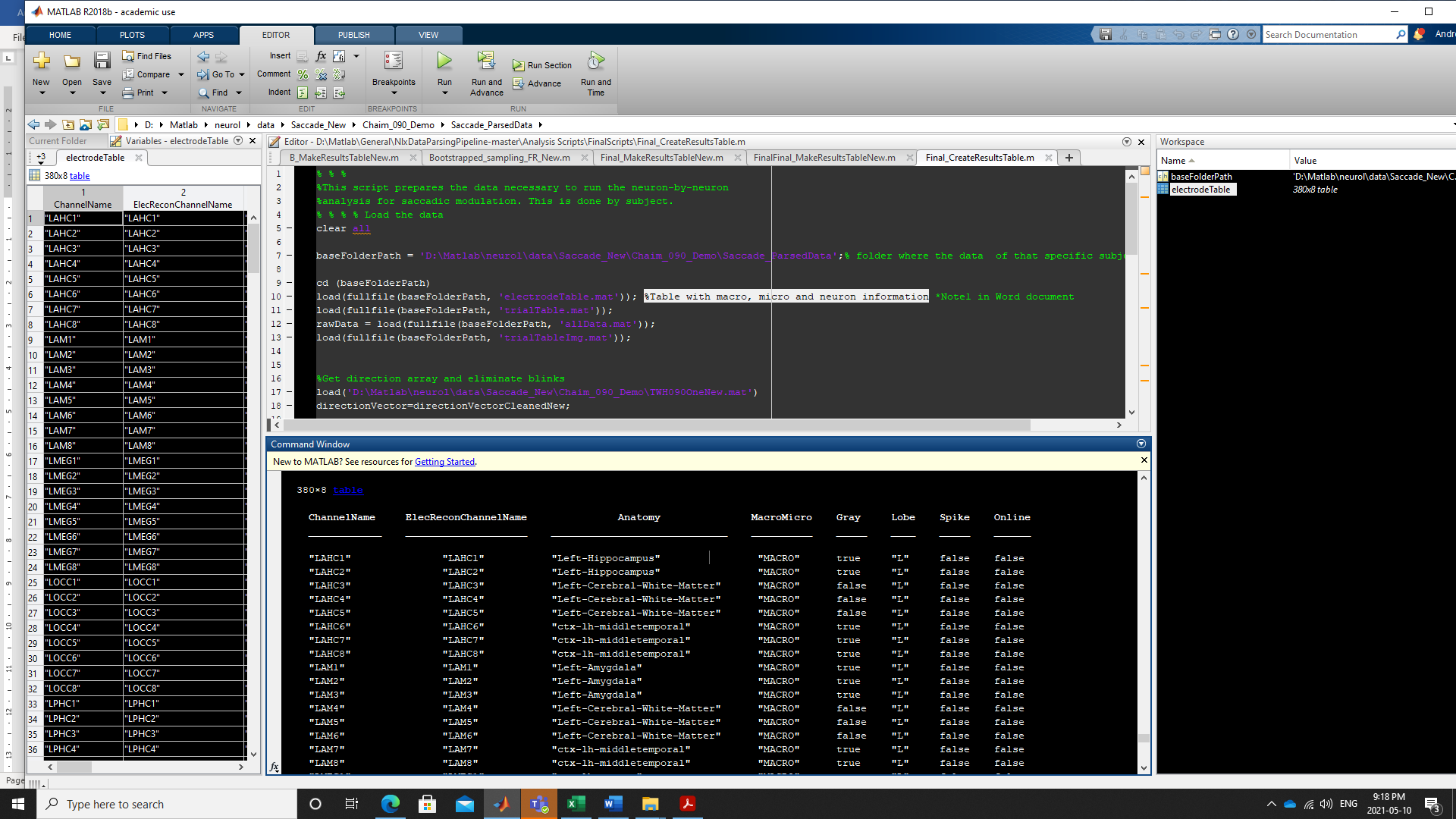
1. **Notes for Final\_CreateResultsTable**

(1st script to run after running the Parsing Pipeline).

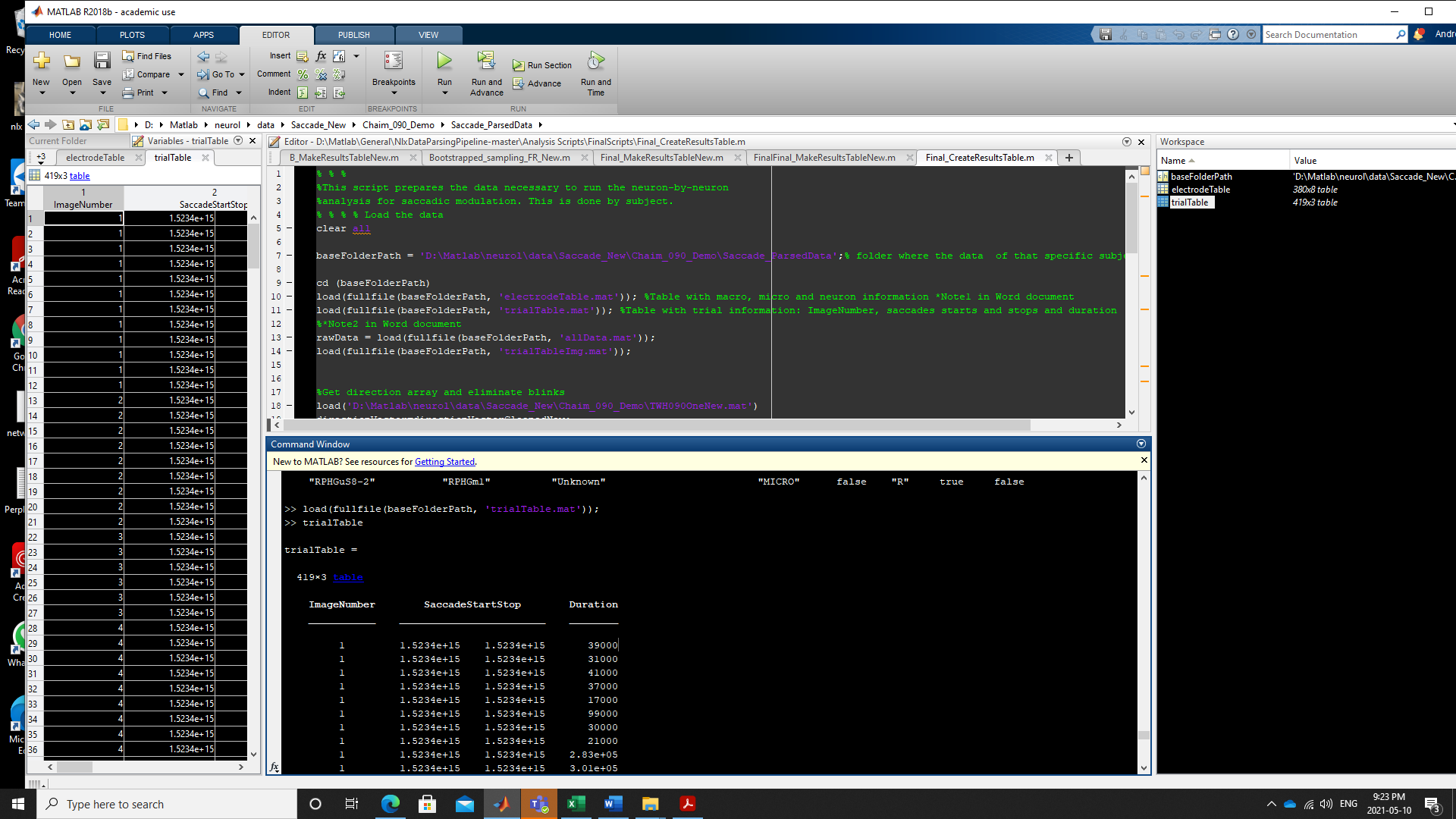
Note1.1: line 10

electrodeTable.mat: %Table with macro, micro and neuron information



Note 1.2: Line 11

trialTable.mat: %Table with trial information: ImageNumber, saccades starts and stops and duration

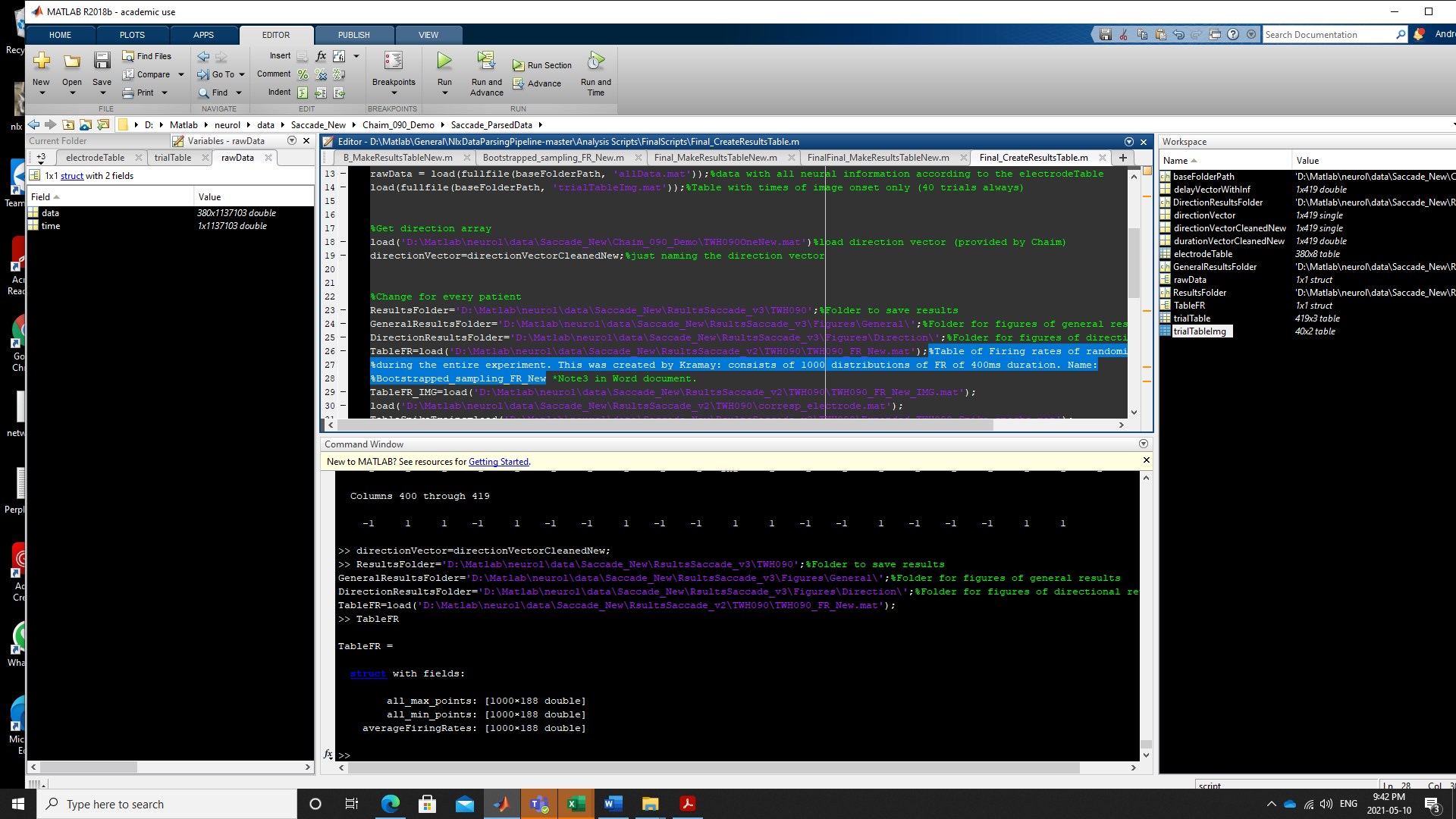


Note 1.3: Line 26

%Table of Firing rates of randomized periods

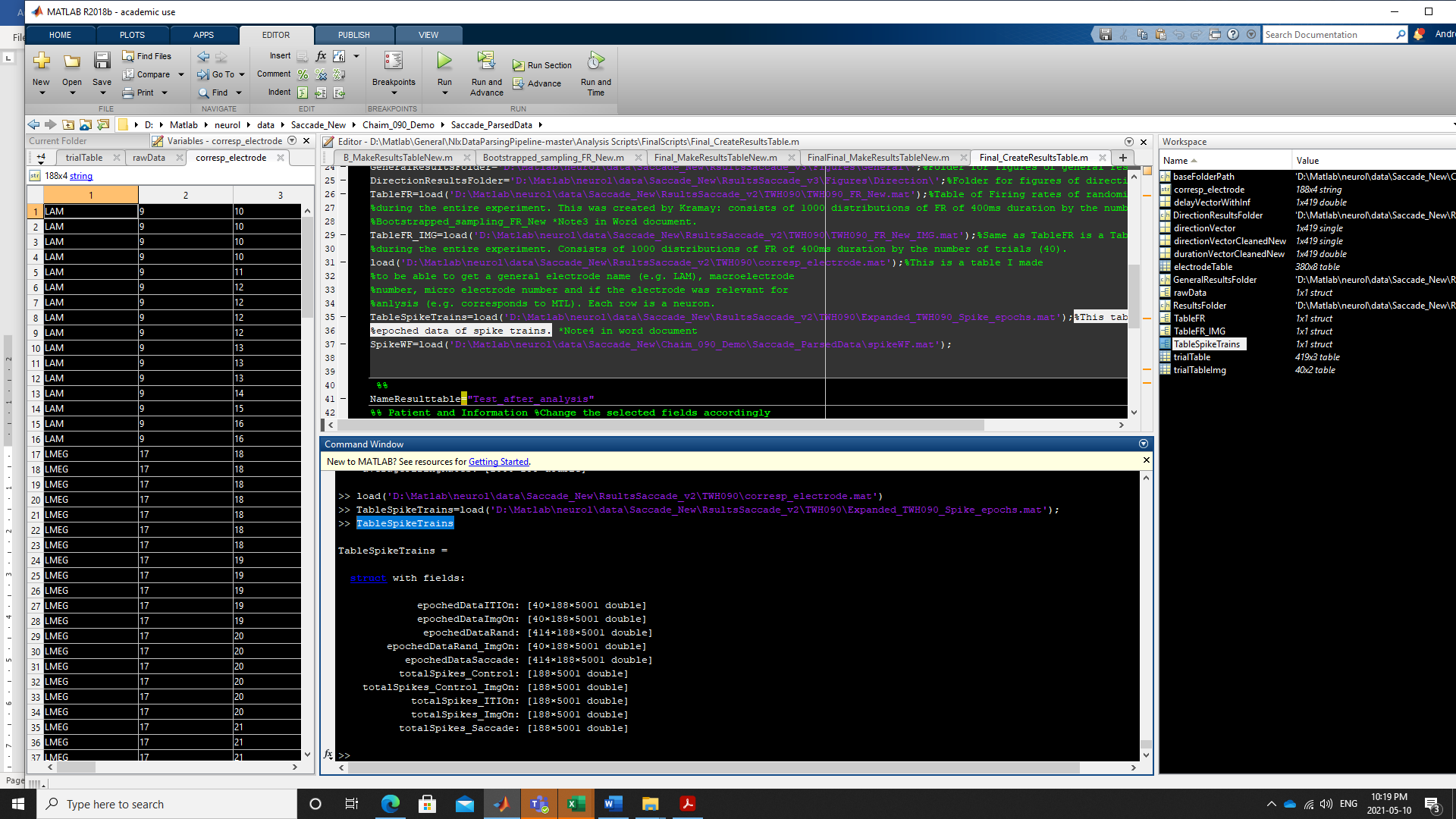
%during the entire experiment. This was created by Kramay: consists distributions of FR of 400ms duration (‘averageFiringRates’). Name:

%Bootstrapped\_sampling\_FR\_New (Included in folder). It also includes distributions of 400ms aligned to Minimum and Maximum values for rebound analysis (‘all\_max\_points’,’all\_min\_points’). Dimensions are iterations (1000) by Neurons in that data set (e.g. 188).



Note 1.4: Line 35

TableSpikeTrains: %This table has epoched data of spike trains.



Where…

epochedDataITIOn: spike trains aligned to ITI onset

epochedDataITIOn: spike trains aligned to Image Onset

\*\*epochedDataRand: spike trains aligned to randomized points during experiment. Similar to what we created with the Bootstrap scripts, but in this case we saved the spike trains and not only the distributions.

\*\*epochedDataRand\_ImgOn: same as epochedDataRand but only for 40 trials as control for ImgOnset.

epochedDataSaccade: spike trains aligned to saccade onset.

The totalSpikes… fields: are the same but already squeezed to make histograms

\*\*Not really used to define a modulation but used for quality control and visualization.

1. **Final\_Final\_MakeResultsTable\_New**

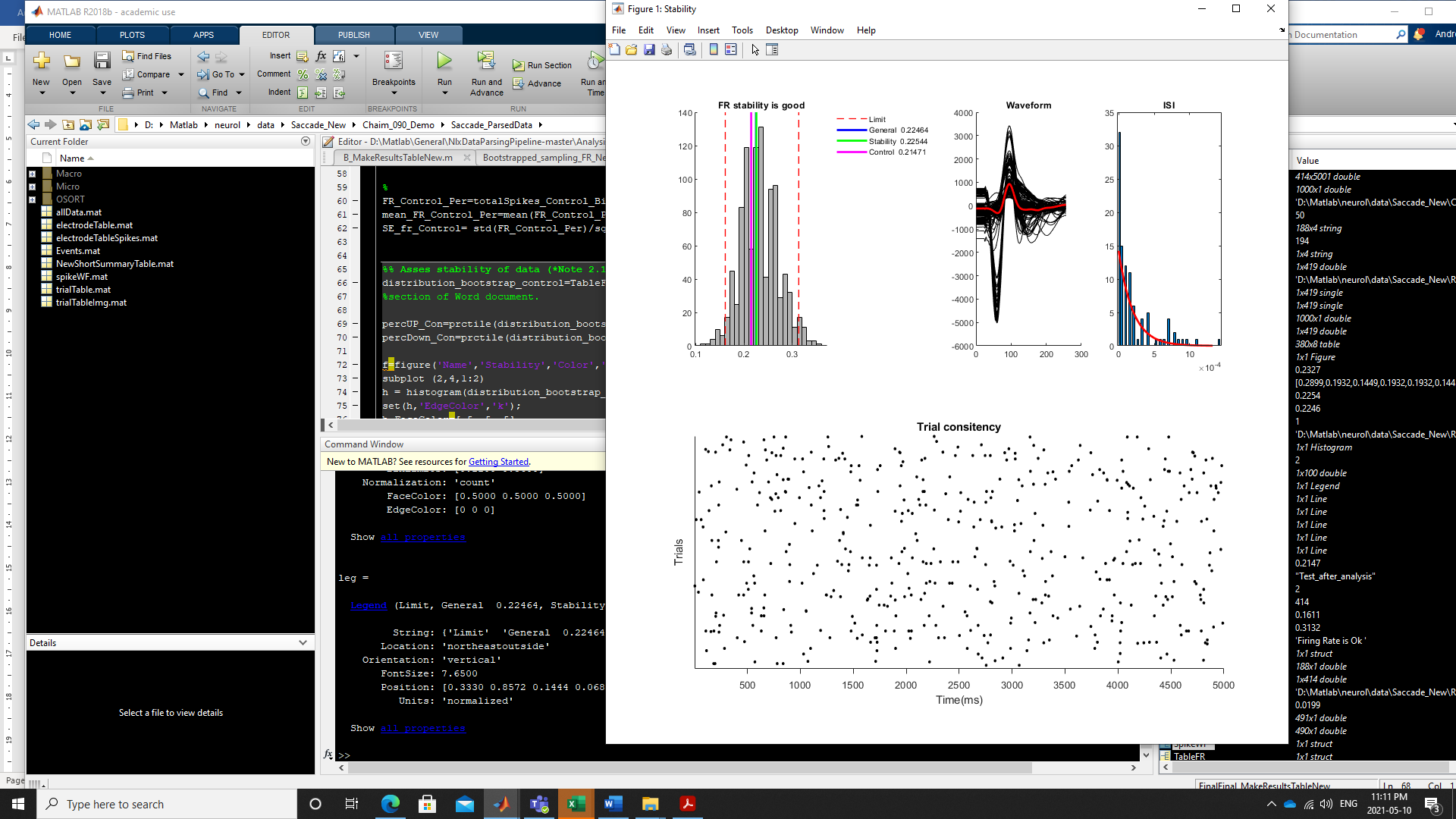
**2.1 Stability assesment**

**For all neurons we assessed the stability of the trials, the general waveform and ISI, to evaluate if it was a neuron worth of analyzing or if it needed to be discarded.**

**We plotted the distribution\_bootstrap\_control (see ‘averageFiringRates’ in previous lines), the mean of that distribution (“General” blue line in plot), the mean of** epochedDataRand (spike trains aligned to randomized points during experiment. Similar to what we created with the Bootstrap scripts, but in this case we saved the spike trains and not only the distributions) aligned to different periods to evaluate stability.

We also plotted the waveforms (black), compared to the specific neuron we are evaluating (red).

If the three lines had similar values it was considered to be a neuron with good stability and the analysis continued.



**\***\*Note 2.2 Line 148

Calculations of mean firing rate in saccade periods

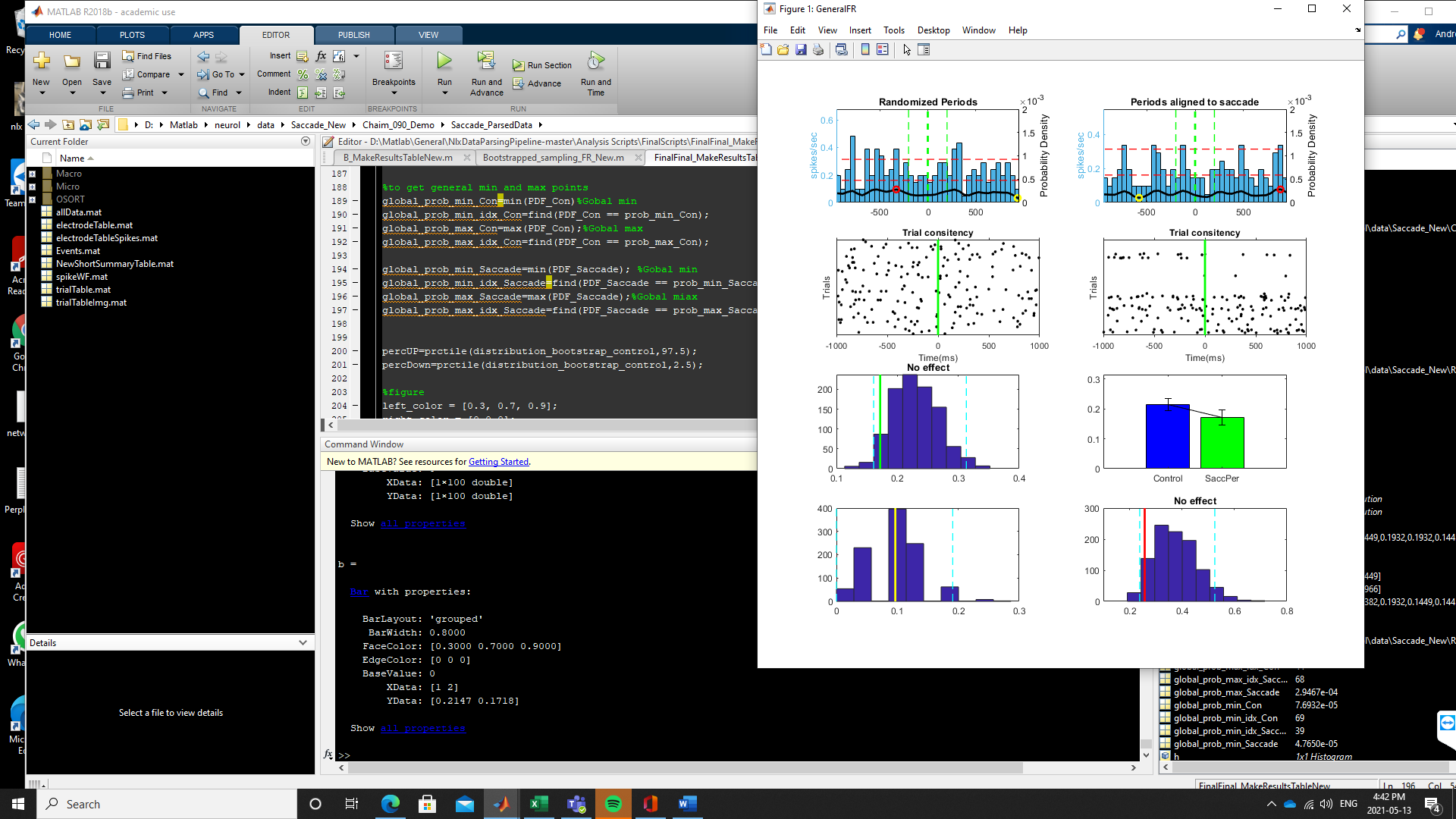
FR\_Saccade\_Per=totalSpikes\_Saccade\_Binned\_persec(46:54); %%%%%calculations of spike train during the saccade period%%%%%%%%%

mean\_FR\_Saccade\_Per=mean(FR\_Saccade\_Per);%mean firing rate during

**\*Note 2.3. Line 153 General effects**

**A neuron showing no effects, but not very good stability, see raster on saccade periods**

**Control (for visualization)**

**Saccade periods were unstable (neuron was rated as lower confidence but taken for analysis and considered to be not modulated).**

**This graph was used as final decision to count it a neuron as modulated or not modulated**

**Analysis of rebound effects (with minimum and maximum values). For more info see below.**

\*Note 2.4 Line 163 Probability Distribution

For the probability density I used the matlab PDF function

<https://www.mathworks.com/help/stats/working-with-probability-distributions.html>

This was taken from different sources:

<https://www.fieldtriptoolbox.org/tutorial/spike/> (although I did not use fieldtrip I did use this for principles of spike analysis)

[t, colSpike\_Saccade] = find(allSpikeTrains\_Saccade);% trials by time

timeStamps\_ms\_Saccade = colSpike\_Saccade-halfTrialTime;% find timestamps where neuron spiked

fitDist\_Saccade = fitdist(timeStamps\_ms\_Saccade,'Kernel','BandWidth',50)% to do de Probability Densitity you need the fitdist first

PDF\_Saccade = pdf(fitDist\_Saccade,x);%struct done

\*Note 2.5 Line 311 Rebound effect

For the rebound effect we used the Bootstrap\_sampling script (created with Kramay) that creates the distributions of the randomized periods. In that script we detected min and maximum points and created distributions of 400ms around those periods.

for iteration = 1:1000

randomTimeIdx = round((maxTime-minTime).\*rand(NumberofSaccades,1) + minTime);

%r\_range = [min(randomTimeIdx) max(randomTimeIdx)]

randomTime=rawData.time(randomTimeIdx);

%Epoch Data

epochedDataRand=createEpochsWithDuration(rawData.data, rawData.time, randomTime, time\_bef\_cont,...

time\_aft\_cont, relevantElectrodeIndexSpike);

totalSpikes = squeeze(sum(epochedDataRand, 1));

for i = 1:size(totalSpikes, 1)

totalSpikesBinned(i,:) = histcounts(find(totalSpikes(i,:)), 0:50:400);

end

[max\_point, max\_point\_idx] = (max(totalSpikesBinned, [],2));

all\_max\_points(iteration, :) = max\_point'./NumberofSaccades;

% all\_max\_points\_idx(iteration, :) = max\_point\_idx'.\*25;

[min\_point, min\_point\_idx] = (min(totalSpikesBinned, [],2));

all\_min\_points(iteration, :) = min\_point'./NumberofSaccades;

averageFiringRates(iteration, :) = sum(totalSpikesBinned, 2)./NumberofSaccades;

toc

end

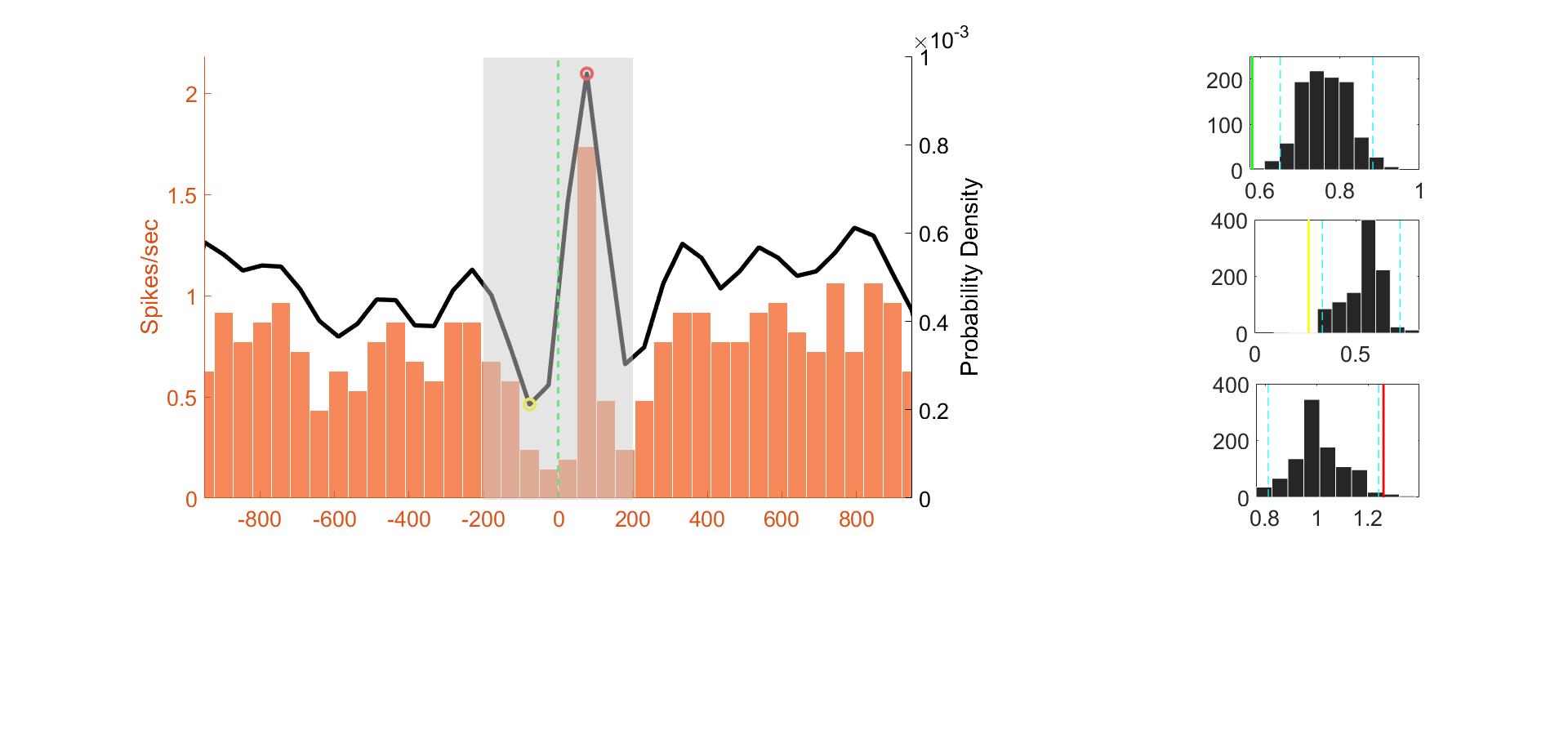
cd(ResultsFolder)

save(NameFRtable,'all\_max\_points','all\_min\_points','averageFiringRates')

After that I ran the script that evaluates effects neuron by neuron.

* In the Rebound part I plot “all\_max\_points” and “all\_min\_points” and calculate the significance (percentiles).
* I create the spike trains for the saccade periods
* I calculate the min and max points and create spike trains around those periods
* Then I calculate the average of those spike trains and compare it with the control periods (“all\_max\_points”/ “all\_min\_points”)

Average of spike trains created around min points (yellow dot)



Average of spike trains created around max points (red dot)

all\_max\_points

all\_min\_points

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Evaluate if there is a rebound effect

FR\_Min\_Per=totalSpikes\_Saccade\_Binned\_persec(prob\_min\_idx\_Saccade-1:prob\_min\_idx\_Saccade+1);

mean\_FR\_Min\_Per=mean(FR\_Min\_Per);

SE\_fr\_Min=std(FR\_Min\_Per)/sqrt(length(FR\_Min\_Per));

FR\_Max\_Per=totalSpikes\_Saccade\_Binned\_persec(prob\_max\_idx\_Saccade-1:prob\_max\_idx\_Saccade+1);

mean\_FR\_Max\_Per=mean(FR\_Max\_Per);

SE\_fr\_Max=std(FR\_Max\_Per)/sqrt(length(FR\_Max\_Per));

distribution\_bootstrap\_Min=TableFR.all\_min\_points(:,Neuron)/.05;

mean\_min=mean(distribution\_bootstrap\_Min);

SE\_min= std(distribution\_bootstrap\_Min)/sqrt(length(distribution\_bootstrap\_Min));

percUP\_Min=prctile(distribution\_bootstrap\_Min,97.5);

percDown\_Min=prctile(distribution\_bootstrap\_Min,2.5);

distribution\_bootstrap\_max=TableFR.all\_max\_points(:,Neuron)/.05;

mean\_max=mean(distribution\_bootstrap\_max);

SE\_max= std(distribution\_bootstrap\_max)/sqrt(length(distribution\_bootstrap\_max));

percUP\_Max=prctile(distribution\_bootstrap\_max,97.5);

percDown\_Max=prctile(distribution\_bootstrap\_max,2.5);

subplot (4,2,7)%MIN

hist(distribution\_bootstrap\_Min)

hold on;

line1 = lineplot(percUP\_Min, 'v', '--c', 'LineWidth', 1);

line2 = lineplot(percDown\_Min, 'v', '--c', 'LineWidth', 1);

lineFR1=lineplot(mean\_FR\_Min\_Per, 'v', 'y', 'LineWidth', 2);

if mean\_FR\_Min\_Per < percDown\_Min

title ('Significant Decrease');

elseif mean\_FR\_Min\_Per > percUP\_Min

title ('Significant Increase');

elseif mean\_FR\_Min\_Per >= percDown && mean\_FR\_Min\_Per <= percUP\_Min

title ('No effect')

end

subplot (4,2,8)%MAX

hist(distribution\_bootstrap\_max)

hold on;

line1 = lineplot(percUP\_Max, 'v', '--c', 'LineWidth', 1);

line2 = lineplot(percDown\_Max, 'v', '--c', 'LineWidth', 1);

lineFR1=lineplot(mean\_FR\_Max\_Per, 'v', 'r', 'LineWidth', 2);

if mean\_FR\_Max\_Per < percDown\_Max %%%%%%these lines was used to Classify as Rebound or not%%%%%%

title ('Significant Decrease');

elseif mean\_FR\_Max\_Per > percUP\_Max

title ('Significant Increase');

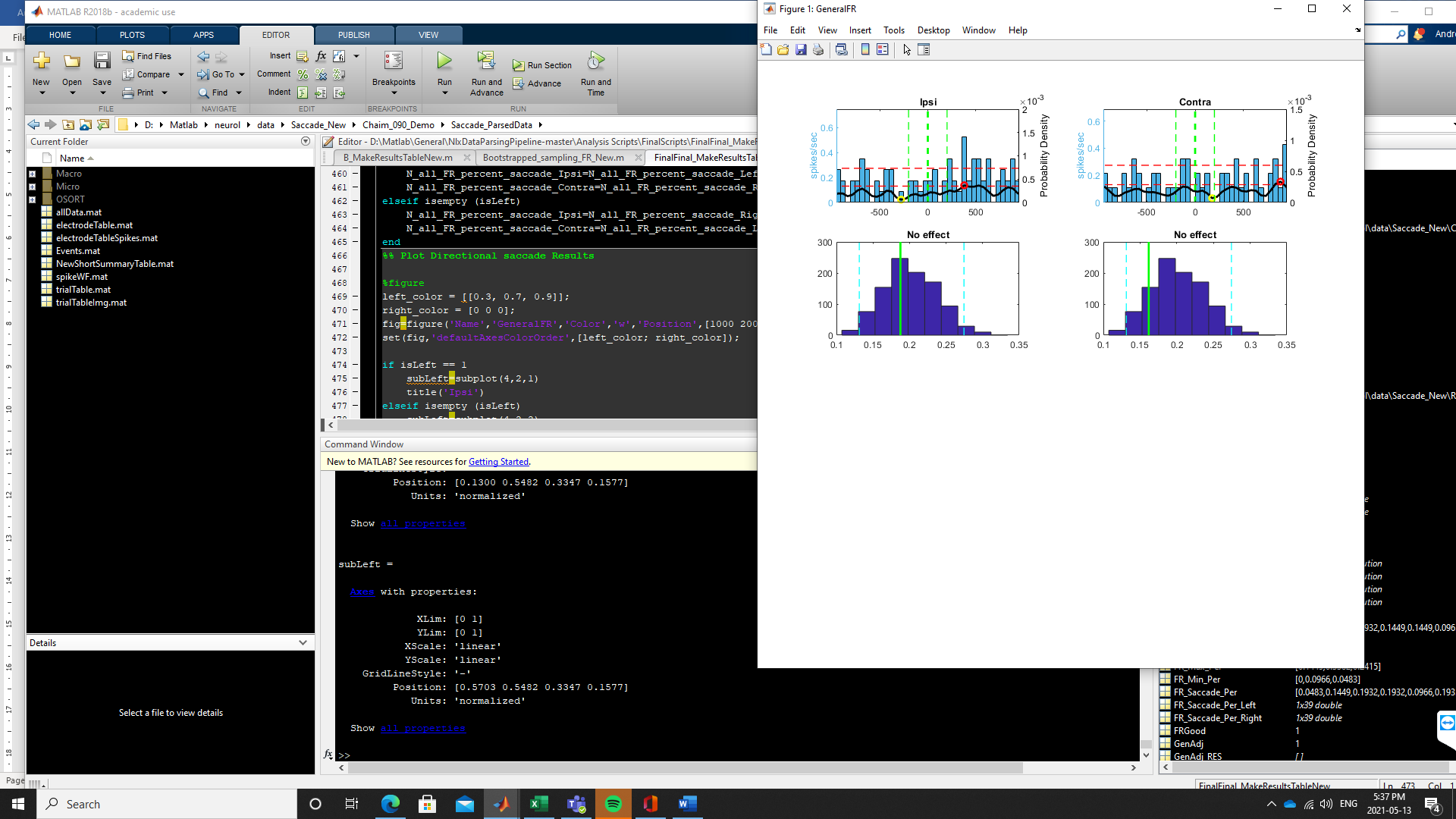
elseif mean\_FR\_Max\_Per >= percDown && mean\_FR\_Max\_Per <= percUP\_Max

title ('No effect')

end

\* Note 2.6 Line 465

Directionality. Same that for general effects decision was made according to bottom plot. The selection of Ipsi or Contra was defined in previous lines.



Decision of modulation based on these values

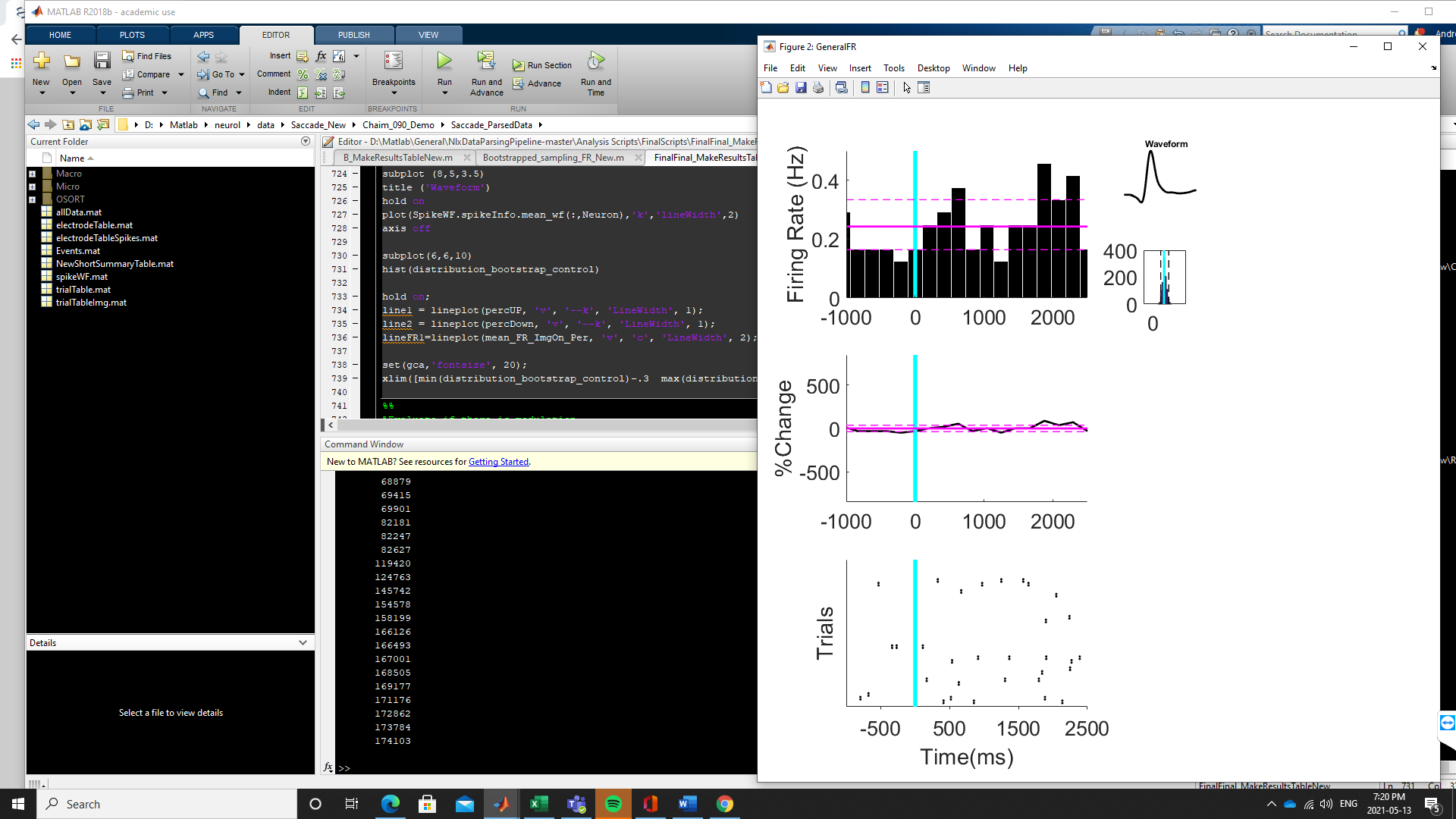
\* Note 2.7 Line 608

Reference 45 (note that it is wrong in the paper): “We counted spikes in a 200–1700ms window relative to stimulus onset. MS neurons were selected based on a significant difference between correctly identified novel and familiar stimuli in this period (p<0.05, two-tailed, boostrap comparison of means with 1000 runs). A MS neuron was FS if the mean if all familiar trials was larger than all novel trials and NS otherwise. VS neurons were selected using a 1×5 ANOVA with the factor visual category (1–5) based on the identical spike counts and with p<0.05”.

We describe as follows: Single unit responses to image onset were analyzed in a window between 200 and 1700ms after image onset, in line with previous single-unit literature [46]. Neurons were marked as modulated by image onset if their mean firing rate in this window significantly increased or decreased compared to the corresponding null distribution (Figure 4A).

**I think this matches the code**

**Note 2.8 (example figure Image onset per neuron)**



**Decisions of modulation were based on this**

**The distribution in this case consists of 1000 samples of FR at 40 random points (equivalent to 40 img onsets) and 1500ms (equal the amount of time used for the analysis of image onset modulation 200-1700ms)**