Course Module
Spectral analysis of spike responses

Jan Grewe

Eberhard Karls Universität Tübingen, Institut für Neurobiologie, Auf der Morgenstelle 28E, 72076 Tübingen
mail: jan.grewe@uni-tuebingen.de

Synopsis

Sensory systems serve to transduce information about the outside world into a code the nervous system can digest. In this module we will analyze data from sensory neurons in the electrosensory system of the weakly electric fish (*Apteronotus leptorhynchus*).
We will investigate the relations between the presented stimulus and the neuronal response with measures in the time and frequency domain. We will use cross-correlations, spike-triggered averages (reverse correlations), then turn to the power spectrum, convolution in frequency and time domain. Finally, we may assess the coding performance by means of the coherence function and the mutual information.

Supplemental Material

With this course module we provide an Introductory Lecture *Spectral analysis of sensory responses* and data files for practical analysis.

Requirements

Practical work on this course module requires that you run Matlab Version 7 or higher. Supplemental data files are required for data analysis.

Introduction

There are several ways to characterize neuronal responses. Many of the classical methods characterize the responses in the temporal domain. Since the late 90's information theory and methods in the frequency domain were introduced to the neuroscience. Nowadays methods like power or coherence spectra are very widespread and belong to the standard methods applied. Many methods are readily available in analysis environments like MATLAB to be used "out of the box". Nevertheless, some details need to be taken into consideration when applying them.

In the following exercises we will analyze spiking responses of sensory neurons in the electrosensory system of the weakly electric fish *Apteronotus leptorhynchus*. These neurons, the so called p-units, code for changes in the electric field amplitude surrounding the fish. The animal uses this information for navigation and prey-detection as well as communication purposes.

Spiking responses of sensory neurons in the weakly electric fish

The **'data_p-unit.mat'** and **'data_ampullary.mat'** Matlab files contain variables with the neuronal responses and the respective stimuli for so called p-unit electroreceptors as well as ampullary receptors. The responses originate from (semi-)intracellular recordings of such neurons. The cells respond to modulations of the electric field surrounding the fish. The stimuli are thus the vectors of such random modulations. Stimulus duration was 10 s and the same stimulus has been presented for a number of trials.

The responses have been sampled at 20 kHz and the 'responses_strong' and 'responses_weak' variables contains a matrices of zeros and ones in which '1' marks the occurrence of a spike. The first dimension represents time, the second dimension the trials. 'strong' and 'weak' denote the intensity of the stimulus.

1. Taking a look at the raw data

The first step is, of course, a look at the raw data to get an impression of stimulus and responses and assess the relationship between both.

Task:

1. Create a plot that shows the stimulus and the responses (use the subplot command). For the spiking responses of the p-units create create e.g. a peri-stimulus-time-histogram (PSTH), or a rasterplot.

2. Create a PSTH with the Kernel-Density approach. i.e., each 'one' in the binary representation of the responses is replaced by a Gaussian kernel. What is the appropriate bandwidth of the kernel? Try to find a kernel width that seems plausible for you.

3. Use the MISE kernel density optimization for this. Just use the **'sskernel.m'** function to find the optimum bandwidth.

4. Use the cross-correlation (use MATLAB's '**xcorr.m'** function) between stimulus and response to get a feeling on the timescales on which stimulus and responses are coupled. What is the delay?

2. The power spectrum

The power spectrum is a widely used way of characterizing neurophysiological data in the frequency domain. It shows for each frequency bin the relative contribution to the structure of the overall data. Before analyzing real data we will "play" around with it.

The Power spectrum is defined as:

$$P(\omega) = |F(\omega)|^2,$$

with $F(\omega)$ the Fourier spectrum. $||$ denote the absolute value which is the magnitude or the "complex modulus" of the complex Fourier coefficients.

The Fourier spectrum can be calculated using the MATLAB **fft.m** function. It returns a vector of 2N+1 elements where N is the length of the data stretch analyzed in samples. Each element in this vector contains the Fourier coefficients from the zero frequency in regular steps to the Nyquist frequency and then a mirror image of the negative frequencies. Usually, only the first N+1 values are of interest. The frequencies represented in the Fourier spectrum depend on both the sampling rate and the size of the data stretch analyzed. Besides the zero frequency the next lowest resolvable frequency is given by the segment duration (e. g. 1024 Hz sample rate and a one second stretch of data: lowest frequency is 1 Hz) The highest frequency is the Nyquist frequency (sampleRate/2 Hz). This leads to the fact that an estimation of a signals power spectrum does **not** become more accurate if longer data stretches are analyzed! For each frequency bin the accuracy stays the same it is just that the frequency is better resolved with more frequency bins. There are two ways by which the accuracy can be increased: (I) the spectrum is calculated for the full amount of data and summing neighboring frequency bins, or (ii) calculating spectra for several segments of the data and then averaging across these. Both ways sacrifice resolution for accuracy. We will use the second approach for it is computationally less expensive.

a) Effect data segmentation
Before analyzing real data we will estimate the power spectra of artificial data that is assembled from several sinusoids and contains some additive Gaussian noise.
Load the **'surrogateData.mat'** file. There is 10 seconds of data sampled at 4096 Hz.

Tasks:
1. Plot the data and try to detect by visual inspection which frequencies are contained.
2. Create power spectra of the data which frequencies can be identified?
3. Use data stretches of different lengths. What is the effect of this data segmentation on the spectrum?

Note: We do not care for correct normalization of the power spectrum or for windowing of segments...

b) Power spectra of p-unit responses
Calculating power spectra of spike responses is a little bit more difficult. We'll do it in several steps.

Tasks:

1. Create a power spectrum of the stimulus to see the relevant frequency range.
2. Create a power spectrum of an individual response trace. Why do the responses have power in a range the stimulus does not?

In the case of these datasets (and it is quite commonly done this way) spike times are

represented by ones in the response matrix. This is an approximation of a Dirac pulse which has a white spectrum.

Thus, in order to remove these effects the ones have to be replaced by something smoother, usually a Gaussian kernel.

3. Convolve the response traces with a Gaussian kernel and calculate the power spectra. (You can use the **'gaussKernel.m'** function to generate a kernel and apply it to the data using MATLAB's **'conv.m'** function). Plot the data as a function of time. Use the optimal kernel as estimated above. Create a power spectrum of the responses. What does it show?
4. Apply a set of kernels (smaller and wider than the optimum kernel) and create the power spectra. What effects do you observe?
5. Now apply a windowing function to the segments and observe the effect on the spectrum. (Windows can be created using the MATLAB's **hann**, **hamming**, … functions.)

3. Coherence spectrum

The coherence allows relating stimulus and response. The coherence is a measure that estimates how well e.g. the stimulus can be estimated from the responses using the optimal linear filter.

$$\gamma^2 = \frac{\langle S*R \rangle \langle R*S \rangle}{\langle S*S \rangle \langle R*R \rangle} \ ,$$

where R and S are the Fourier spectra of response and stimulus, respectively, * denotes the complex conjugate and <> indicate averages across segments.

The coherence assumes values between 0 and 1 while 0 indicates that there is no linear relation between stimulus and response and 1 indicates perfect a linear relationship. The complex conjugate can be calculated using MATLAB's **'conj.m'** function.

Tasks:
1. Implement the coherence function and create coherence spectra for p-unit data. Keep in mind that you have to split the data into at least two segments. What can be observed?
2. If there is a deviation from 1? What might be the reason?
3. Calculate the "expected" coherence spectra between the average response and the individual responses and plot the average of these. What might this be good for?

4. Spike-triggered average and reverse reconstruction

In these tasks we will use the spike-triggered average (STA) and the reverse reconstruction methods. Both methods allow to assess which features of the stimulus are represented in the neuronal responses.

a) *R*everse reconstruction

Reverse reconstruction is the attempt to reconstruct e.g. the stimulus by filtering the responses with the so called reverse filter, H(f). This filter can be estimated by:

$$H_{rev}(f) = \frac{\langle R*(f) \cdot S(f) \rangle}{\langle S*(f) \cdot S(f) \rangle} \ ,$$

with R(f) and S(f) the Fourier spectra of response and stimulus, respectively. * denotes the complex conjugate and <> indicate across segment averaging. This filter can now be used to create an estimate of the stimulus.

$$S_{est} = H_{rev}(f) \cdot R(f)$$

Tasks:
1. Estimate the reverse filter and filter the responses with it. Compare the estimated stimulus with the original.
2. Estimate the forward filter, which can be analogously calculated by flipping R and S in the equation above. Use only half of the data and then apply the filter it to estimate the response to the second half. Compare estimated and real response.

b) Spike-triggered average (STA)
The spike-triggered average is a representation of the average stimulus that precedes a spike. It can be used to reconstruct the stimulus from the responses. To create an STA one simply averages the stimulus that precedes all spikes.

1. Create the STA of the a response and and plot it.
2. Reconstruct the stimulus by convolution of the spike train with the STA.
3. Use only fractions of the responses to estimate the STA and use it to reconstruct the stimulus that led to the other parts of the data. Compare the estimated stimulus with the original one.

5. All-spikes vs. synchronous spikes (optional)
Here we will investigate the effects of different readout strategies on the information that can be extracted from the sensory input. In the first strategy a hypothetical postsynaptic neuron integrates all incoming spikes and its output is simply the sum of the presynaptic inputs. In the second scheme the postsynaptic neuron fires a spike only if the spikes in the inputs are synchronous.

Tasks:
1. Estimate the stimulus-response coherence of a hypothetical postsynaptic neuron that sums the inputs of the two neurons. We will simulate this by using pairs of responses. Calculate the average coherence for all combinations.
2. Do the same for a neuron that fires only if the spikes in the inputs are synchronous. Use different levels of synchrony. What is the effect of the different "synchrony" windows?
3. Do the same thing for the ampullary data.
4. Combine more than two input signals...

Acknowledgements