Simulating Ocular Dominance with Unsupervised Learning Rules

Cathy Chen
Instructor: Dr. Zahra Aminzare

1. Background

1.1. Biological Systems

The visual system transmits information about stimuli perceived by the eyes through networks of neurons. These networks transfer signals from the photoreceptors that capture light to the neurons in visual cortex that further process these signals [10]. Some neurons, such as those in layer 4 of the human primary visual cortex (V1), display ocular dominance by responding preferentially to inputs from a certain eye [7]. Some animals (such as cats and monkeys) have cortical maps composed of ocular dominance columns, in which alternating columns of neurons in the visual cortex respond preferentially to inputs from the same eye.

1.2. Unsupervised Learning

These cortical maps emerge through self-organization rather than some externally imposed structure. Unsupervised learning allows a network to learn from inputs based on internal rules. In contrast to supervised learning, which requires labels provided by an external "teacher", unsupervised learning does not involve external labels [2]. As such, we use unsupervised learning rules to model the development of ocular dominance maps in the visual cortex.

2. Related Work

2.1. Learning Rules

Researchers have devised a number of unsupervised learning rules to model preferential neural responses, and here we give a brief overview of these rules used in our simulations and their characteristics. This subsection and the next are largely taken from [2].

We model a neuron's firing rate as a weighted combination of its inputs. The weights correspond to synapses between neurons, and the inputs correspond to activations of neurons that synapse onto the modeled neuron. For a given input vector $\mathbf{u}$ and weight vector $\mathbf{w}$, the neuron's firing rate v changes according to

$$\tau_r \frac{dv}{dt} = -v + \mathbf{w} \cdot \mathbf{u} \tag{1}$$

In our simulations, we assume that the firing rate reaches its steady state (we justify this assumption by noting that biological synaptic plasticity occurs much more slowly than firing rate dynamics), so we directly take the firing rate v as

$$v = \mathbf{w} \cdot \mathbf{u} \tag{2}$$

.

To mimic biological systems, we constrain the neural activity v to non-negative values.

In our simulations, we model the learning of the weights $\mathbf{w}$ using the following learning rules:

2.1.1. Basic Hebbian Learning This learning rule follows Hebb's "fire together, wire together" rule, which states that synapses strengthen between neurons that tend to activate together. This rule takes the form

$$\tau_w \frac{d\mathbf{w}}{dt} = v\mathbf{u} \tag{3}$$

Using our assumption in (2), we approximate this rule as

$$\tau_w \frac{d\mathbf{w}}{dt} = (\mathbf{w}\cdot\mathbf{u})\mathbf{u} \tag{4}$$

As an alternative to separately computing the affect of each input pattern $\mathbf{u}$, we can approximate Hebbian learning by computing the average affect of all the input patterns presented to the neuron. Then we substitute (3) with $\tau_w \frac{d\mathbf{w}}{dt} =< (\mathbf{w}\cdot\mathbf{u})\mathbf{u} >$. We let $\mathbf{Q} =< \mathbf{uu} >$, and write the averaged Hebbian learning rule as

$$\tau_w \frac{d\mathbf{w}}{dt} = \mathbf{Q}\cdot\mathbf{w} \tag{5}$$

In this form, Hebbian learning causes the magnitude of the weight vector $\mathbf{w}$ to constantly grow, producing weight vectors that become arbitrarily large:

$$\begin{aligned}\tau_w \frac{d|\mathbf{w}|^2}{dt} &= 2\mathbf{w}\cdot\frac{d\mathbf{w}}{dt} \\ &= 2\mathbf{w}\cdot(\mathbf{w}\cdot\mathbf{u})\mathbf{u} \\ &= 2(\mathbf{w}\cdot\mathbf{u})^2 \\ &\geq 0\end{aligned}$$

2.1.2. Synaptic Normalization To prevent weights that become arbitrarily large in magnitude, we can impose weight normalization in our model. Biologically, this corresponds to competition between synaptic weights.

Subtractive Normalization In this model, we normalize synaptic weights by setting a hard constraint on the sum of the weights. (In this model, we constrain weights to non-negative values). This rule takes the form

$$\tau_w \frac{d\mathbf{w}}{dt} = v\mathbf{u} - \frac{v(\mathbf{n}\cdot\mathbf{u})\mathbf{n}}{N_u} \tag{6}$$

where $\mathbf{n}$ is a vector of ones and N_u is the length of the input vector $\mathbf{u}$.

In this model, the sum of weights $\mathbf{n} \cdot \mathbf{w}$ remains fixed:

$$\begin{aligned}
\tau_w \frac{d(\mathbf{n} \cdot \mathbf{w})}{dt} &= \mathbf{n} \cdot (v\mathbf{u} - \frac{v(\mathbf{n} \cdot \mathbf{u})\mathbf{n}}{N_u}) \\
&= \mathbf{n} \cdot v\mathbf{u} - \frac{\mathbf{n} \cdot v(\mathbf{n} \cdot \mathbf{u})\mathbf{n}}{N_u}) \\
&= v\mathbf{n} \cdot \mathbf{u}(1 - \frac{\mathbf{n} \cdot \mathbf{n}}{N_u}) \\
&= v\mathbf{n} \cdot \mathbf{u}(1 - \frac{N_u}{N_u}) \\
&= v\mathbf{n} \cdot \mathbf{u}(1-1) \\
&0
\end{aligned}$$

Oja's Rule We can also impose synaptic normalization locally. In this model, we update weights according to

$$\tau_w \frac{d\mathbf{w}}{dt} = v\mathbf{u} - \alpha v^2 \mathbf{w} \tag{7}$$

where $\alpha > 0$.

In this form, the weights compete with each other and stabilize over time:

$$\begin{aligned}
\tau_w \frac{d|\mathbf{w}|^2}{dt} &= 2\mathbf{w} \cdot \frac{d\mathbf{w}}{dt} \\
&= 2\mathbf{w} \cdot (v\mathbf{u} - \alpha v^2 \mathbf{w}) \\
&= 2\mathbf{w} \cdot (\mathbf{w} \cdot \mathbf{u})\mathbf{u} - 2\mathbf{w}\alpha v^2 \mathbf{w}) \\
&= 2v^2 - 2\alpha v^2 \mathbf{w}^2) \\
&= 2v^2(1 - \alpha \mathbf{w}^2)
\end{aligned}$$

which has a steady state at $\mathbf{w}^2 = \frac{1}{\alpha}$

2.2. Cortical Maps

To simulate cortical maps composed of many neurons, we add recurrent connections between the postsynaptic neurons in our model. These connections induce competition between neural activity, which helps prevent the neurons from developing the same responses to presynaptic input. This is important because it allows the neurons to develop different response preferences.

To simulate these connections, we add recurrent connections to the neural activity dynamics, which becomes

$$\tau_r \frac{d\mathbf{v}}{dt} = -\mathbf{v} + \mathbf{W} \cdot \mathbf{u} + \mathbf{M} \cdot \mathbf{v} \tag{8}$$

where $\mathbf{M}$ describes the recurrent connections between postsynaptic neurons.

In this case, our approximation of $\mathbf{v}$ becomes

$$\mathbf{v} = \mathbf{W} \cdot \mathbf{u} + \mathbf{M} \cdot \mathbf{v} \tag{9}$$

If $(\mathbf{I}-\mathbf{M})^{-1}$ is invertible, then we can define $\mathbf{K} = (\mathbf{I}-\mathbf{M})^{-1}$ and write $\mathbf{v}$ as

$$\mathbf{v} = \mathbf{K} \cdot \mathbf{W} \cdot \mathbf{u} \tag{10}$$

Upon defining $\mathbf{v}$, we can apply the learning rules from (2.1) to simulate the development of cortical maps.

2.2.1. Competitive Hebbian Learning To strengthen competition between postsynaptic neurons, we can use nonlinear recurrent connections. For instance, the competitive Hebbian learning model uses nonlinear competition to implement long-range inhibition and short-range excitation. This model promotes similar response preferences between neurons that are closer together, and competition (which induces different response preferences) between neurons that are farther apart.

In this model, we model activation as follows:

$$\mathbf{z} = \frac{(\mathbf{W} \cdot \mathbf{u})^{\delta}}{(\mathbf{1}^{\top} \cdot \mathbf{W} \cdot \mathbf{u})^{\delta}} \tag{11}$$

$$\mathbf{v} = \mathbf{M}\mathbf{z} \tag{12}$$

2.3. Existing Simulations

Currently, there exist open-source projects to create larger-scale simulations of topographic maps in the brain. For instance, the software package Topographica allows multiple layers of large networks of neurons [1].

Furthermore, researchers have investigated various learning rules for ocular dominance. For instance, [3] surveyed pattern-based and developmental models of ocular dominance and compared the results to experimental data from macaque monkeys. Others, such as [6] have proposed and simulated specific mathematical models of ocular dominance.

3. Motivation and Goals

The goal of this project is to model ocular dominance using unsupervised learning rules and to analyze changes to these models. We simulate ocular dominance in single neurons as well as cortical maps of ocular dominance, and experiment with various learning rules and parameters.

This project builds upon the work of previous researchers who have designed the various learning rules implemented in this project. By experimenting with model parameters, we hope to gain insight into possible biological implementations of ocular dominance maps.

Furthermore, we wanted to create easily modifiable implementations of these ocular dominance models. Therefore in addition to analyzing various models and parameters, this project involved implementing the models from scratch in a format designed to allow easy experimentation. The appendix (7) contains more details about this component of the project.

4. Methods

We model ocular dominance in single neurons as well as cortical maps of ocular dominance in networks of multiple neurons in the same layer. In all our simulations, we use a learning rate of 0.1, which corresponds to $\tau_w = 10$.

4.1. Ocular Dominance in a Single Neuron

We first model ocular dominance in single neurons; that is, we model the tendency of single neurons to develop a preferential response to inputs from a certain eye.

We model inputs from the left and right eye as a 2×1 vector of correlated random inputs. The eyes' input are generated according to

$$u_l = x + 0.5\sigma_l \tag{13}$$

$$u_r = x + 0.5\sigma_r \tag{14}$$

where x, σ_l, and σ_r are uniform random variables between 0 and 1. We use this scheme to generate inputs to approximate the real-world tendency for both eyes to receive correlated, but slightly different, stimuli.

We then implement each of the learning rules described in (2.1) and simulate the response preferences that develop in single neurons.

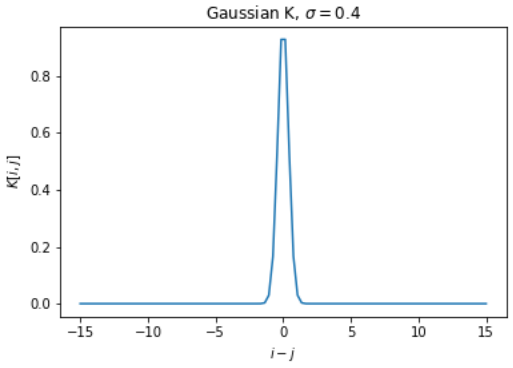
4.2. Cortical Map of Ocular Dominance

We then model the development of cortical maps of ocular dominance using systems of recurrently connected neurons.

We simulate a system with 500 neurons, and experiment with three models of recurrent connections.

First, we test a model in which the recurrent connections between two neurons follow a Gaussian distribution. More specifically, we use

$$K[i,j] = \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-1}{2} \left(\frac{|i-j|}{\sigma}\right)^2 \tag{15}$$

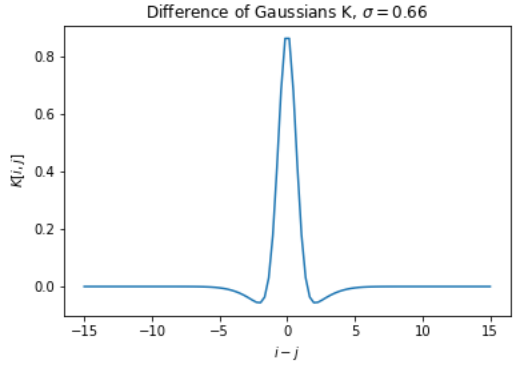


(16)

Second, we test a model in which $\mathbf{K}$ is a difference of Gaussians, as described in [2]. We

implement this as

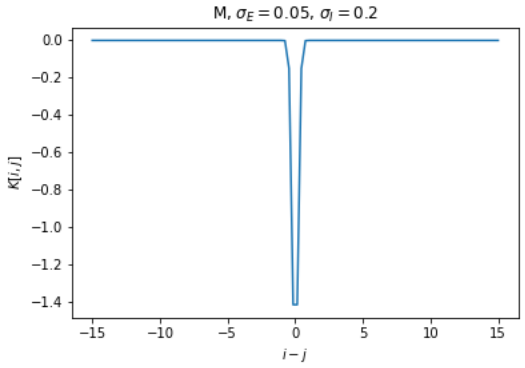
$$K[i,j] = \exp\frac{-(i-j)^2}{2\sigma^2} - \frac{1}{9}\exp\frac{-(i-j)^2}{18\sigma^2} \tag{17}$$



(18)

Third, we test a model that follows the "Mexican-hat" model (approximated by a difference of Gaussians with different σ for excitatory and inhibitory contributions) described in [9]. We implement this as

$$\frac{1}{\sigma_E\sqrt{2\pi}}\exp\frac{(i-j)^2}{2\sigma_E^2} + \frac{1}{\sigma_I\sqrt{2\pi}}\exp\frac{(i-j)^2}{2\sigma_I^2} \tag{19}$$



(20)

5. Results and Discussion

5.1. Ocular Dominance in a Single Neuron

Here, we show the trajectories of left and right eye weight development for a single trial. We also show final weights learned after 10 trials, where we randomly initialize the weights at the beginning of each trial.

5.1.1. Basic Hebb With basic Hebbian learning, we see that both weights grow arbitrarily large (fig 1). Furthermore, we observe that the approximation we describe in (2.1.1) exhibits the same behavior as the original learning rule.

5.1.2. Oja's Rule With Oja's rule, rather than weights growing indefinitely large, we observe that the weight magnitudes display competition. When both weights have large magnitudes, the system moves towards reducing the weights, and the weights eventually stabilize rather than continuing to grow in magnitude (fig 2).

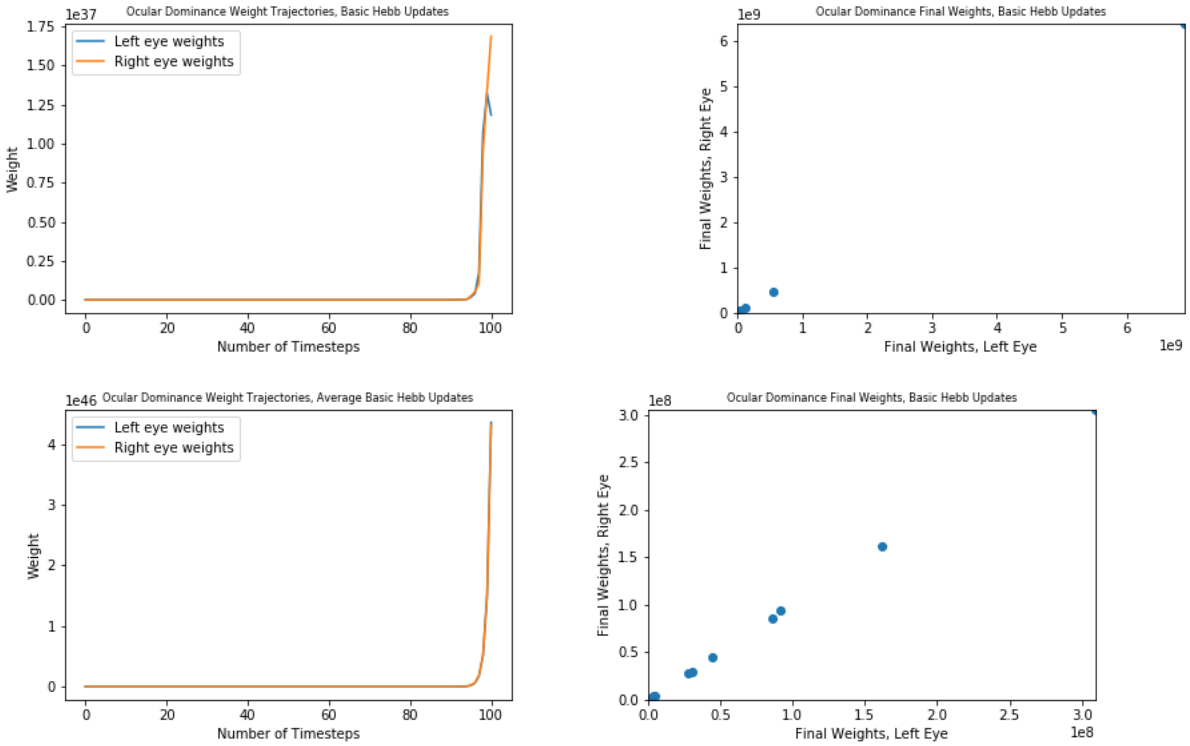


Figure 1: Basic Hebb

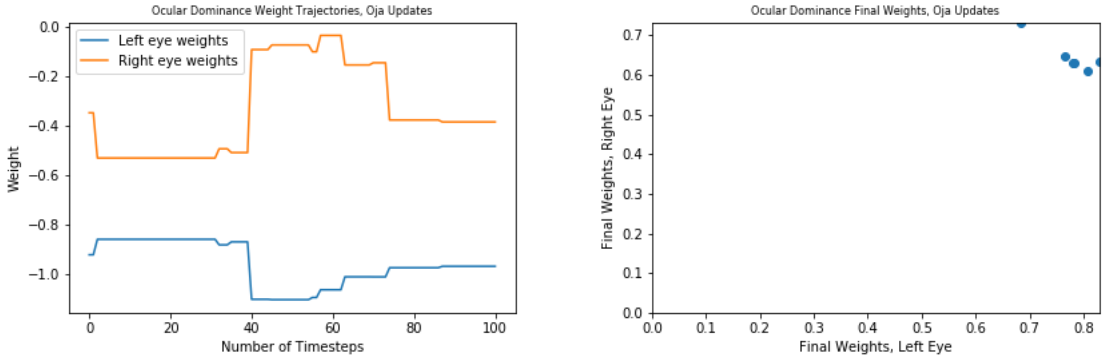


Figure 2: Oja's Rule

5.1.3. Subtractive Normalization With subtractive normalization, we similarly see competition between weight magnitudes, in that a high value in one weight tends to correspond to a low value in the other weight (fig 3). In these simulations, we see ocular dominance start to emerge, where the final weights display near-zero values for one eye and a relatively higher weight for the other (fig 3).

5.2. Cortical Maps of Ocular Dominance

In each simulation, we show the differences between weights to the left eye and to the right eye of 50 neurons in the simulation (in the simulated system, there are a total of 500 neurons).

5.2.1. Gaussian K With a Gaussian **K**, we observe that very small values of σ lead to maps with more changes of ocular dominance preferences, and large values of σ lead to map with more changes between ocular dominance preferences (figs 4,5,6). By looking at the forms of Gaussian

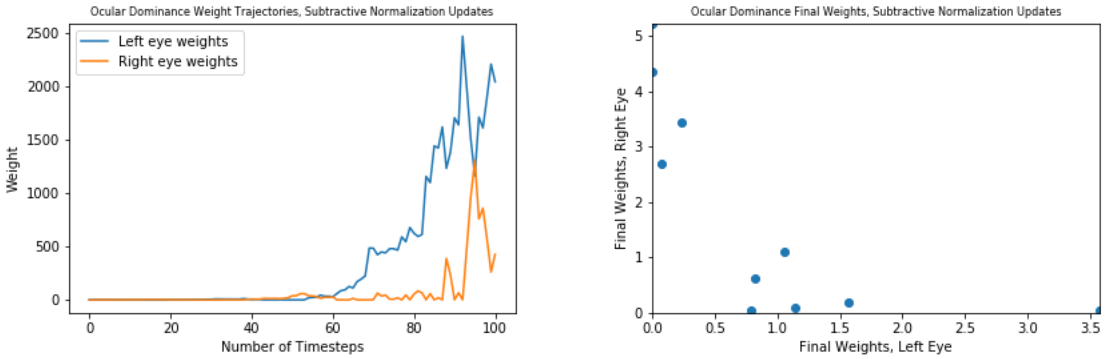


Figure 3: Subtractive Normalization

functions with different σs, we observe that functions with smaller σs correspond to recurrent connections that are more concentrated to a particular neuron's nearer neighbors (fig 7).

Figure 4: Ocular Dominance Map, Gaussian $\mathbf{K}$, Averaged Basic Hebb

Figure 5: Ocular Dominance Map, Gaussian $\mathbf{K}$, Basic Hebb

Figure 6: Ocular Dominance Map, Gaussian $\mathbf{K}$, Subtractive Normalization

5.2.2. Difference of Gaussians K We see a similar pattern with the $\mathbf{K}$ that takes the form of a difference of Gaussians (figs 8,9,10).. In these simulations, when we have $\sigma = 0.1$, the resulting neural preferences fail to show the orderly pattern of preferences that appear in cortical maps of ocular dominance. At the other end, when we have $\sigma = 2$, the resulting neural preferences form a pattern that is dominated by few shifts in ocular dominance. We observe that this follows from the forms of the difference of Gaussians functions (as described in (17)), which we show in (fig 11).

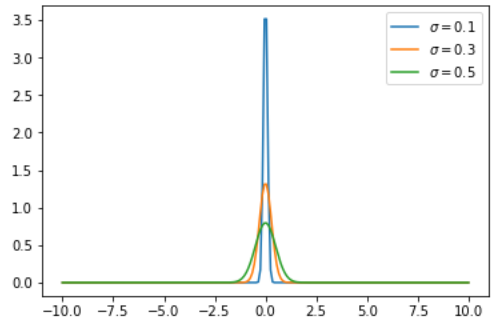


Figure 7: Gaussian functions with different σ

Figure 8: Ocular Dominance Map, Difference of Gaussians $\mathbf{K}$, Averaged Basic Hebb

5.2.3. Competitive Hebbian Learning In our simulations with competitive Hebbian learning, we observe less distinct patterns of ocular dominance preferences (figs 12,12). We were able to find parameters for the matrix of recurrent connections $\mathbf{M}$ that result in networks that resemble cortical maps of ocular dominance, but these maps are much less clear than the ones in the preceding sections.

5.2.4. 2D Cortical Map Since neurons in the brain form cortical maps in more than one direction, we also simulate a 2D cortical map of ocular dominance, using a $\mathbf{K}$ that takes the form of a 2D Difference of Gaussians. We find that this forms a 2D cortical map of ocular dominance similarly displays neuron preferences organized into a cortical map (fig 15).

6. Future Work

In the future, we plan to build upon these simulations to mimic biological experiments of ocular dominance. Specifically, we will perform manipulations to mirror experiments of neurotrophic factors and monocular deprivation.

Researchers have found that applying neurotrophic factors to neural systems prevent the development of cortical maps of ocular dominance, and have interpreted this as resulting from a lack of competition between neurons [5]. By experimenting with the recurrent connections used in this project's simulation, we will mimic a reduction in competition due to neurotrophic factors and investigate the effect of this manipulation on the formation of cortical maps of ocular dominance.

Furthermore, we will study monocular deprivation and recovery. Experimenters have shown that monocular deprivation during a critical period hinders the development of ocular dominance preferences in kittens and rats, and that later binocular reinstatement sometimes allows the animals to recover ocular dominance responses to both eyes [4][8]. By manipulating the inputs used in this project's simulations, we will mimic monocular deprivation during ocular dominance development. With these manipulations, we plan to find a "critical period" in our model that mirrors that of biological systems, and to test the situations in which ocular dominance maps can be reinstated.

Figure 9: Ocular Dominance Map, Difference of Gaussians $\mathbf{K}$, Basic Hebb

Figure 10: Ocular Dominance Map, Difference of Gaussians $\mathbf{K}$, Subtractive Normalization

References

[1] James Bednar. Topographica: Building and Analyzing Map-Level Simulations from Python, C/C++, MATLAB, NEST, or NEURON Components. *Frontiers in Neuroinformatics*, 3:8, 2009.

[2] Peter Dayan and Laurence F. Abbott. *Theoretical Neuroscience*. Draft edition, 2000.

[3] E. Erwin, K. Obermayer, and K. Schulten. Models of orientation and ocular dominance columns in the visual cortex: a critical comparison. *Neural Computation*, 7(3):425–468, May 1995.

[4] Daniel E. Feldman. Synaptic Mechanisms for Plasticity in Neocortex. *Annual review of neuroscience*, 32:33–55, 2009.

[5] Anthony E. Harris, G. Bard Ermentrout, and Steven L. Small. A model of ocular dominance column development by competition for trophic factor. *Proceedings of the National Academy of Sciences of the United States of America*, 94(18):9944–9949, September 1997.

[6] K. D. Miller, J. B. Keller, and M. P. Stryker. Ocular dominance column development: analysis and simulation. *Science (New York, N.Y.)*, 245(4918):605–615, August 1989.

[7] Kenneth D. Miller. Receptive Fields and Maps in the Visual Cortex: Models of Ocular Dominance and Orientation Columns. In *Models of Neural Networks III*, Physics of Neural Networks, pages 55–78. Springer, New York, NY, 1996. DOI: 10.1007/978-1-4612-0723-8_2.

[8] Donald E. Mitchell, Max Cynader, and J. Anthony Movshon. Recovery from the effects of monocular deprivation in kittens. *The Journal of Comparative Neurology*, 176(1):53–63, November 1977.

[9] Taro Toyoizumi and Kenneth D. Miller. Equalization of ocular dominance columns induced by an activity-dependent learning rule and the maturation of inhibition. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 29(20):6514–6525, May 2009.

[10] Jeremy M. Wolfe, Keith R. Kluender, Dennis M. Levi, Linda M. Bartoshuk, Rachel S. Herz, Roberta L. Klatzky, Susan J. Lederman, and Daniel M. Merfeld. *Sensation & Perception*. Sinauer Associates, Inc., Place of publication not identified, 4th edition edition, October 2014.

7. Appendix

The code used to create the simulations described in the previous sections is available online. The code is designed to allow modular definitions of learning rules, recurrent connections, and inputs that can be substituted into the main simulation framework.

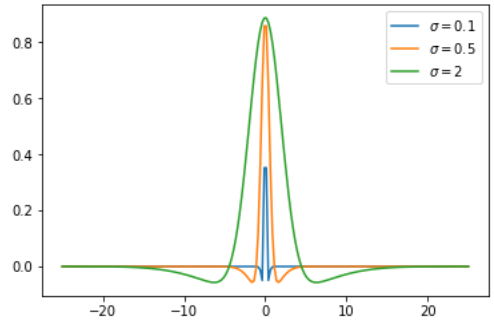


Figure 11: Difference of Gaussians function with different σ

Figure 12: Ocular Dominance Map, Competitive Hebbian Learning, $\delta = 1$

Figure 13: Ocular Dominance Map, Competitive Hebbian Learning, $\delta = 10$

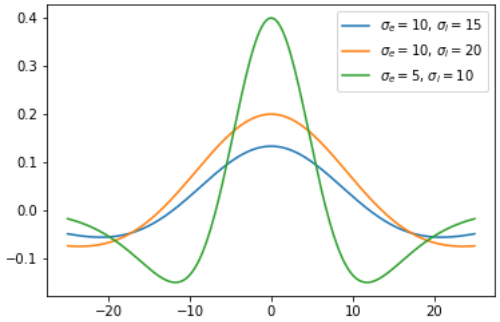


Figure 14: "Mexican-hat" $\mathbf{M}$ functions with different σ_E and σ_I

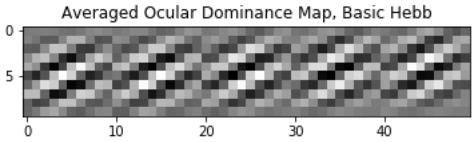


Figure 15: Ocular Dominance Map in 2D, Difference of Gaussians with Averaged Basic Hebb Updates