

# Spike Count Analysis

Spiking activity of neurons is the main focus of the [Steinmetz et al, 2019 paper](#). With the rich dataset that they collected during their experiments many analyses are possible that can link neural activity to behavioral or experimental variables.

This live script is dedicated to working with multi-dimensional data stored in matrix format and how to extract poignant information from large, complex datasets.

## Table of Contents

Slicing Matrices: Finding Neural Spikes.....	2
Subplots.....	5
Extracting Averages from Matrices: Neural Activity in Different Brain Areas.....	8
Averaging Spiking Counts.....	8
Most and Least Active Cells.....	10
Masking: Spike Count Analysis with Trial-level Categorical Variables.....	10

## Download Data

```
userpath(fullfile(fileparts(matlab.desktop.editor.getActiveFilename), "src"))
download_from_sciebo('https://uni-bonn.sciebo.de/s/3ZI1QhRQt7fBaLs', 'data/
2017-12-05_Lederberg.mat')
```

```
Downloading file to data/2017-12-05_Lederberg.mat
Done!
```

```
addpath(fullfile(fileparts(matlab.desktop.editor.getActiveFilename), "data"))
```

## Load in Data

```
data = load("data/2017-12-05_Lederberg.mat")
```

```
data = struct with fields:
    spike_counts: [250x340x698 int8]
    brain_areas: [698x1 string]
    trial: [340x1 int32]
    cell: [698x1 int32]
    time: [250x1 double]
    response_type: [340x1 double]
```

```
spike_counts = data.spike_counts;
brain_areas = data.brain_areas;
trials = data.trial;
cells = data.cell;
time = data.time;
response_types = data.response_type;
```

# Slicing Matrices: Finding Neural Spikes

Spiking activity in this dataset is stored as a 3-dimensional matrix.

Let's slice up this matrix to understand the dimensions of the experiment and explore neural activity.

Code	Description
<code>size(data)</code>	retrieves the shape, or dimensions of a data structure
<code>matrix(3:6)</code>	index the 3rd to 6th elements (inclusive) of a matrix
<code>matrix(4, :, :)</code>	index the 4th element along the first dimension of a matrix
<code>matrix(:, 2:8, :)</code>	index the 2nd to 8th (inclusive) elements along the 2nd dimension of a matrix
<code>matrix(:, :, [1,3,5])</code>	index the 1st, 3rd and 5th elements along the 3rd dimension of a matrix
<code>reshape(matrix, n_rows, n_cols)</code>	reshape a matrix to the dimensions
<code>image(data, 'CDataMapping', 'scaled')</code>	plot image data scaled to the data itself
<code>title("My Beautiful Plot")</code>	set the title of a plot
<code>clf</code>	clears the figure of any old plots

## Example Exercise

What are the dimensions of the data stored in the variable `trials`?

```
size(trials)
```

```
ans = 1x2
      340    1
```

## Exercises

What are the dimensions of the data stored in the variable `cells`?

```
size(cells)
```

```
ans = 1x2
      698    1
```

What are the dimensions of the data stored in the variable `time`?

```
size(time)
```

```
ans = 1x2
      250      1
```

What are the dimensions of the data stored in the variable **spike\_counts**?

```
size(spike_counts)
```

```
ans = 1x3
      250      340      698
```

Which experimental aspects do the dimensions of **spike\_counts** correspond to?

Fill in the blanks in the comments below with either time, trial or cell and the number of datapoints in each.

**Hint-** use the **size** function to help you.

```
% 1st dimension is time - 250 datapoints
% 2nd dimension is trial - 340 datapoints
% 3rd dimension is cell - 698 datapoints
```

What are the dimensions of the data in **brain\_areas** and what do they correspond do? Fill in the blanks below

```
size(brain_areas)
```

```
ans = 1x2
      698      1
```

```
% 1st dimension is cell - 698 datapoints
```

What brain area is cell 3 in?

```
brain_areas(3)
```

```
ans =
"visual cortex"
```

Find the spike counts over time and trials for cell 3. Name this result **spike\_counts\_cell\_3**

```
spike_counts_cell_3=spike_counts(:, :, 3)
```

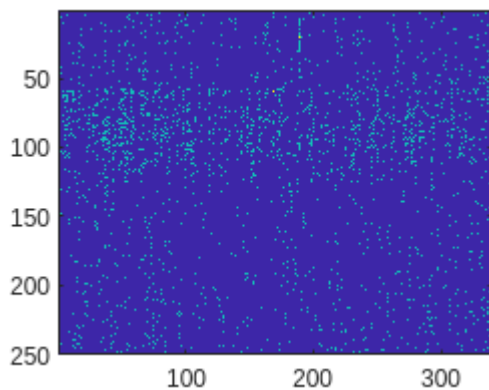
What are the dimensions of **spike\_counts\_cell\_3** ?

```
size(spike_counts_cell_3)
```

```
ans = 1x2
      250      340
```

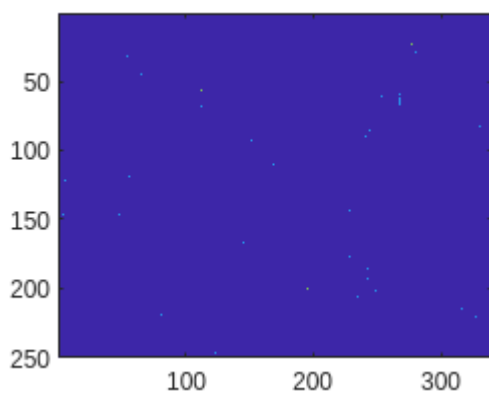
Plot a heatmap **spike\_counts\_cell\_3** and describe what you see in the data.

```
image(spike_counts_cell_3, 'CDataMapping','scaled')
```



Remake the above heatmap but for cell number 30. What differences do you notice and why?

```
dd=spike_counts(:, :, 30);  
image(dd, 'CDataMapping','scaled')
```



What area of the brain are the cells 29-31 located in?

```
brain_areas(29:31)
```

```
ans = 3x1 string  
"hippocam...  
"hippocam...  
"hippocam..."
```

Find the spike counts for the cells 29-31. Name this **chosen\_spikes**

```
chosen_spikes=spike_counts(:, :, 29:31);
```

What are the dimensions of **chosen\_spikes**?

```
size(chosen_spikes)
```

```
ans = 1x3
      250   340     3
```

## Subplots

Below is an example of how to use subplots.

```
subplot(3,1,1) %plot on the first subplot in a 3x1 grid
plot([1,2,3])

subplot(3,1,2) % change to the second subplot
plot([4,5,6])

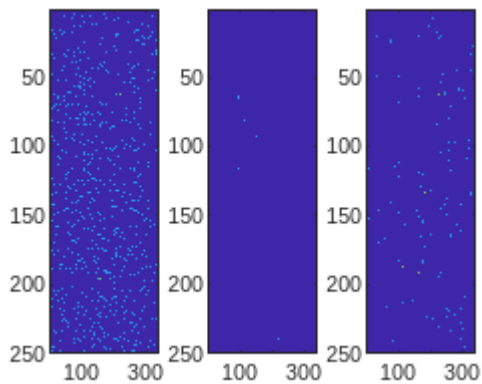
subplot(3,1,3) % 3rd subplot
plot([7,8,9])
```

There are 3 cells in **chosen\_spikes** , make a heatmap of each cell's spikes as an individual subplot

```
subplot(1,3,1)
image(chosen_spikes(:,:,1), 'CDataMapping','scaled')

subplot(1,3,2)
image(chosen_spikes(:,:,2), 'CDataMapping','scaled')

subplot(1,3,3)
image(chosen_spikes(:,:,3), 'CDataMapping','scaled')
```

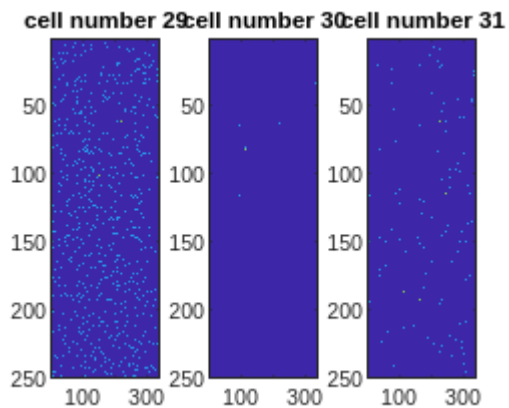


Add a title to each subplot showing the cell number

```
subplot(1,3,1)
image(chosen_spikes(:,:,1), 'CDataMapping','scaled')
title("cell number 29")

subplot(1,3,2)
image(chosen_spikes(:,:,2), 'CDataMapping','scaled')
title("cell number 30")
```

```
subplot(1,3,3)
image(chosen_spikes(:,:,3), 'CDataMapping','scaled')
title("cell number 31")
```



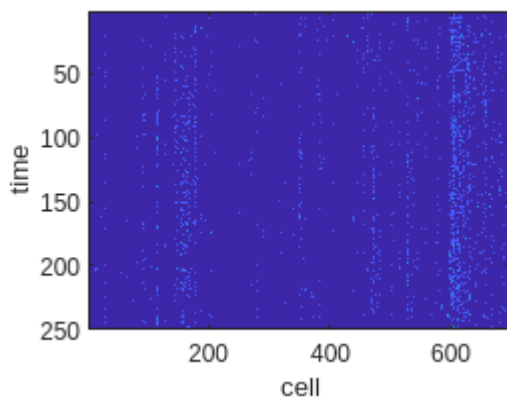
### Example Exercise

Find the spike data from `spike_counts` that represents trial number 99

```
spike_counts(:,99,:)
```

Reshape the data from trial number 99 into a 2D matrix and plot it as an image

```
dd=reshape(spike_counts(:,99,:), 250,[]);
clf
image(dd, 'CDataMapping','scaled')
ylabel('time')
xlabel('cell')
```



Select the data from trials 10 and 334 from `spike_counts`. Name this `spike_counts_chosen_trials`

```
spike_counts_chosen_trials=spike_counts(:,[10,334],:)
```

What are the dimensions of **spike\_counts\_chosen\_trials** ?

```
size(spike_counts_chosen_trials)
```

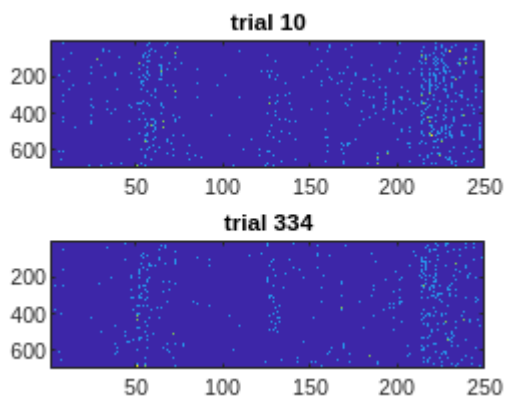
```
ans = 1x3  
      250      2      698
```

Using subplots, visualise the data in **spike\_counts\_chosen\_trials**

Below is an example of how to use subplots.

```
subplot(2,2,1) %plot on the first subplot in a 2x2 grid  
plot([1,2,3])  
  
subplot(2,2,2) % change to the second subplot  
plot([4,5,6])  
  
subplot(2,2,3) % 3rd subplot  
plot([7,8,9])
```

```
clf  
subplot(2,1,1)  
image(reshape(spike_counts_chosen_trials(:,1,:),698,[]),  
      'CDataMapping','scaled')  
title("trial 10")  
  
subplot(2,1,2)  
image(reshape(spike_counts_chosen_trials(:,2,:),698,[]),  
      'CDataMapping','scaled')  
title("trial 334")
```



Do you notice a difference between the three chosen trials? They are the trials with the most, closest to average and least number of spikes respectively! But shown like this, we can hardly tell them apart.

It's time to analyse the spike data in more depth!

## Extracting Averages from Matrices: Neural Activity in Different Brain Areas

Averages are a helpful way of reducing the dimensionality of data to extract essential statistics from large and complex datasets.

In this section we will be averaging our spike counts matrix to investigate which brain areas are the most active.

<u>Code</u>	<u>Description</u>
<code>mean(matrix, "all")</code>	calculate the mean over all elements in a matrix
<code>mean(matrix, 2)</code>	calculate the mean over the 2nd dimension of a matrix
<code>mean(matrix, [1,3])</code>	calculate the mean over the 1st and 3rd dimensions of a matrix
<code>unique(data)</code>	find unique elements in <b>data</b>
<code>histogram(data, "DisplayName", "label")</code>	plot a histogram with a given legend label
<code>legend</code>	display legend
<code>reshape(matrix, 1, [])</code>	reshape a matrix into an array
<code>[max_val, index] = max(data)</code>	find the maximum value and its index in <b>data</b>

### Example Exercise

Calculate the mean spike count per time bin. ie. over the 1st dimension

```
mean(spike_counts, 1)
```

### Exercises

#### Averaging Spiking Counts

What is the mean spike count in all time bins for all cells over all trials?

```
mean(spike_counts, "all")
```

```
ans = 0.0406
```

What is the mean spike count over time and trials?

```
mean(spike_counts, 3)
```

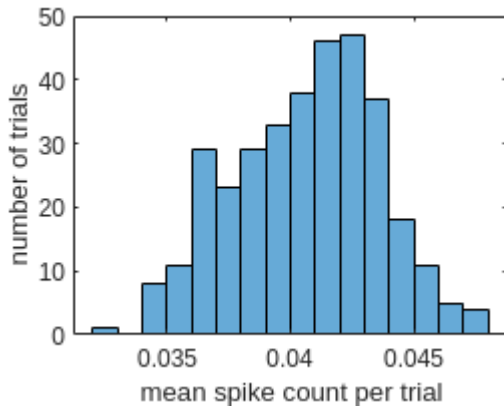
What is the mean spike count over cells?



```
mean(spike_counts,[1,2])
```

Plot the histogram of mean spike count per trial as a histogram

```
clf
histogram(mean(spike_counts, [1,3]))
xlabel('mean spike count per trial')
ylabel('number of trials')
```



What is the mean spike count per cell? Name the result `mean_per_cell`

```
mean_per_cell=mean(spike_counts, [1,2]);
```

What dimensions does `mean_time_trials` have?

```
size(mean_per_cell)
```

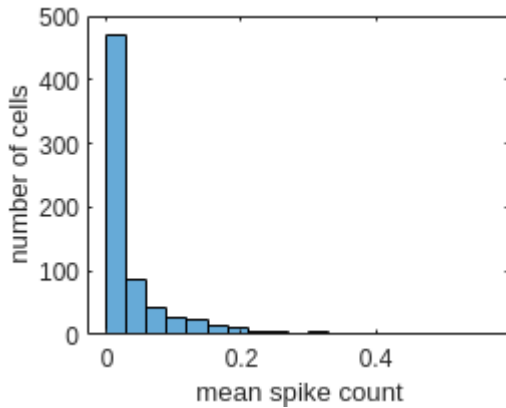
```
ans = 1x3
      1      1    698
```

Index all of the elements in `mean_time_trials` with `(:)` - what is the shape of the result?

```
mean_per_cell(:)
```

Visualise `mean_time_trials` as a histogram. **Hint** - first you will need to reshape the matrix

```
clf
mean_array = mean_per_cell(:);
histogram(mean_array)
xlabel("mean spike count")
ylabel("number of cells")
```



## Most and Least Active Cells

Which cell has the largest mean spike count over all trials and time bins?

```
[max_mean_spike_count, cell_num] = max(mean_array)
```

```
max_mean_spike_count = 0.5537
cell_num = 604
```

Which brain area does this cell belong to?

```
brain_areas(cell_num)
```

```
ans =
"thalamus"
```

Which brain area hosts the cell that spiked the least?

```
[min_mean_spike_count, cell_num] = min(mean_array)
```

```
min_mean_spike_count = 0
cell_num = 6
```

```
brain_areas(cell_num)
```

```
ans =
"visual cortex"
```

## Masking: Spike Count Analysis with Trial-level Categorical Variables

As we have seen before, in addition to brain area, we also have access to several categorical variables such as response type and stimulus contrast level. Let's see whether the pattern of neural activity would change for different values of these categorical variables.

Code	Description
<code>my_matrix(:, :, array=="value")</code>	apply a mask to a matrix, where <b>array</b> has a specified <b>value</b>

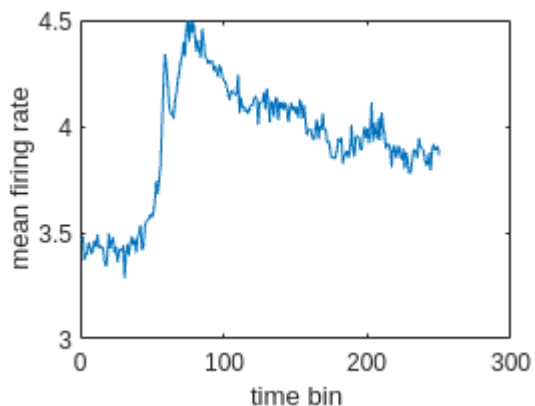
<code>plot(data, "DisplayName", "label")</code>	plot data with a given label in the legend
<code>legend</code>	show plot legend
<code>title("My Beautiful Plot")</code>	set the title of a plot
<code>clf</code>	clears the figure of any old plots

Let's start by turning the spike counts into firing rates.

```
dt = 0.01;
firing_rate = spike_counts/dt;
```

**Example Exercise:** Plot the mean firing rate (averaged across neurons and trials) over time, taking all neurons.

```
plot(mean(firing_rate, [2,3]))
xlabel('time bin')
ylabel('mean firing rate')
```

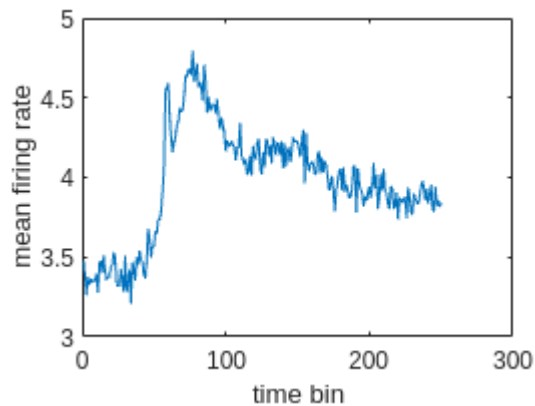


## Exercises

Plot the mean firing rate only for trials where the response type is -1. **Hint** - use the `response_types` variable

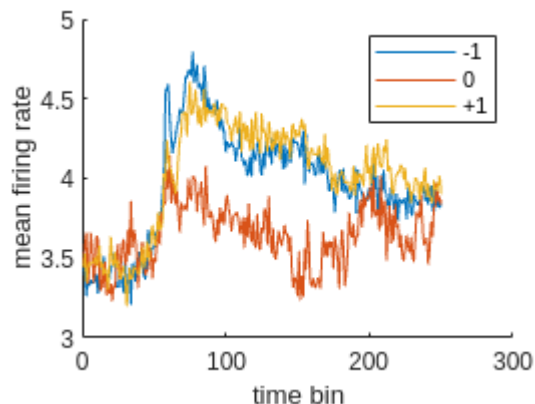
```
firing_rate_response_neg_1 = firing_rate(:,response_types==-1,:);
plot(mean(firing_rate_response_neg_1,[2,3]))

xlabel('time bin')
ylabel('mean firing rate')
```



Add to the previous plot the other two response types: one line for each response type (-1,0,+1). Did the population activity change between the response types?

```
clf
hold on
plot(mean(firing_rate(:,response_types==-1,:),[2,3]), "DisplayName","-1")
plot(mean(firing_rate(:,response_types==0,:),[2,3]), "DisplayName","0")
plot(mean(firing_rate(:,response_types==1,:),[2,3]), "DisplayName","+1")
hold off
xlabel('time bin')
ylabel('mean firing rate')
legend
```



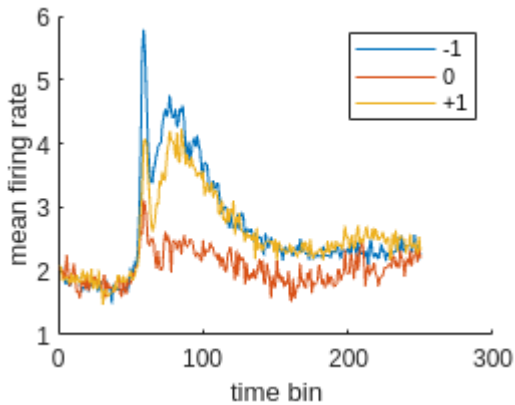
Make the same plot as the previous exercise, but only for neurons in the visual cortex. What do you notice?

```
clf
hold on
plot(mean(firing_rate(:,response_types==-1,brain_areas=="visual cortex"),[2,3]), "DisplayName","-1")
plot(mean(firing_rate(:,response_types==0,brain_areas=="visual cortex"),[2,3]), "DisplayName","0")
```

```

plot(mean(firing_rate(:,response_types==1,brain_areas=="visual cortex"),
[2,3]), "DisplayName", "+1")
hold off
xlabel('time bin')
ylabel('mean firing rate')
legend

```



Using subplots, create four plots in a single figure, each plotting showing the the analysis in the previous exercise for a different brain area. Specify the brain area in the title.

Below is an example of how to use subplots.

```

subplot(3,1,1) %plot on the first subplot in a 3x1 grid
plot([1,2,3])

subplot(3,1,2) % change to the second subplot
plot([4,5,6])

subplot(3,1,3) % 3rd subplot
plot([7,8,9])

```

```

clf
subplot(2,2,1)
region = "visual cortex";
hold on
plot(mean(firing_rate(:,response_types==-1,brain_areas==region),[2,3]),
"DisplayName", "-1")
plot(mean(firing_rate(:,response_types==0,brain_areas==region),[2,3]),
"DisplayName", "0")
plot(mean(firing_rate(:,response_types==1,brain_areas==region),[2,3]),
"DisplayName", "+1")
title(region)
hold off

subplot(2,2,2)
region = "thalamus";
hold on

```

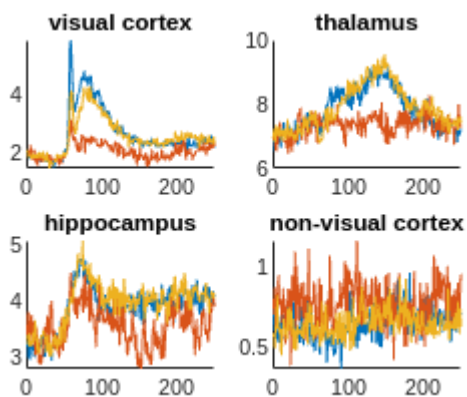
```

plot(mean(firing_rate(:,response_types==-1,brain_areas==region),[2,3]),
      "DisplayName", "-1")
plot(mean(firing_rate(:,response_types==0,brain_areas==region),[2,3]),
      "DisplayName", "0")
plot(mean(firing_rate(:,response_types==1,brain_areas==region),[2,3]),
      "DisplayName", "+1")
title(region)
hold off

subplot(2,2,3)
region = "hippocampus";
hold on
plot(mean(firing_rate(:,response_types==-1,brain_areas==region),[2,3]),
      "DisplayName", "-1")
plot(mean(firing_rate(:,response_types==0,brain_areas==region),[2,3]),
      "DisplayName", "0")
plot(mean(firing_rate(:,response_types==1,brain_areas==region),[2,3]),
      "DisplayName", "+1")
title(region)
hold off

subplot(2,2,4)
region = "non-visual cortex";
hold on
plot(mean(firing_rate(:,response_types==-1,brain_areas==region),[2,3]),
      "DisplayName", "-1")
plot(mean(firing_rate(:,response_types==0,brain_areas==region),[2,3]),
      "DisplayName", "0")
plot(mean(firing_rate(:,response_types==1,brain_areas==region),[2,3]),
      "DisplayName", "+1")
title(region)
hold off

```

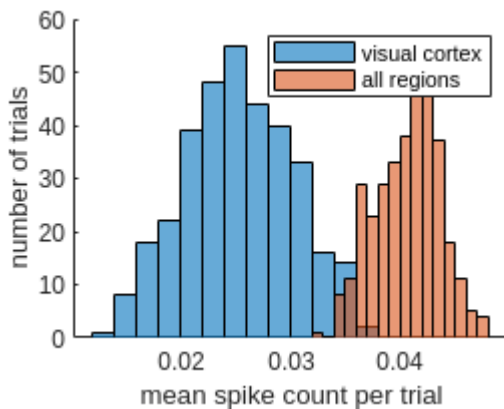


Plot a histogram showing the mean spike count per trial for all cells and for cells only in the visual cortex. **Hint-** use `hold on...` `hold off` to display multiple histograms on the same plot

```

visual_cortex_spike_count = spike_counts(:, :, brain_areas=="visual cortex");
clf
hold on
histogram(mean(visual_cortex_spike_count, [1,3]), "DisplayName", "visual
cortex")
histogram(mean(spike_counts, [1,3]), "DisplayName", "all regions")
hold off
legend
xlabel('mean spike count per trial')
ylabel('number of trials')

```



Remake the above plot, plotting the spike counts in the visual cortex and another brain area or areas of your choice.

Feel free to plot as many areas as you want!

```

visual_cortex_spike_count = spike_counts(:, :, brain_areas=="visual cortex");
thalamus_spike_count = spike_counts(:, :, brain_areas=="thalamus");
clf
hold on
histogram(mean(visual_cortex_spike_count, [1,3]), "DisplayName", "visual
cortex")
histogram(mean(thalamus_spike_count, [1,3]), "DisplayName", "thalamus")
hold off
legend
xlabel('mean spike count per trial')
ylabel('number of trials')

```

