Acquisition and Analysis of Neural Data

Bernstein Center for Computational Neuroscience Berlin
Lecturer: Prof. Dr. Richard Kempter
Tutors: Naomi Auer, Gaspar Cano and Paula Kuokkanen

Analytical Problem Set 1 — Deadline: Tuesday 20th April, 2021 at 23:55

Please scan and upload your solution on Moodle as a single PDF file by the due date. Make sure that all text is clearly readable. For example, you could use the 'Adobe Scan' application on your mobile phone. Each student must submit their own solution. Do not forget to write your name and to mention if you collaborated with someone. Please specify how much time you needed to finish the exercise.

Convolution and Dirac delta

The convolution of two functions $u(t)$ and $v(t)$ is defined as

$$(u*v)(t) = \int_{-\infty}^{\infty} u(t-s)v(s)\,\mathrm{d}s\,. \tag{1}$$

Convolution is used widely in mathematics, physics, and engineering, e.g., in signal processing and linear systems' analysis. In neuroscience, it could be used to estimate firing rates from discrete spike times. With the following exercises, you will get an intuitive understanding of the convolution operation.

1. Graphical convolution (3 points)
The response $A = x*h$ of a linear, time invariant, system is given by the convolution of the input signal $x(t)$ and the system's impulse response $h(t)$. Here, we perform a "graphical" convolution between x and h, by computing the area under the product of the two functions. Consider the following functions:

$$x(t) = \begin{cases} 0 & \text{for } t < 0 \\ 3 & \text{for } 0 \leqslant t \leqslant 3 \\ 0 & \text{for } t > 3 \end{cases} \quad \text{and} \quad h(t) = \begin{cases} 0 & \text{for } t < 0 \\ 4 & \text{for } 0 \leqslant t \leqslant 1 \\ 0 & \text{for } t > 1\,. \end{cases}$$

(a) Sketch $x(t-s)$ and $h(s)$ as a function of s for a fixed, generic value of t. Identify the intervals of t for which the system has a different linear response. For each interval, sketch again $x(t-s)$ and $h(s)$ as a function of s. This will help you to compute the system's response.

(b) Compute the system's response $A(t) = \int_{-\infty}^{\infty} x(t-s)h(s)\,\mathrm{d}s$ and sketch the result.

2. Properties of the convolution (3 points)
Prove the following properties of the convolution:

(i) Commutative: $(u*v)(t) = (v*u)(t)$
(ii) Distributive: $(u*(\alpha v+\beta w))(t) = \alpha\,(u*v)(t) + \beta\,(u*w)(t)$ for all $\alpha,\beta \in \mathbb{R}$
(iii) Associative: $(u*(v*w))(t) = ((u*v)*w)(t)$

where $*$ stands for convolution. Hint: use the definition of the convolution and change of variables.

3. The Dirac delta (4 points)
The Dirac delta is often used to describe analytically the spike times of a neuron. This exercise gets you acquainted with the Dirac delta defined as a limit of functions. First, remember that the derivative of a function is defined as

$$\frac{\mathrm{d}}{\mathrm{d}t}x(t) = \lim_{h\to 0} \frac{x(t+h)-x(t)}{h} \quad . \tag{2}$$

The Dirac delta $\delta(t)$ can be constructed from a rectangular function with infinitely small width and infinitely large height.

(a) Define a rectangle $R(t)$ with width A and height $1/A$ using a difference of two Heaviside step functions. Use only rescaled and shifted versions of the Heaviside step function

$$u(t) = \begin{cases} 0 & \text{for} \quad t < 0 \\ 1 & \text{for} \quad t \geq 0 \end{cases} \tag{3}$$

(b) Define a new function $D(t)$ as the limit of $R(t)$ when the width A is arbitrarily small. Write out the function $D(t)$ explicitly.

(c) Show that for $D(t)$ the following properties hold:

  (i) $D(t) = 0$ for $t \neq 0$

  (ii) $\int_{-\infty}^{\infty} D(t)\,\mathrm{d}t = 1$

  (iii) $\int_{-\infty}^{\infty} f(t-s)D(s)\,\mathrm{d}s = f(t)$.

  For the first property, plot the function $D(t)$ and explain. For the third property, use a change of variables and the definition of the derivative. The three properties above can be used to define a Dirac delta, i.e., $\delta(t) := D(t)$.

(d) Can you think of other ways of constructing a function that fullfills the three properties in (c)? No proof required.

Numerical Problem Set 1 Deadline: Tuesday 20$^{\text{th}}$ April, 2021 at 23:55

Please upload your solution on Moodle by the due date. Return both a Python code file (or notebook) and a PDF or HTML file including figures and code. Please use clear, descriptive variable names and comments. Each student must submit their own files. Do not forget to write your name and to mention if you cooperated with someone. Please also indicate how much time you needed to complete the exercise.

Estimating firing rates from spike data

1. Firing rate estimation and convolution (10 points)

The goal of this exercise is to learn how to estimate firing rates from spike train data. To start, you need to download ExampleSpikeTimes1.dat from the Moodle course page.

This .dat-file can be loaded in Python with the numpy function loadtxt(). The file contains a variable named SpikeTimes, which is a vector of spike times t_i (in ms) with $i = 1, 2, \ldots, n$ in response to a stimulus with a duration of $T = 4000$ ms. The temporal resolution of the spike times is 0.1 ms. Hint: it is convenient to convert the data in seconds. Here, as in all other exercises, use numpy arrays and avoid loops as much as possible.

Your task is to estimate the firing rate with different methods. Hint: Your plots should be organized similar to Fig. 1.4 on page 12 of the Dayan and Abbott book.

(a) (1 point) Plot the raw spike train. Have a look at the pyplot function eventplot. You may need to set the axes limits manually for correct visualization.

(b) (3 points) Construct spike-count histograms with non-overlapping bins of widths $\Delta t = 20$, 50 and 150 ms, and normalize them to obtain a firing rate in spikes/s. Hint: Have a look at the pyplot function hist and the weights parameter.

(c) (5 points) Use the window functions defined below to estimate an approximate firing rate

$$r_{\text{approx}}(t) = \sum_{i=1}^{n} w(t - t_i) = \int_0^T \mathrm{d}\tau\, w(t-\tau)\rho(\tau)$$

  where $\rho(t) := \sum_{i=1}^{n} \delta(t - t_i)$ is the neural response function and T is the trial length. Hint: Have a look at the analytical exercises on the convolution and the Dirac delta on this sheet. You can use the numpy function convolve. Make sure you understand the differences between the convolution modes full and same, particularly regarding time offsets and boundary effects. You could plot the spikes on top of the rates to check that you obtained the correct time offset. Firing rates shall be plotted in units of spikes/s.

(i.) Estimate the rate with a rectangular window

$$w(\tau) = \begin{cases} 1/\Delta t & \text{if } -\Delta t/2 \leq \tau < \Delta t/2 \\ 0 & \text{otherwise} \end{cases}$$

with widths of $\Delta t = 20$, 50 and 150 ms. The rectangular window is *centered* at each spike, i.e., the filtering is non causal.

(ii.) Estimate the rate with a Gaussian window

$$w(\tau) = \frac{1}{\sqrt{2\pi}\sigma_\mathrm{w}} \exp\left(-\frac{\tau^2}{2\sigma_\mathrm{w}^2}\right)$$

with $\sigma_\mathrm{w} = 10$, 20 and 50 ms. Here the Gaussian is centered at each spike (non causal filtering).

(iii.) Estimate the rate with an alpha function

$$w(\tau) = \left[\alpha^2\, \tau\, \exp(-\alpha\tau)\right]_+$$

where $1/\alpha$ is a time scale and $[x]_+$ is the half-wave rectification function. Use $1/\alpha = 10$, 20 and 50 ms. Here the rate starts rising only *after* a spike is emitted, i.e., the filter is causal. Why would one want to use a causal rather than a non-causal filter?

(d) (1 point) Calculate the spike count-rate $r = \frac{n}{T} = \frac{1}{T}\int_0^T \mathrm{d}t\, \rho(t)$.

2. Extra: estimation of firing rate and trial averaging (3 extra points)

(a) (1 extra point) Estimate firing rates for the spike data in ExampleSpikeTimes2.dat to see if your code is flexible enough to handle it. Spike timings are in milliseconds, the trial length is $T = 10$ s, and the resolution is 0.1 ms.

(b) (2 extra points) Download ExampleSpikeTimes3.dat and calculate first the trial-averaged neural response and then the trial-averaged firing rate (only one convolution). You can use a filter of your choice. Load the data using the numpy function loadtxt() with the keyword parameter delimiter=';'. The file contains a matrix of spike times in milliseconds. The dimensions are [m x n, where m=100 is the maximum spike-time index and n=1000 is the number of trials. Note: each trial has a different number of spikes, the remaining entries are filled with nan. The trial length is $T = 1$ s, the resolution is 0.1 ms.

Naomi Auer	Phone: 2093 98407	Email: naomi.auer at posteo.de	Office: Haus 4, Room 106
Gaspar Cano	Phone: 2093 98407	Email: gaspar.cano at hu-berlin.de	Office: Haus 4, Room 106
Paula Kuokkanen	Phone: 2093 98407	Email: paula.kuokkanen at hu-berlin.de	Office: Haus 4, Room 106