Signal theory overview and data viz

Christopher G. Wilson, Ph.D.
Associate Professor Physiology and Pediatrics
Center for Perinatal Biology

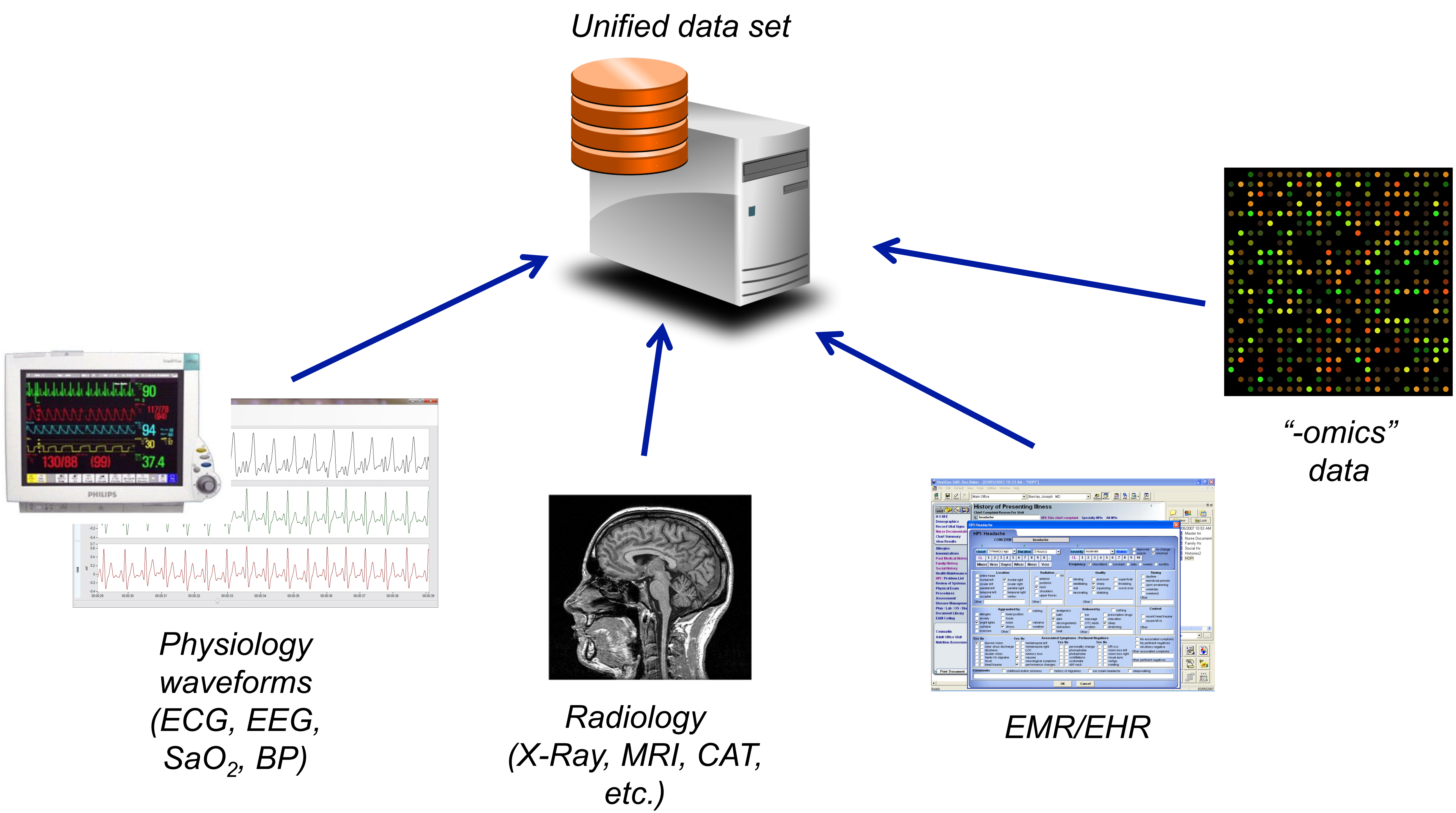
LOMA LINDA UNIVERSITY

Big Data Analytics, April 16th, 2015

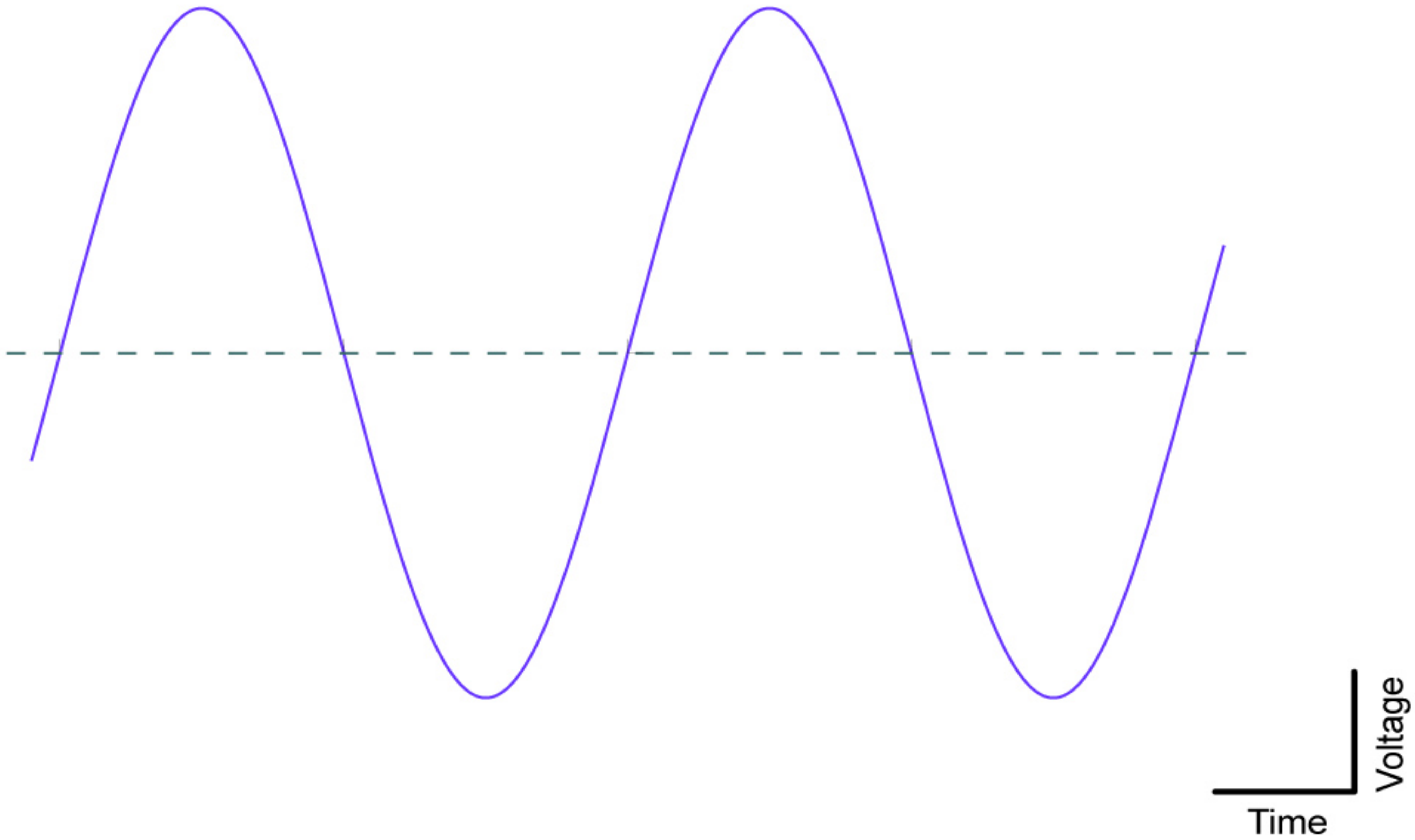
Outline

» Moving from the analog to the digital world
» Nuts and bolts of sampling
» Data takes up space!
» Time-series plots
» Moving from the time-domain to the frequency-domain
» Ways to represent variability in data

Data is multi-modal



Analog signals are continuous…

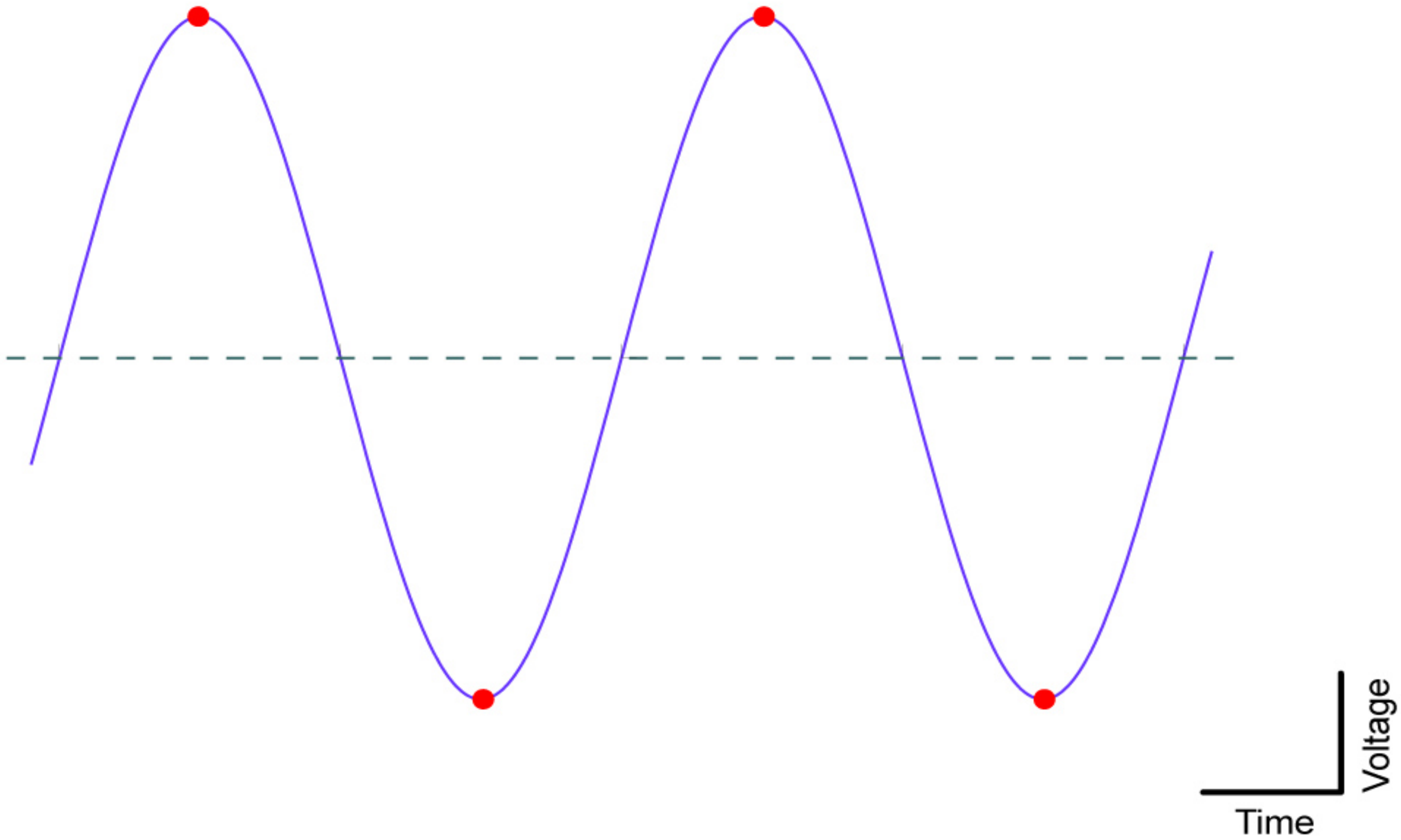


Nyquist-Shannon "Criterion"

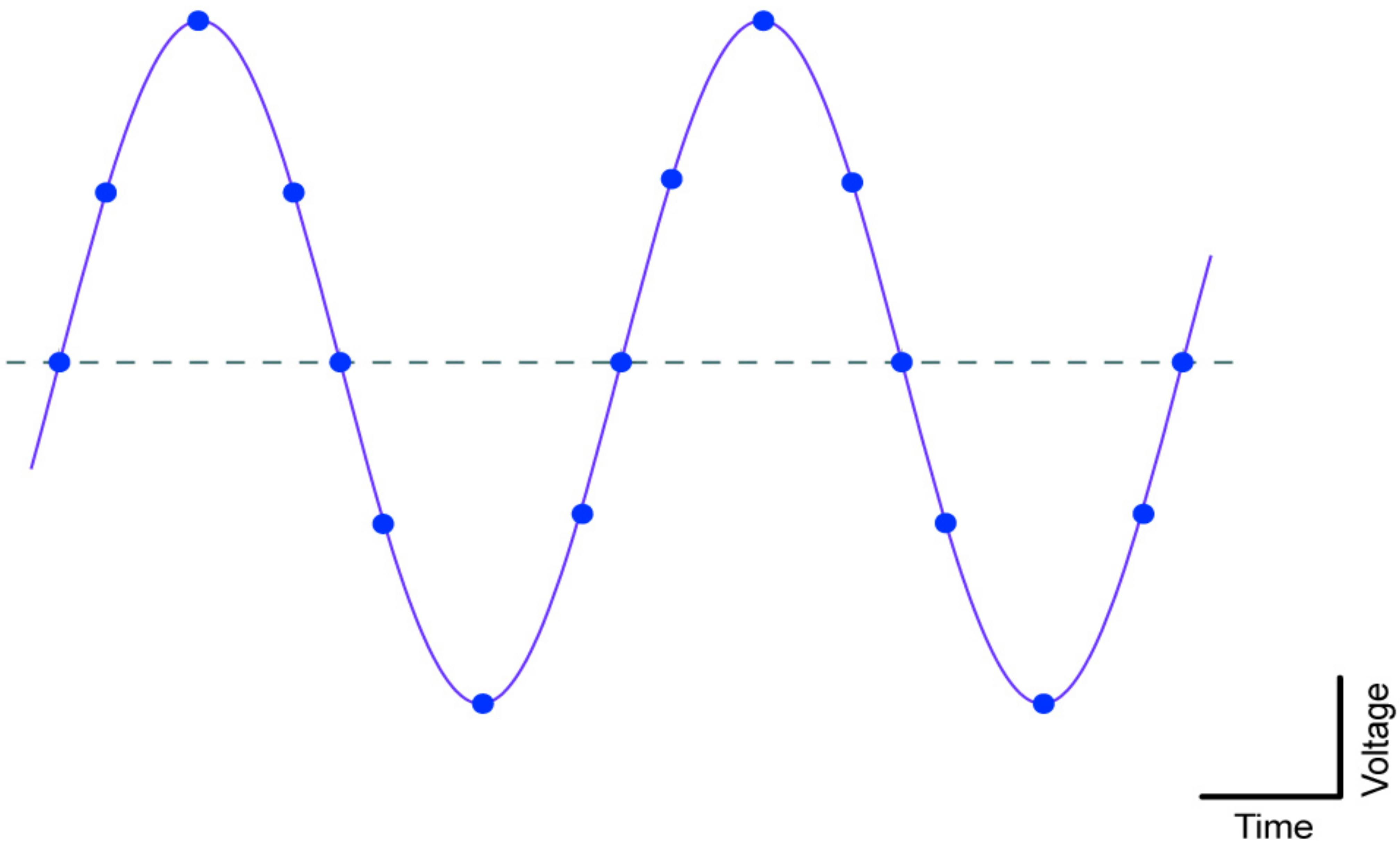
- The ***Nyquist–Shannon sampling theorem*** is a fundamental result in signal processing. Sampling is the process of converting a signal (for example, a function of continuous time or space) into a numeric sequence (a function of discrete time or space). The theorem states:
  - ~ *If a function x(t) contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced 1/(2B) seconds apart*.

- This means that a *band-limited* analog signal that has been digitally sampled can be *perfectly* reconstructed from a sequence of samples if the sampling rate exceeds 2×B samples per second, where B is the highest frequency of interest contained in the original signal.

And sampled at 2x their highest frequency…



But it's better to sample more!



All that data adds up!

- Storage space required = (# of channels) × (sampling rate) × (recording time)
- If we record respiration, ECG, and Pulse-Ox at a very slow sampling rate (50 samples per second).
- And four channels of EEG (1000 samples per second).
- Over 12 hours of continuous monitoring we would collect ***~200 Megabytes*** of data for a single patient!

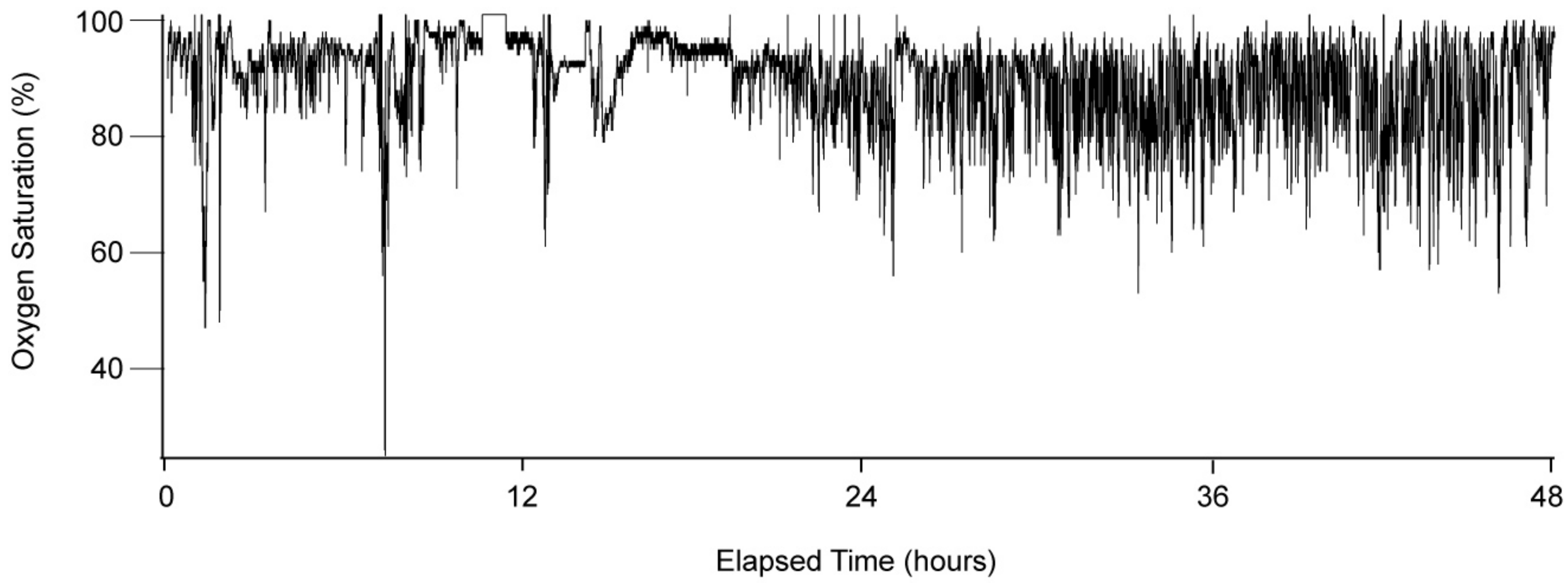
Long-term Data Storage

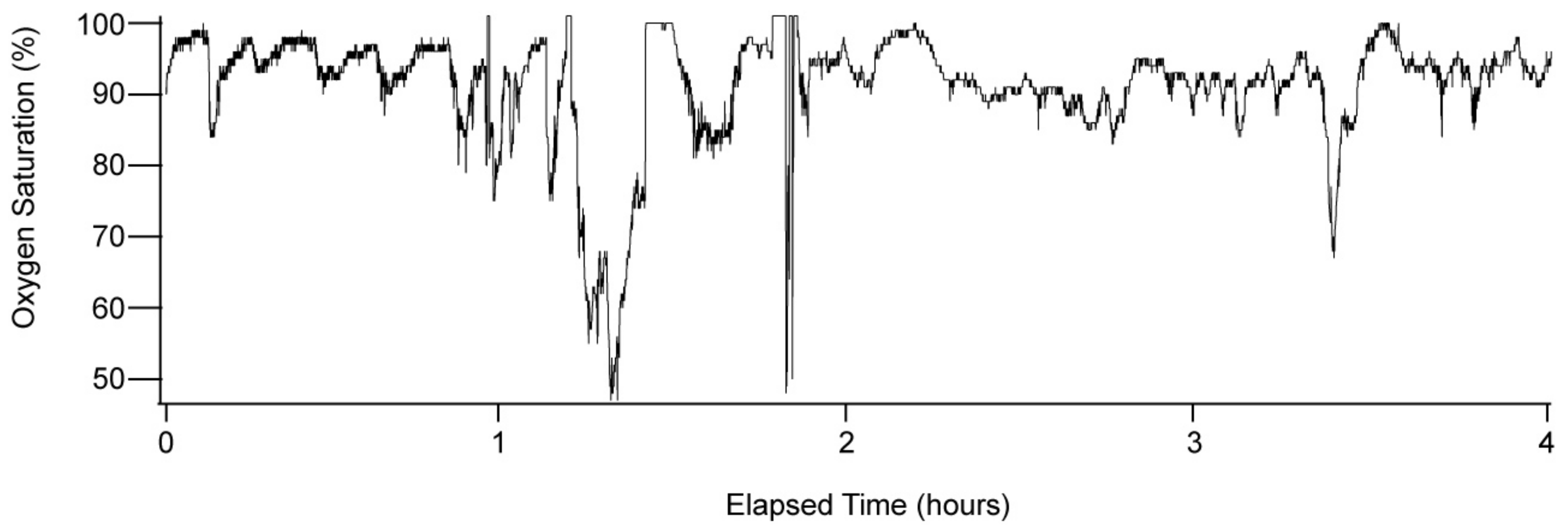
» Luckily disk storage is now very cheap (approximately $80/terabyte).

» However, with 100s of patients in the hospital per year, even with only a few hours of limited recording per patient, the data will become prohibitive to manage locally.

» Computer operating systems that can handle large datasets in memory have only recently become more common (32 bit versus 64 bit).

Example of Long-term Acquisition





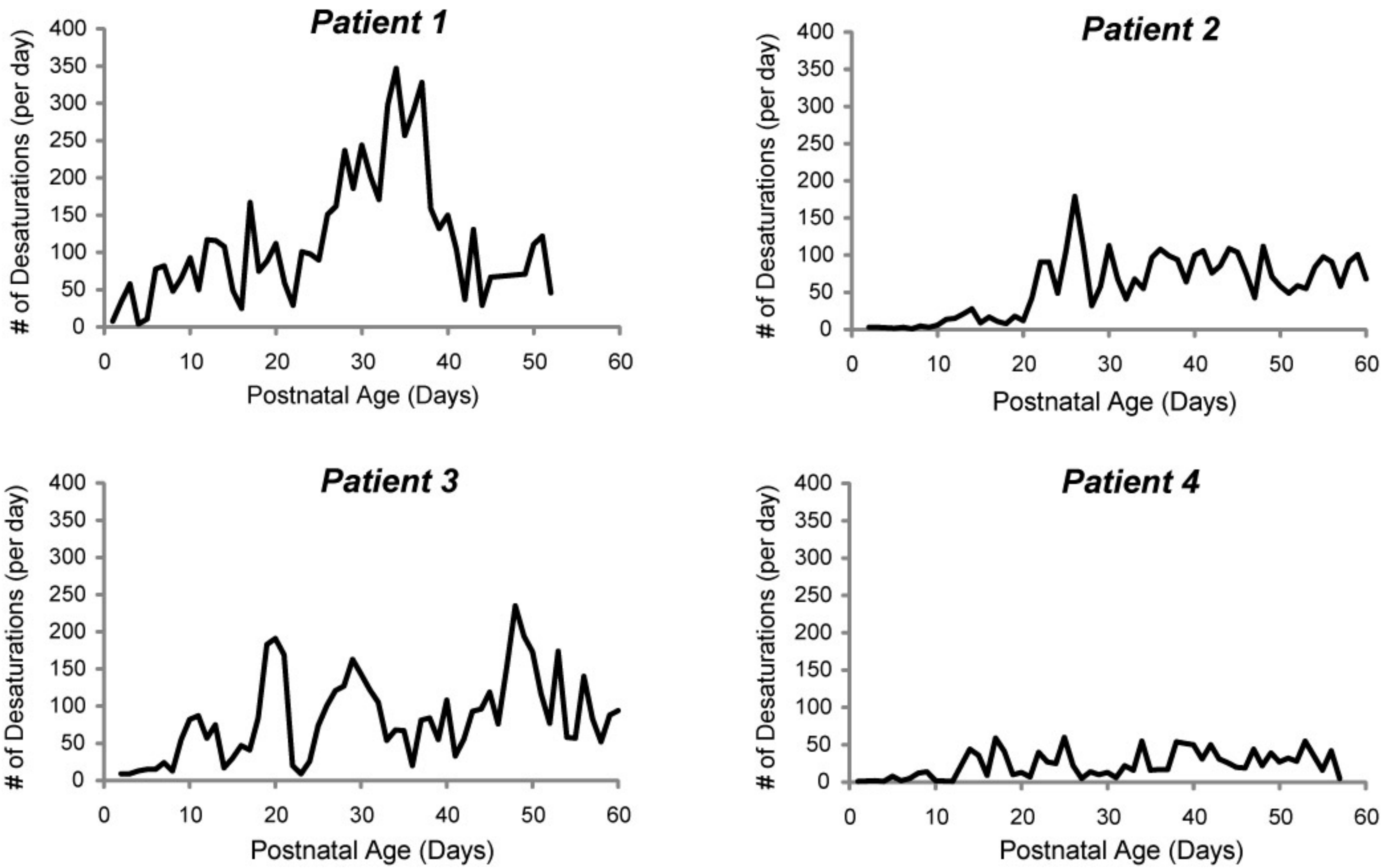
Why filtering is important

- » To avoid *aliasing* your acquired signal, you need to bandpass the signal and have the appropriate sampling frequency to assure that you have an accurate representation of your analog signal in your digital world.
- » There are a variety of filters to help you with this problem: low-pass, high-pass, band-pass, notch, non-linear, etc.
- » Understanding how your data has been filtered is KEY to obtaining the signals that you are interested in.

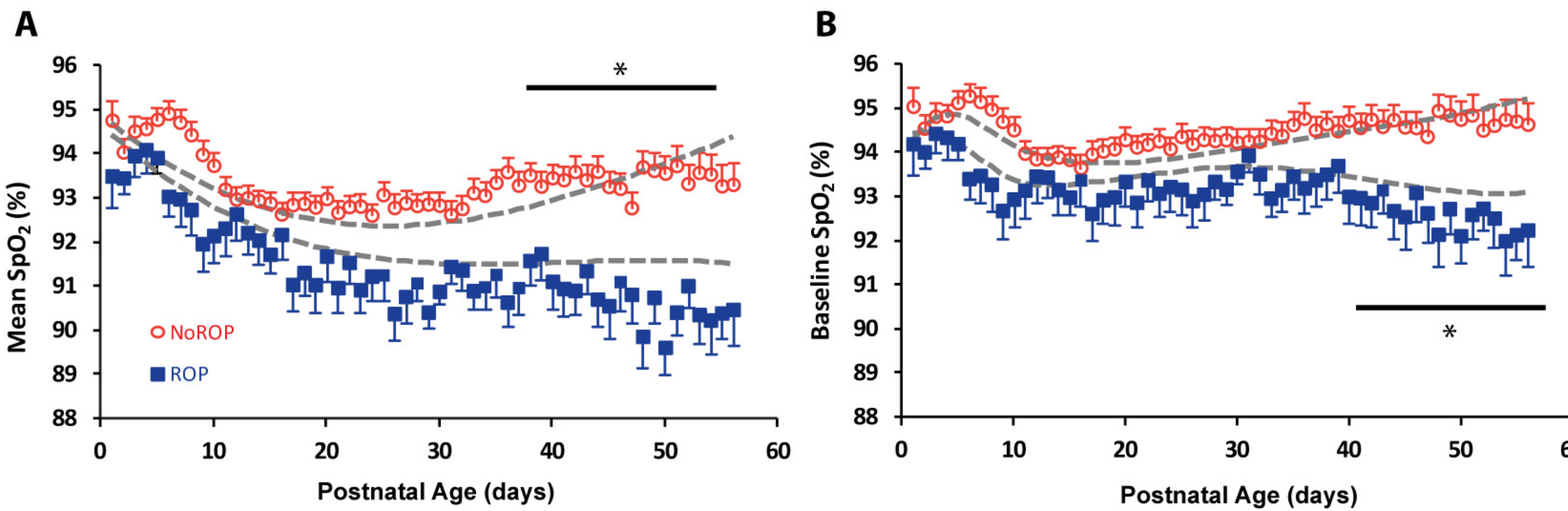
Artifact sources

» Patient moves, dislodging the finger cuff
» Patient is moved by transport to another location
» Equipment malfunction
» Movement artifact
  ~ These sources of artifact can happen with any signal source!

Time series data



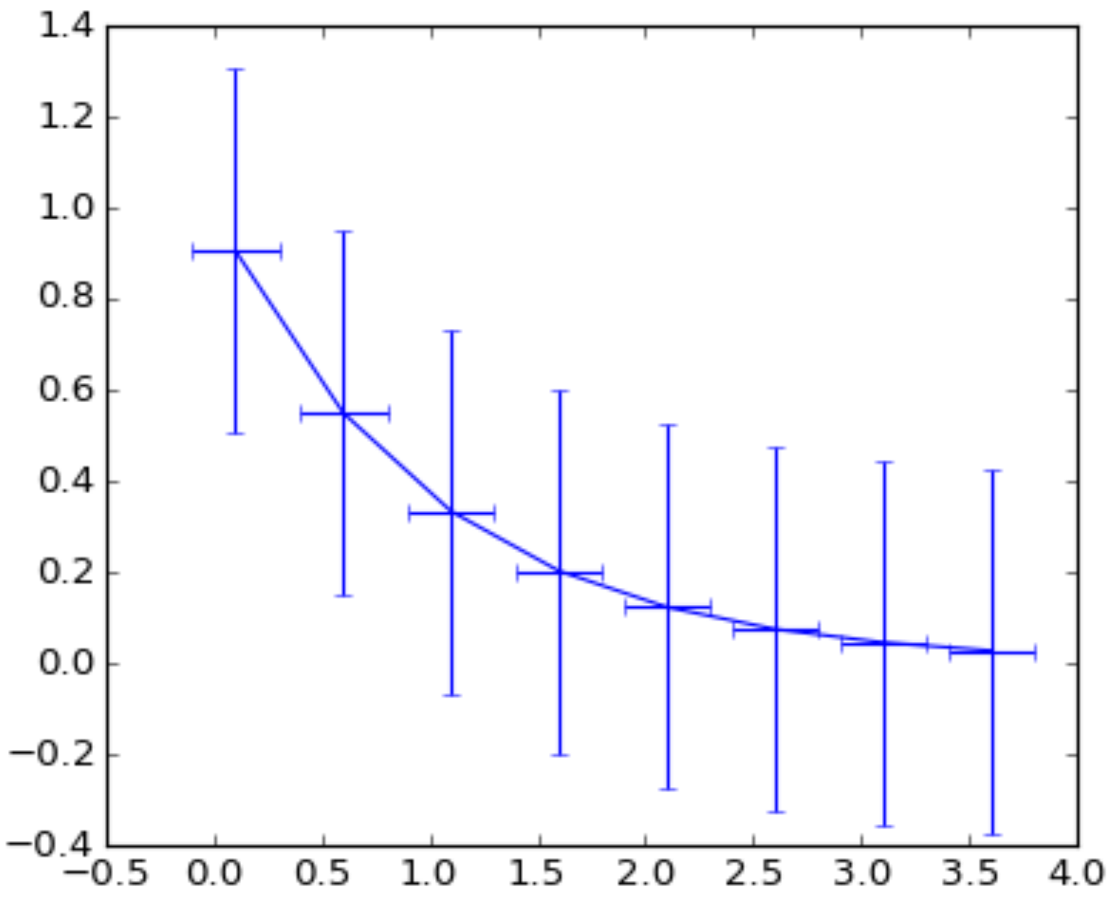
Mean and Baseline saturations differ in ROP and mild/noROP groups



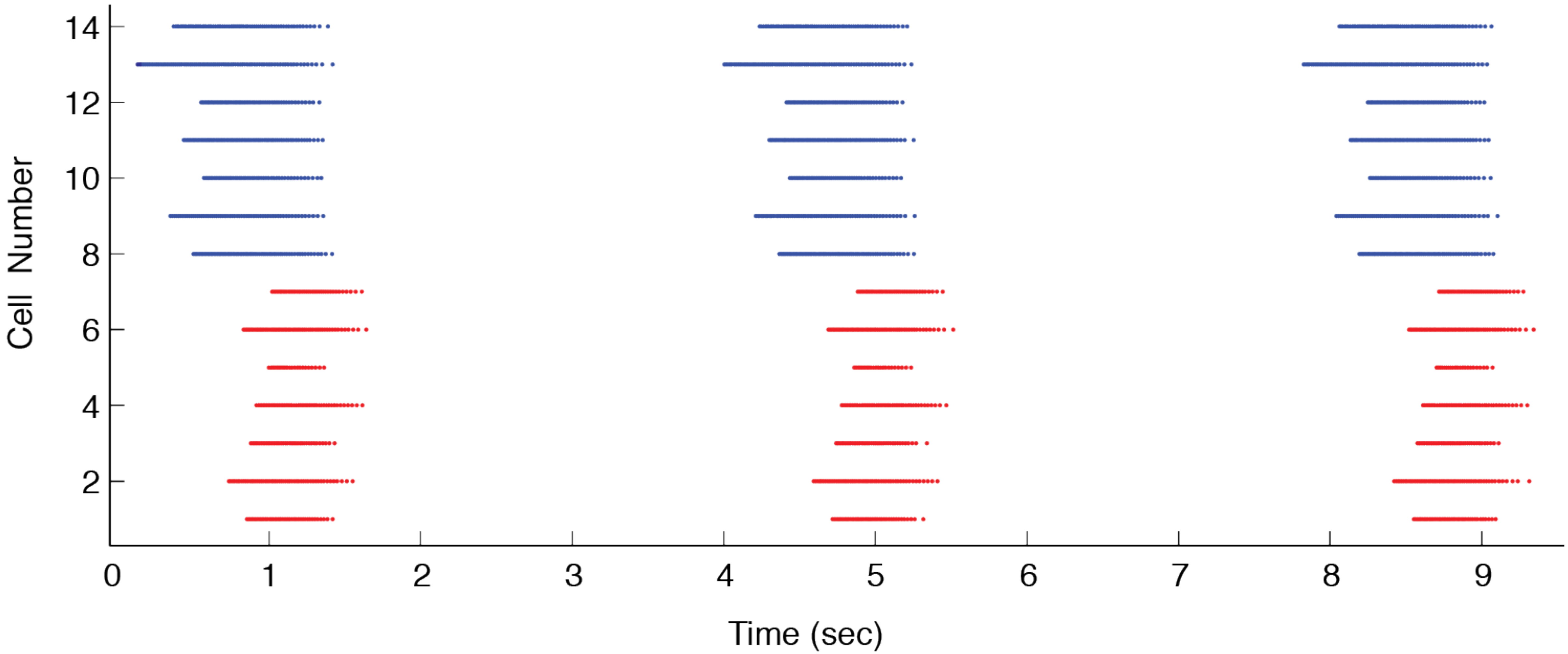
$* p < 0.05$

Line plots with error bars

```
import numpy as np
import matplotlib.pyplot as plt

# example data
x = np.arange(0.1, 4, 0.5)
y = np.exp(-x)

plt.errorbar(x, y, xerr=0.2, yerr=0.4)
plt.show()
```



Raster plots of multiple events over time



Christopher Fietkiewicz

The Fourier Transform in a couple of lines….

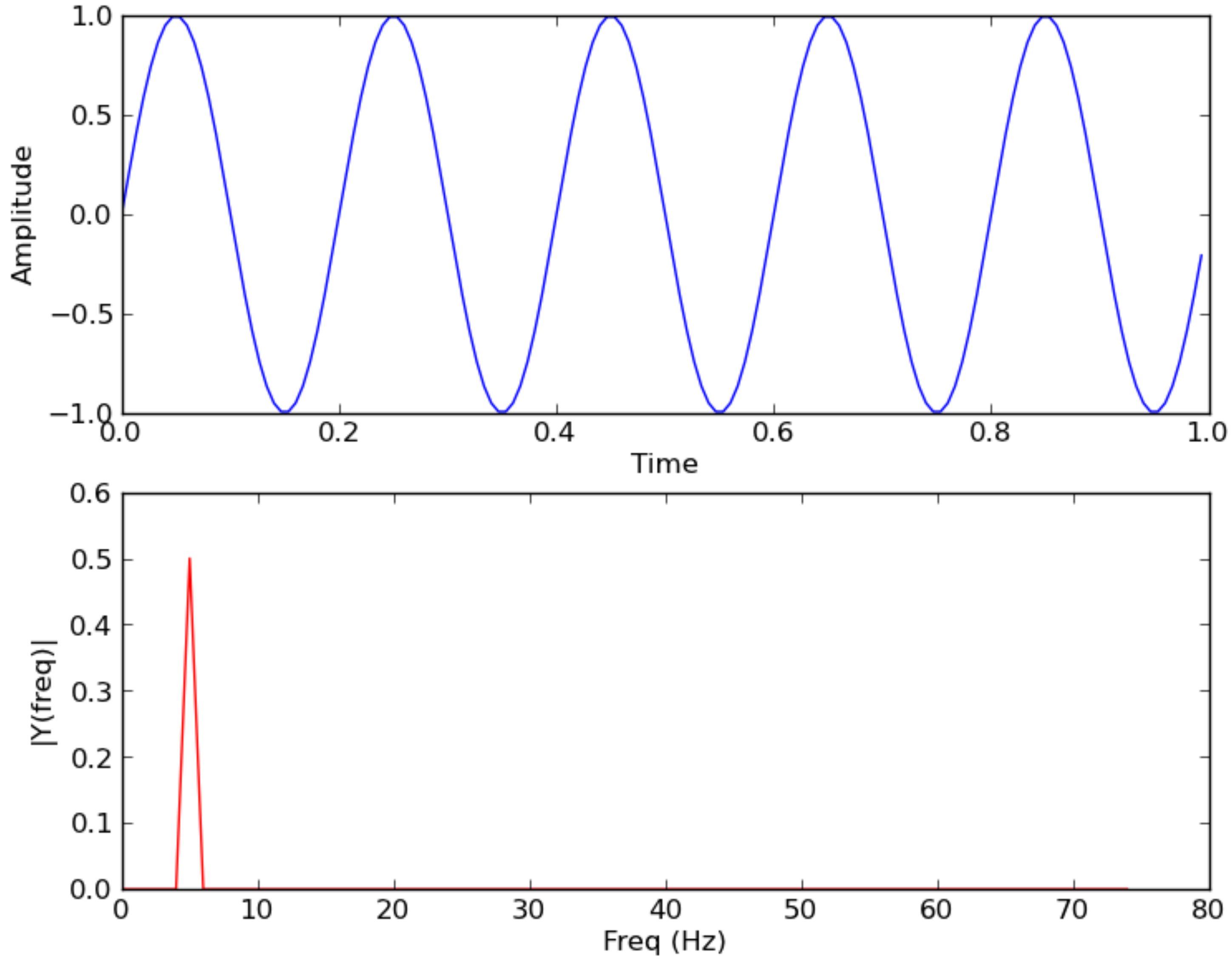
$$X_k = \frac{1}{N}\sum_{n=0}^{N-1} x_n e^{i2\pi k \frac{n}{N}}$$

To find the energy at a particular frequency, spin your signal around a circle at that frequency, and average a bunch of points along that path.

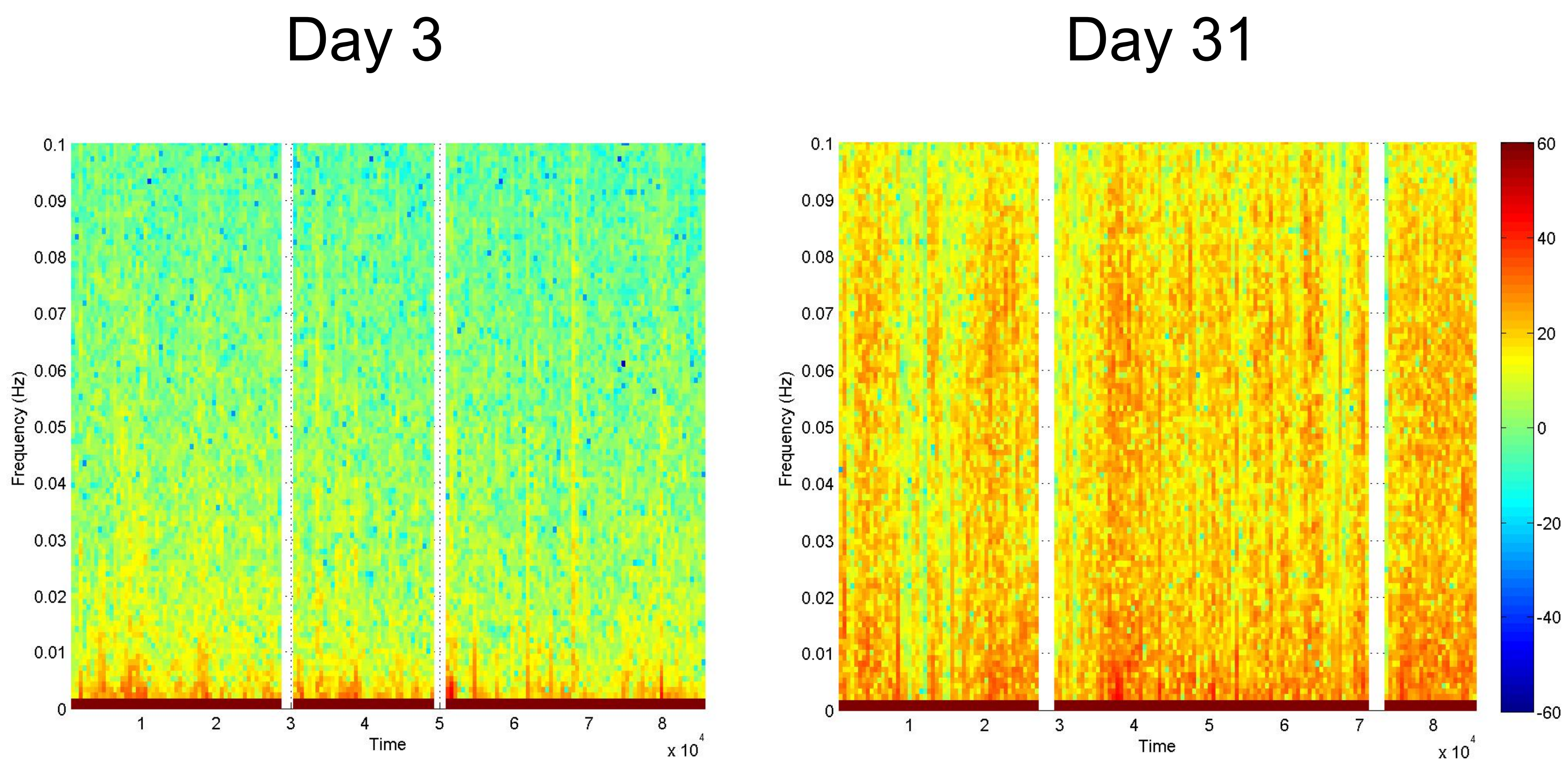
Courtesy of Stuart Riffle!

http://blog.revolutionanalytics.com/2014/01/the-fourier-transform-explained-in-one-sentence.html

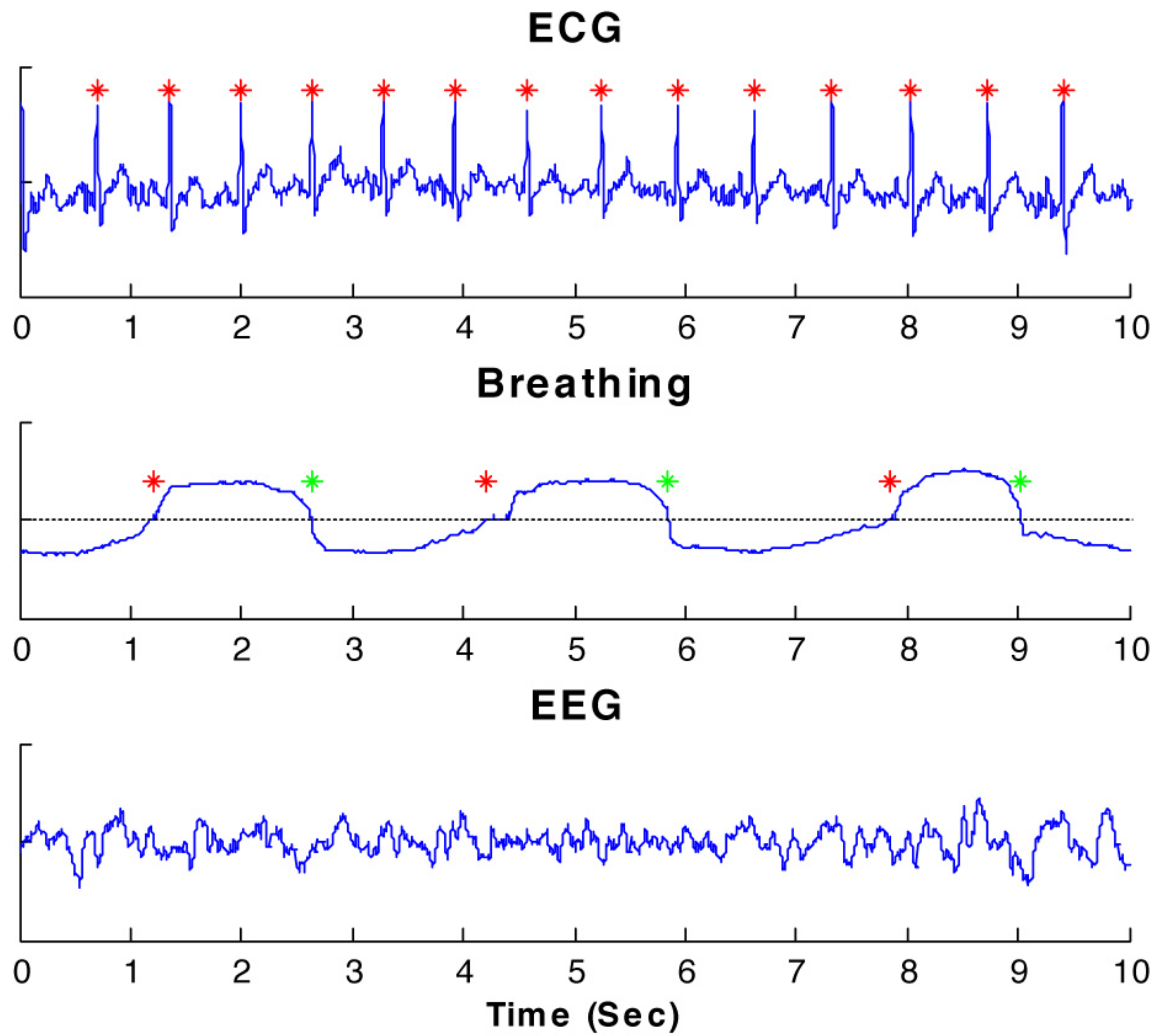
What the Fourier Transform looks like



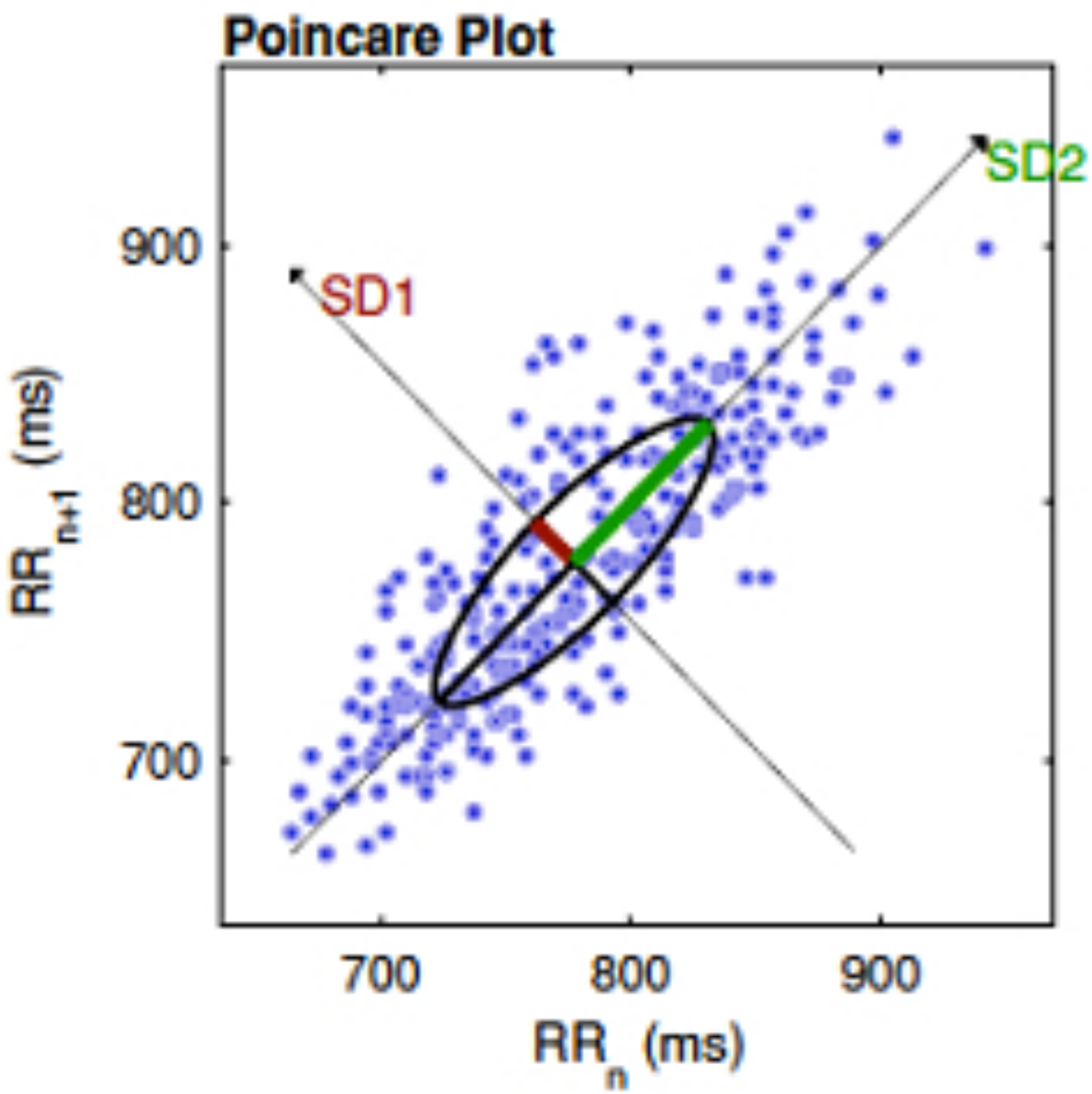
Time varying spectral density



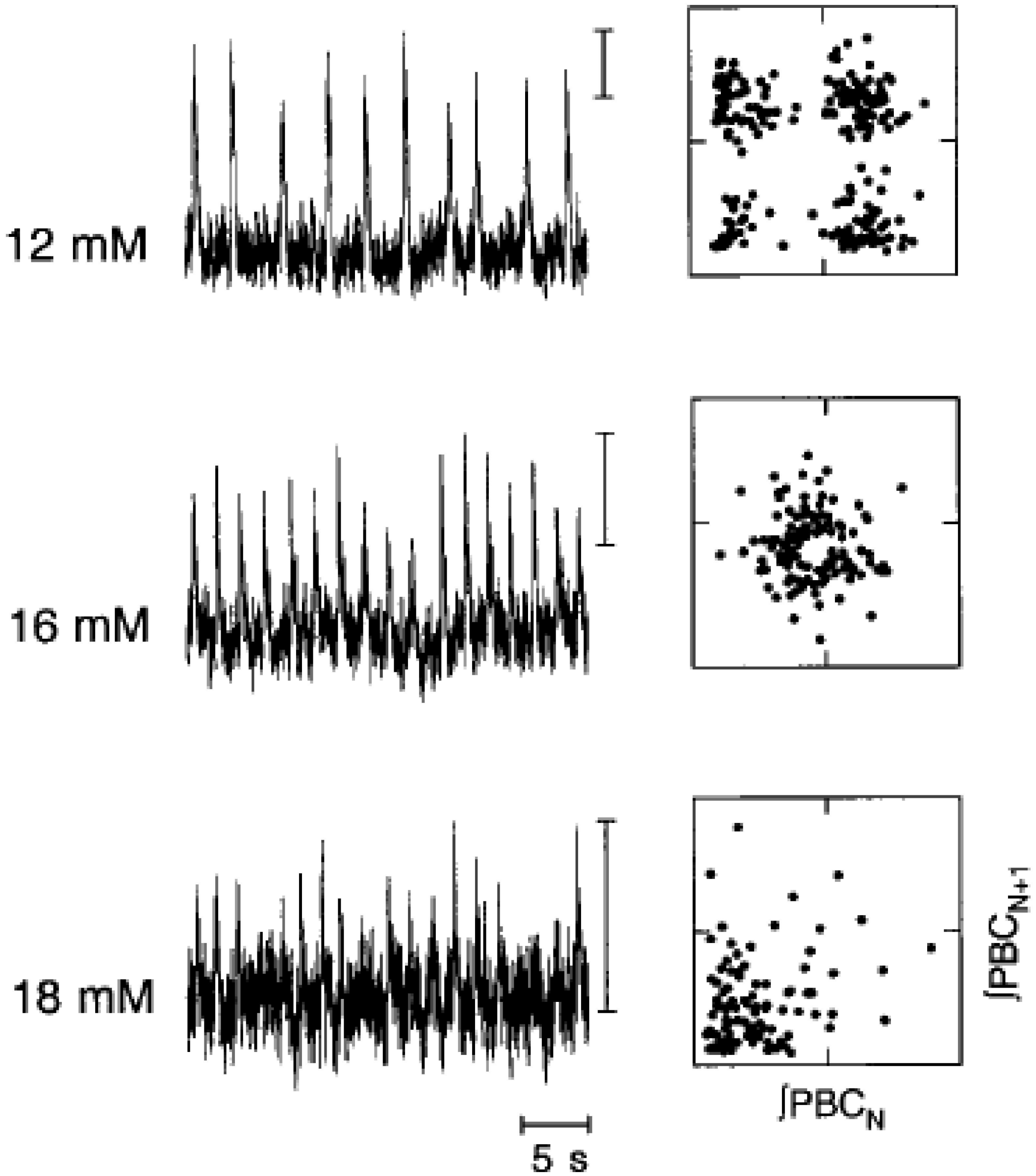
Measuring data variability

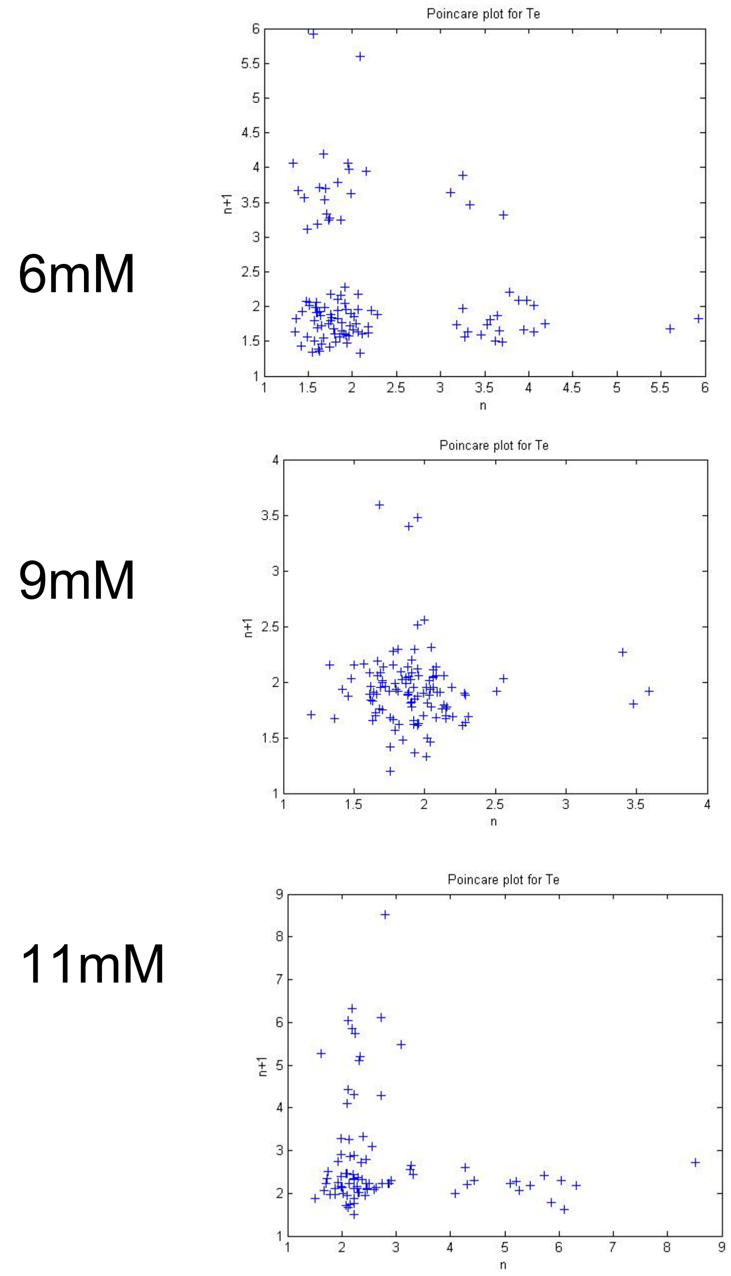


Poincarè metrics



Poincarè maps of burst timing/amplitude

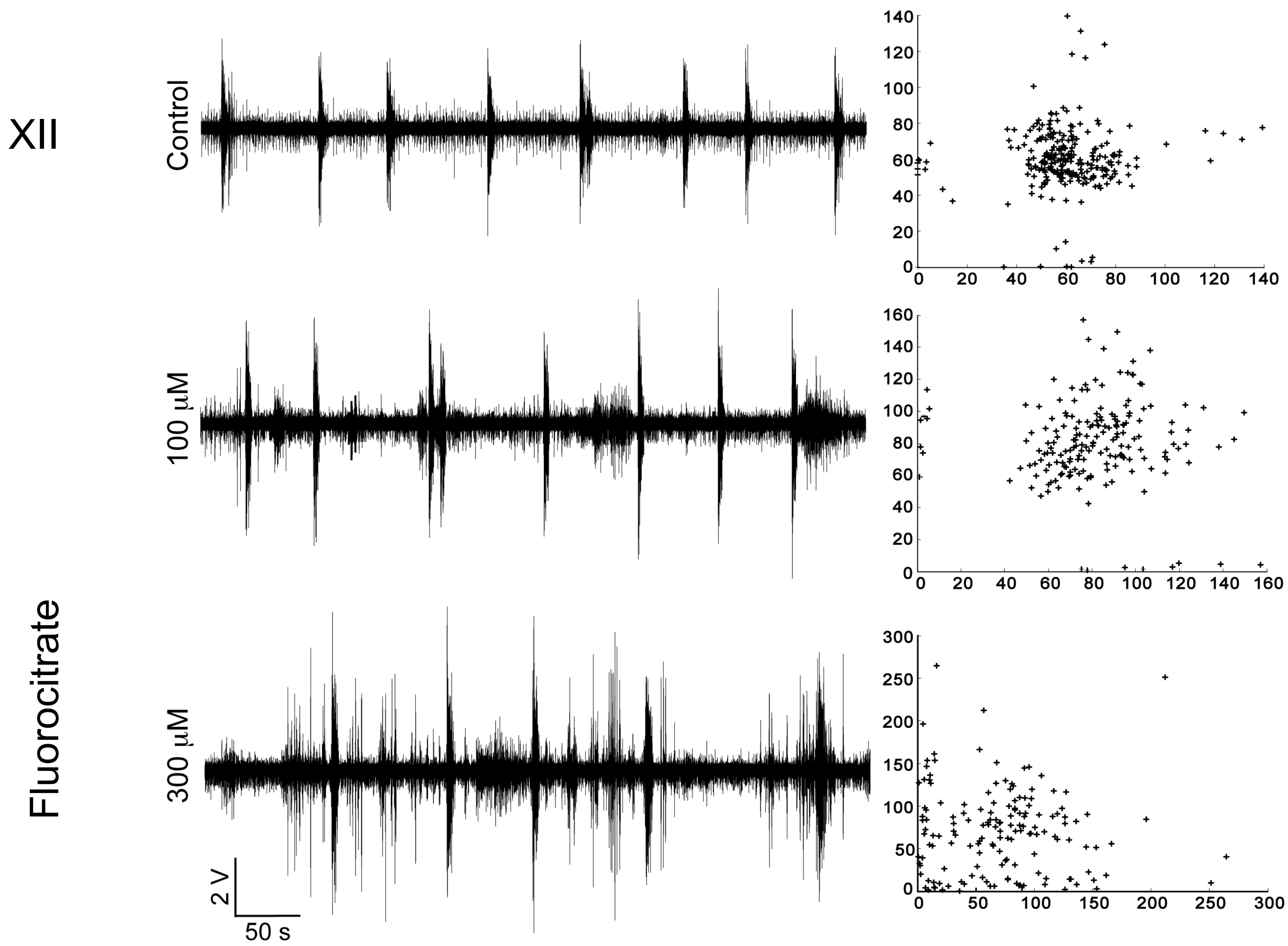




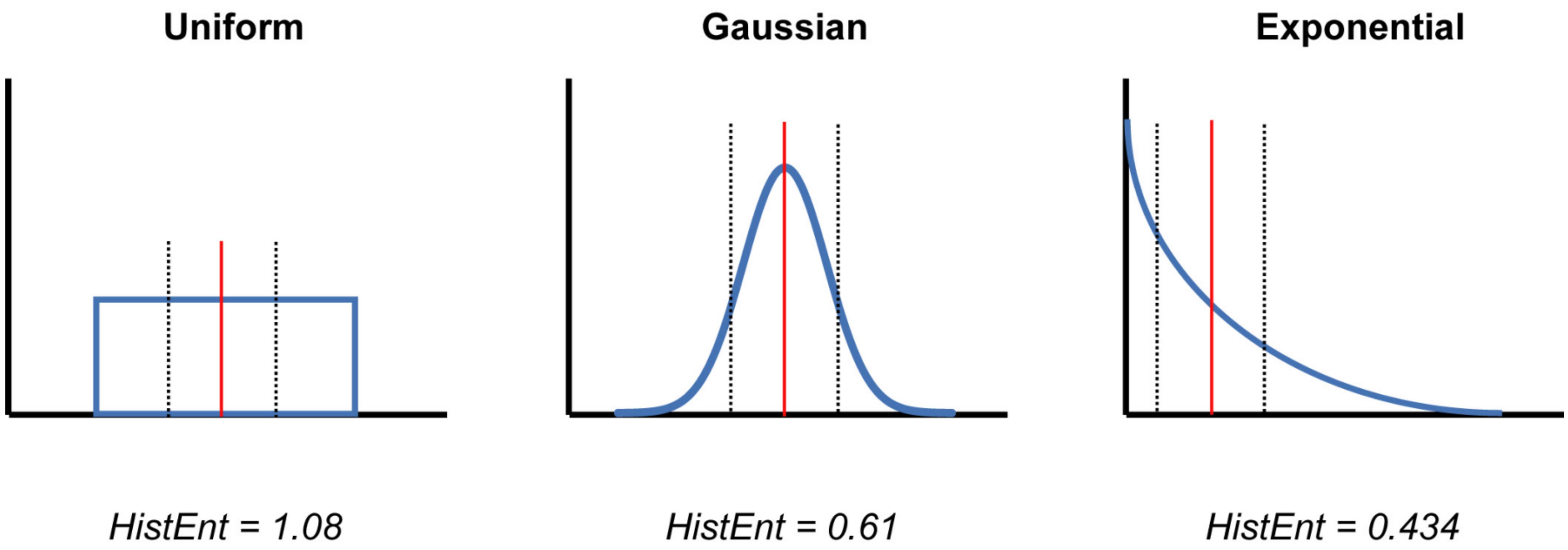
from Del Negro, et al. 2002, *Biophysical Journal*

* Transition from mixed-mode, to quasiperiodic, to chaotic oscillations

Astrocyte "poisoning" disrupts periodicity

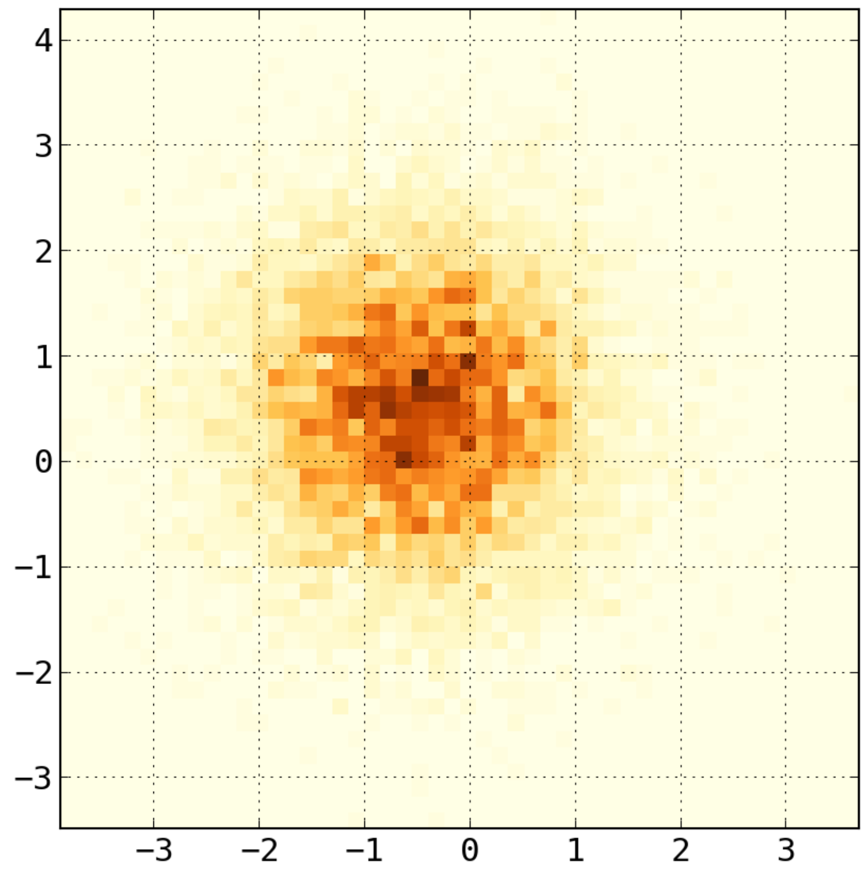


Shannon/Histogram Entropy as a measure of "complexity"



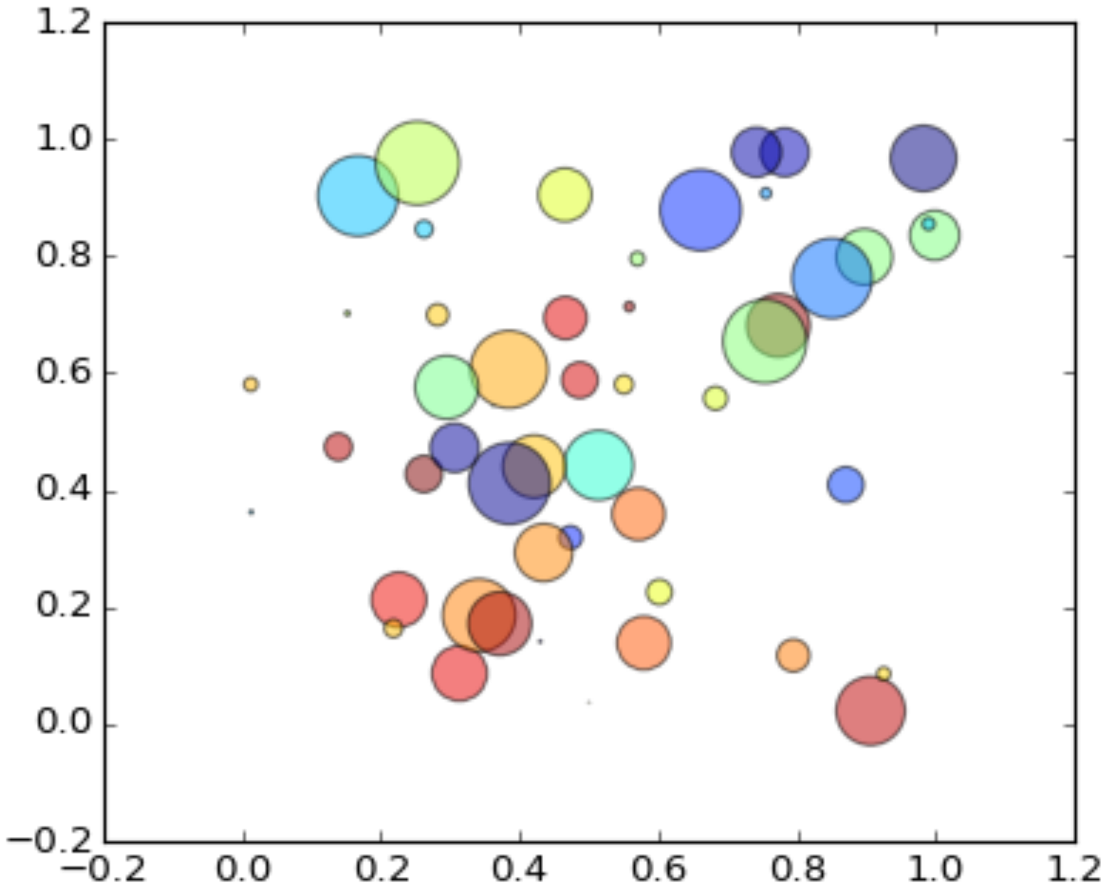
Heatmaps

```
import numpy as np
import numpy.random
import matplotlib.pyplot as plt

# Generate some test data
x = np.random.randn(8873)
y = np.random.randn(8873)

heatmap, xedges, yedges = np.histogram2d(x, y, bins=50)
extent = [xedges[0], xedges[-1], yedges[0], yedges[-1]]

plt.clf()
plt.imshow(heatmap, extent=extent)
plt.show()
```



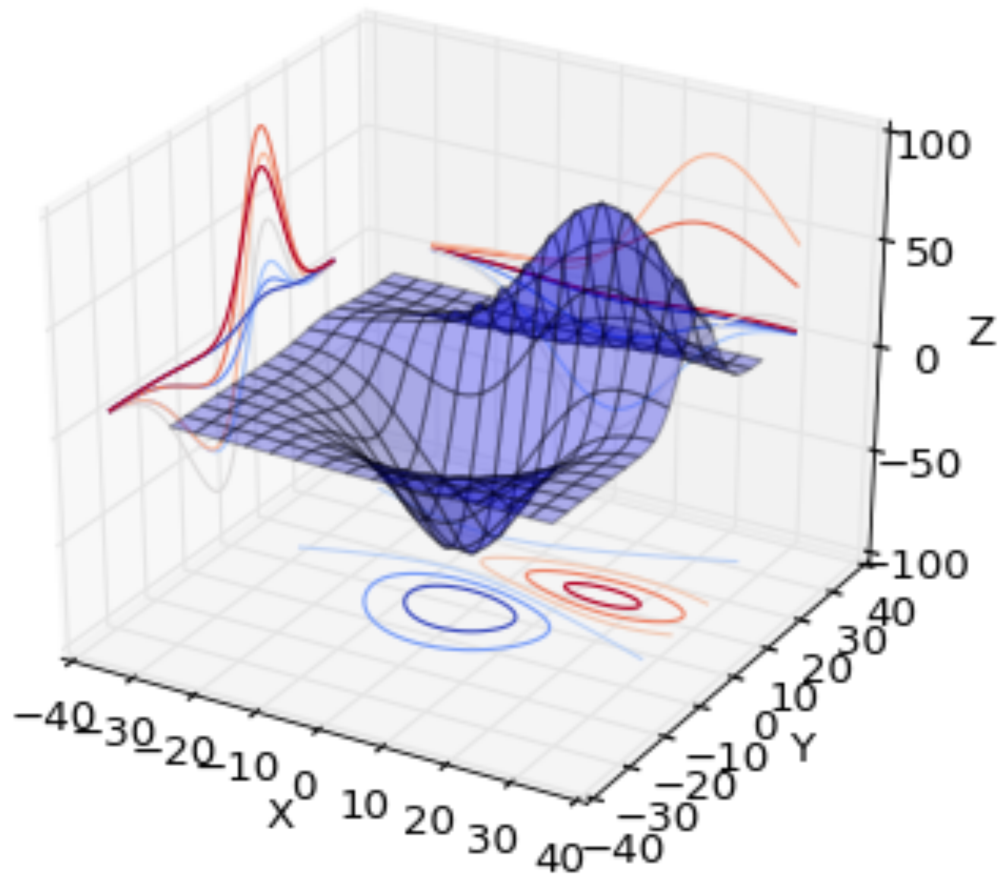
Scatterplots

```
import numpy as np
import matplotlib.pyplot as plt


N = 50
x = np.random.rand(N)
y = np.random.rand(N)
colors = np.random.rand(N)
area = np.pi * (15 * np.random.rand(N))**2 #
0 to 15 point radiuses

plt.scatter(x, y, s=area, c=colors,
alpha=0.5)
plt.show()
```



3D contour map

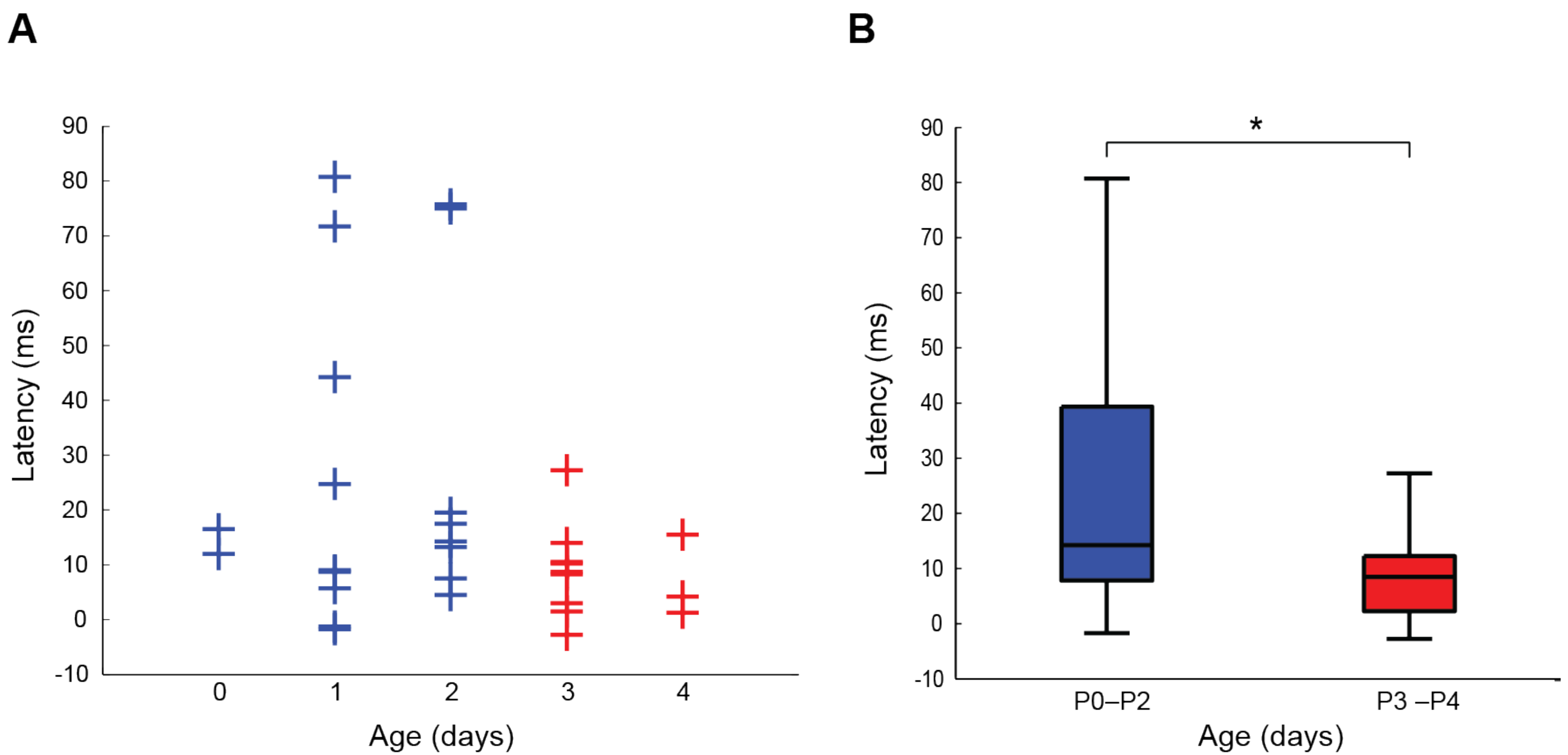
```
from mpl_toolkits.mplot3d import axes3d
import matplotlib.pyplot as plt
from matplotlib import cm

fig = plt.figure()
ax = fig.gca(projection='3d')
X, Y, Z = axes3d.get_test_data(0.05)
ax.plot_surface(X, Y, Z, rstride=8,
cstride=8, alpha=0.3)
cset = ax.contour(X, Y, Z, zdir='z',
offset=-100, cmap=cm.coolwarm)
cset = ax.contour(X, Y, Z, zdir='x',
offset=-40, cmap=cm.coolwarm)
cset = ax.contour(X, Y, Z, zdir='y',
offset=40, cmap=cm.coolwarm)

ax.set_xlabel('X')
ax.set_xlim(-40, 40)
ax.set_ylabel('Y')
ax.set_ylim(-40, 40)
ax.set_zlabel('Z')
ax.set_zlim(-100, 100)

plt.show()
```



Category data plots

Individual data points

Box-whisker plots



Christopher Fietkiewicz