**Homework for spike-field coherence. Due Dec 5, 2019.**

1. Load the file **spikes-LFP-2.mat**, available at the GitHub repository 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.
   1. Visualize the data. What rhythms do you observe? Do you detect associations between the LFP and spikes?
   2. Plot the spectrum versus frequency for these data. Are the dominant rhythms in the spectrum consistent with your visual inspection of the data?
   3. Compute and display the spike-field coherence. Do you find evidence for spike-field coherence?
   4. Describe (in a few sentences) your results, as you would to a colleague or collaborator?
2. Load the file **spikes-LFP-3.mat**, available at the GitHub repository 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.
   1. Visualize the data. What rhythms do you observe? Do you detect associations between the LFP and spikes?
   2. Plot the spectrum versus frequency for these data. Are the dominant rhythms in the spectrum consistent with your visual inspection of the data?
   3. Compute and display the spike-field coherence. Do you find evidence for spike-field coherence?
   4. Describe (in a few sentences) your results, as you would to a colleague or collaborator?
3. 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 random; 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, 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. We define these parameters as follows:  
  
 K = 100; #Define no. of trials.  
 N = 1000; #Define no. of samples per trial.

dt = 0.001; #Define sampling interval.

Now, let’s define the synthetic data. We create in each trial a field, which here will be a sinusoid; and a spike train, which here will be drawn from a Bernoulli distribution with a probability *p* of a spike in each sampling interval.

﻿y = np.zeros([K,N]); #Matrix to hold field data.

n = np.zeros([K,N]); #Matrix to hold spike data.

﻿for k in np.arange(K): #For each trial ...

y[k,:] = np.sin(2.0\*np.pi\*np.arange(N)\*dt \* 10)+0.1\*np.random.randn(1,N);

n[k,:] = np.random.binomial(1,0.01,N)

In this code, the frequency of the sinusoid is set to 10 Hz, and the probability of a

spike in each sampling interval is set to p=0.01. These choices are reasonable yet arbitrary. With these synthetic multiscale data defined, repeat the analysis you performed on the data above. In particular,

* 1. Visualize the data. What rhythms do you observe? Do you detect associations between the LFP and spikes?
  2. Plot the spectrum versus frequency for these data. Are the dominant rhythms in the spectrum consistent with your knowledge of the data?
  3. Compute and display the spike-field coherence. Do your results match your expectations?
  4. Describe (in a few sentences) your results, as you would to a colleague or collaborator?

1. Let’s now consider a simulation in which we expect a nonzero spike-field coherence. To produce a multiscale interaction, we must introduce a relation between the spikes and the field. We do so by making the probability of a spike in each time interval a function of the field. Let’s fix the number of trials (K=100), the number of samples per trial (N=1000), and the sampling interval (dt=0.001) at the same values used in the first simulation example. Then let’s define the spike and field data for each trial.

﻿ f = 0.01; #Parameter for scaling of rate.

b = 1; #Parameter for background spiking.

y = np.zeros([K,N]); #Matrix to hold field data.

n = np.zeros([K,N]); #Matrix to hold spike data.

for k in np.arange(K): #For each trial ...

# ...define the LFP as a 10 Hz sinusoid + noise.

y[k,:] = np.sin(2.0\*np.pi\*np.arange(N)\*dt \* 10)+0.1\*np.random.randn(1,N);

# ...draw spikes from a Bernoulli distribution,

p = f\*(b+np.exp(y[k,:])); #...with probability dependent on LFP

n[k,:] = np.random.binomial(1,p,N);

Note that the probability *p* of a spike in each time interval depends on three factors: (1) an overall scaling term *f*, (2) a baseline level of probability *b*, and (3) the exponentiated field *exp(y)* . We exponentiate the field *y* so that the probability is always positive. In this way, the probability of a spike depends on the field.

With these synthetic multiscale data defined, repeat the analysis you performed on the data above. In particular,

* 1. Visualize the data. What rhythms do you observe? Do you detect associations between the LFP and spikes?
  2. Plot the spectrum versus frequency for these data. Are the dominant rhythms in the spectrum consistent with your knowledge of the data?
  3. Compute and display the spike-field coherence. Do your results match your expectations?
  4. Describe (in a few sentences) your results, as you would to a colleague or collaborator?