**Computer challenges for the spike-field cohernece**

1. Load the file **spikes-LFP-2.mat** available at the [GitHub repository](https://github.com/Mark-Kramer/Case-Studies-Python/tree/master/5.%20Cross-Frequency%20Coupling) into Python. You will find three variables. The variable *y* corresponds to the LFP data, in units of millivolts. The variable *n* corresponds to simultaneously recorded binary spiking events. The variable *t* corresponds to the time axis, in units of seconds. Both *y* and *n* are matrices, in which each row indicates a separate trial, and each column indicates a point in time. Use these data to answer the following questions.

a. Visualize the data. What rhythms do you observe? Do you detect associations between the LFP and spikes?

b. Plot the spectrum versus frequency for these data. Are the dominant rhythms in the spectrum consistent with your visual inspection of the data?

c. Compute and display the STA and FTA. Do you find evidence for associations

between the two types of data?

d. Compute and display the spike-field. Do you find evidence for spike-field coherence?

e. Describe your results, as you would to a colleague or collaborator.

1. Load the file **spikes-LFP-3.mat** available at the [GitHub repository](https://github.com/Mark-Kramer/Case-Studies-Python/tree/master/5.%20Cross-Frequency%20Coupling) into Python. You will find three variables. The variable *y* corresponds to the LFP data, in units of millivolts. The variable *n* corresponds to simultaneously recorded binary spiking events. The variable *t* corresponds to the time axis, in units of seconds. Both *y* and *n* are matrices, in which each row indicates a separate trial, and each column indicates a point in time. Use these data to answer the following questions.

a. Visualize the data. What rhythms do you observe? Do you detect associations between the LFP and spikes?

b. Plot the spectrum versus frequency for these data. Are the dominant rhythms in the spectrum consistent with your visual inspection of the data?

c. Compute and display the spike-field. Do you find evidence for spike-field coherence?

d. Describe your results, as you would to a colleague or collaborator.

1. In this question, we consider a simple example that illustrates the fundamental features of spike-field coherence. Let’s consider the case in which the field is a sinusoid plus Gaussian noise, and the spike train is drawn from a Bernoulli distribution. For a Bernoulli distribution, the probability of a spike at any time is not related to previous spiking behavior. In this case, we also assume no relation between the field and point process. Therefore, we expect to find no spike-field coherence. Let’s simulate some synthetic data in Python, compute the spike-field coherence, and see what we find.

As a first step, create 100 trials of multiscale data, each trial of 1 s duration

with a sampling rate of 1000 Hz. Then define the synthetic data. To do so, create in each trial a field, which here will be a sinusoid; and a spike train, which here will be drawn from a binomial distribution with a probability *p* of a spike in each sampling interval:

n = 1

p = 0.01

s = np.random.binomial(n, p, 1000)

With these synthetic multiscale data defined, repeat the analysis we performed in class. In particular,

a. Visualize the data. What rhythms do you observe? Do you detect associations between the LFP and spikes?

b. Plot the spectrum versus frequency for these data. Are the dominant rhythms in the spectrum consistent with your visual inspection of the data?

c. Compute and display the spike-field. Do you find evidence for spike-field coherence?

d. Describe your results, as you would to a colleague or collaborator.

Do the results of each method match your expectations? In particular, do you expect to observe spike-field coherence between these simulated data?