Integrate and Fire

Balazs B Ujfalussy

25/09/2017

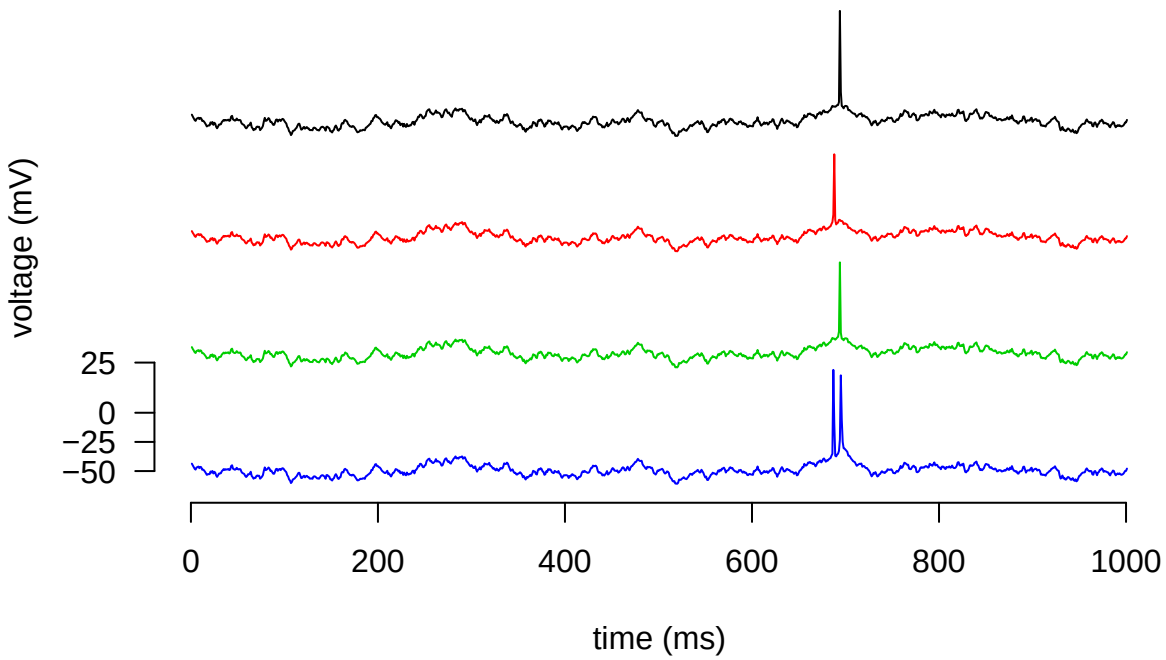
This is a demo for illustrating the response of different neuronal models to somatic current injections.

```
source('./IF_sim.R', chdir=T)
source('./Izhikevich_sim.R', chdir=T)
source('./HH_sim.R', chdir=T)
```

The models are compared with experimental data from *The quantitative single-neuron modeling competition*. In this experiments a cortical pyramidal neuron was stimulated by injection of randomly fluctuating currents of various amplitudes. Current was injected and voltage responses were recorded at the soma. For more details, see Fig. 1 of Jolivet et al., **The quantitative single-neuron modeling competition** Biol Cybern (2008) 99:417–426.

The experimental data is loaded from the `Data` folder - Jolivet_resp.RData and `Jolivet_stim.RData`. First we load the data and hen plot a short example of it.

```
load("./Jolivet/Jolivet_stim.RData")
load("./Jolivet/Jolivet_resp.RData")
matplot(t(resp[1:4+4*2,3000:4000] - c(0, 1, 2, 3)*100), t="l", lty=1, axes=F, xlab="time (ms)", ylab="v
```

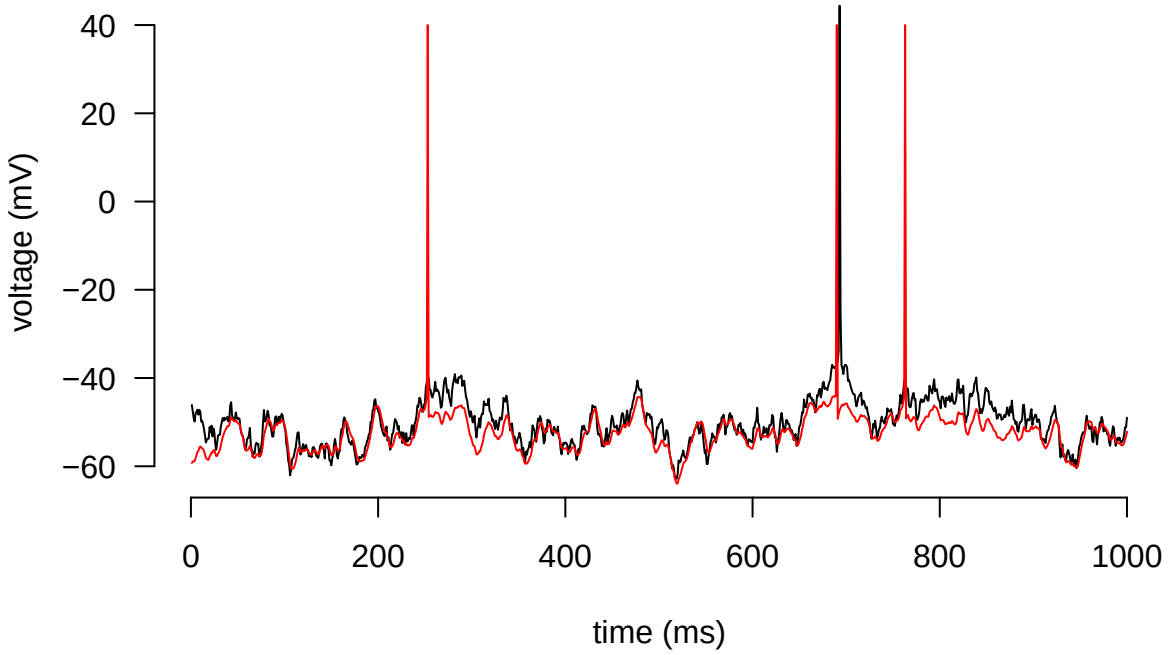


There are 8 different stimulus, and 4 recordings with each - so we have altogether 32 stimulus response pair. The stimuli differ in their mean and variance, otherwise they are filtered Gaussian noise.

Next, we select a 1s long portion of the data - both the stimulus and the response. We select the 9th stimulus response pair, which has a large input variance but low mean. We will inject the same current to an integrate and fire neuron and record its response. We adjusted the parameters of the IF neuron to roughly match the real neuron's response. Importantly, the IF neuron has only a couple of intuitive parameters, including `tau`: time constant of the integration, ms; `Rm`: membrane resistance, GOhm; `v.rest`: resting potential, mV; `v.init`: initial value for the voltage; `v.reset`: reset after spike; `v.spike`: peak of the spike; `v.threshold`:

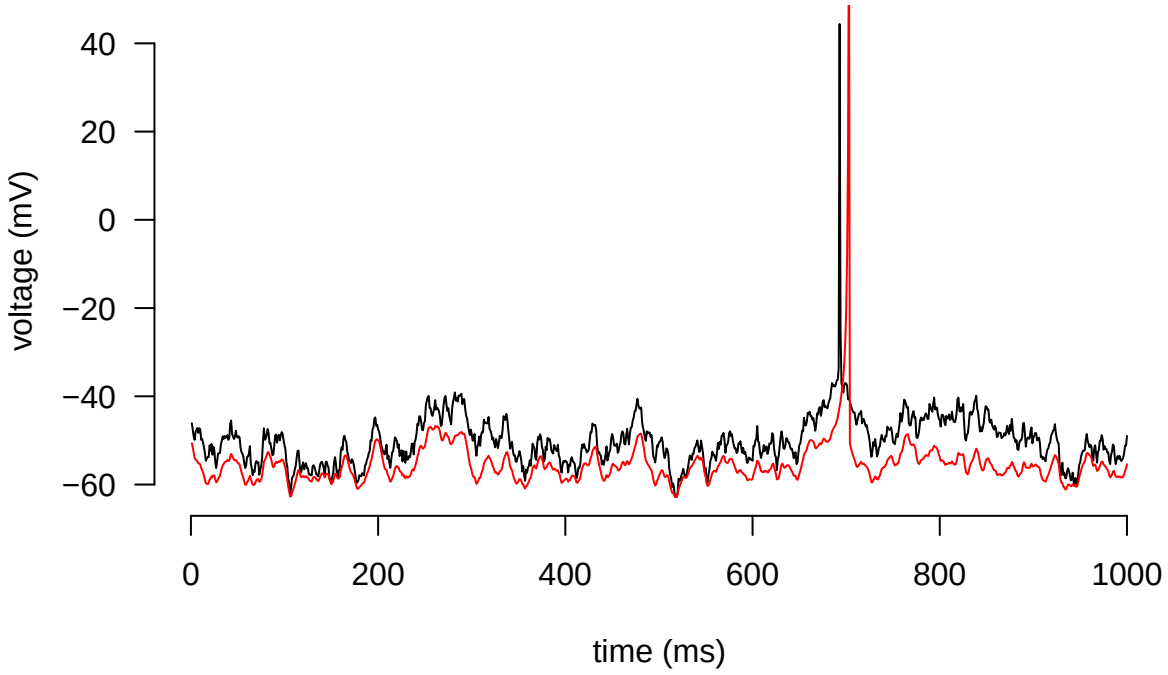
spike threshold. The parameters are set to their default value, except when changed explicitely.

```
r.cell <- resp[9,3001:4000 ]
input <- stim[9, 3001:4000]
v.IF <- sim.IF(I=input, v.rest=-60, Rm=0.045, tau=10, v.threshold=-44, v.reset=-50)
plot(r.cell, t="l", lty=1, axes=F, xlab="time (ms)", ylab="voltage (mV)"); axis(1); axis(2, las=2)
lines(v.IF, col=2)
```



We repeat the same experiment now using an Izhikevich neuron. This neuron is different from the IF neuron as it also simulates the spikes, not only the subthreshold response. It has a higher number of parameters, including: `tau`: time constant of the integration, ms; `Rm`: membrane resistance, GOhm; `v.rest`: resting potential, mV; `v.init`: initial value for the voltage; `v.reset`: reset after spike; `v.threshold`: spike threshold; `v.peak`: peak of the spike; `a`,`b`,`c`: parameters of the slow variable, controlling adaptation or bursting. The parameters are set to their default value, except when changed explicitely.

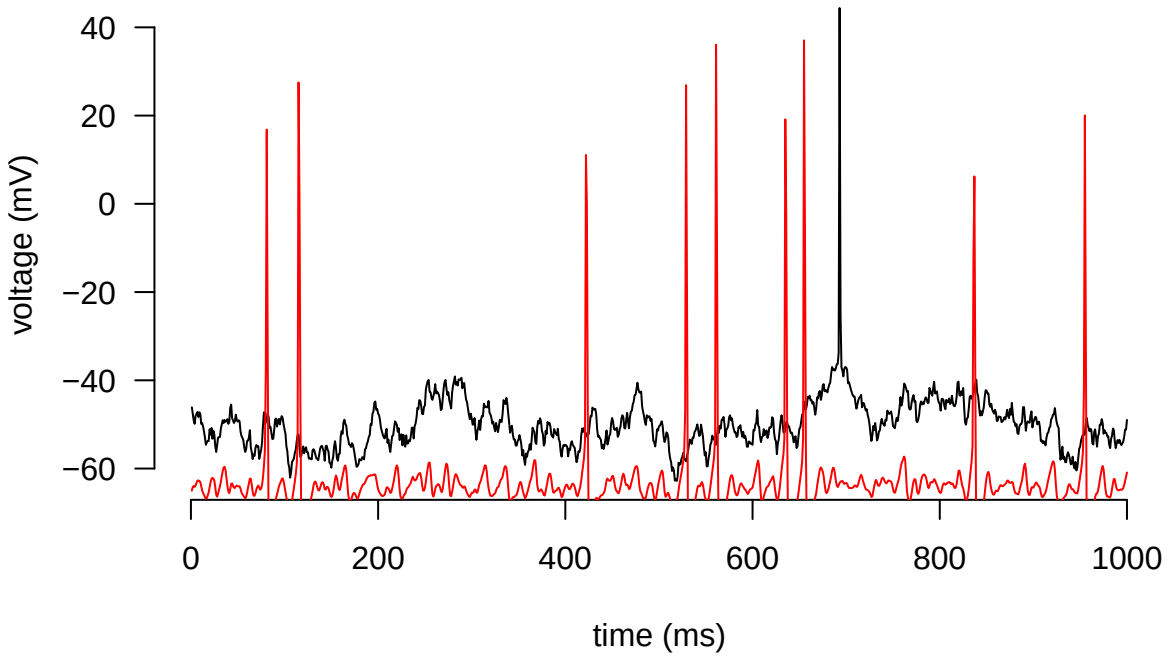
```
v.Iz <- sim.Izhikevich(I=input, Rm=0.045, tau=10, v.init=-50)
plot(r.cell, t="l", lty=1, axes=F, xlab="time (ms)", ylab="voltage (mV)"); axis(1); axis(2, las=2)
lines(v.Iz, col=2)
```



We repeat the same experiment now using a Hodgkin-Huxley cell - the HH cell has many parameters, including the activation and inactivation curves of the Na and the K channels, their conductances, K and Na reversal potentials, membrane capacitance, leak conductance. Importantly, these parameters have a biophysical interpretation, but they do not always have an intuitive, simple effect on the neuronal response.

```
I.ext <- approxfun(1:1000, input/1000/1000/10, method = "linear", rule = 2) # this is necessary to prov
params <- c(gK=gK, gNa=gNa, gL=gL, cm=cm, E.Na=E.Na, E.K=E.K, E.L=E.L) # parameters of the system.
state <- c(v=-65, m=.053, h=.596, n=.317) # initial state of the system
times <- 1:1000
v.HH <- ode(y = state, times = times, func = sim.HH, parms = params)
```

```
plot(r.cell, t="l", lty=1, axes=F, xlab="time (ms)", ylab="voltage (mV)"); axis(1); axis(2, las=2)
lines(v.HH[,1], v.HH[,2], col=2)
```



Homework

1. Look at the response of the three different model neurons! Which model is the most / least accurate? How can you measure the accuracy of the models, i.e., the match between the model's resposne and the real neuron's resposne? Calculate this metric for the three models and compare it with your intuition! [4p]
2. Select a stimulus response pair, and change the parameters of the integrate and fire (IF) model to match the actual data! Show how the accuracy metric changes after finding the best parameters? (You don't need to use a real optimiser, just try to change the parameters of the IF neuron intuitively!) [4p]
3. Select a stimulus response pair, and change the parameters of the Izhikevich neuron or the Hodgkin-Huxley cell to match the actual data! Show how the accuracy metric changes after finding the best parameters? (You don't need to use a real optimiser, just try to change the parameters of the IF neuron intuitively!) [4p]
4. Summarise your experience: Which of the three models was the easiest to fit? [2p] Which model yielded the best fit to the data (smallest training error / best accuracy)? [2p] What was the secret of the winning model? [2p] What are the typical errors you observe, and how could you correct them? [2p]