Experiment 0

GRAPHICAL AND STATISTICAL ANALYSIS

Apparatus

For this analysis, you will need a computer with an installation of Matlab or GNU Octave[1]. A glossary for doing the analysis with Excel is given at the end.

Data

The experiments in this laboratory require taking several quantitative readings and analyzing their results. Typically, we will vary a parameter, the "independent variable" and record values of a parameter that has changed as a result, the "dependent variable". We end up with data pairs.

Examples:

1. We may apply a force F on a spring (perhaps by loading it with a mass), and record the resulting extension of the spring x. The data should follow $F = kx$, Hooke's law, where k is the spring constant. The independent variable is F and the dependent variable is x. We record data pairs $\{(F_n, x_n)\}$, where $n = 1 \ldots N$ enumerates the readings.

2. We may apply a voltage difference V between the terminals of a resistor, and record the current i as a result. The data should follow Ohm's law, $V = iR$, where R is the resistance. The independent variable is V and the dependent variable is i. We record data pairs $\{(V_n, i_n)\}$, where $n = 1 \ldots N$ enumerates the readings. There are power sources that allow you to select a current, and then you would read the voltage as a dependent variable so the voltage is not always the independent variable.

One stores the data in two vectors

```
i = [ 0.09, 0.21, 0.30, 0.42 ];
V = [ 2.00, 3.80, 6.12, 8.32 ];
```

[1]GNU Octave is a free and open-source implementation of Matlab.

Graphing

To analyze the results graphically, we plot the data pairs, putting on the horizontal axis the independent variable, and the dependent variable on the vertical axis. Graphs are described in the text as "dependent vs. independent", or more shortly as "dependent(independent)".

For the spring example, the graphing instruction would read : "plot x vs. F", or "plot $x(F)$", upon which you would plot F on the horizontal axis, and x on the vertical axis. Note that calling something "x-axis" in this and many other contexts is ambiguous. Often, when one writes "x-axis", really one should write "horizontal axis". This is particularly confusing in this example, since the x-axis in this example is the vertical axis!

Data is plotted with symbols, not with lines, unless you have a very large number of data points. For the $i(V)$ example, to plot the data, issue the command (the last parameter is the symbol to be used)

```
hold off;
plot(i, V, 'x');
```

We start the command with 'hold off' because we want this plot command to start a clean new plot. You will need to label the axes

```
xlabel('i (A)');
ylabel('V (V)');
```

Statistical Analysis

Individual readings often contain a measurement uncertainty, and that uncertainty is often stochastic in nature, *i.e.* it causes the value to fluctuate around a mean. We are often interested in the mean, and quantify the fluctuations to determine how certain we are of the reading. The average is defined as

$$\langle x \rangle = \frac{1}{N} \sum_{n=1}^{N} x_n \tag{1}$$

while the standard deviation is defined as

$$\sigma_x = \sqrt{\frac{1}{N-1} \sum_{n=1}^{N} (x_n - \langle x \rangle)^2} \tag{2}$$

For the $i(V)$ data, we could calculate the resistance for every single point

```
R = V./i;
```

and then statistically analyze these values

```
mean(R)
# 20.132
std(R)
# 1.70
```

and one would conclude that from this data we find that $R = 20.13 \pm 1.70\ \Omega$. However, this analysis is somewhat flawed. When we calculate the ratio of $V./i$, the values at low current have a larger proportional error than the values at higher current, and those ratios will end up having a larger uncertainty. Yet when we calculate the mean and standard deviation we weigh them equally. A better approach is to do regression analysis, a.k.a. a "least-squares fit" or simply "fit".

Regression Analysis

To do a first-order fit to data pairs $\{(x_n, y_n)\}$, we calculate the coefficients of the line

$$y = ax + b \tag{3}$$

that minimizes the error

$$\epsilon^2 \equiv \frac{1}{N-2}\sum_n (y_n - ax_n - b)^2 \tag{4}$$

The name "first-order" fit derives from the order of the polynomial that is fit to the data. To fit a parabola, hence, we would do a "second-order fit".

For our data, because we are interested in the slope of $V(i)$ instead of $i(V)$, we calculate the line

```
p = polyfit(i, V, 1);
```

The slope and offset are stored in the variable p, and we assign the first element (the slope) to the variable R,

```
p
# 19.610256     0.059385
R = p(1);
```

To calculate the error in the coefficients, we calculate ϵ from equation 4 as well as

$$\delta \equiv N\sum_{n=1}^{N} x_n^2 - \left(\sum_{n=1}^{N} x_n\right)^2 \tag{5}$$

and then the error σ_a in the slope (a) and the error σ_b in the intercept (b) are given by

$$\sigma_a^2 = N\frac{\epsilon^2}{\delta} \quad , \quad \sigma_b^2 = \frac{\epsilon^2}{\delta}\sum_{n=1}^{N} x_n^2 \tag{6}$$

which for our data is calculated as

```
n = length(i);
epsilon = sqrt(sum((V-p(1)*i-p(2)).^2)/(n-2));
delta = n*sum(i.^2) - (sum(i))^2;
sigmaa = sqrt(n*epsilon^2/delta)
# 1.3253
sigmab = sqrt(epsilon^2*sum(i.^2)/delta)
# 0.37404
```

So we conclude

$$R = 19.6103 \pm 1.3253\ \Omega \qquad \leftarrow \text{too many digits!}$$

We should pay attention to the significant figures here. The uncertainty of the slope is 1.3253 Ω, so it does not make sense to keep this many significant figures in R. We typically round the value R to the smallest digit or two as indicated by the uncertainty. So we write

$$R = 19.6 \pm 1.3\ \Omega \qquad \text{or} \qquad R = 20 \pm 1\ \Omega$$

Now we plot the trend line in the figure with the data

```
hold on
plot(i, p(1)*i+p(2), '-');
```

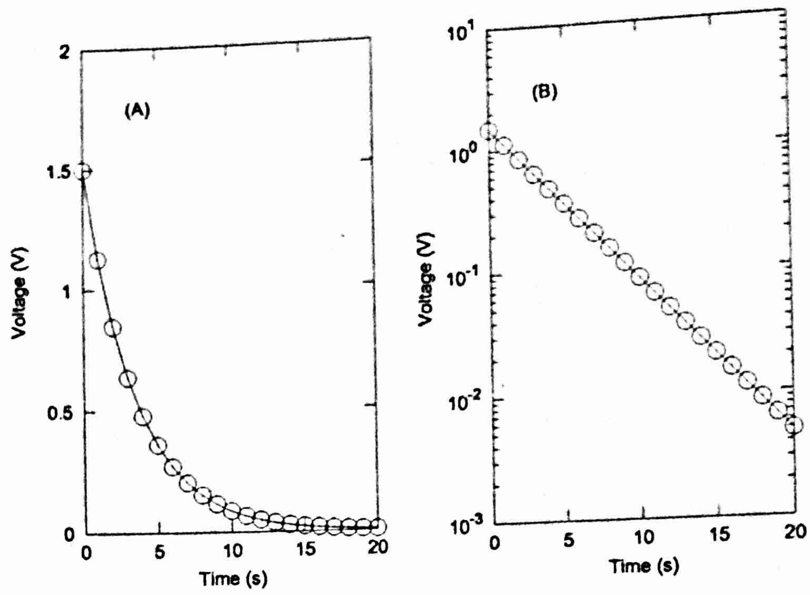

***Figure 1:** Analysis of exponential signal on (A) linear and (B) semi-logarithmic graph.*

We issued the 'hold on' command so the plot goes on top of the previous plot without starting a clear new plot. We should add legends and titles to the plot

```
title('A resistor');
legend('First data set', [num2str(R) '+/-' num2str(sigmaa)])
```

To save the figure, we can run

```
print('figure1.pdf', '-dpdf');
print('figure1.png', '-dpng');
```

which will generate a PDF and a PNG file of the plot.

Semi- logarithmic analysis

We often encounter signals that depends exponentially on a parameter, rather than linearly. One such example we will encounter in Experiment 6.

Say the data follows the following exponentially decaying curve

$$V(t) = V_0 e^{-t/\tau} \tag{7}$$

and τ is the decay constant, in seconds. An example is shown in Fig. 1. If we plot the data on a linear plot (A), you see that the signal decays rapidly

```
plot(t, V, 'x')
```

but if we plot it on a semi-logarithmic plot, taking the log of the vertical axis, we get the right plot (B)

```
semilogy(t, V, 'x')
```

If we wish to extract the value of τ, we need to do a curve fit. We use the fact that the data follows a linear dependence once we take the log of the voltage to do the analysis, since

$$\log(V) = \log(V_0) - \frac{t}{\tau} \equiv p_1 t + p_2 \quad .$$

So we do a fit of

```
p = polyfit(t, log(V), 1)
```

and the value of τ is extracted from p_1, and the factor V_0 is extracted from p_2

```
tau = -1/p(1)
# 3.5
V0 = exp(p(2))
# 1.5
```

and we can plot the fit in the graph

```
semilogy(t, V0*exp(-t/tau), '-')
```

Propagation of uncertainty

When we measure a physical signal as a function of several independent variables, we can calculate how accurate our physical signal is once we know how accurate the input is. The method of keeping track of these errors is known as "propagation of uncertainty".

We can use equation 6 to calculate errors in the slope, but how would such an error translate into quantities derived from the slope?

Let's say we get R from the slope, but we are interested in $G = 1/R$. Then, we can write

$$\frac{1}{R+\sigma_R} = \frac{1}{R(1+\sigma_R/R)} \approx \frac{1}{R}\left(1 - \frac{\sigma_R}{R}\right) \equiv G - \sigma_G \quad ,$$

with (we use the absolute value, so drop the sign)

$$\sigma_G \equiv \frac{\sigma_R}{R^2} = G\frac{\sigma_R}{R} \quad .$$

Generalizing, let us say we have a measurement h as a function of input parameter x, we use a Taylor expansion up to first order

$$h(x+\sigma_x) \approx h(x) + \frac{\partial h(x)}{\partial x}\sigma_x \equiv h(x) + \sigma_h \quad ,$$

hence

$$\sigma_h = \frac{\partial h(x)}{\partial x}\sigma_x \quad .$$

If there are more than one parameter, and we assume that uncertainties are uncorrelated, then we can use Pythogoras' theorem to get the resultant error, *i.e.*

$$\sigma_h^2 = \left|\frac{\partial h}{\partial x}\sigma_x\right|^2 + \left|\frac{\partial h}{\partial y}\sigma_y\right|^2 + \ldots \tag{8}$$

Say we have done an experiment to determine the gravitational acceleration g using a pendulum, we would measure the frequency of oscillation f and the length l. Then the relation

$$f = \frac{1}{2\pi}\sqrt{\frac{g}{l}}$$

is solved for g

$$g = 4\pi^2 f^2 l \quad .$$

If the frequency we measured was $f = 1.59$ Hz and the pendulum length was $l = 0.10$ m, we find

$$g = 9.98 \text{ m/s}^2 \quad .$$

We have two uncertainties, one for the length σ_l and one for the frequency σ_f. Using equation 8, we write

$$\sigma_g^2 = \left|\frac{\partial g}{\partial f}\sigma_f\right|^2 + \left|\frac{\partial g}{\partial l}\sigma_l\right|^2 = \left(8\pi^2 f l \sigma_f\right)^2 + \left(4\pi^2 f^2 \sigma_l\right)^2$$

If we had a 1 mm accuracy for measuring length, so $\sigma_l = 1.0 \times 10^{-3}$, and a 0.03 Hz accuracy for f, so $\sigma_f = 0.03$, we find

$$\sigma_g^2 = 0.14 + 0.010 \quad \rightarrow \quad \sigma_g = 0.39 \text{ m/s}^2 \quad .$$

Hence our final answer is

$$g = 9.98 \pm 0.39 \text{ m/s}^2 \quad .$$

Most of the uncertainty came from the uncertainty in f, so if we would like to improve our accuracy, we should focus on getting better readings for f first.

Glossary for using Excel

Doing statistical and graphical analysis with Excel should be familiar to you from previous labs. For completeness, these are the Excel functions you can use to implement the various calculations

Equation Number		Excel function
1	$\langle x \rangle$	average(*cells*)
2	σ_x	stdev(*cells*)
3	$y = ax + b$	linest(*dependent cells*, *independent cells*, true, true)

Procedure

1. Use the following data to determine R and its uncertainty, plot the data, the fit, label the axes, and add the appropriate legend.

i (A)	V (V)
0.57	3.30
1.01	6.16
1.50	6.73
2.02	11.98
2.53	13.60
3.02	15.09
3.59	18.41
4.05	21.19
4.50	22.95
5.02	26.98

2. We have the following data for $V(t)$ that follows an exponential dependence of the form of equation 7. Plot V vs. t on a semi-logarithmic plot, put V on the log axis, fit it, extract the decay time τ and the prefactor V_0 and plot the fit in the same graph.

t(s)	V (V)
0	3.1
1	2.0
2	1.3
3	0.8
4	0.5
5	0.3
6	0.2
7	0.1