Poisson Process Practical Work

Sergio Peignier[1], Alexandre Foncelle

INSA Lyon, Biosciences department
BF2I Laboratory (UMR0203)
INSA Bâtiment Louis Pasteur - 11 avenue Jean Capelle
69621 Villeurbanne Cedex
`sergio.peignier@insa-lyon.fr`

1 Homogeneous Poisson process generator

1 Implement a Poisson process generator, using the `numpy.random.exponential` pseudo-random number generator. This function should receive as an input the parameter λ, and T the time during which the signal is recorded.
2 Implement a function that receives as an input a Poisson process realization, and returns the corresponding counting at a given time t.
3 Implement a function that estimates the λ parameter, of a list of Poisson process realizations, received as input.
4 Test the previous function using your Poisson process generator.

2 Modeling spike trains using Poisson process

2.1 Context

Along time, neurons receive and fire action potentials as a way to propagate information. Spike timing is an interesting element to characterize neural communication and is often used in modelling neural networks. One possible way to model the spikes arrivals, is to represent them using a random arrival process. In this practical work we will use a non-homogeneous Poisson process, to model the spike arrivals. The data that will be used in the context of this practical work, has been studied extensively in [1]. The authors' goal was to compare the neural activity of "Parkinsonian"[1] and "control" rats. The data analyzed here, has been collected on control individuals, using the following experimental protocol:

A rat underwent a surgery, in order to use an electrode to stimulate a region in the *cortex region (C)*, and another electrode to record the spikes delivered by a single neuron from the *substantia nigra pars reticulata region (SNR)*, during one second. In order to get different replicates, this procedure has been repeated 50 times.

In Figure 1 we schematize the different paths between C and SNR. Nevertheless all these paths are not systematically used, and this leads to different patterns in the neuron spikes generation.

[1] Parkinson-like behaviour could be induced by killing specific neurons.

2.2 Dataset

In this practical work, we are going to study and model the signals from two control individuals.

- The files `SHAM4_artefact_timing.txt` and `SHAM5_artefact_timing.txt`, contain the electrode stimulus time stamps, for individuals SHAM4 and SHAM5 respectively.
- The files `SHAM5_spike_timing.txt` and `SHAM4_spike_timing.txt`, contain the spikes time stamps, for individuals SHAM4 and SHAM5 respectively.

2.3 Questions

- Load the dataset.
- Implement a function to split the 50 replicates. You can record the replicates in a list of arrays:
  $[[t_{elec}^{(rep_1)}, t_{spike_1}^{(rep_1)}, t_{spike_2}^{(rep_1)}, \ldots], [t_{elec}^{(rep_2)}, t_{spike_1}^{(rep_2)}, t_{spike_2}^{(rep_2)}, \ldots], \ldots]$
- For each replicate set the 0 at the electrode stimulus.
- Only keep the epochs $s \in]0, 0.1]$ (between 0 and 100ms).
- Represent in the same plot, the spikes along time, using a different line for each replicate, as shown in Figure 2.
- Concatenate the signals and build a histogram, as in Figure 3.
- Using the counts produced by the histogram, estimate the rate $\lambda(t)$ at each interval[2]. To do so, you should use the properties of a Poisson counting process.
- In [1], the authors created a Poisson process simulator, to produce synthetic surrogate data, in order to fit properly the parameters of a neural network model, representing the system depicted in Figure 1. We are not going to fit any neural network model, ain't Nobody Got Time for That! But we are going to make a non-homogeneous Poisson process simulator, to model the neuron firing process that has been recorded.
  - Propose two methods to simulate the non-homogeneous Poisson process: one should be intuitive and, the other should rely on the properties of subdivision of Poisson processes.
  - Implement any one of the two methods (I suggest the second one).
  - Simulate 1000 replicates, and plot the corresponding histogram.
  - Compare the histogram with the real data one.
- Compare the patterns obtained for the two neurons, which are the paths followed in each case?

References

1. Foncelle, A.: Data-driven computational modelling for some of the implications of dopamine in the brain: from subcellular signalling to area networks. Ph.D. thesis, Université de Lyon; INSA Lyon (2018)

[2] If you use a 100 bins histogram for a 100ms signal, this is somehow like inferring a different $\lambda(t)$ for each 1ms interval.

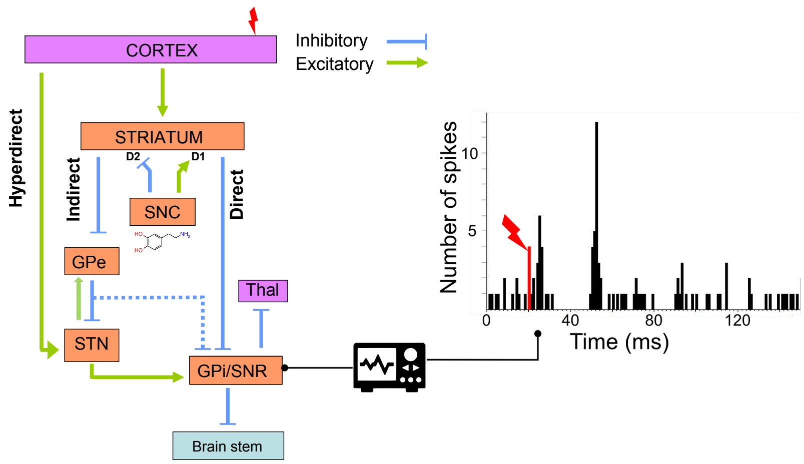

Fig. 1. Three main pathways between C and SNR. The fastest is the hyperdirect, then the direct and finally the indirect.

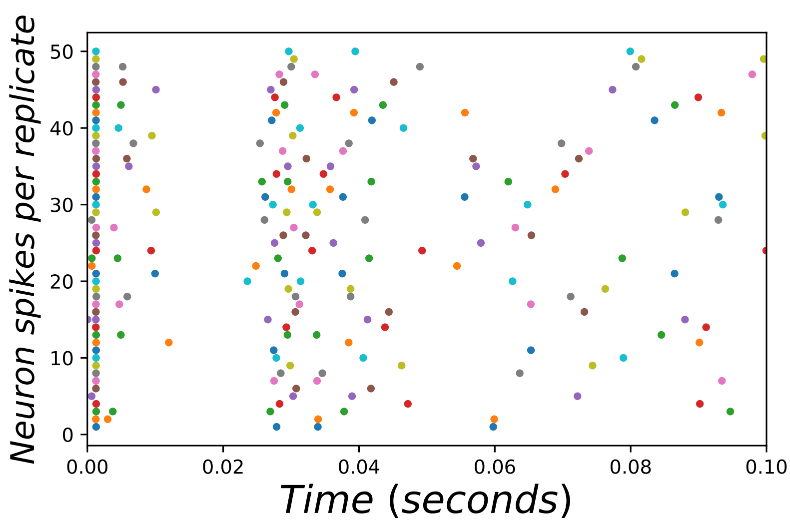

Fig. 2. Dots represent spikes along time, and each line and color represents a different replicate.

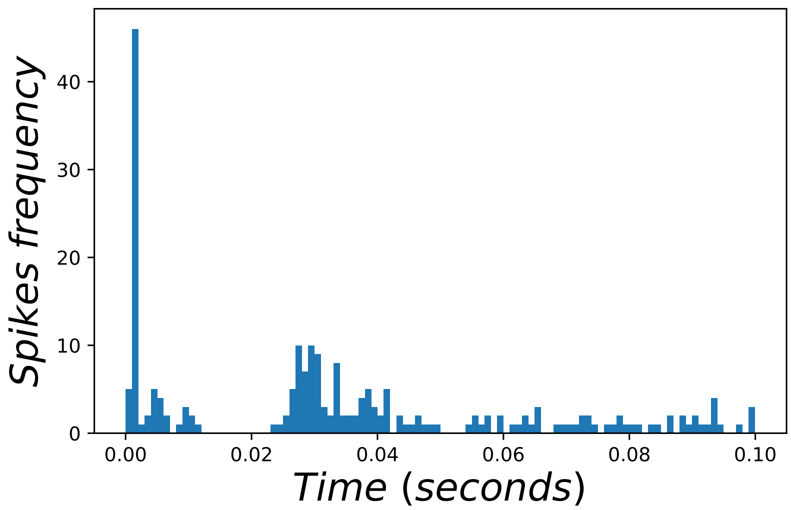

Fig. 3. Number of spikes per time interval (100 bins)