small) has meaningful predictive ability. Once more data is collected we can rerun these experiments and expect our results to improve.

5.2. Experimental Benefits

A common problem is to identify genes that affect neuron function. Currently, researchers perform gene function studies by creating mutations or transgenic animals [4]. These experimental approaches are extremely time consuming and expensive. By identifying a subset of genes that predict synapse connections, we can we limit the number of necessary experiments, and experimentally quantify the importance of the 'top' genes we identified.

5.2.1. Upregulation and Downregulation

Upregulation (Downregulation), commonly known as "Gain of Function/Loss of Function" (GOF/LOF) is a process that increases (decreases) a receptor's sensitivity level, resulting in increase (decrease) in the cell's response to a stimulus. Current research utilizes the popular GOF/LOF method to analyze specific gene functions [5].

Up (down) regulation is commonly done using gene over-expression / knockdown, in which the expression levels of a specific gene are drastically reduced using CRISPR technology [6]. An experiment is then performed to quantify the impact on proteins related to that gene and the level of impact can suggest the importance of the gene. This experiment is repeated for many genes present in the neuron.

Using our machine learning models, we aim to identify the genes that contribute most to the formation of synaptic connections. By formulating a subset of important genes, we could knockdown or over-express only genes from this subset, substantially reducing the number of experiments to perform and improving efficiency.

Additionally, there is significant demand for understanding the formation of cell recognition molecules that contribute to synaptic connections. Although the molecular mechanisms of neural connections have been studied for decades, we only know a small list of genes that encode cell recognition molecules [7]. By identifying a subset of genes that contribute to synaptic connections and performing over-expression / knockdown on this subset, we can help discover which genes lead to the formation of cell recognition molecules.

5.2.2. Theories for Synaptic Specificity

According to the predominant theory of code matching, a specific level of gene expression in two neurons will guarantee

the presence of a synaptic connection between the neurons. Some recent works have suggested that gene expression does not directly correspond to the presence of a synaptic connection between two neurons, but rather increases the likelihood of a connection. While these broad hypotheses cannot be tested using popular gene function studies, computational methods such as the ones in this paper can help to determine which of these theories is more likely based on the data collected.

5.3. Future Work

Our data set, as described in Section 2, is a weighted network with features on each node. In our experiments we choose to represent each data point with a single feature vector and discrete label to match the expected input of the classical machine learning models described in Section 3.2. In recent years, more complex models have been popularized to handle network data, such as the highly influential graph neural network (GNN) model [8]. To meaningfully run these complex models require large-scale data, significantly exceeding the current size of our data set.

Thus, a sub problem would entail collecting more data points by mapping neuron transcriptome data to adult connectome data for different cell types. Currently, only 40-50 neuron cell types are known by their respective morphology and locations within the brain. Further, many cell types are being sequenced for their gene expression levels, and PCA results have clustered many more neurons into classes [9]. Each of these classes represents a cell type, classified by common marker genes. In order to add more data points, we need to learn the location, morphology and synaptic partners of these cell types. One way in which this is done is to map the genetic transcriptome data of a class of cell types to a continuous timeline. In other words, given a random neuron's gene expression levels, identify the time in the fly's lifespan from which those expression levels were taken. Once all genes expression levels are mapped to their respective times, we will be able to learn the cell type's interaction with other cell types at a given time.

6. CONCLUSION

We applied five standard machine learning models to a data set consisting of gene expression levels of neurons to predict the presence of synaptic connections. Using the full data sets we were able to get significant regression and classification results. By applying feature selection methods, we were able to identify a subset of genes that could significantly contribute to the processes related to the formation of synaptic connections. We expect that once more data is collected, the accuracies attained by our methods could improve significantly, and we suggest that our results could be of value for future study of neuron wiring patterns.

7. REFERENCES

[1] Burkhard Poeck, Susanne Fischer, Dorian Gunning, S Lawrence Zipursky, and Iris Salecker, "Glial cells mediate target layer selection of retinal axons in the developing visual system of drosophila," *Neuron*, vol. 29, no. 1, pp. 99–113, 2001.

[2] Shin-ya Takemura, C Shan Xu, Zhiyuan Lu, Patricia K Rivlin, Toufiq Parag, Donald J Olbris, Stephen Plaza, Ting Zhao, William T Katz, Lowell Umayam, et al., "Synaptic circuits and their variations within different columns in the visual system of drosophila," *Proceedings of the National Academy of Sciences*, vol. 112, no. 44, pp. 13711–13716, 2015.

[3] Christopher M. Bishop, "Pattern recognition and machine learning," *Springer: Berlin/Heidelberg, Germany*, 2006.

[4] Dennis G Ballinger and Seymour Benzer, "Targeted gene mutations in drosophila," *Proceedings of the National Academy of Sciences*, vol. 86, no. 23, pp. 9402–9406, 1989.

[5] P Rorth, Kornelia Szabo, Adina Bailey, Todd Laverty, Jay Rehm, Gerald M Rubin, Katrin Weigmann, Marco Milán, Vladimir Benes, Wilhelm Ansorge, et al., "Systematic gain-of-function genetics in drosophila," *Development*, vol. 125, no. 6, pp. 1049–1057, 1998.

[6] Matthias Schlichting, Madelen M Díaz, Jason Xin, and Michael Rosbash, "Neuron-specific knockouts indicate the importance of network communication to drosophila rhythmicity," *Elife*, vol. 8, pp. e48301, 2019.

[7] Thomas C Südhof, "Towards an understanding of synapse formation," *Neuron*, vol. 100, no. 2, pp. 276–293, 2018.

[8] Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun, "Graph neural networks: A review of methods and applications," *arXiv preprint arXiv:1812.08434*, 2018.

[9] Yerbol Z Kurmangaliyev, Juyoun Yoo, Javier Valdes-Aleman, Piero Sanfilippo, and S Lawrence Zipursky, "Transcriptional programs of circuit assembly in the drosophila visual system," *Neuron*, 2020.