Zachary McNulty

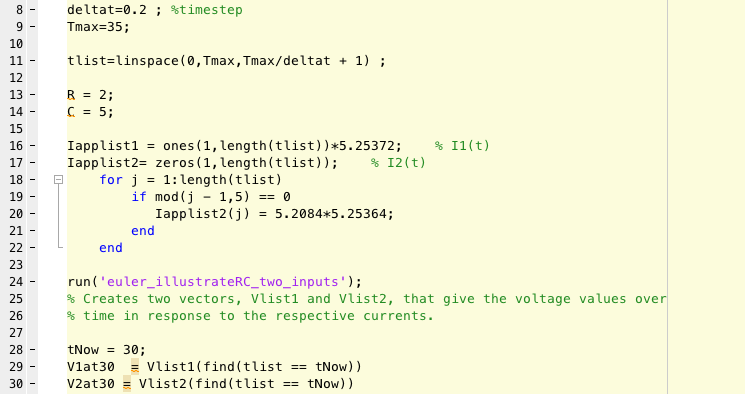
AMATH 342

HW # 3

**Section 1: Filtering of inputs: what matters in driving the membrane response?**

**1.1)**

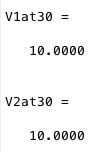
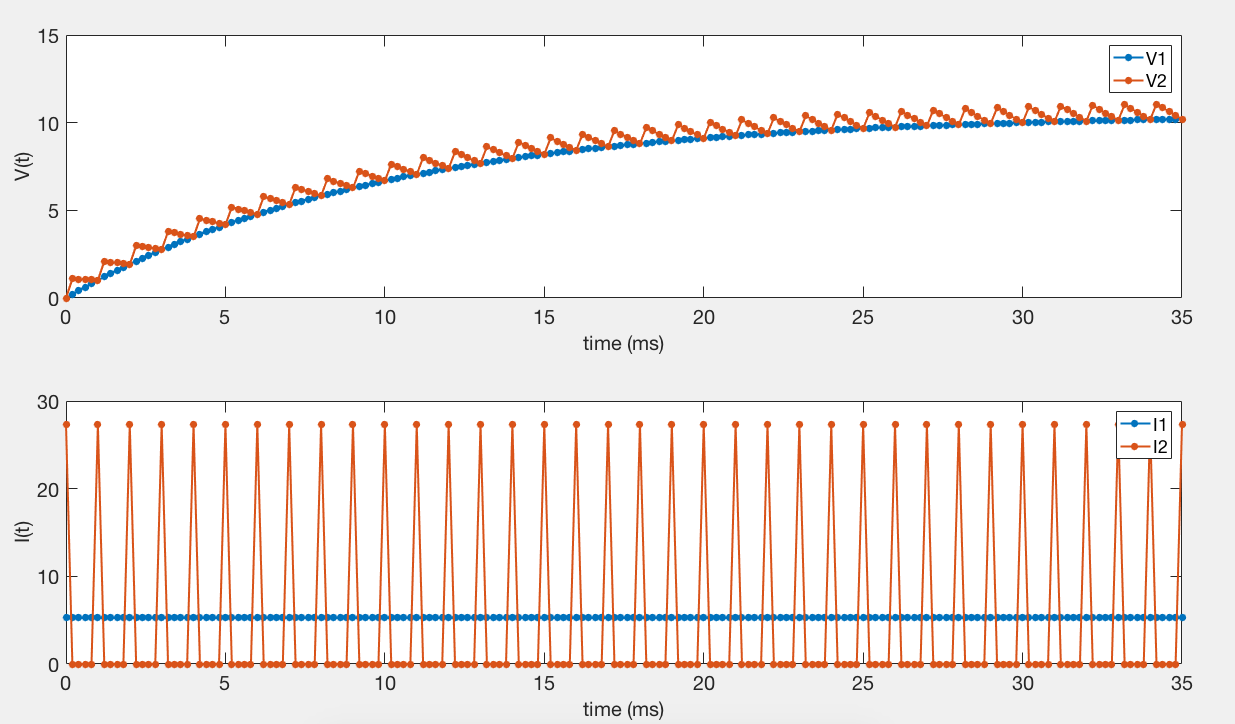
In this first section, I hope to develop two very different currents that generate identical voltage values after a given period of time, tNow. This first section generates the current functions, Iapplist1 (a constant current) and Iapplist2 (a periodic step-current), as functions of time. Once these are developed, the voltages they generate over time Vlist1 and Vlist2 (from line 24) are analyzed at tNow to show that the generated the same voltage value at that time.



**euler\_illustrateRC\_two\_inputs.m:** This code (not written by me) uses the differential equation relating dV/dt to the current and numerically integrates this function in order to develop values of voltage over time, V(t). It does so by calculating the derivative at a point, moving forward a small amount of time in that direction, updating all its values, and repeating. I do this for both currents, and then I plot these functions of current and voltage over time.



Below we see graphs of the two currents, Iapplist1 and Iapplist2, over time as well as the voltages they created respectively. We see that, although the two current functions vary immensely, the functions of voltage they generate over time are very similar, and in fact the two currents generate identical voltage values at the point of interest, tNow = 30 ms.



**1.2)**

**V(t) =**

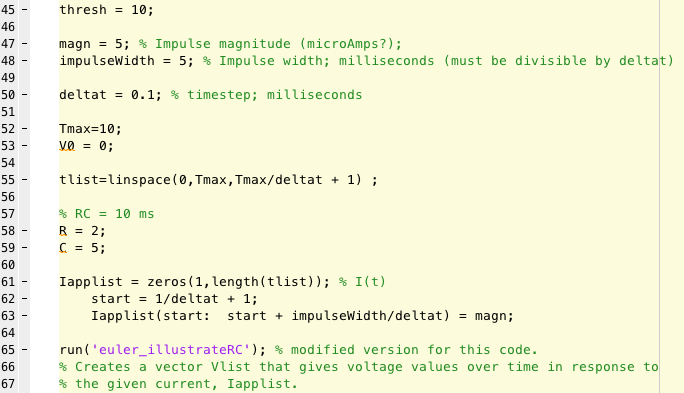
The explicit solution for V(t) involves the integral of the current, weighted differently at different points of time, over a preceding time interval. As such, V(tNow) represents the weighted sum of the current values at discrete points of time preceding tNow. Thus, it is not the individual current values at each point in time that matter, but rather only their weighted sum is important. Like any sum, there are many ways to reach this value: the definite integral of a low, long-lasting constant function can be the same as that of one which has large peaks, but is near zero elsewhere. Thus, a large but short-lived current can have the same effect as a low, consistent current if their weighted sums over time are equivalent. As such, the explicit formula for V(t) suggests there are many different currents that can create the same value for V(tNow).

The two currents I chose follow this same line of reasoning. One is a consistent, low current (I(t) 1) while the other is a periodic step current (I(t) 2). The step-current has periods of high current followed by zero current, and thus has an average current value fairly close in magnitude to the constant current, so that the sum of the two currents over time is near identical. This produces very similar V(t) values over time. The two functions are then scaled by some constant value in order to ensure they both have the appropriate value of V(tNow) = 10 mV.

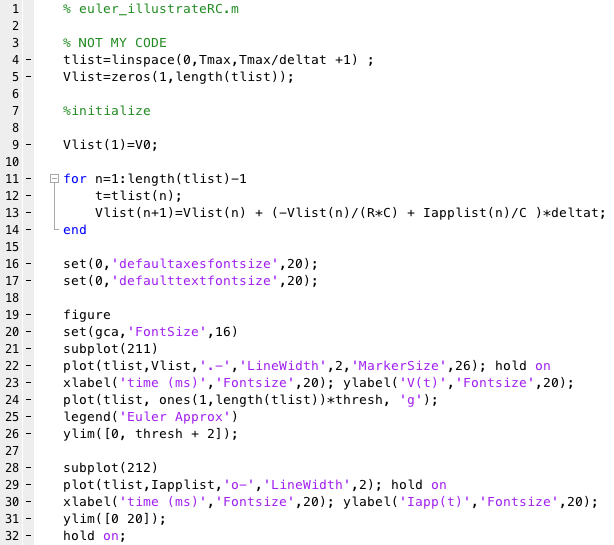
**Section 2: do impulses summate linearly, sublinearly, or superlinearly?**

**2.1) Current-Driven (RC) Circuit**

In this section, I explore how specific impulses of current can influence the voltage of a cell over time, and how several identical impulses arriving at the same time sum together to create an overall change in voltage. To do so, I simply generate a step current with an arbitrary width of 5 ms and magnitude of 5 µA (line 61-63). I then calculate V(t) created by the current (line 65; see below for description of what ‘euler\_illustrateRC’ does).

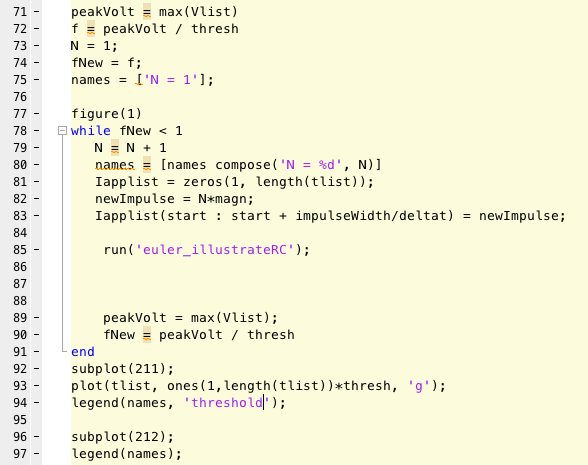
****

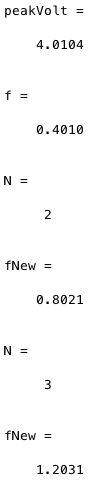
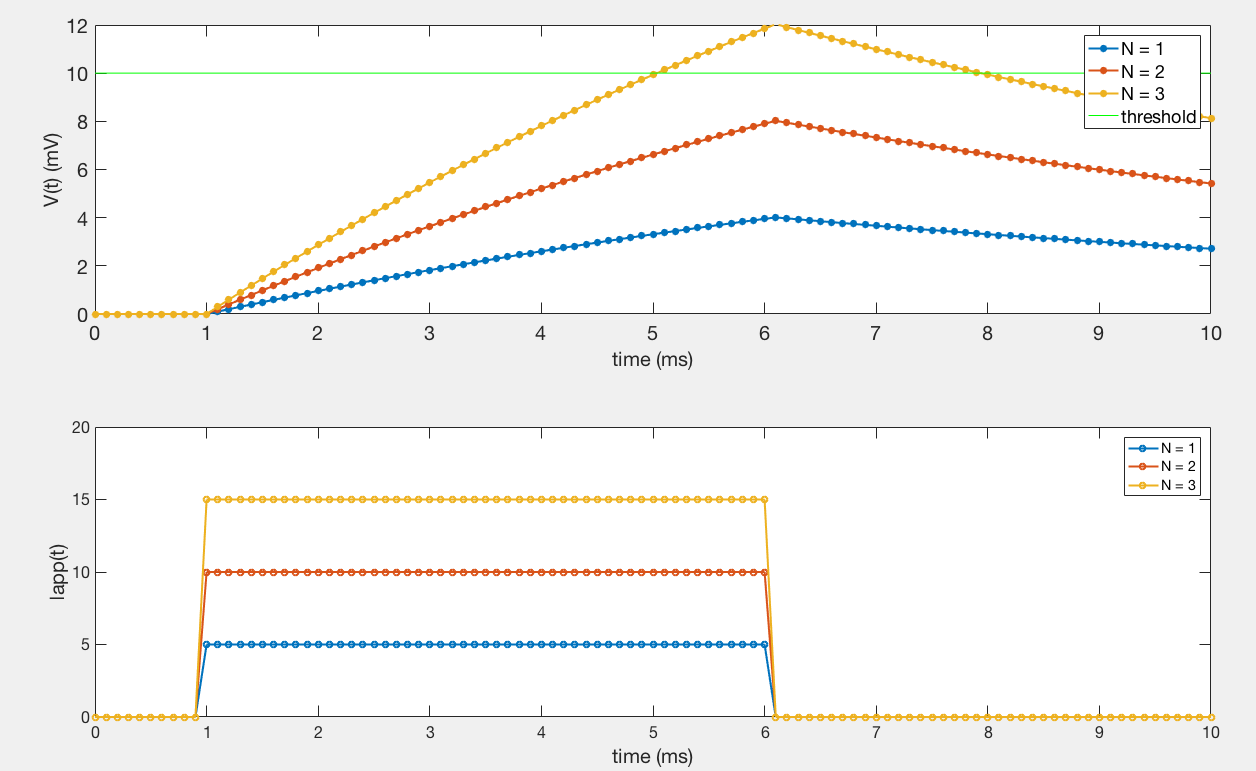
**euler\_illustrateRC.m:** To calculate V(t), I numerically integrate (in the same manner as described previously) the function dV/dt which depends on the applied current among other things (line 4-14). I then plot V(t) in one plot and plot its respective current function I(t) in a separate plot (line 19-32).

****

To see how close the current impulse drove the voltage V(t) to threshold, I find the max voltage value reached (peakVolt) and determine what fraction of the threshold this represents (f). If f >= 1, that implies threshold was reached (line 71 -75).

Now I will explore how multiple of these impulses might sum together to create a new voltage. In this case, when multiple impulses arrive simultaneously, it is as if a single impulse of current came with the same magnitude as all the individual currents combined. Thus, N impulses of magnitude “magn” create a single current of magnitude N\*magn (line 82). I then repeat what I did previously with the new current magnitude, and keep track of how the fraction (max voltage) / threshold, stored here in fNew, changes as N increases (lines 78-91). I continue this process until my voltage finally reaches threshold (i.e. fNew finally reaches 1). Lastly, to get a better visualization of the current impulse summation, I plot the impulses all in a single plot, and I plot all their respective generated voltages together in another plot (line 92-97).

****

****

As we see in the lower graph, the impulses are simply scalar multiples of the original impulse, with each successive one a single magnitude higher than the last. Interestingly enough, the voltage values associated to these functions also seem to share some sort of linearity, as the peaks seem fairly regularly spaced from each other. This observation is confirmed by looking at how fNew, the fraction of the threshold the max voltage is, changes as N changes. As N increases by 1, it seems fNew increases by around 0.4. To put it more specifically, it appears there exists a relationship between fNew and N such that fNew = N \* f. This suggests that each additional impulse of the given magnitude increases the voltage by the same amount as the initial one, suggesting V(t) is a linear function in terms of the applied current

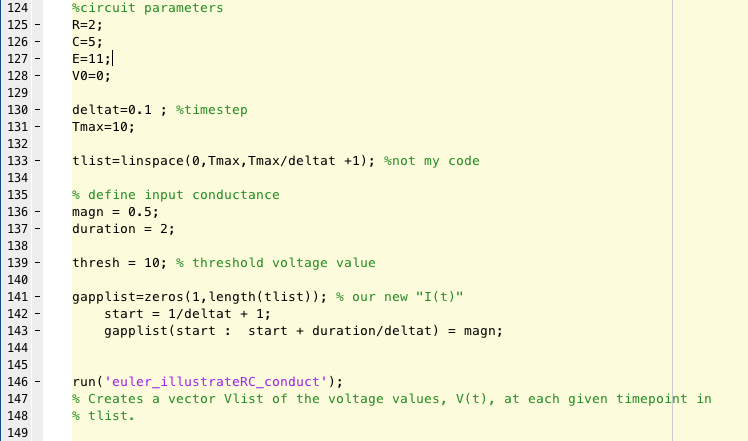
(i.e. I(t) 1 + I(t) 2 = V(t) 1 + V(t) 2). Thus, we see here that the lowest value of N that will drive the voltage over threshold for this given impulse is N = 3. To reach threshold, we know fNew must be greater than or equal to 1, as fNew is the proportion of threshold reached, so N is naturally related to f by the fact that N is the smallest integer such that N \* f >= 1.

We can arrive at the same conclusion by looking at the explicit formula for V(t) in terms of I(t):

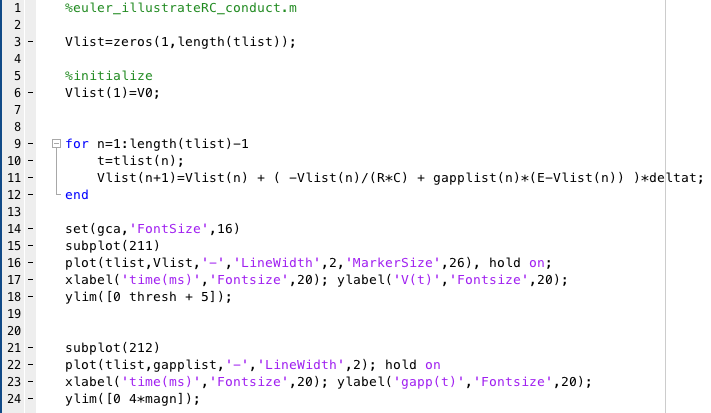
V(t) =

**2.2) Conductance-Based Input Current**

We have seen that for the current-driven model, V(t) is a linear function of the applied current, and as such the voltage generated by any combination of currents is simply the linear some of the voltages generated by each individual current. I will now investigate whether the same holds true for Conductance-Based Input Currents. I start by generating my initial Conductance “impulse” (gapplist; line 141-143) and calculating the generated voltage V(t) (line 146).

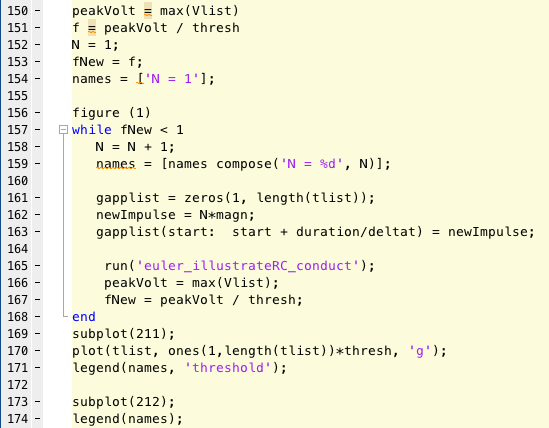
****

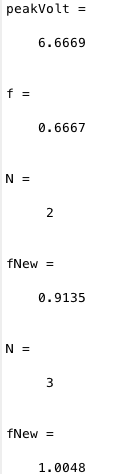
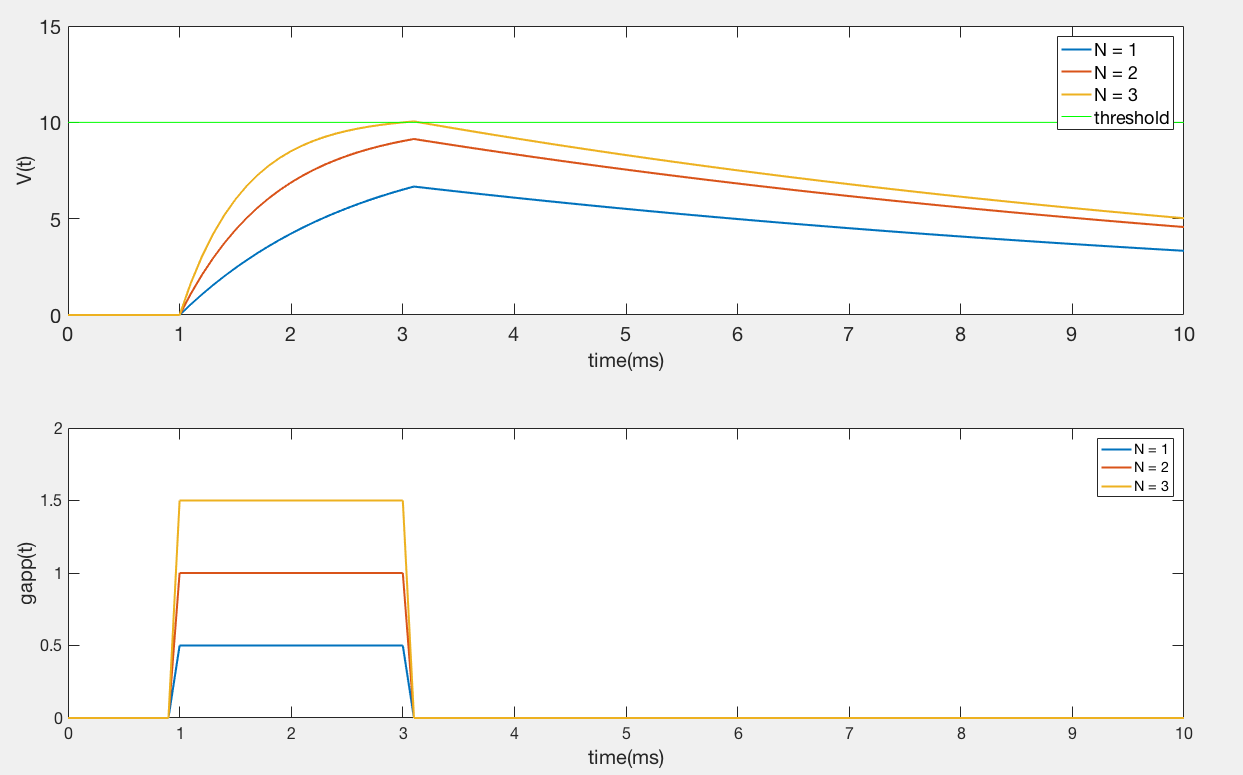
**euler\_illustrateRC\_conduct.m:** To calculate V(t), I apply the same techniques of numerical integration I described previously, but this time I use a new equation for dV/dt that considers the effects of conductance rather than a direct applied current like before. I plot the generated voltage V(t) and, in a separate plot, I plot its respective conductance.

****

In order to get some metric of comparison, I again calculate the proportion of threshold reached at the max of V(t)

(f; line 150). To evaluate how these conductance impulses sum together to generate a new voltage, I repeat the same procedure with new conductance magnitudes representing the linear sum of multiple impulses of the original magnitude arriving simultaneously (i.e. what voltage values does conductance impulses of 2 times the original magnitude, 3 times, …, N times the original magnitude generate) (line 157-168). For each of these, I again calculate the proportion of threshold reached, fNew. I plot all the generated voltages (V1(t), V2(t), …) in a single plot and all the conductances (g1(t), g2(t), …) in another plot so we can see how these conductance’s sum to create new voltages.

****

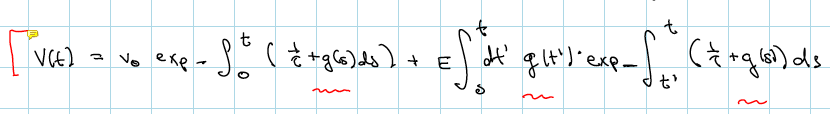
****

As we see in the lower plot, the conductance impulses match what we would expect from linear combinations of a single impulse of a given magnitude, and we see an even separation between each of the impulses. However, the voltage values do not seem to be as linearly related. There is a large jump from the baseline as a result of the single conductance impulse, but as more impulses keep getting added the added benefit of each additional impulse gets less and less. We see this same trend in the values of f, fNew (N = 2), and fNew (N = 3). The first impulse reaches 0.67 of the way to threshold, while the next one only provides an additional 0.2468 of the way to threshold, and the third impulse provides a measly 0.0913 of the way to threshold. N = 3 is still the lowest value of N that allows us to reach threshold, but we could not have predicted this as readily from f as we did before (our previous linear model for the RC circuit would have predicted N = 2). N is still related to f in the sense that as N increases, the proportion of threshold increases as well, but not so linearly.

According to this example, a better approximation of N would be the lowest number of terms such that

f + f \* (0.37) + f \* (0.37)^2 + …>= 1 (this is a very rough estimate at what the formula may be and is likely not truly accurate, but it provides some insight into this problems lack of linearity).

Unlike the Current-Driven RC model, these voltages do not seem to share this property of linearity. The reason for this can be seen in the explicit formula for V(t) in the conductance-based circuit model:

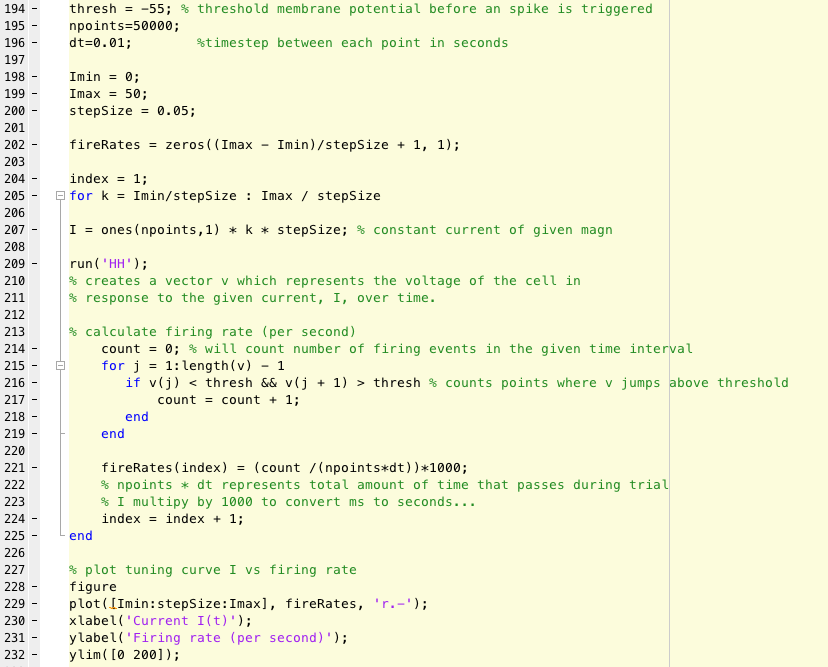
****

The exponential function is nonlinear! Any linear sum of conductance impulses, described in g(s), will be transformed by the exponential function into a nonlinear combination, and thus will certainly not come out to a linear change in V(t).

**Section 3: Hodgkin-Huxley Model**

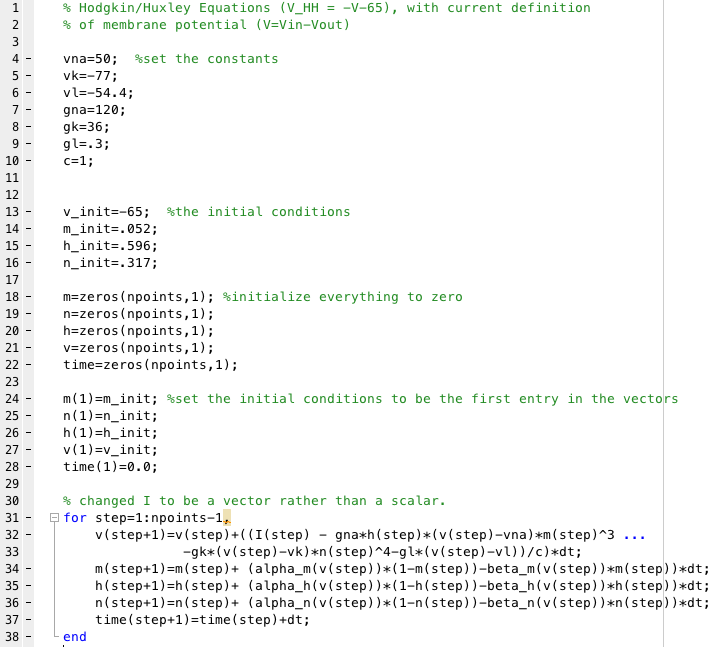
**3.1) Firing Rate -- Current Tuning Curve**

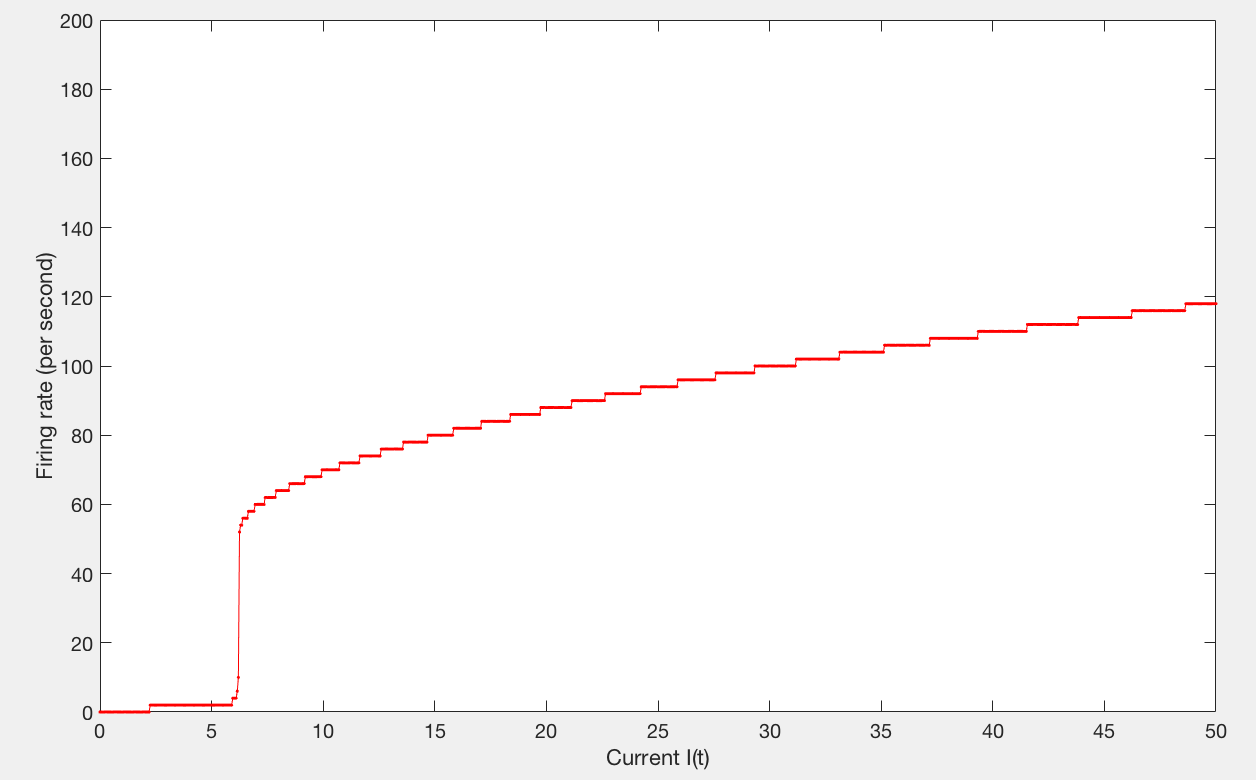
To get a tuning curve relating an applied, constant current I(t) to the firing rate, I can use the Hodgkin-Huxley model to approximate the voltage v(t) over time of a given cell in response to such a current, and repeat this test with a wide range of current magnitudes (lines 205-225). To determine the firing rate from the voltage values over time, I choose a given threshold voltage value above which I know an action potential will be triggered. In this case, I chose a threshold of -55 mV. After that, I simply step through all the values of v(t) and find all the points in time where at time t, v(t) was below threshold and at time t+1, v(t) was above threshold (i.e. all the times v(t) crosses threshold) (line 214-219). This avoids the issue of possibly double counting a spike when threshold is crossed again on the way down from the peak of the spike by strictly considering this “less than” before “greater than” condition. This will give me a count for the number of spikes that occurred in the given trial, and this can easily be converted into a firing rate by dividing the count by the duration of the trial (line 221). Lastly, I plot the tuning curve to get a look into the response of the cell to varying current magnitudes.

****

**HH.m:** To calculate the voltages generated by a given applied current, I apply the Hodgkin-Huxley model and use numerical integration as described previously to step through each time point and approximate the values of v(t).

(NOT MY CODE).

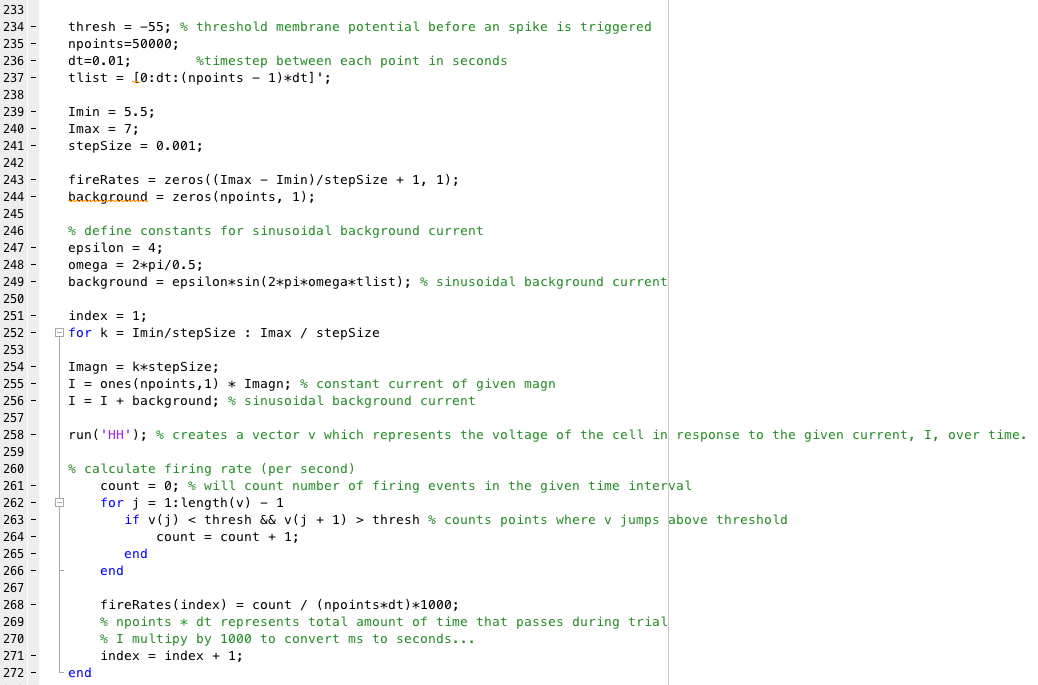
****

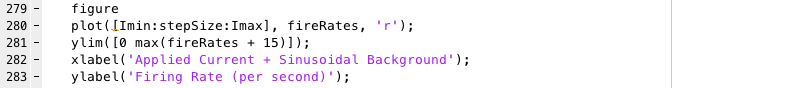
****

Above is the tuning curve for a cell responding to a wide-range of constant currents of a specific magnitude. We see a sharp rise in the response around a current magnitude of 6-7 µA, so this suggests there is some kind of threshold value to be reached before the current is enough to trigger a measurable response. After this threshold, the increase in firing rate is fairly consistent as the current increases, although it seems like the firing rate may be approaching some sort of plateau at much larger current magnitudes.

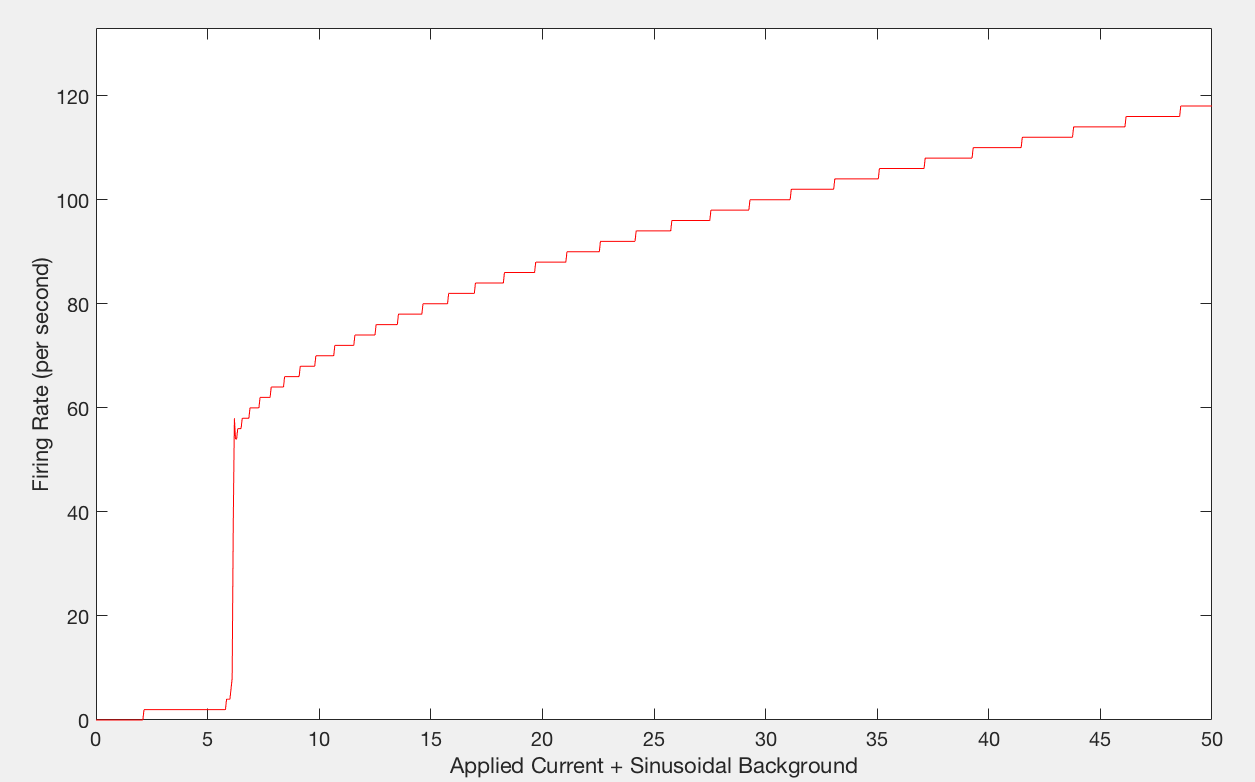
**3.2) Tuning Curve with Sinusoidal Background Current**

This code does exactly the same thing as the code in the previous section with one small alteration. I define a sinusoidal background current that will act as some sort of background noise to the simulation. I define the magnitude and frequency of such a background current (line 251-253) and then simply added in this background to each one of my currents from the previous step. The most interesting results of such a background noise will occur at the threshold value I previously mentioned as at this point there is a large jump in the firing rate over a relatively small change in current magnitude. Thus, there is a chance a positive background might trigger a vast increase in the firing rate if that original current magnitude was insufficient, while negative interference might cause a current magnitude that normally fires consistently to fall below threshold and thus fall flat. I chose the constants epsilon and omega in such a way to maximize my chance of this occurring, mostly through trial and error. Lastly, I simply plot the tuning curve (line 279-283).

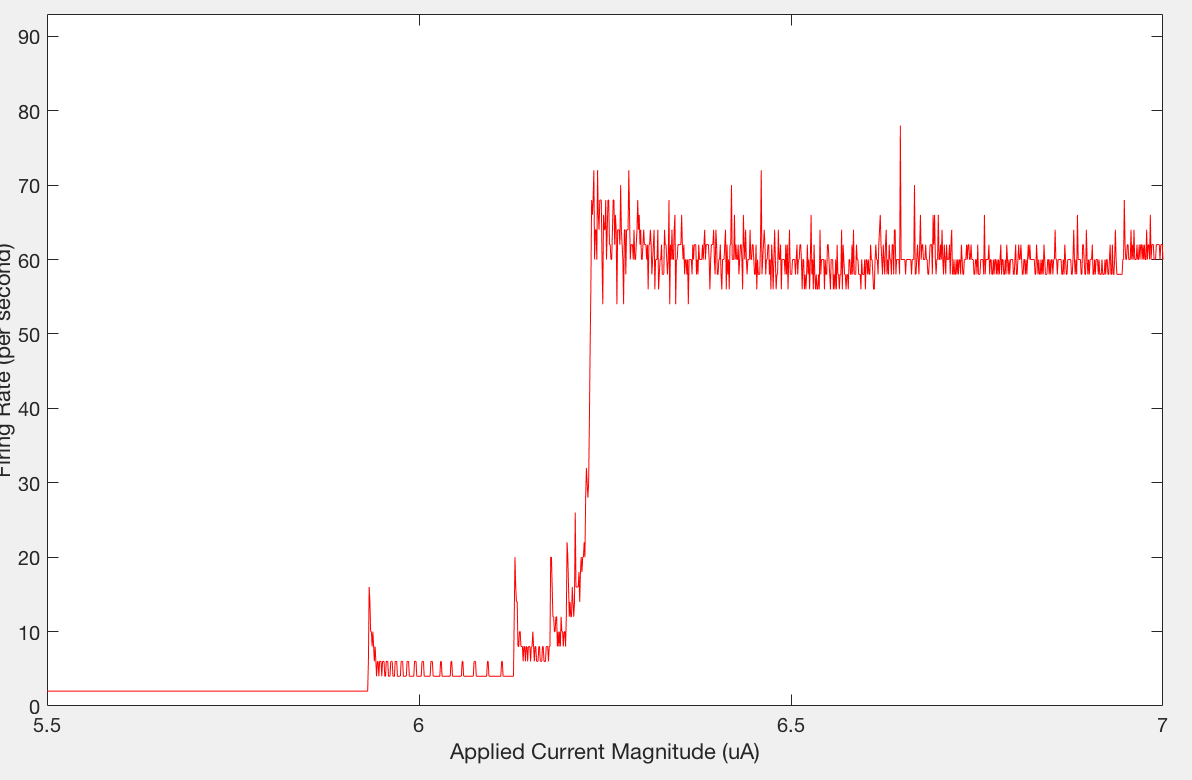


****

Overall, the main tuning curve looks relatively the same. The background sinusoidal current has a zero mean and a relatively small magnitude compared to many of the right most values, so it is no surprise that this portion of the graph is relatively unaffected. As I mentioned before, the truly interesting changes will occur close to threshold.

****

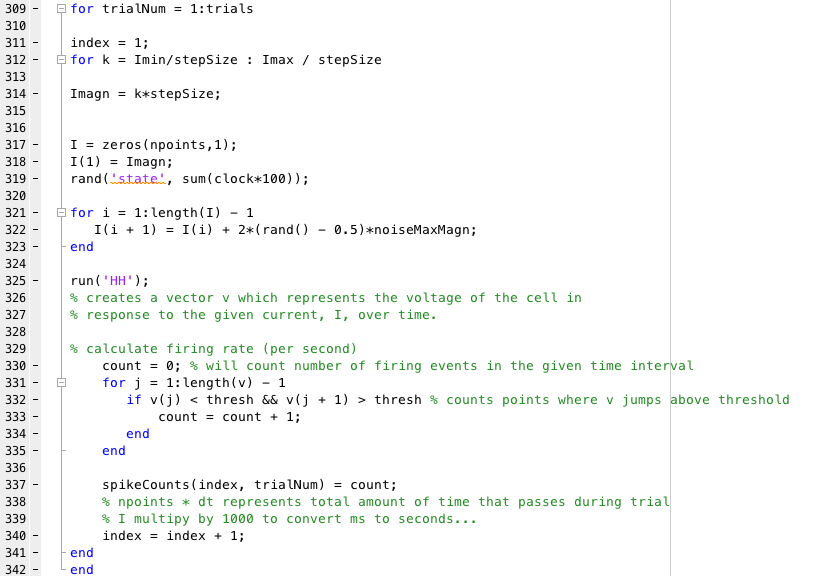
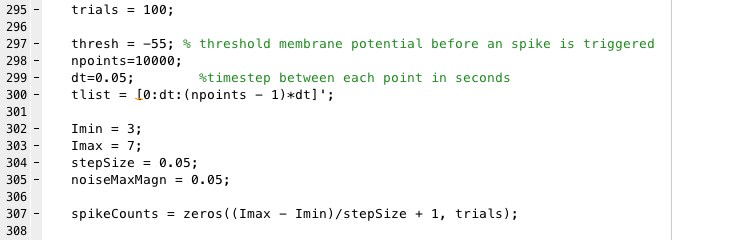
Below, we take a look at values specifically close to the threshold value of ~6-7 µA I previously mentioned. Unlike before without the background current, the rise at threshold is much less smooth. We see a significant amount of spiking earlier than usual, a result of the positive areas of the background current boosting current values typically too low to be above threshold. Furthermore, we see marked decreases in the firing rate while before there was a near constant increase in firing rate with an increase in current. These areas of decreased firing rate highlight negative interference of the background current, where it is on the low side of its oscillation.

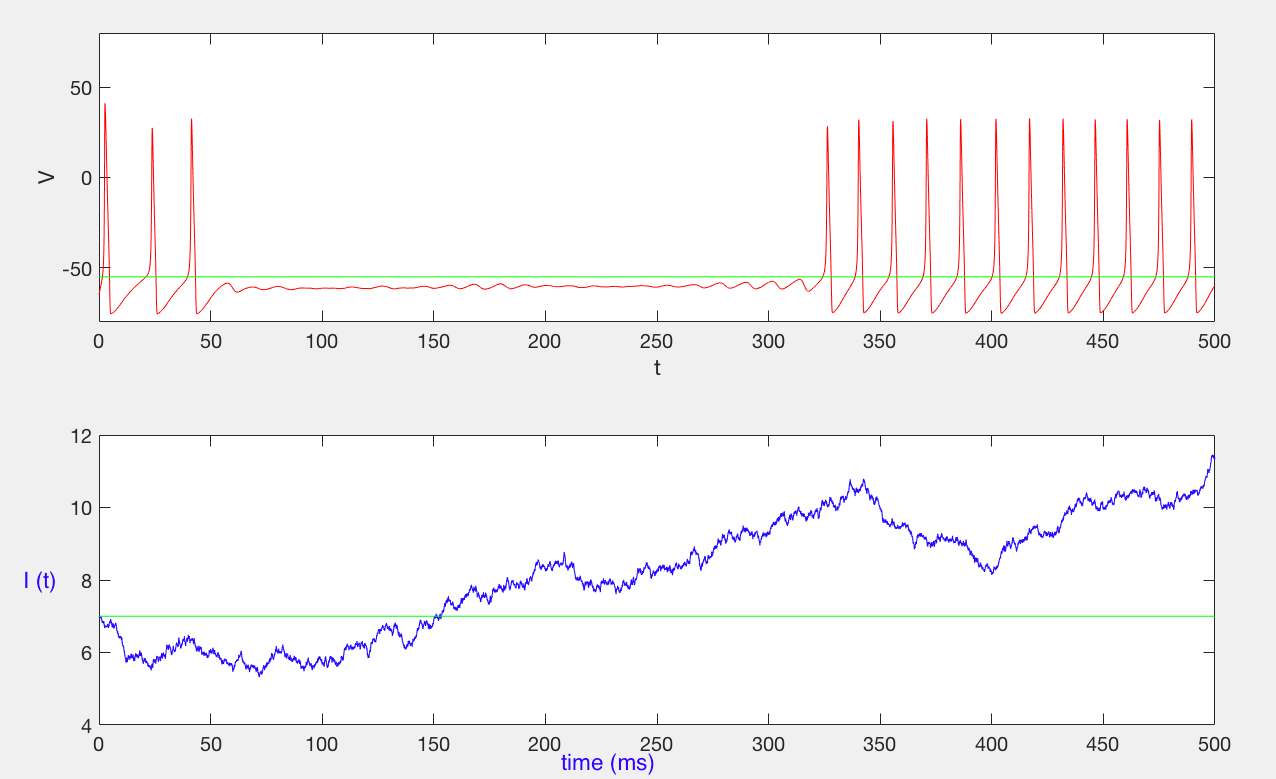
****

**3.3) Adding A Bit of Noise!**

I chose to add noise to the experiment by adding random numbers to current at each point of time such that noise from past time points is still incorporated into future time points (the current at time t + 1 is equal to current at time t + noise; this way the noise can build off of each other). This noise is a random number between –noiseMaxMagn to noiseMaxMagn, and thus has a zero mean so that I(t) + noise will have a long-term average equal to the magnitude of the constant applied current.

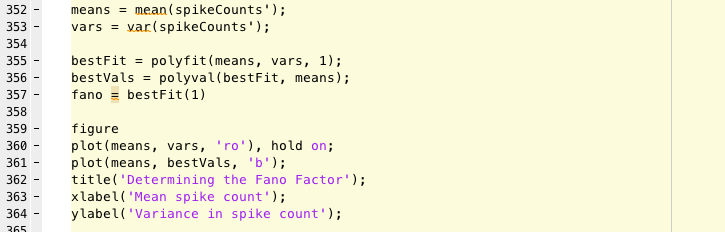
This code does the same as the previous code, with two small changes. Firstly, it takes an extra input, the number of trials, and repeats the calculations of v(t) and firing rate numerous times for each current magnitude in order to get a look at the kind of variation the noise is producing. Next, it adds the noise term as previously described (line 305, 321-323).

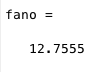
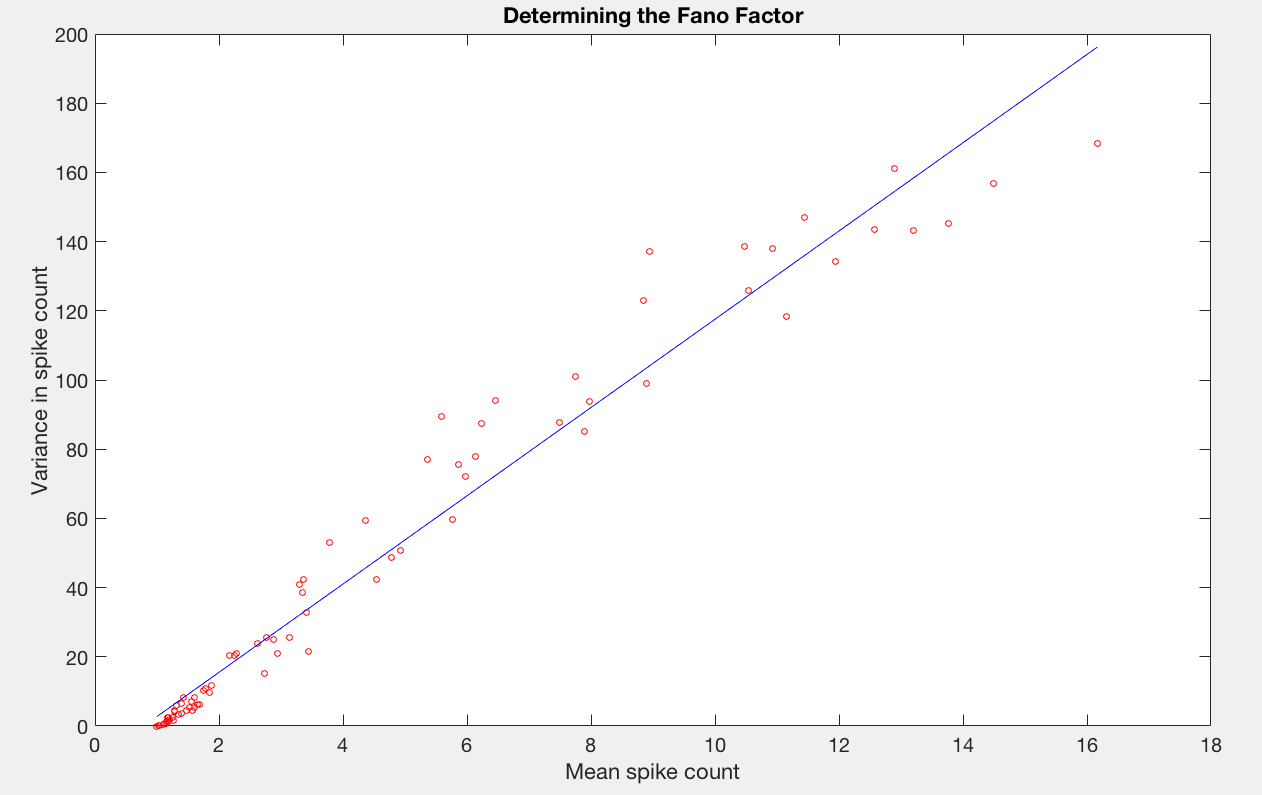




Above, I have provided an example of the noisy current (this plot comes from HH.m; described in an earlier section). We see in the lower plot that the current often drifts away from the magnitude of the applied current as time goes on as a result of the noise, but the average value of I(t) is still fairly close to the magnitude of the applied current. The first plot shows the effects this noise can have on the firing rate, creating areas of inactivity/activity rather than a consistent firing rate (or no firing at all) we see with just constant currents.

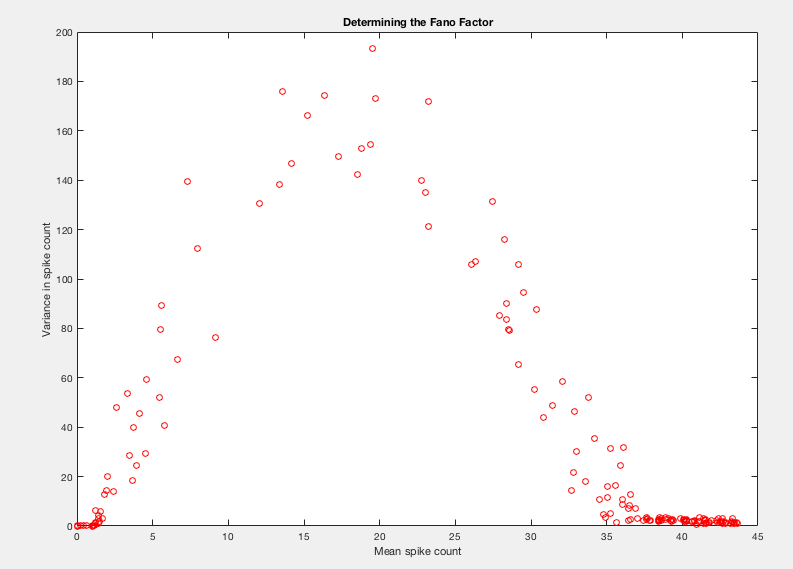
To see how this noise creates variabilty, I will calculate the fano factor of such a process. To do so, I simply look at the relationship between the variances and the means of my spike counts (line 352-353). Then, to calculate the fano factor I find the approximate ratio of variance to mean by finding slope of the best fit line through my points (line 355-357). I plot this information to get a better visual interpretation of what is going on.

****

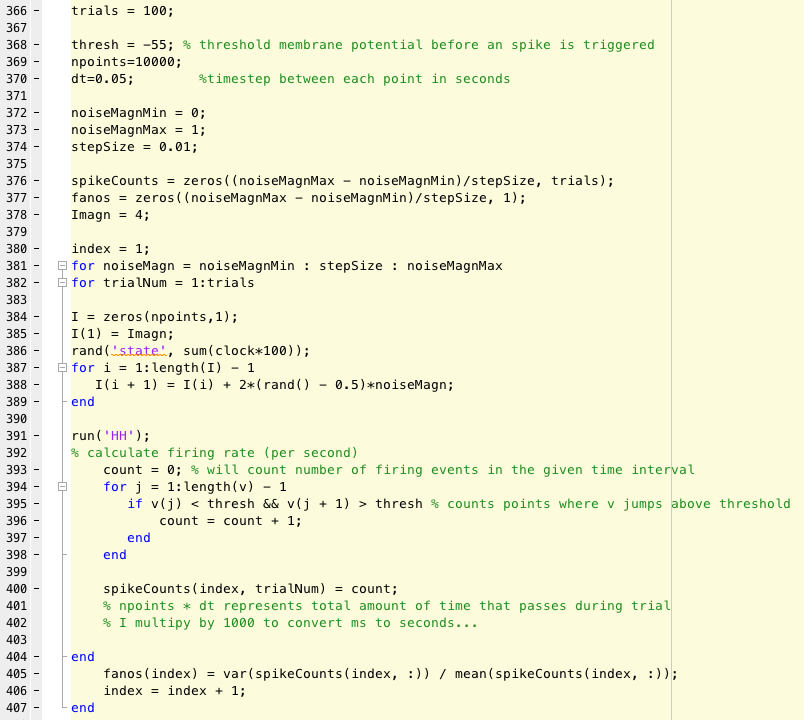


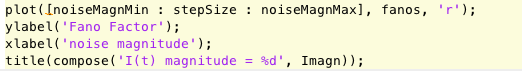
For this fano factor, I mostly considered points near the threshold for repetitive firing, as this is the region where any kind of noise can have a major impact (as we saw with the sinusoidal background current in the last section). If the current is too high above threshold, the relatively small magnitude noise will contribute little to the firing rate as this condition will fire constantly regardless of the noise simply because the noise is not strong enough to drop it below threshold. If the current is too low, no firing can happen as the relatively small magnitude noise is rarely enough to jump above threshold on its own. Thus, there is very little variation due to noise sufficiently above and below the threshold, creating a fano factor of zero (I added the full graph over a wide range of applied currents at the bottom of the page for a visual, but my analysis from here on will be solely near threshold, with applied currents around 3-7 µA).

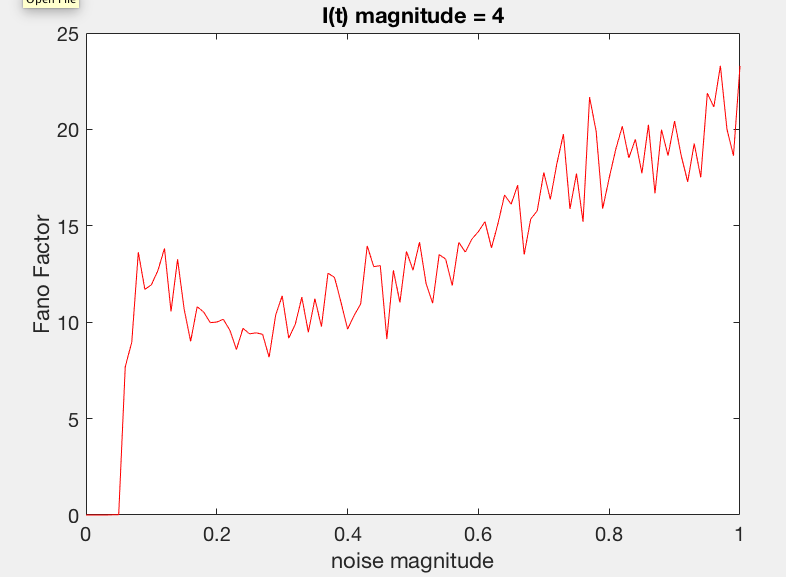
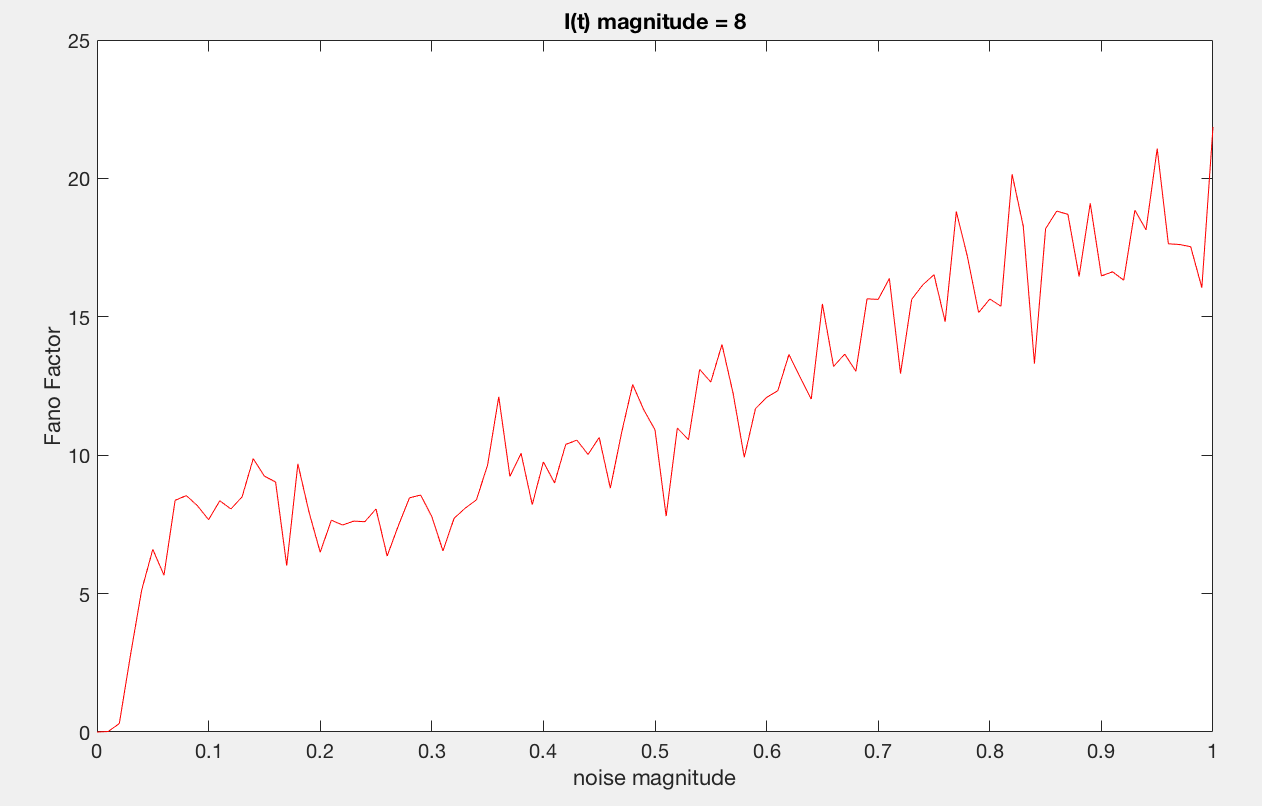
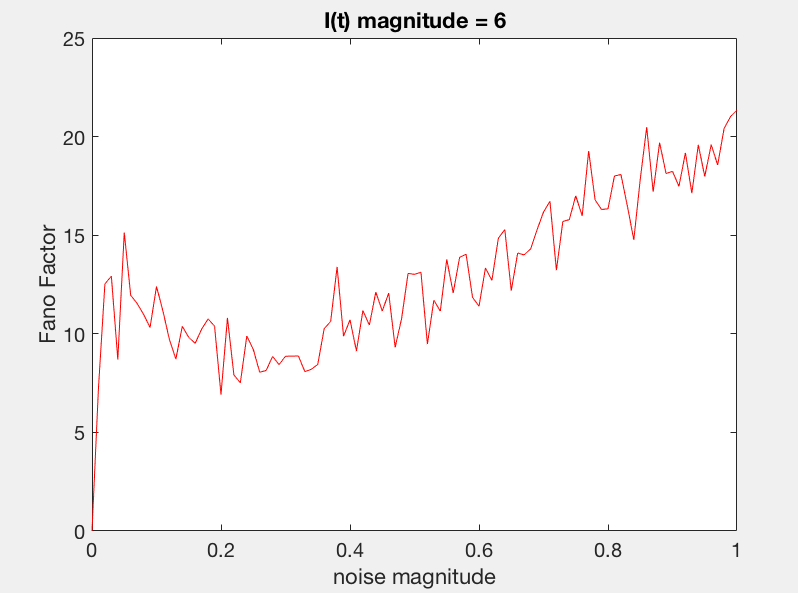
Near this threshold, the noise can create so much variation, leading to a high fano factor, because a firing event is such an all-or-nothing response. If the noise, by chance, brings the current above threshold, we see very consistent firing, else we see practically none. Thus, there is lots of variation from the mean, and this variation leads to a large variance when it gets squared as a result of the calculation.



Lastly, I will analyze how the noise magnitude influences the fano factor. This time, rather than progressing through a series of applied current magnitudes, I choose a fixed current magnitude and step through numerous noise magnitudes (line 381). I add noise to my applied current and calculate the spike count as previously described. At each given noise magnitude, I run numerous trials and compare the spike count means across these trials with the variance in order to calculate the fano factor (line 405). Lastly, I plot the fano factors against the noise magnitudes that generated them in order to see if there are any trends.







As these graphs highlight, there seems to be a positive trend between the noise magnitude and the fano factor. More noise appears to lead to a higher variance to mean ratio. This makes sense as the noise, as it is zero mean itself, across many trials should average out more or less and leave the mean spike count alone. The variance trial to trial, however, will likely increase as the noise becomes a larger fraction of the applied current: there are more trials that jump across threshold when they normally would not have. If the variance increases while the mean stays relatively the same, the fano factor will increase.