

---

## Table of Contents

NENS 230: Assignment #2 Solution .....	1
1) The Compound Action Potential .....	1
2) The Strength-Duration curve .....	2
3) Conduction Velocity .....	3
4) Smoothing a Noisy Trace .....	4

## NENS 230: Assignment #2 Solution

Sergey Stavisky (sstavisk@stanford.edu)  
Sep 27 2015

```
clear % Good practice is to start with a clean workspace for a given analysis
```

### 1) The Compound Action Potential

We'll make a recording of the compound action potential, and graph it below.

```
% Load the actionpotential.mat data file
% (1 line)
% Hint: Recall that the syntax is 'load filename.mat' (but without quotes)
load actionpotential.mat;

% Adjust the time vector so that (t == 0) is the first point at which the voltage
% (1 or 2 lines)
% Hint: You want to offset (i.e. subtract from) the time vector by the first time
% the voltage first stops being equal to zero.
time = time - time(find(voltage ~= 0, 1));

% Adjust the time vector so that it is in milliseconds as opposed to seconds
% (1 line)
time = time*1e3; %could have also done elementwise multiplication. Multiplying ve

% Make a plot on figure 1 of the recorded voltage vs. time.
% Label your axes appropriately, and turn the grid on
% (~7 lines)
figure(1);
plot(time, voltage, 'k-');
xlabel('Time (ms)');
ylabel('Voltage (mV)');
title('Compound Action Potential');
grid on;
set(1, 'Name', 'Compound Action Potential')

% Determine the minimum and maximum voltage recorded.
% (4 lines)
tmax = time(voltage == max(voltage));
vmax = voltage(voltage == max(voltage));
tmin = time(voltage == min(voltage));
vmin = voltage(voltage == min(voltage));
```

---

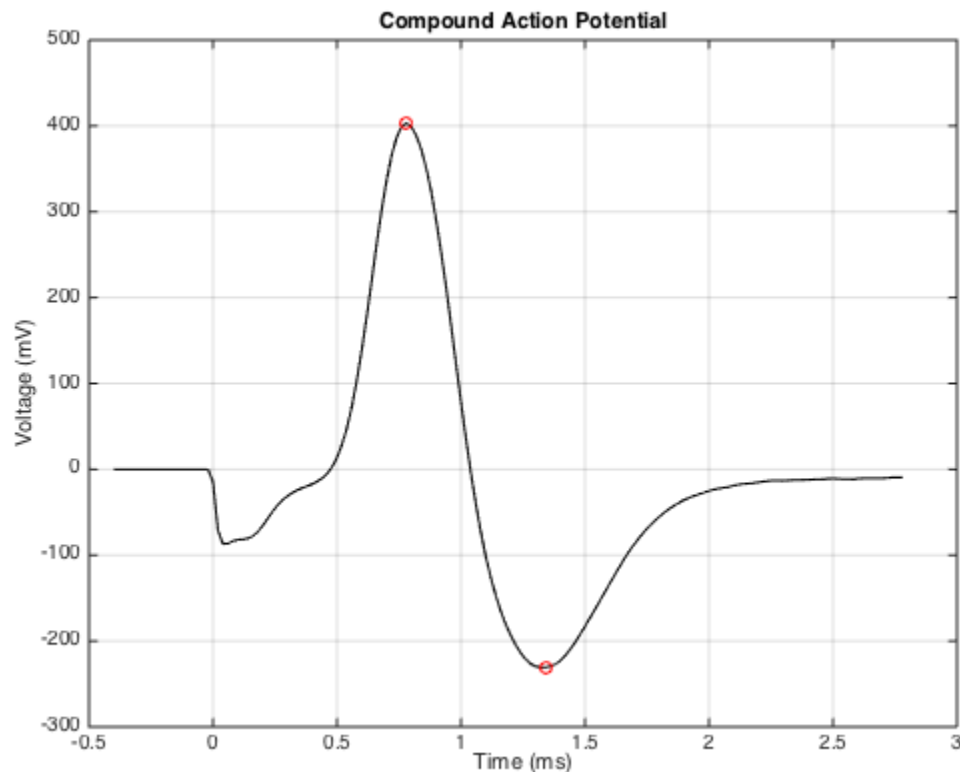
```

% Plot when these minimum and maximum happen
% (2 more lines)
hold on
plot(tmax,vmax,'ro')
plot(tmin,vmin,'ro')

% Display this information as output with fprintf -- format as you like as long
% as it's reasonable.
% (~2-4 lines)
fprintf('Maximum voltage is %smV and occurs at %sms.\n', mat2str(vmax),mat2str(tma
fprintf('Minimum voltage is %smV and occurs at %sms.\n', mat2str(vmin),mat2str(tmi
% Note: I used mat2str( ) here in case there happen to be more than one
% identical max or min voltage values. In that case, the earlier code will
% return multiple values and the then this printout code using mat2str( )
% will print the whole vector. Had I e.g. used this command:
% fprintf('Maximum voltage is %.2fmV and occurs at %.2fms.\n', vmax, tmax);
% Only one of the max/min points would be printed.
%This is a subtle, "edge case".

```

Maximum voltage is 403.1982mV and occurs at 0.78ms.  
Minimum voltage is -230.6976mV and occurs at 1.34ms.



## 2) The Strength-Duration curve

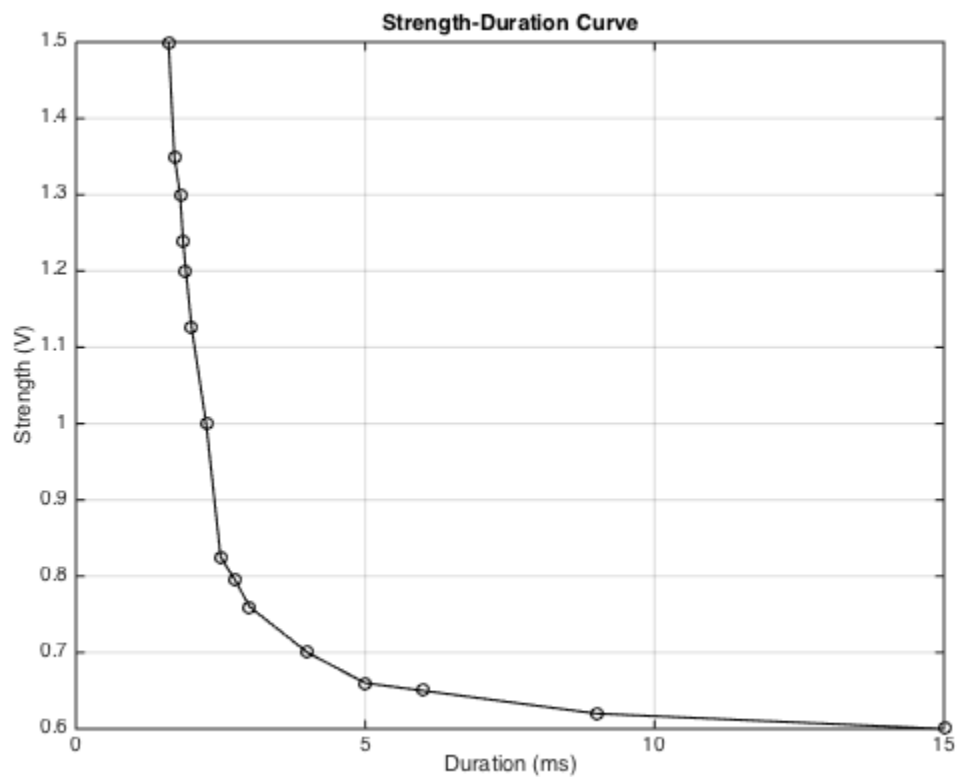
We'll make a strength-duration curve of whether the nerve responds to incr

---

stimulation, and graph it below:

```
% Load the data from the file pulsedata.csv, skipping the first header line
% Hint for skipping lines: look at the documentation for csvread, and note
% what the optional second argument does.
% (1 line)
pulseData = csvread('pulsedata.csv',1);

% Plot a strength duration curve on figure 2 of the strength/voltage (y-axis) vs.
% Label your axes appropriately, and turn the grid on. Use circle markers and a so
% Don't forget to make a new figure using the figure command, or you'll mess up yo
% figure.
% (~6 lines)
figure(2);
plot(pulseData(:,2), pulseData(:,1), 'ko-');
xlabel('Duration (ms)');
ylabel('Strength (V)');
title('Strength-Duration Curve');
set(2, 'Name', 'Strength-Duration Curve')
grid on;
```



### 3) Conduction Velocity

We'll make four measurements of conduction velocity from the provided data (1 line)

Warning: data is BIG. Don't try to print it to the screen, your command wi

---

will be covered with numbers. Call `size( traces )` to get your bearings.  
Remember, `size( )` returns `[numRows, numColumns]`

```
load recordings.mat
```

```
% find maximum indices
% (2 lines)
% Hint: Look up the behavior of the max() function if you provide it two
% output arguments. Also read about how it handles a matrix as input.
[values, indices] = max(traces);
maxTimes = traceTime(indices);

% estimate velocity
% (1 line)
% Hint: ./ is 'element-wise' division, which is what you want. You should
% get a length 4 vector. If you did / (no dot), you asked for a least squares
% regression and will have a single-element (scalar).
velocity = distances*0.01./maxTimes; % note conversion from cm to meters

% compute the mean and standard deviation of our measurement
% (2 lines)
meanVelocity = mean(velocity);
stdVelocity = std(velocity);

% Display this information as output with fprintf
% (1 to a few lines)
fprintf('Conduction velocity: mean: %4.4f m/s, std. dev.: %4.4f m/s.\n', meanVelocity, stdVelocity);

Conduction velocity: mean: 0.3236 m/s, std. dev.: 0.0097 m/s.
```

## 4) Smoothing a Noisy Trace

\* This problem is extra credit \* i.e. If you've never programmed before and worked long to get through 1-3, feel free to not do this one.

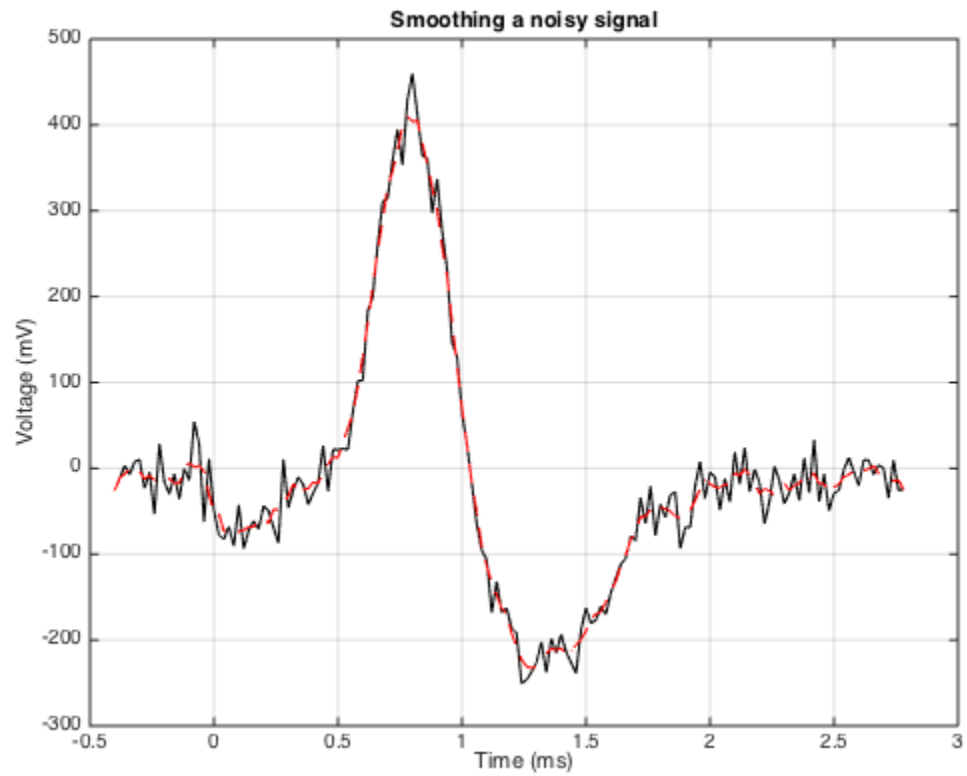
```
% Make a noisy voltage trace by adding Gaussian noise
% (1 line)
% Make a noisy voltage trace by adding Gaussian noise
% (1 line)
% Hint: add random noise by adding randn( ) to voltage.
% e.g., to generate a 1x3 vector of gaussian noise of std 5,
% use 5*randn(1,3)
noisyVoltage = voltage + 25*randn(size(voltage)); % size( ) already returns correct size

% Smooth this voltage trace
% Hint: use the smooth( ) command
% (1 line)
smoothVoltage = smooth(noisyVoltage);

% Make a plot of the two traces (noisy with a solid black line, smoothed with a dashed red line)
% (~6 lines)
figure(3);
plot(time, noisyVoltage, 'k-', time, smoothVoltage, 'r--');
xlabel('Time (ms)');
```

---

```
ylabel('Voltage (mV)');  
title('Smoothing a noisy signal');  
set( 3, 'Name', 'Smoothing')  
grid on;
```



*Published with MATLAB® R2014b*