APSC 1001 MATLAB Curve Fitting Tools

Randy Schur

October 1, 2015

1 Sampling Data

An valuable part of your toolbox of engineering skills is testing, data collection, and analysis. Much of what scientists and engineers study was originally discovered by experimentation and the analysis of experimental data. The ability to decipher some meaning from the data you collect is something you will rely on throughout your engineering career.

1.1 Signal Analysis

Many textbooks have been written on the analysis of different types of signals, and we won't go into great detail in this class. Some common examples of applications that demand signal analysis are circuit design and radio frequency (RF) work in electrical engineering, image processing in computer science, wind or water tunnel experiments in mechanical or aerospace engineering, and earthquake movement in civil engineering.
Often, the signals we see are repeating. We call this *periodic*, meaning that the signal is repetitive and has a measurable period, or length of time between cycles. The simplest example of a periodic signal is a sine wave. In fact, any periodic signal may be replicated using only sine (or cosine) waves of different frequencies! This is a surprising and extremely important result in many fields of mathematics and engineering. Breaking a signal up into its component frequencies is called a Fourier Transform, named after Joseph Fourier who (sort of) discovered that periodic, or *sinusoidal*, functions can be approximated by sine waves. It is something that you will likely discuss in several classes during your studies.

1.2 Statistical Approximations

Not all data that we collect is from a periodic signal. Sometimes we have a single process, and we test it multiple times while changing one or more variables. An example of a data collection task like this might be testing for when a particular component fail. You might vary the loading on that component, and run the test multiple times. Once you have collected data from multiple runs of the experiment, you might try to find the how the failure of the component depends on the loading. You are looking for a *function* for failure in terms of loading. Because we can't perfectly control any experiment, there will be some element of randomness to the data you collect. However, using some statistical tools, it it often possible to find a relationship between the two (or more) variables that you observing. This process is called *curve fitting*.

2 Curve Fitting in MATLAB

MATLAB provides several built in functions that can help with curve fitting. We are going to use a tool with a nice GUI called cftool. See the example below to learn how to use this tool.
One way we can test the tool is to make up a function, inject some randomness in order to model 'experimental' data, and then see if we can find the original function.
Type the following into MATLAB.

```
>> x = linspace(0,100,101); %create initial data
>> y_actual = 2.5*x + 17; %function of x
>> rnums = 15*randn(1, 101); %random numbers with a zero mean (no bias)
>> y_test = y_actual + rnums; %create 'test' data.
```

Now we'll use the curve fitting toolbox to see if we can get the correct function from our test data.

```
>> cftool(x, y_test)
```

This brings up the curve fitting GUI.