Membrane Potential and Stochastic Ion Channels

Homework: This assignment is to be done individually and uploaded **BEFORE** the start of lab. We will post videos on Canvas that cover the coding syntax you need in order to complete this assignment. Videos should be available over the weekend between Week 1 and Week 2.

Organize your code and answers clearly in two .Rmd files, 1) GHK assignment, 2) Ion channel assignment. Enter all answers to boldface questions as comments in the code.

This is an individual assignment, but you are allowed to work together in groups and discuss coding and answers. That said, you are responsible for all of the material in this homework assignment. ***DO NOT COPY from anyone that you work with****.* ***You are NOT allowed to share code****. You need to write the code and answer the questions yourself. Try the coding yourself first before seeking help.*

Be sure to include your name in the file name as follows: lastname_firstname_labday .Rmd. Also type your full name as the first comment in your .Rmd file. Upload the two .Rmd and 2 knitted files to Canvas before the start of your lab.

Due: BEFORE the start of your lab period in week 2
Worth: 5 points

If you have any questions as you work on this assignment, please do not hesitate to email Dr. Haddadian (haddadian@uchicago.edu). Best of luck!

Part One: Goldman–Hodgkin–Katz Equation – First .Rmd file

Introduction

The voltage across animal plasma membranes is generated primarily by the flux of ions across that membrane. How easily ions pass through (diffuse across) a membrane is determined by the permeability of that ion. In neurons, the permeability of the membrane to various ions is primarily controlled by the opening and closing of ion-selective channels.

The GHK voltage equation was developed in the 1940s. It relates ion permeabilities and concentrations to a steady-state membrane voltage, the **reversal potential**, such that:

$$E_{rev} = \frac{RT}{F} \ln\left(\frac{\left(P_{Na}Na_o\right) + \left(P_K K_o\right) + \left(P_{Cl}Cl_i\right)}{\left(P_{Na}Na_i\right) + \left(P_K K_i\right) + \left(P_{Cl}Cl_o\right)}\right)$$

Where

- R is gas constant 8.314 J/mol-K
- T is absolute temperature 310 K
- F is Faraday's constant 96500 coul/mol
- Na_o and Na_i are the external and internal concentrations of Na^+, respectively, mM
- K_o and K_i are the external and internal concentrations of K^+, respectively, mM
- Cl_o and Cl_i are the external and internal concentrations of Cl^-, respectively, mM

- P_{Na}, P_K, and P_{Cl} are the respective membrane permeabilities of Na^+, K^+ and Cl^-. Absolute permeabilities (cm/s) are very hard to determine experimentally, therefore permeabilities are often expressed as ion permeability relative to another ion (unitless).

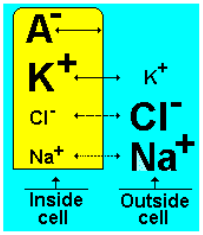
1. ***What do we call this membrane potential (in physiological terms)? Describe what is going on with the ionic currents when the membrane is at the calculated potential. Put the answer as a comment in your .Rmd file for GHK.***

Programming GHK Equation

Please open R and program the GHK equation (make GHK equation a **general function** that you are able to call for the remainder of the lab). You will need to first define constants (listed above), and the parameters (ion concentrations and permeabilities (table below)).

Membrane Permeabilities	*External Concentrations*	*Internal Concentrations*
$P_{Na}=1$	$Na_0=460$ mM	$Na_i=50$ mM
$P_K=20$	$K_0=10$ mM	$K_i=400$ mM
$P_{Cl}=5$	$Cl_o=540$ mM	$Cl_i=100$ mM

Physiological conditions for giant squid axon.



In general concentration differences between inside/outside of cell

Programmed thus far: constants, parameters, and defined the GHK function. Now run the code with the parameters defined in the table above. Answer questions 2-3 as comments in your code.

2. ***What value do you get for E_{rev}? (Remember units)***
3. ***What do we call this value of the membrane potential that you calculated in #2?***

Part Two: Probability and Stochastic Behavior – Second .Rmd file

Introduction

In the macromolecular world that we inhabit in our normal lives, the behavior of objects is deterministic. We can shut the door and it stays shut. We don't have to consider the possibility of the door being open at some later time, providing we haven't touched it. This is very different from the world of molecules. Molecules inside cells are in a constant state of motion. They jiggle and shake and behave in ways that are only predictable in a statistical sense. We say that at the molecular level behavior is stochastic. An enzyme molecule, for example, doesn't cleave substrate molecules at a uniform rate like a factory process, instead it jerks along fitfully depending on random encounters with substrate molecules and the buffeting it receives from surrounding water

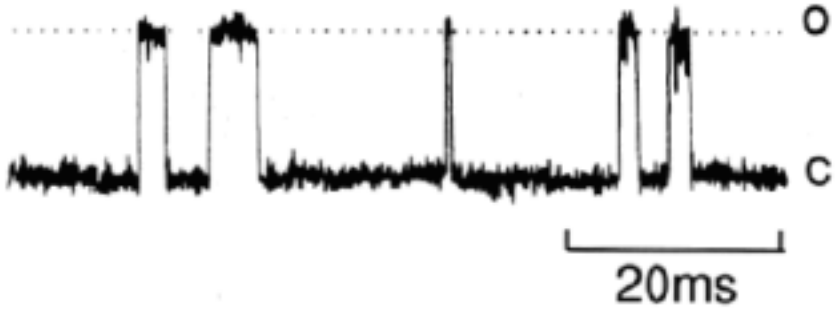
molecules. Though this process is jerky and stochastic it can be characterized by a mean (average) rate.

A particularly striking example of stochastic behavior can be seen in a special family of biological molecules. These molecules are called ion channels and they occur in the membranes of all living cells. Channels are big, complicated proteins consisting of several interlocking subunits, each of which crosses the cell membrane, thereby connecting the inside of the cell with the outside medium. Channels can typically exist in one of two states, either open or closed. In the open state the subunits move apart a little, allowing ions to pass in and out of the cell.

Different channel molecules allow different ions to pass. Some channels allow sodium ions to pass; others allow potassium or chloride ions to pass. The movement of ions through channel molecules is the signal that activates the cells of your immune system to fight disease and provides the signals in nerve cells that allow you to read this page. There are a huge number of human diseases caused by genetic defects in channels. These diseases, which are now known as channelopathies, run all the way from muscular dystrophy to cystic fibrosis to chronic heart conditions to kidney diseases. For many biologists, the really cool thing about these molecules is that, because ions are charged, the movement of ions can be easily measured as tiny electric currents passing across the membrane. By measuring the presence or absence of a tiny electric current, we can monitor the behavior of a single multimeric protein in real time!

Some kinds of channels have an open probability (the probability that it is "open" at a given moment) that depends on the voltage across the cell's membrane. When the voltage across the membrane is small, the channel has a high probability of being in the open state, and when the voltage across the membrane is large, it has a low probability of being open.

Channel molecules are quite like coins in a coin-flip experiment. The coin can land in two states, heads or tails, and the channel can exist in two states, open or closed. Just as we cannot say for sure that we will get a head on the next toss, we cannot say for sure that at the next instant a channel molecule will be in the open state. But we can say that with a fair coin we have a 50% chance of the trials turning up heads, and likewise we can give a probability of a channel being in the open state, though of course this is not usually 50%! Here is the record of a single channel molecule flipping stochastically between closed (C) and open (O) states.



What do we mean by random?
When we look at a record like this, it is pretty obvious that the openings and closings of the channel are random; in fact, this is what we mean when we say the behavior is stochastic. We have to be

careful though, when we use the word random, to be quite clear exactly what we mean. Let us suppose that by "random" we mean that at every instant the state of the channel -- either closed or open -- is determined by a random variable, in other words, a coin toss. We can check this idea out to see if it is likely to be true! **That is, we can make a conjecture and test it using a simple mathematical model. The conclusions that we reach from our test tell us something about biology.**

Simulating a Coin Toss

As we mentioned above, the simplest random event is a coin toss -- it has two possible outcomes, heads or tails. If the coin is fair, then we get each about half the time, and if the coin is unfair, it will fall to one side more often. One-way to simulate this is with a sequence of 0's and 1's: a 1 represents heads, and a 0 represents tails (or vice versa). So, our task of simulating a coin toss now becomes constructing a random sequence of 0's and 1's.

Once we have a **random number from runif(1),** we can use a **Boolean logic test** to convert that random number into either a zero or a one. If we assign a variable, say y, to be the Boolean expression 0.1>= 0.5, then y will be assigned a value of zero. Conversely if y is assigned to be the Boolean expression 0.7 >=0.5, then y will be assigned a value of one.

PROGRAM the coin toss in R

Create two variables, one variable for the number of flips of the coin (=10), and the other variable for the result of the flip (which will be composed of either 0 or 1 once the Boolean logic test is invoked– comment whether the 0 or 1 is heads/tails). When you create the variable for the result of the flip, you need to zero out the entries first, before you fill the flip result variable with the decisions of the Boolean logic test with each flip of the coin.

4. ***What is the size of the flip result array?***

Create a scheme that will invoke the Boolean logic test so that based on a random number that is chosen, each coin flip decides whether the entry in flip result will be a 0 or a 1. This is decided based on the random number compared with a probability. Use the probability that 50% of the time the coin lands on heads and 50% of the time the coin lands on tails (hint: you will use a for loop and if/else statements).

Find the average value of the flip result variable (use ***mean()*** in R).

5. ***Record the average value of the flip result given 10 flips of the coin (enter as a comment). Is it 0.5? Write as a comment in your code what the mean value is for each run (re-running the code 3 times).***

1) Plot the results of flip result (0's or 1's) versus the number of flips. Remember to include axes labels and title (use xlim and ylim to define the range of the x- and y-axes). **2)** Also put the mean on the graph using ***abline*** function – you will make a horizontal line of the value of your calculated mean. 3) Put a legend on the graph with the value of the calculated mean using ***legend("topleft", bty="n", legend=paste("mean=",average),text.col=c(2)).***

6. ***Comment on the findings of the graph? If you re-run the code 3 times, does the profile of the graph change? Comment on the changes.***

If the number of flips were increased, then the mean should become closer to 0.5. Go back and try to convince yourself of this by changing the value of the number of flips of the coin to 100 and 1,000 checking and recording the mean of flip result each time. Make sure to plot the flip result vs the number of flips and including all the items as discussed in # 5.

7. ***Write a sentence explaining why larger numbers of flips give a mean closer to 0.5.***