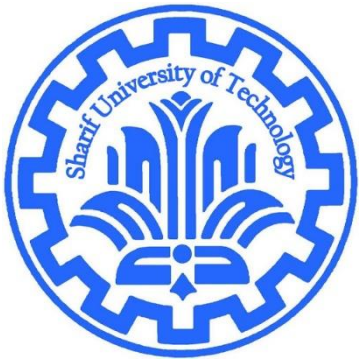


In the Name of God

Neuroscience of Learning, Memory, Cognition

Project Report

Isolation of Relevant Visual Features from Random Stimuli for Cortical Complex Cells

Armin Panjhepour - 98101288

Aliasghar Pourghani – 98101299

- 1.Project Introduction:

1. This project is a implementation of the article below:

Isolation of Relevant Visual Features from Random Stimuli for Cortical Complex Cells

Jon Touryan,[1] **Brian Lau,**[2] **and Yang Dan**[1,2]
[1]*Group in Vision Science, University of California, Berkeley, California 94720, and* [2]*Division of Neurobiology, Department of Molecular and Cell Biology, University of California, Berkeley, California 94720-3200*

The purpose of this article is to understand neural circuits in visual processing for **complex** neurons by extracting the features of a complex visual input. In this article, they have given the cat, spatiotemporal bars as stimuli and recorded responses of its complex cells in primary visual cortex and then, applied **spike-triggered correlation** analysis to the stimuli. Then they were able to isolate a small number of relevant features from a large number of null features in the stimuli. This small number of relevant features excite neuron effectively but null features response a little even they have no response. Thus, for each of this complex neurons, visual inputs can be decomposed into two distinct types of features:

1- relevant features 2- null features

So, How we can decompose complex visual inputs into sets of basic features? Several methods have been used for this purpose such as principal component analysis(PCA), independent component analysis, etc. For neurons with **linear** stimulus-response relationship, we can estimate relevant visual features can be identified by estimating their linear receptive fields using **spike-triggered average** of the stimulus. The resulting receptive fields can largely work account for neuronal responses to complex spatiotemporal stimulus but in the visual cortex, most of the neurons are complex neurons with non-linear

stimuli-response relationship which spike-triggered average is not capable of characterizing. So in this article, they have used **spike-triggered correlation** to construct the basis set for complex cells which they`ll segregate into two distinct types, a small number of relevant features and a large number of null features.

2. Complex neurons:

Neurons with non-linear stimuli-response relationship. According to the **MATERIALS AND METHODS** in the article, by recording using tungsten electrodes, cells were classified as simple if their receptive fields had clear on and off subregions and if the ratio of the first harmonic to the DC component of the response to an optimally oriented drifting grating was > 1 and all other cells were classified as complex. We have 61 complex neurons in this experiment.

3. Spike-triggered Correlation Analysis:

In general, if cell is sensitive to certain features in the visual stimuli, the spike-triggered stimulus ensemble should exhibit a different probability distribution from the entire stimulus ensemble in presence of that feature. in this method we study changes in the variance of the distribution to see if any change occurred in presence of a feature.

Practically, to find relevant features:

1. Calculate Spike-triggered correlation matrix:

we calculate correlation of spike-triggered inputs with themselves, and average them over all spike-triggered inputs:

$$C_{\mathrm{m,n}} = \frac{1}{N}\sum_{i=1}^{N} S_{\mathrm{m}}(i) S_{\mathrm{n}}(i),$$

which $C_{m,n}$ is the (m,n) element in spike-triggered correlation matrix.

2. Calculate eigen-values and eigen-vectors of the Spike-triggered correlation matrix.
3. Create proper number(5 is enough) of control correlation matrices and calculate the eigen-values and eigen-vectors of the mean of them. Then the confidence interval for the eigen-values was set at mean $\pm$ 10.4 SD. Eigen-vectors with eigen-values outside of the confidence interval were considered significant.

4. Separating spike triggered stimulus and other stimulus

To compare all stimulus with spike-triggered stimulus, we use their image on significant eigen-vectors of the correlation matrix, and as it will be explained in coming parts, if this image were bigger than a threshold, it is more likely to be spike-triggered stimuli than normal stimuli. Also in next parts we will see there is significant difference in histogram of these to set of stimulus, especially in their average(spike-triggered has bigger average, but both have normal distribution). Another way to compare two distribution is t-test, which is used in part 3.

5. Segregation between two types of visual features:

We tried explain the results and avoid explaining experiment steps, etc. Experiments results, indicate that every neuron has a few significant eigen-vectors, most of them (47 out of 60) has 2 significant eigen-vectors These two eigenvectors exhibited separate on and off spatial subregions, resembling the receptive fields of simple cells. 3 out of 60 has one significant eigen-vector. In a few cases (3 of 60), we found only one significant eigenvector for each complex cell; these vectors also exhibited spatiotemporal profiles resembling

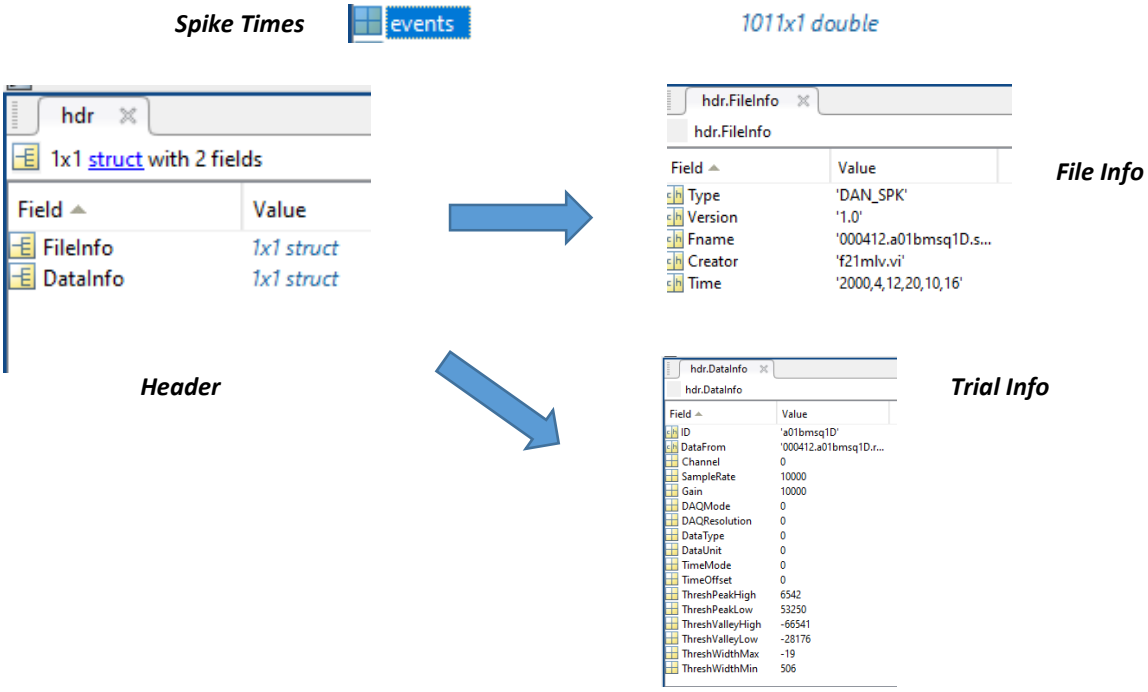
simple cell receptive fields. In the remaining cases, more than two eigenvalues reached significance. However, these additional eigenvectors (corresponding to third, fourth, . . . , largest eigenvalues) tended to exhibit much less spatiotemporal structure than the first two eigenvectors, and their eigenvalues were much smaller, suggesting less functional importance.

WARNING: on lines, 60-62-221, you should change this address, `cd C:\Users\Utel\Desktop\Neuroscience_Project`**, to the address which you have unzipped our project folder. Unless you get errors while running.**

- 2.Introduction to Data set:

1. “sa0/log” files contents:

Each sa0 file contains a vertical vector called “events” which has the spike times of that trial and a struct called header(“hdr”), which has some information of that specific trial and some information about the file. For example these are contents of “000412.a01bmsq1D.sa0”



log files contains information about the stimulus. For example its name, testing type, Frame rate, directions given in that trial, width and height of the bar, directions sequence, etc.

2. Read “sa0/msq1D” Data:

fget_spk.m in MatlabFunctions directory is used for reading “sa0” files and **tview.m** is for reading “tune” files.

```
function outputStruct = Func_ReadData(neuronCode)
```

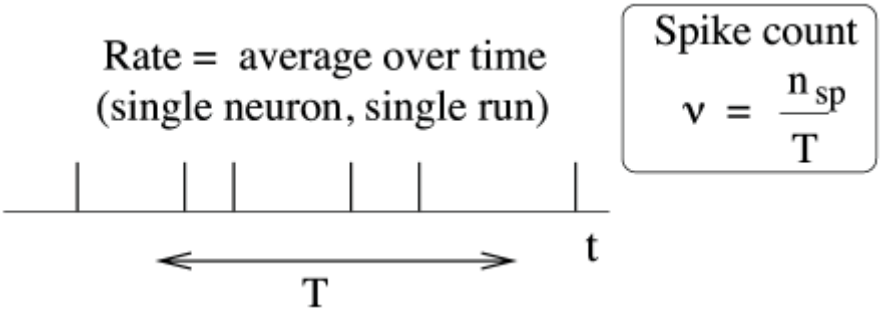
This function, gets neuron code in the input and return a struct which contains spike times of the target neuron(events)

and header of each trial. For example for the neuron, “000412.a01”:

1x4 struct with 2 fields

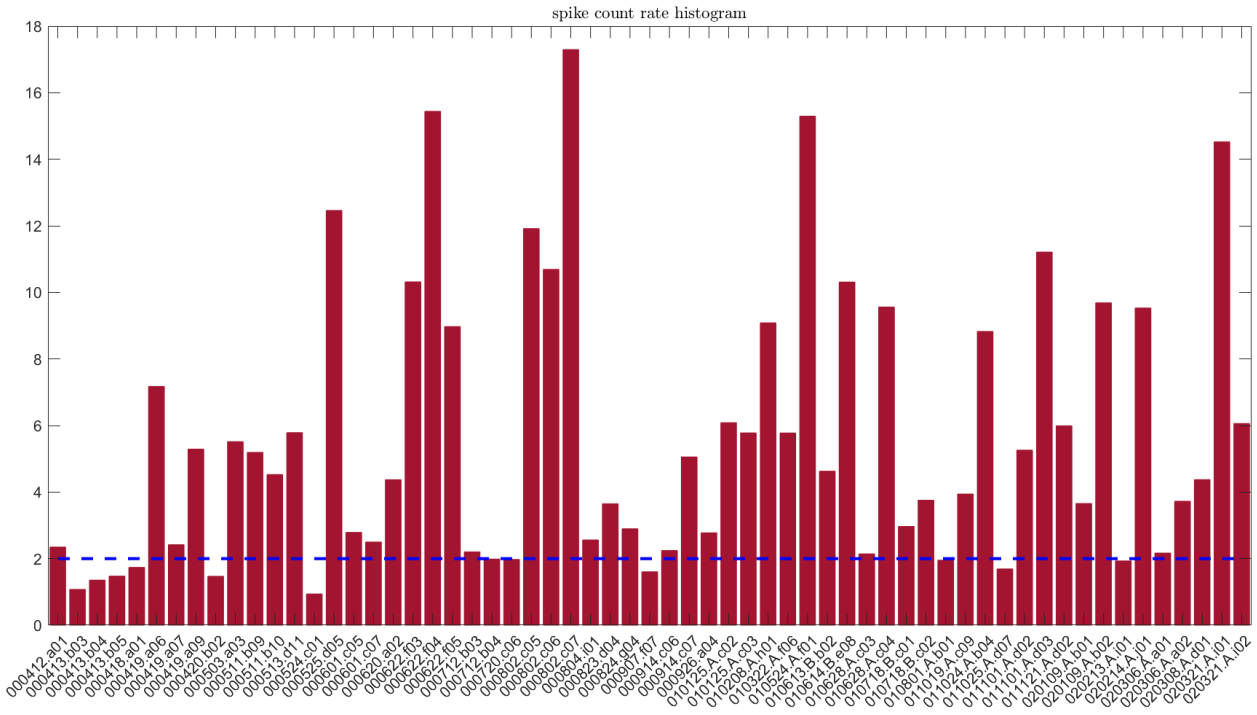
Fields	events	hdr
1	1011x1 double	1x1 struct
2	1274x1 double	1x1 struct
3	1396x1 double	1x1 struct
4	1472x1 double	1x1 struct

3. Spike Count Rate:



We average number of spikes over all trials and over T which is the period of each trial. T = 32767 / 59.721395

```
function SCR = plotSpikeCountRate(neuronCode,msq1Dstruct,T,numberOfFrames)
    % SCR = spike count rate
    SCR = 0;
    for i=1:length(msq1Dstruct)
        SCR = SCR + (length(msq1Dstruct(i).events));
    end
    SCR = SCR/(length(msq1Dstruct)*T*numberOfFrames);
end
```



neurons with spike count rate less than 2:

lessThanTwoSCRsNeuronCodes

12x1 string

	1	2
1	000413.b03	
2	000413.b04	
3	000413.b05	
4	000418.a01	
5	000420.b02	
6	000524.c01	
7	000712.b04	
8	000720.c06	
9	000907.f07	
10	010801.A.b01	
11	011025.A.d07	
12	020213.A.i01	

4. Spike-Triggered Inputs:

```
function [targetExperiment stimuliExtraction] =
Func_StimuliExtraction(neuronCode,experimentID,msq1D,T,key,controlVec)
```

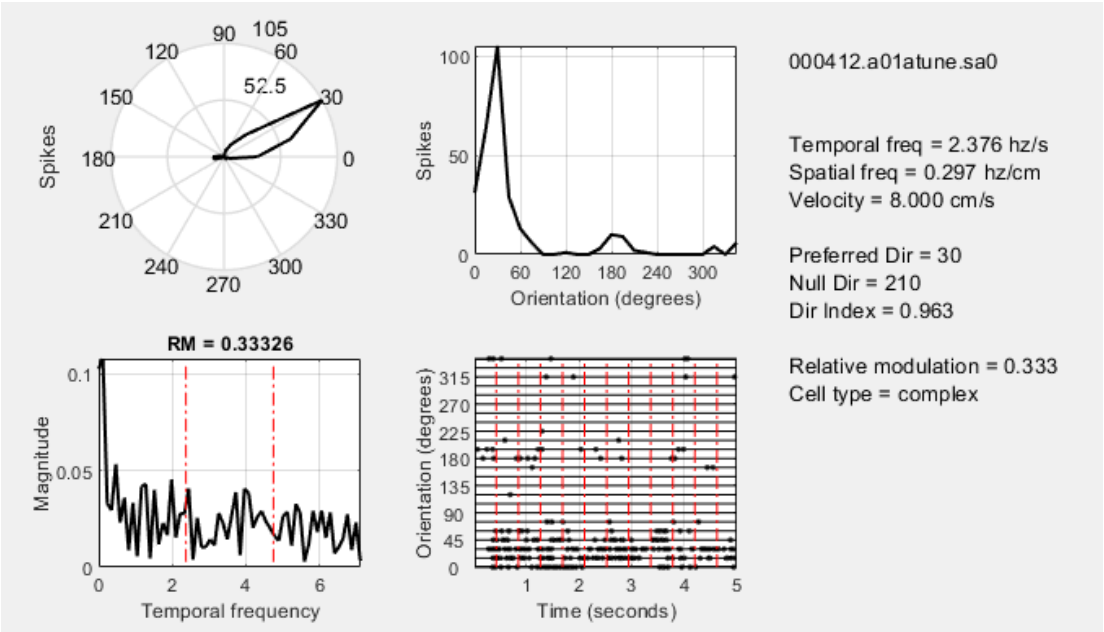
"Func_StimuliExtraction" which gets neuron code, experiment ID, the input stimuli(msq1D), Period(T=1/59.721395), a key for selecting the algorithm which if it is 1, it finds the spike-trigger inputs for `controlVec` which is the random vectors for creating control correlation matrices that we`ll see soon and if it is 0, it finds the spike-trigger inputs for the output of the neurons.

Now we need this for our neuron responses, for example, for the neuron ,"000412.a01" and the experiment, "a01emsq1D", our stimuliExtracted matrix is:

Which the 3rd dimension is the number of spikes of the following neuron in that specific trial.

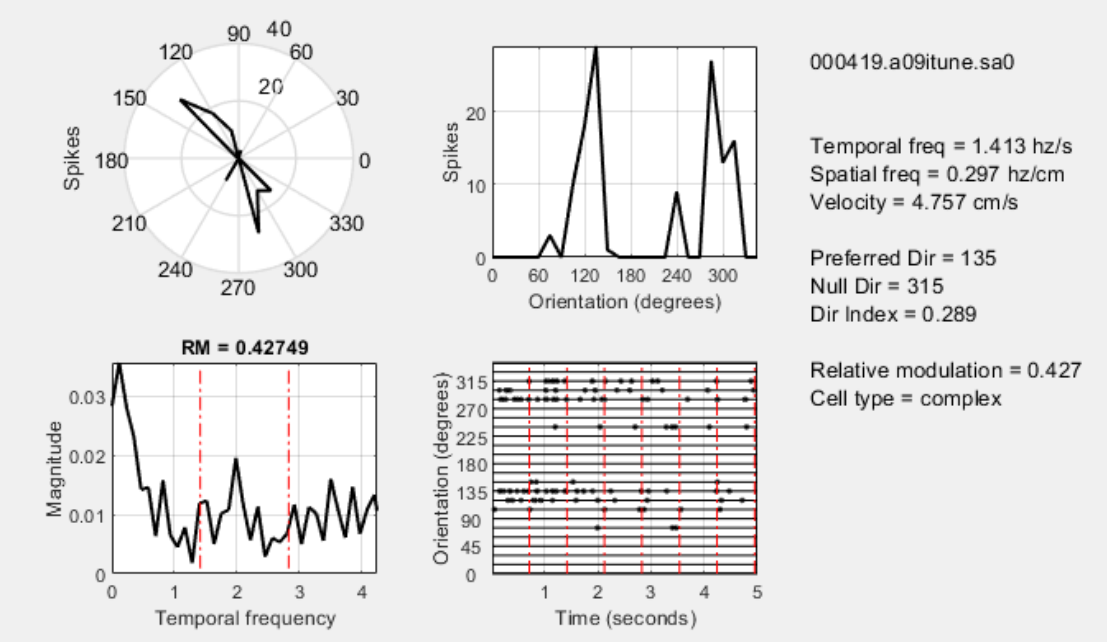
5. **tview:** The function gets log file name in the input.

- neuron "000412.a01":

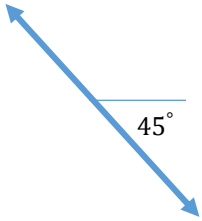


As all the plots show, this neuron is most sensitive to the angle of 30 degrees.(a little sensitivity to the 210 degrees, too)

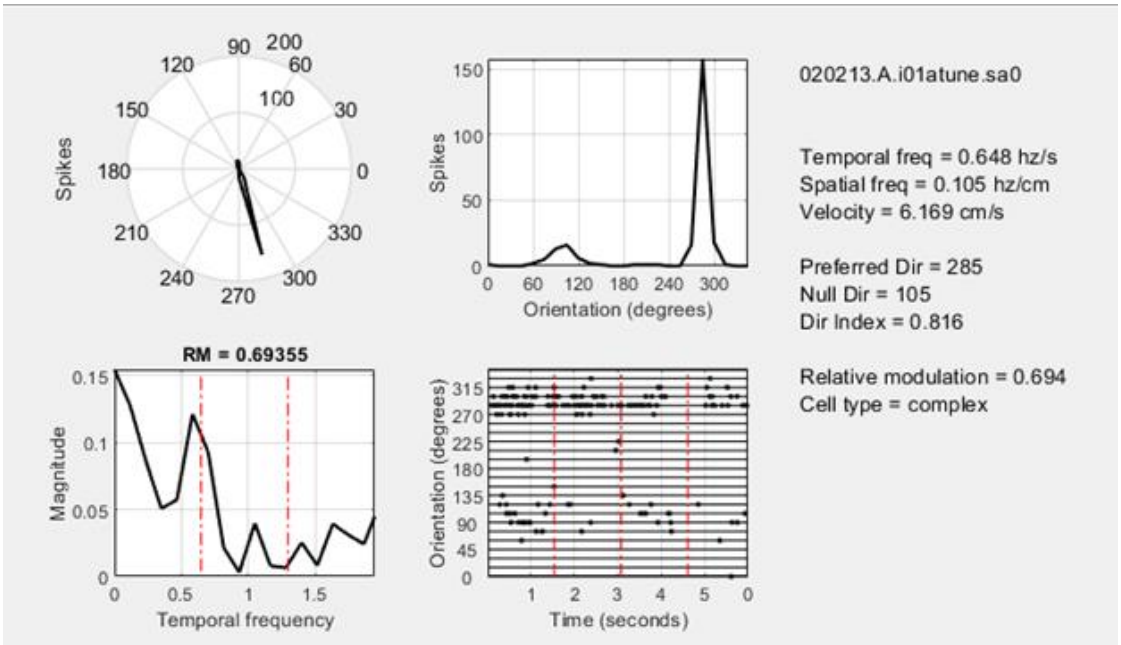
- neuron “000419.a09”:



This neuron is most sensitive to two angles, around 135 and 315 degrees. As you`ve realized, their difference is about 180 degrees and thus, the neuron is sensitive to this axis:



- neuron "020213.A.i01":



This neuron is most sensitive to the angle around 280 degrees. Although it is a little sensitive to 100, too. The difference of this two angles is around 180 degrees again but the neuron is mostly sensitive just to one direction which is 280 degrees.

- he way that the best directions for each neuron has found, is probably some experiments which they show the cat different bars in different angles and capture neurons` spiking. The project sheet called this experiments tuning.

- 3.Working with classic spike-triggered average method

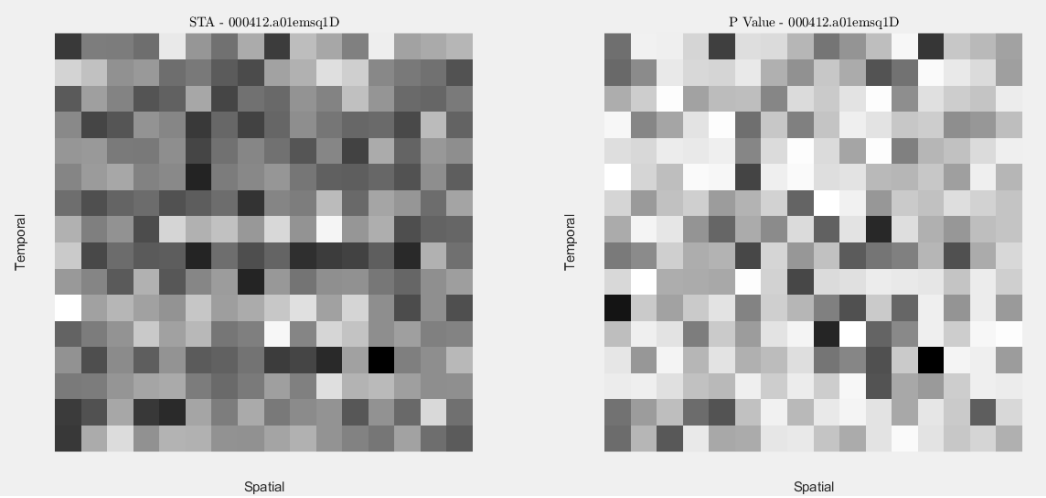
First we select a neuron, and because more than one experiment is performed on some neurons, we choose one of the experiments too (if there were more than one).

<u>1. Spike-triggered Average:</u>

a) find inputs that caused spike (assume output is response of instant input, also note that every input is a 16×16 matrix, means 16 frames and 16 bars)

b) average all inputs that specified above.

to implement these steps in Matlab, we defined a function named `Func_StimuliExtraction` to find inputs that caused spikes, Then we average all these inputs and plot them with `imshow()` and `mat2gray()` (to have better constrast.)
The left picture is the result:



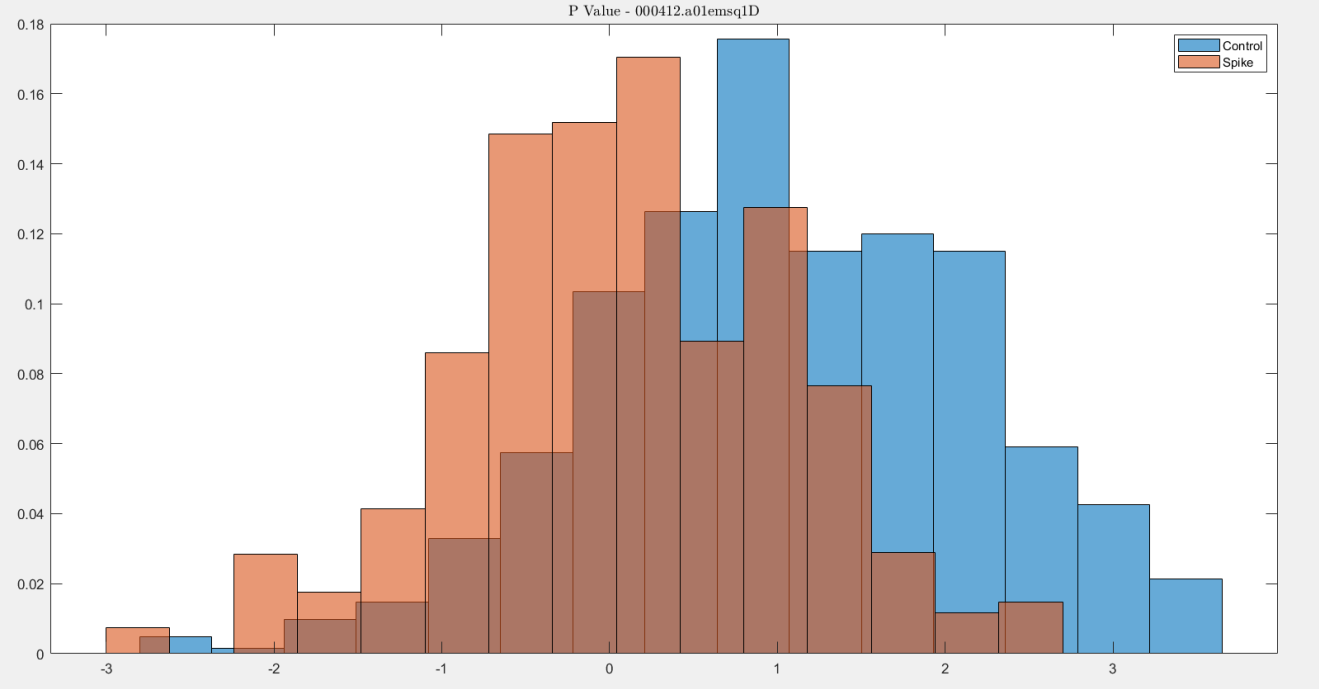
2. T-test:

T-test: A t-test is a type of inferential statistic used to determine if there is a significant difference between the means of two groups, which may be related in certain features.
to calculate P-value, we've used `ttest2()` function on 256 cell of the 16×16×N matrix.
the result is plotted above. (the right picture)
As we see in the picture, there is a significant difference between the spike-triggered inputs and zero, explains that inputs must have large image on SPA to cause spike.

3. To find images of the stimulus on the SPA, we used inner product, for example:

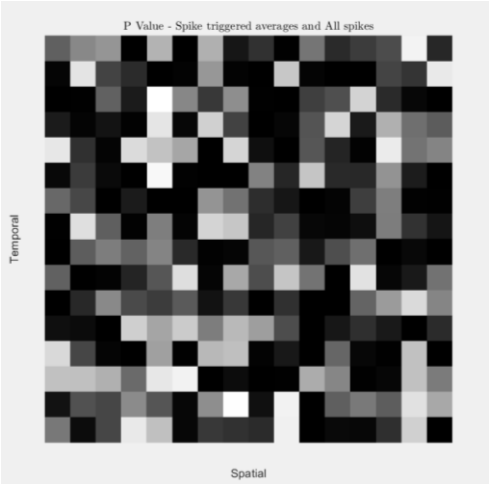
```
allStimulusImage(i) =
sum(normalStimulus(:,:,i).*spikeTriggeredAveraged,'all');
```

then we plotted distribution of these images with histogram function:



as we expected, both distributions are almost Normal and mean of the Spike-triggered inputs are higher.
because inputs that are more similar to the receptive field of the neuron, are more likely to cause spike.

4. We performed T-test between all cells of the all inputs and inputs that caused spike, to find out if there is a significant difference between these matrices.



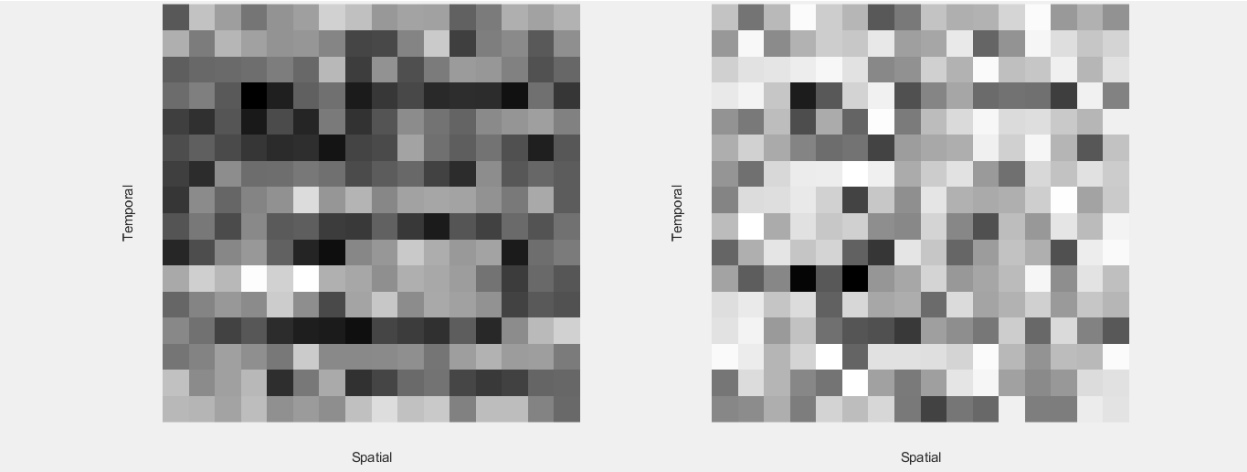
As we see there are some bright areas, means there are some areas in spike-triggered inputs that are important to neuron and are slightly different in spike-triggerd inputs than other spikes.

5. We fit gaussian functions on both of the histograms we saw in part 3 using their mean and standard deviation and find the place which the two gaussians cross each other. This would be the threshold. For the stimulus ont the right side of this threshold, we`ll have spike and on the left, we don`t.

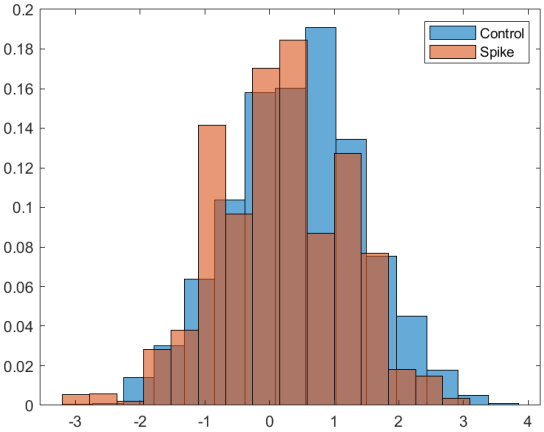
6.

Same 4 previous questions for neuron "000503.a03", and experiment, "a03cmsq1D":

Question 1 and 2:(STA and P Value)

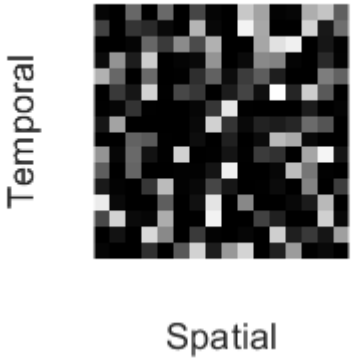


Question 3:



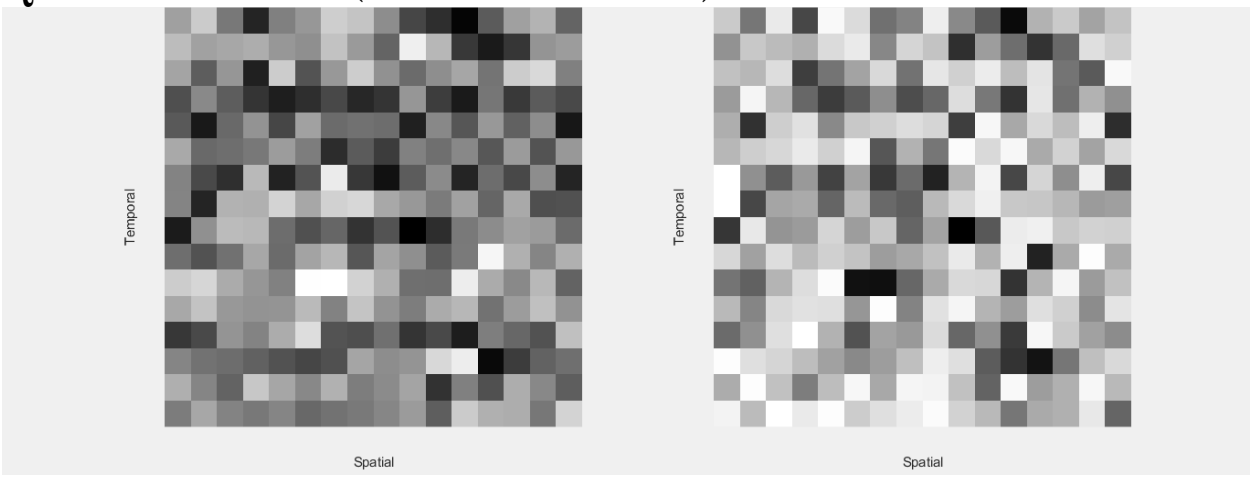
Question 4:

P Value - Spike triggered averages and All spikes

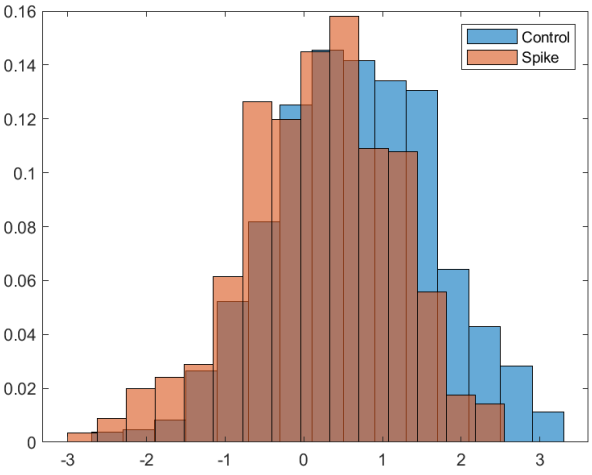


Same 4 previous questions for neuron “000926.a04”, and experiment, “a04bmsq1D”:

Question 1 and 2:(STA and P Value)

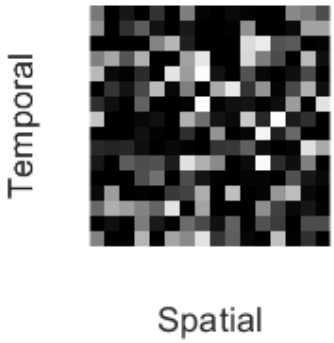


Question 3:

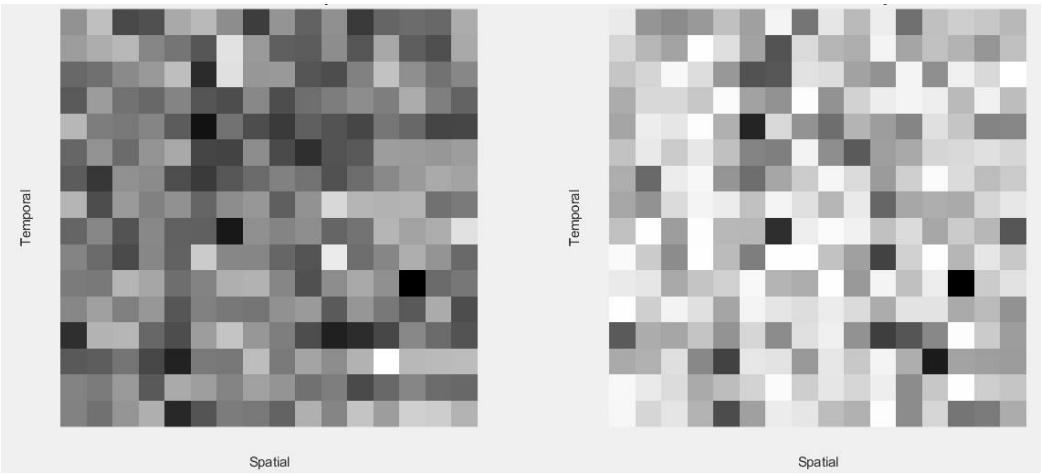


Question 4:

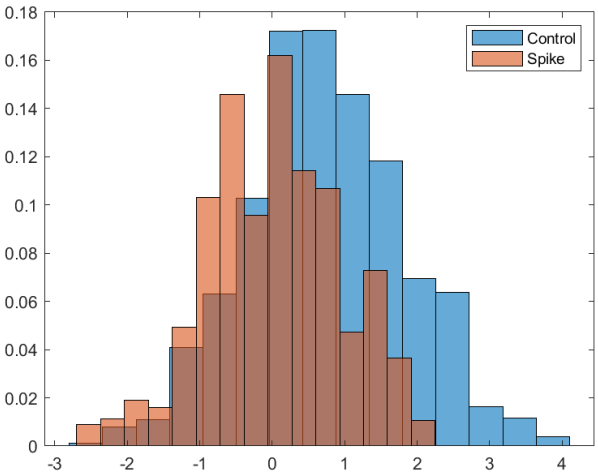
P Value - Spike triggered averages and All spikes



Same 4 previous questions for neuron “020308.A.d01”, and experiment, “A.d01emsq1D”:
Question 1 and 2:(STA and P Value)

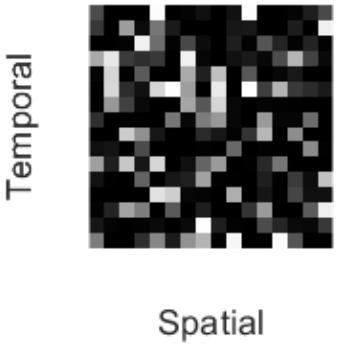


Question 3:

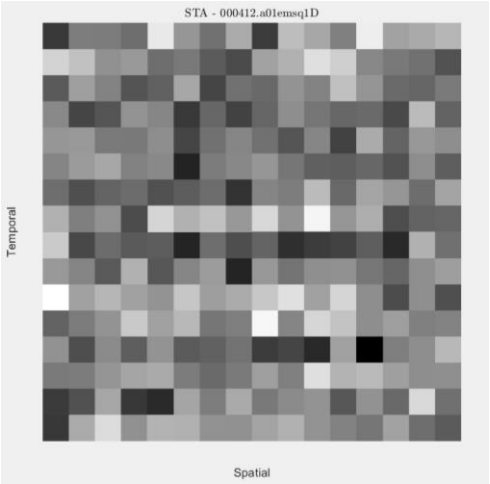


Question 4:

P Value - Spike triggered averages and All spikes



7. As we studied in the paper too, Spike-triggerd average cannot characterize the stmiulus-response relation, so we didn't expect any special pattern in STA.
As we see below, STA seems a random matrix and approves our expectation:



- the estimated receptive field(STA) is sth random.

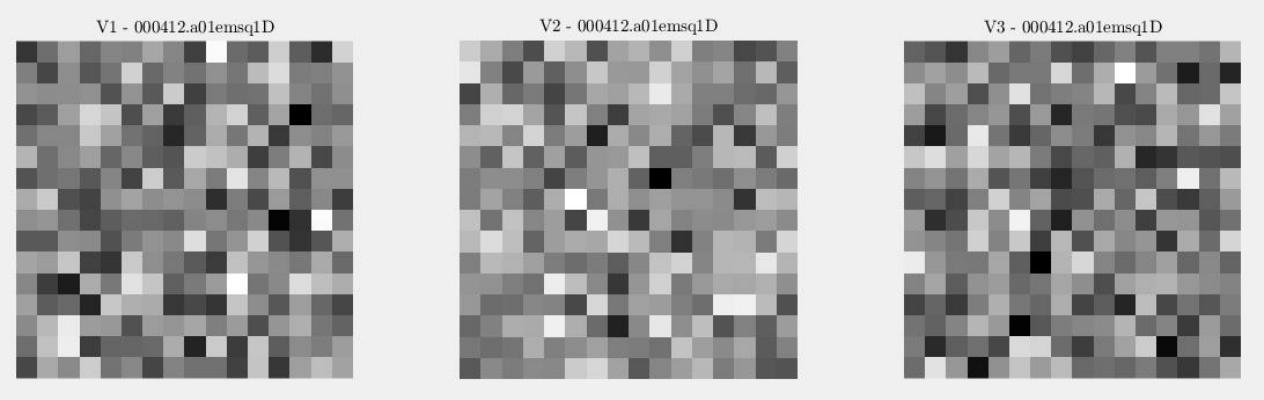
- 4.working with spike-triggered correlation method:

First we choose a neuron, and because more than one experiment is performed on neurons, we choose one of the experiments, too.

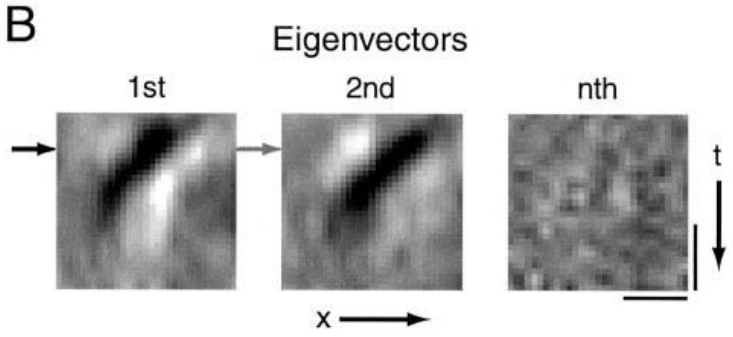
1. To find eigen-vectors, first we calculate correlation matrix using `correlationMatrixCalculator()` function (this function is explained below). then we use `eig()` function to calculate eigen-values and eigen-vectors.

- explanation of the `correlationMatrixCalculator()` function :

```
for i = 1:size(inputSignal,3)
                correlatio nMat = correlationMat +
        inputSignalvert(:,:,i)*inputSignalhor(:,:,i);
end
```



from right to left, we have our top 3 eigen vectors, which as you see, the more the eigen value of that eigen vector, the more meaningful its pattern is. A better result of this implementation from article:



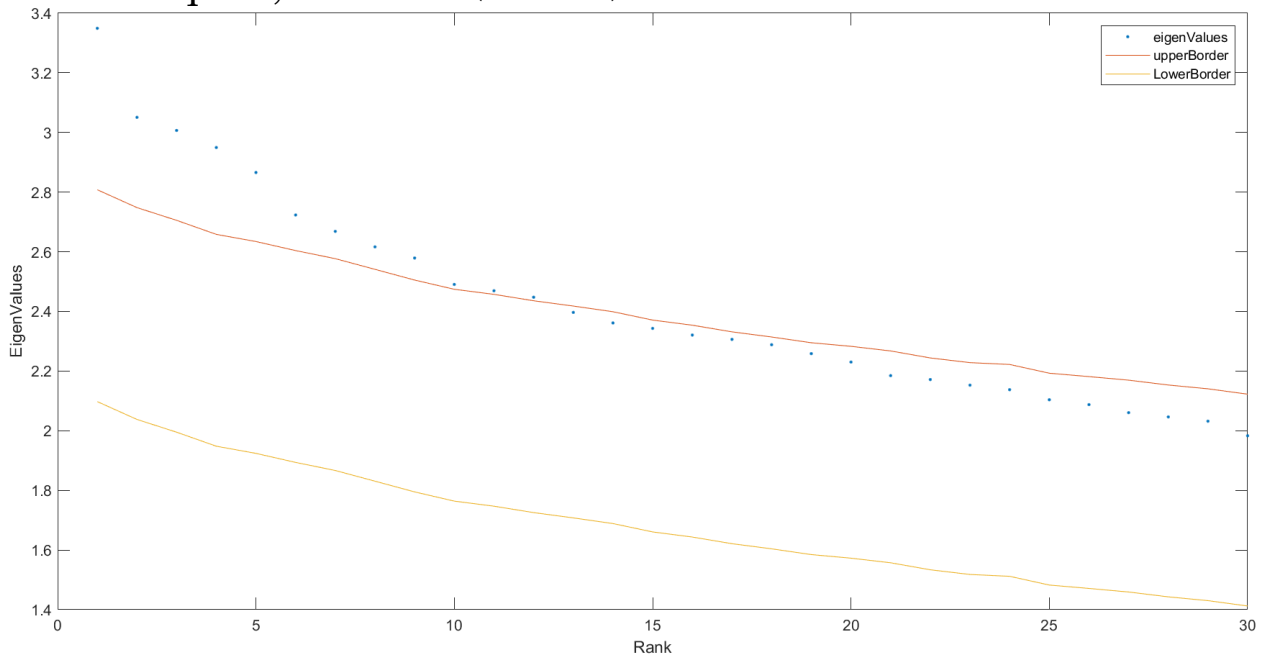
2. Confidence interval:

To find confidence interval, we need control correlation matrices, as it mentioned in the paper, too, 5 control correlation matrices are enough.(control correlation matrix is spike-triggered correlation of the random spike trains which has same number of spikes with ouput). Then we calculate their eigen values and get a mean over 5 eigen values we have and we come up with a meanEigenControlVector. With this meanEigenControlVector we can create our confindance intervals depend on this formula:

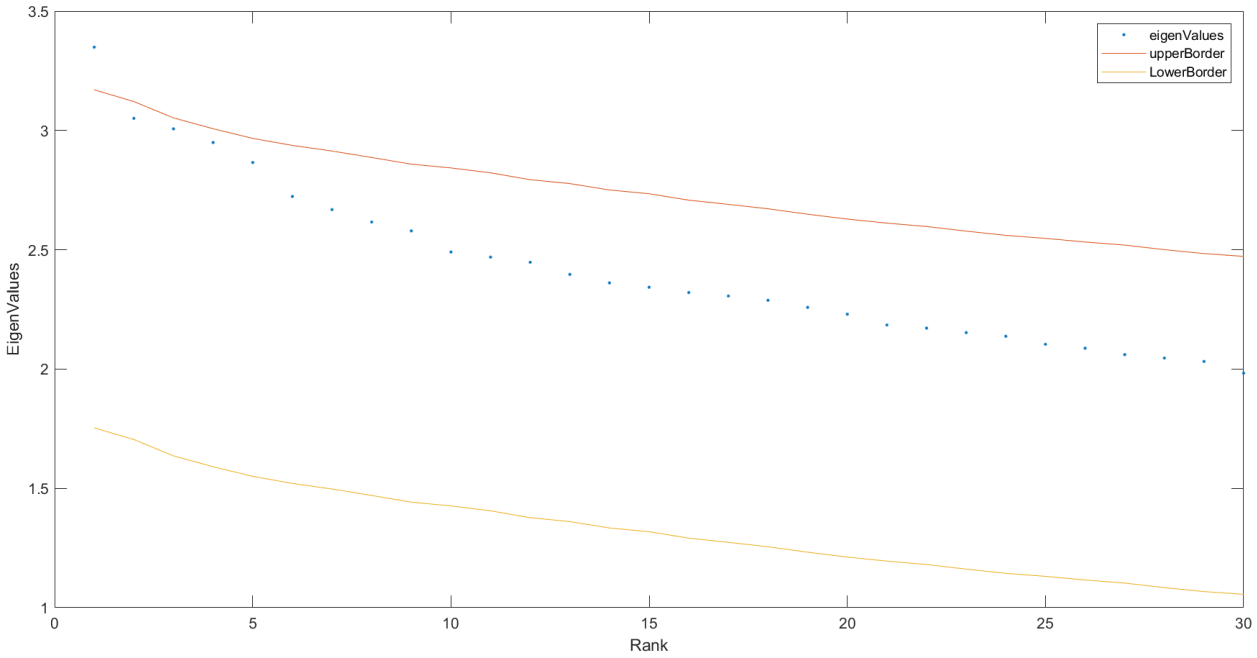
$$CI = \bar{x} \pm z\frac{s}{\sqrt{n}}$$

Which CI is our confidence intervals, $\bar{x}$ is the meanEigenControlVector, s is the standrad deviation meanEigenControlVector, n is the size of our sample which is 256 and z could be 5.2, 10.4 and 20.8 depend on the neuron you`ve choosed.(defualt values = 10.4).

For example, for the neuron “000412.a01” and the experiment, “a01emsq1D”, we have:(z=10.4)



As we can see, with the z = 10.4, 11 eigen-values are out of control confidence interval and specifies 11 relevent features. If we increase the z to 20.8 we get a very better result:



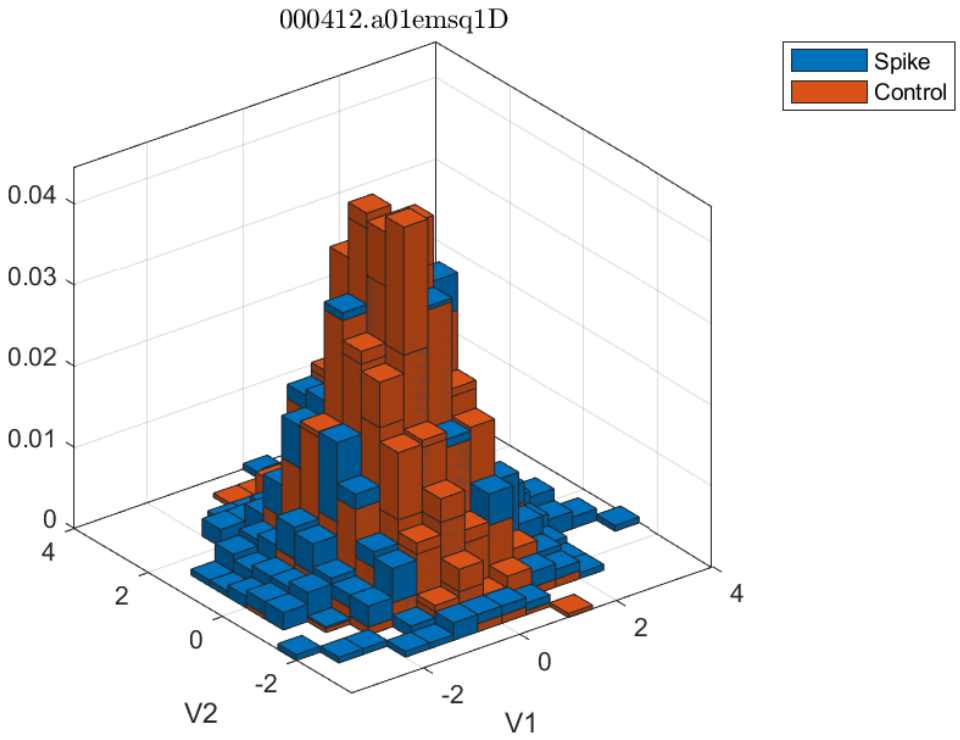
We can see that we have a eigen value outside of the control interval, so we have a relevent eigen vector(feature).

3. explainations are in question1 and 2.

4. To find images of the stimulus on the STC, we used inner product, for example:
this part of code calculates image of all spikes on biggest eigen-vector.

```
allStimulusImageOnEigenVector1 = normalStimulus*v1;
```

then we do these steps on spike-triggered inputs and all inputs on second biggest eigen-vector too.
at last we use `histogram2()` function to plot the 2-dimensional distribution of the images:



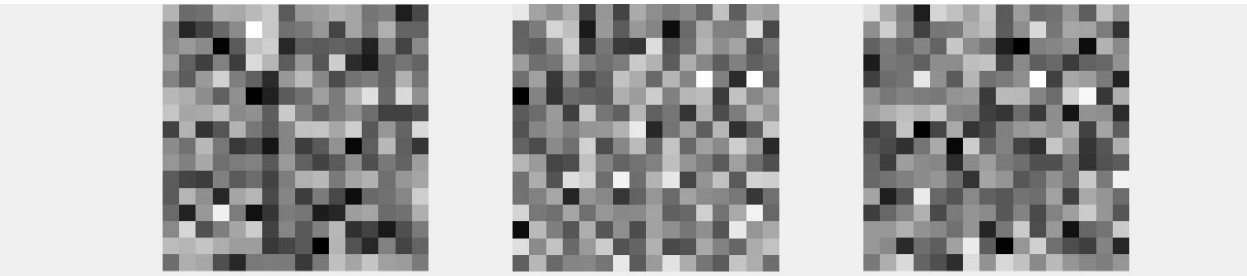
Just like part 3.3, we expected that the spike-triggered inputs have bigger images on the significant eigen-vectors (as it is shown in picture above)

Also it shows that the special set of inputs changed distribution of output significantly, therefore, size of image of the input on significant vector can be a measurement to see if the input is in receptive field of the neuron or not.

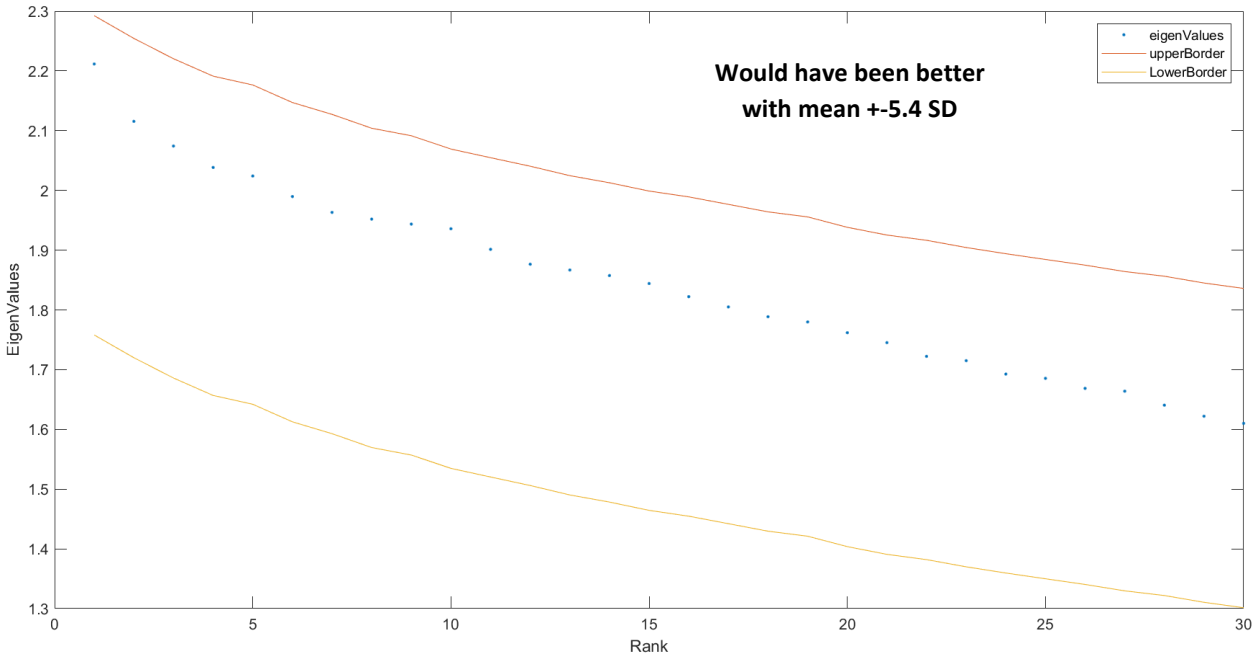
5.

6. Same 4 previous questions for neuron "000503.a03", and experiment, "a03cmsq1D":(V1,V2,V3)

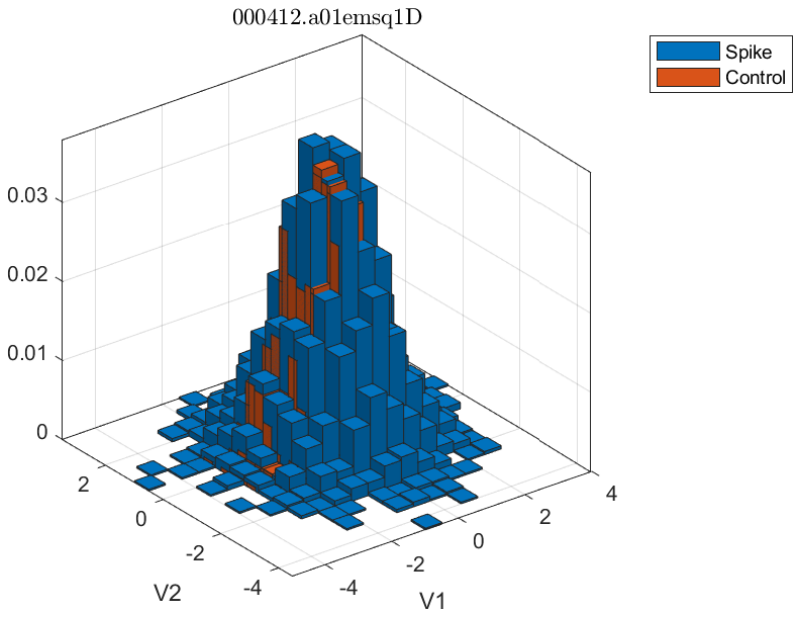
Question 1:



Question 2:

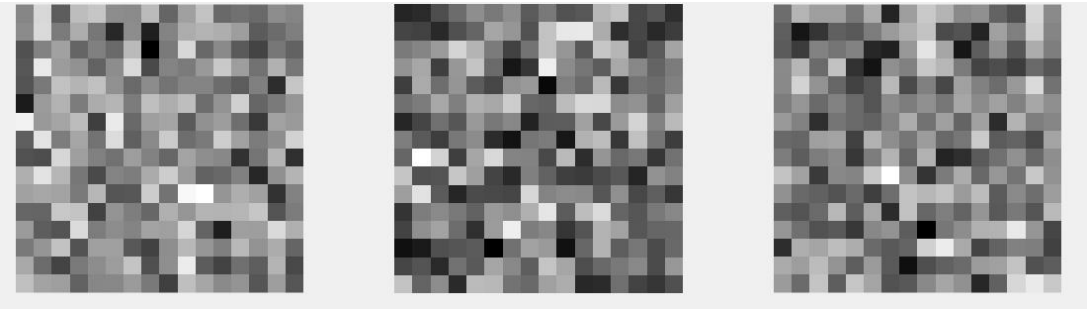


Question 4:

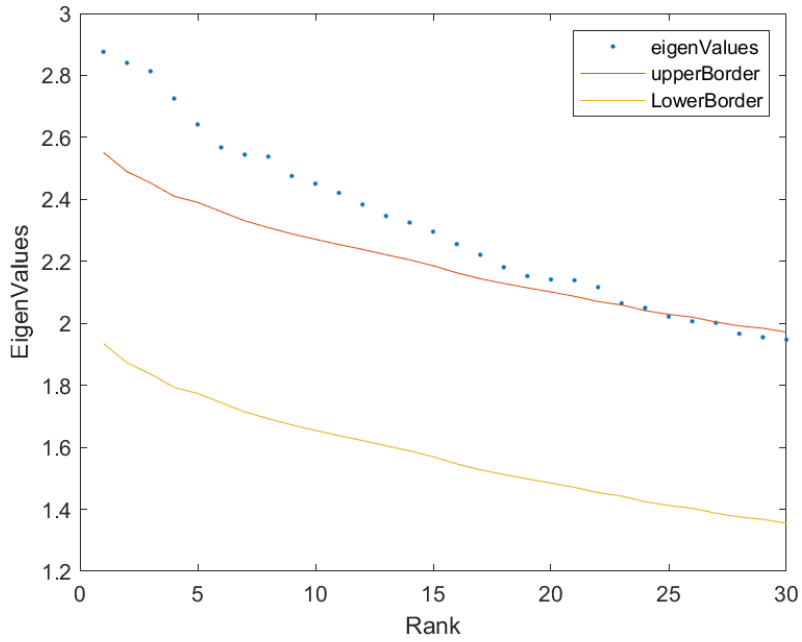


Same 5 previous questions for neuron “000926.a04”, and experiment, “a04bmsq1D”:

Question 1:

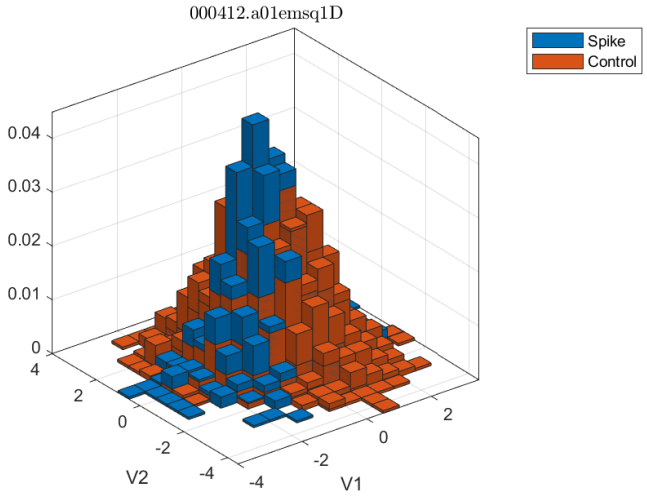


Question 2:



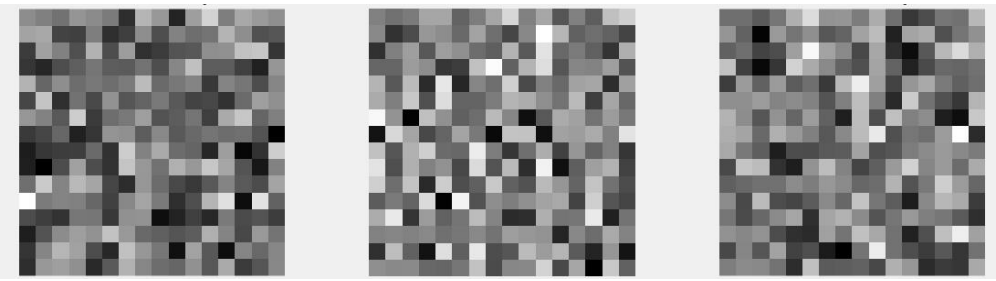
Would have been better with mean +- 20.8 SD

Question 4:

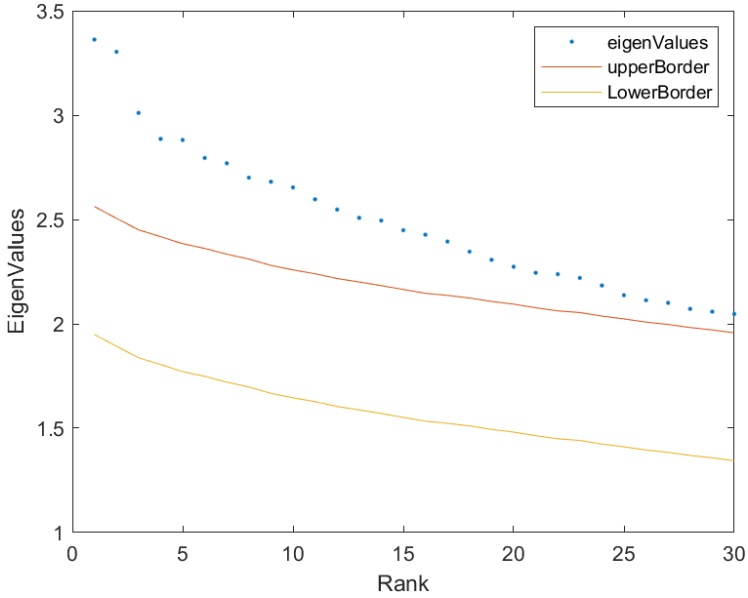


Same 5 previous questions for neuron “020308.A.d01”, and experiment, “A.d01emsq1D”:

Question 1:

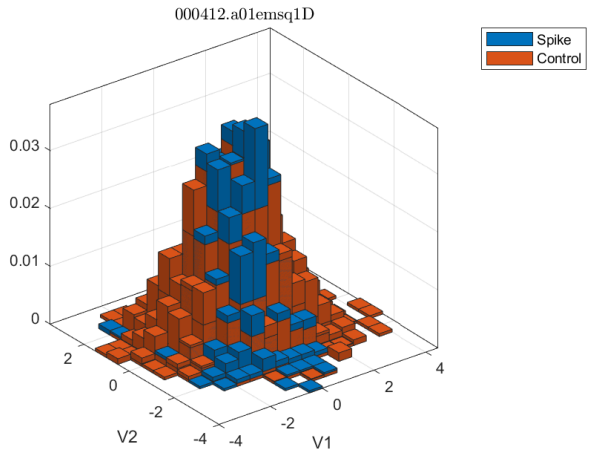


Question 2:



Would have been better with mean +- 20.8 SD

Question 4:



7. Here we choose a neuron with one significant eigen-value (and eigen-vector), therefore we expect to see a special pattern in this eigen-vector. Although the spatiotemporal structure of the non significant eigenvector appears to be random, the significant eigenvectors have a meaningful pattern. (picture is attached in part.4.1). Since the neuron is complex, STC method can characterize the neuron better than STA. Also the result of the STC is more meaningful than STA (which was just a random matrix)

- 5.Designing a question from anywhere of Neuroscience:

- *Principle Component Analysis*

PCA is very useful and common method in neuroscience. here we try to get familiar with some of its properties:
The main idea behind PCA method is to map a vector x in p dimensional space into a smaller vector z in a q dimensional space.(z approximates x)
here we only consider linear mappings:

$$z = Ex$$

which E is a p×q matrix.

our goal is to reconstruct the original vector using E^T (we don't do it here completely, because it would be so long!). So all PCA does is rotating the system to a new one.
reconstruction error formula:

$$\|\mathbf{x}^i - \hat{\mathbf{x}}^i\|^2 = \|\mathbf{x}^i - \mathbf{E}^T\mathbf{E}\mathbf{x}^i\|^2 = (\mathbf{x}^i - \mathbf{E}^T\mathbf{E}\mathbf{x}^i)^T(\mathbf{x}^i - \mathbf{E}^T\mathbf{E}\mathbf{x}^i)$$

1. Here we want to simplify this formula a little in a special case and show it in terms of E and covariance matrix C.
we know:
a) q = 1
b) ||E|| = 1

2. assume v_i i = 1, ..., p are eigen vectors of C, also we know E is a 1 by p vector, therefore E can be written as linear combinations of v_i^T s, with w_i i = 1, …, p coefficient.
what is the expression of total reconstruction error in terms of λ_i and w_i.(where λ_i is defined as: $\mathbf{C} = \sum_i \lambda_i \mathbf{v}_i \mathbf{v}_i^T$)

1.

- total error over the hole data :

$$\sum_{i=1}^{N} \|x^i - \hat{x}^i\|^2 = \sum_{i=1}^{N} (x^i - E^T E x^i)^T (x^i - E^T E x^i) \quad , \quad \underline{E E^T = 1 \ , \ E x = x^T E^T}$$

Simplification $\rightsquigarrow$ $\sum_{i=1}^{N} (x^i - E^T E x^i)^T (x^i - E^T E x^i) = \sum (x^T - x^T E^T E)(x - E^T E x)$

$$= \sum [x^T x - x^T E^T E x - x^T E^T E x + x^T E^T E E^T E x]$$

$$= \sum [x^T x - x^T E^T E x] = \sum [\|x\|^T - E x x^T E^T]$$

$$= N [\langle \|x\|^2 \rangle - E \langle \|x\|^2 \rangle E^T] = N [Tr C - E C E^T] \quad , \quad C \text{ is the covariance matrix}$$

$$, \ Tr C = \sum C_{ii} \ .$$

2.

- $C = \sum_i \lambda_i v_i v_i^T$ and $E = \sum_i w_i v_i^T$

$\Longrightarrow$ $Tr C = \sum_i \lambda_i Tr(v_i v_i^T) = \sum_i \lambda_i (v_i^T v_i)$, $E C E^T = \sum_{ijk} w_i v_i^T \lambda_j v_j v_j^T w_k v_k$

v_i orthonormal $\rightsquigarrow$ So , $v_i^T v_j = \delta_{ij}$ and $v_j^T v_k = \delta_{jk}$ $\Longrightarrow$ $Tr C = \sum_i \lambda_i$

,

$$E C E^T = \sum_i w_i^2 \lambda_i$$

total error $\rightsquigarrow$ $N [Tr C - E C E^T] = N [\sum_i \lambda_i - \sum_i w_i^2 \lambda_i] \xrightarrow{\|E\| = 1}$

$$1 = E E^T = \sum_{ij} w_i v_i^T w_j v_j = \sum_{ij} w_i \delta_{ij} w_j = \sum_i w_i^2$$