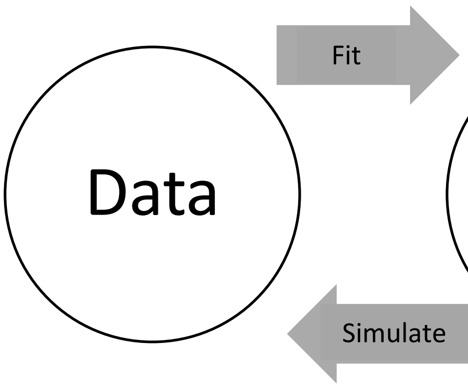
BIOE 198MI Biomedical Data Analysis. Spring Semester 2018.
Lab 7: Data Fitting

Background



If we know have a mathematical **model** for a system, then we can **simulate** the behavior of the system and generate any number of **data** points. This has been the situation that we have been working in for many of the previous labs.

However, if we don't know the underlying model, but we have measurements or **data** from the system, that data can be **fit** to a general **model** in order to understand the behavior of the system and specific underlying parameters.

Fitting to a Linear Model

Let's suppose that we have an temperature sensor, where at different temperatures the sensor yields different amounts of current, according to the following relationship:

$$I = m \cdot T + b$$

where *I* is the current in amps, *m* = 5 A/K, and *b* = 1 A.

We can generate a plot showing this relationship with the following segment of code:

```
T = 0:5:30;
m = 5
b = 1
figure(1)
I = m.*T + b;

subplot(1,2,1)

g=plot(T,I,'k-o');
set(g,'MarkerSize',10)
title('Noisy Data')
xlabel('Temperature (K)'); ylabel('Current (A)')
legend('ground truth','Location','Northwest')
    ax = gca;
    ax.FontSize = 18;
```

However, in reality any measurement would have some associated noise or error associated with it. We can simulate the noise by adding a random offset to each data point with the following segment of code:

```
subplot(1,2,2)
% +noise
I_noise = I + 7.5.*randn(length(I),1)';
g=plot(T, I_noise, 'r.');
set(g,'MarkerSize',20)
title('Noisy Data')
xlabel('Temperature (K)'); ylabel('Current (A)')
legend('noisy data','Location','Northwest')
    ax = gca;
    ax.FontSize = 18;
```

Now, the fitting challenge is: Using <u>only the noisy data</u>, can you extract out the parameters *m* and *b*?

<u>Evaluating Guesses</u>

We can guess any reasonable value for *m* or *b*, that looks like a good fit for the data (i.e. using something like the code below), but how do we know if our guess is good or not? How do we compare different guesses?

```
figure(2)

g=plot(T, I_noise, 'r.');
set(g,'MarkerSize',20)

m_guess = 4; b_guess = 2;

I_guess = m_guess.*T + b_guess;

hold on

g=plot(T,I_guess,'b-o');
set(g,'MarkerSize',10)

hold off

legend('noisy data', 'guess model', 'Location', 'Northwest')
title('Random Guess')
xlabel('Temperature (K)'); ylabel('Current (A)')
    ax = gca;
    ax.FontSize = 18;
```

Introduction to *lsqcurvefit*

From the MATLAB documentation:

```
x = lsqcurvefit(fun,params0,xdata,ydata)
```

starts at `x0` and finds coefficients `x` to best fit the nonlinear function `fun(params,xdata)` to the data `ydata` (in the least-squares sense). `ydata` must be the same size as the vector (or matrix) `F` returned by `fun`.

Important things to note:

- `fun` can be defined as a separate function file or in-line
- `params0` is a vector of initial guesses for the parameters defined in the model function
- `xdata` and `ydata` are the vectors of data that you are trying to fit the model to

So, this can be applied to fit our noisy temperature data with the following lines code:

```
%% Section 3: Introduction to lsqcurvefit
clear all;
load NoisyTempData.mat

xdata = T;           % input xdata
ydata = I_noise; % input ydata
params = lsqcurvefit(@linear_fn, [2 7], xdata, ydata);

% Alternative: uncomment the following line to use anonymous function
% params = lsqcurvefit(@(params,xdata)params(1)*xdata+params(2), [2 7],
xdata, ydata);

m_fit = params(1); b_fit = params(2);
T_vec = 0:50;
I_fit = m_fit*T_vec + b_fit;

figure(4)

g=plot(T, I_noise, 'r.');
set(g,'MarkerSize',5)

hold on
g = plot(T_vec,I_fit,'b-');
set(g,'MarkerSize',7)
hold off

legend('noisy data', 'fitted model', 'Location', 'Northwest')
title('Optimized Fit')
xlabel('Temperature (K)'); ylabel('Current (A)')
    ax = gca;
    ax.FontSize = 18;
```

Follow-up exercise:

Can you fit the data in `DecayData1.mat` to the following general model?

$$I = A\, e^{-k*t} + b$$

What do you get for *A*, *k*, and *b*?

Homework Assigment

Download the *BacterialGrowthRates.mat* file from the course website.

Using the following model for population growth,

$$P(t) = \frac{K}{1 + C\, e^{-rt}}$$

where *P* is the population at any given time (units of number of bacteria), *K* is the carrying capacity of the system (units of number of bacteria), *C* is a scaling coefficient, and *r* is the growth rate (units of number of bacteria per day), fit the three datasets to this model, and answer the following questions:

1. What are your fitted parameters for each dataset?
2. How good are your final fits?
3. Which of the three populations has the fastest growth rate?