SUGGESTIONS FOR LAB REPORTS—January 28, 2011

1. A Typical Outline for a Scientific Paper

A typical scientific paper is divided into sections, subsections, and subsubsections. The global outline for a typical scientific paper looks like the following. Note, however, that there are wide variations from this, which depend on content, subject matter, and individual style.

1. Title. Should encapsulate the contents and meaning.

2. Abstract. A short summary of the paper including the most important results. Purpose is to tell a prospective reader whether it is worth spending more time on article.

3. Introduction. Sets the context: summary of the current state of knowledge, how that state can be improved, what this work does to advance the field. What did you hope to accomplish? Briefly, what did you accomplish?

4. Observations or Experiments. What you observed, who did it, when you did it, what equipment you used, how you recorded data, any particulars or peculiarities.

5. Data Analysis. The theory according to which you analyze the data, how you actually did the analysis, the results of your analysis. Provide the essential numbers—the distillation of your original data (often millions of numbers) into a set of essential numbers or results. This is what we mean by "data reduction"!

6. Interpretation. What the results mean in terms of astrophysics or your previous state of knowledge. How your results relate to specific issues that were mentioned in the introduction.

7. Conclusion. A summary of important results and points made in the paper, including pointing to the particlar sections so that the reader can easily learn more detail. What aspects are lacking? How would you have done things better? Prospects for future work.

2. Scientific and Interpretive Issues

These are *most important* because they related directly to your scientific and experimental work and interpretation.

- When you derive a result or calculate something it's important to be *self-critical*. This is known as a *reality check*. Various forms of reality check include the following (a limited list):
    1. Generate fake data. Run your software on them (*note the plural use of "data"!*) and check for consistency.

2. When doing a least-squares fit, plot the *data*, overplot the *fitted curve*, and plot the *residuals*. The data and fitted curve should look similar. The residuals should exhibit no systematic trends and should look like noise clustered around zero. If not, why not?
3. Before deriving a result with fancy numerical techniques you should first make a guess, using your physical intuition, about what the answer is. If your fancy numerical technique gives something wildly different, then
   (a) Your physical intuition is no good, which means you don't understand the basic fundamentals.
   (b) Your numerical technique or software is no good.

   Which is it? (Or is it both????) Talk to people, ask questions, or whatever, but *resolve these discrepancies!*

- When you plot some data, *look at the plot and think about what you see.* For example, when you observed the Sun with the interferometer, the Campanile shadowed the dishes and the signal went away for some time. Ask yourself: what happened to the data during that time? In your lab report, such things are worth comments!
- Abstracts should contain essential information—*including the important numbers that you derive.*

3. Grammar, etc.

Some grammatical-type issues:

- The word 'data' is *plural.* Use it as you would use the word 'datapoints'. The singular of data is *datum.* Use it as you would use the word 'datapoint'. For example:
  1. The data *indicate* (not *indicates!*) that the system doesn't work... Similar to saying "The datapoints indicate..."
  2. This datum *is* a bad measurement and we will discard it.
- Capitalize proper names. This includes 'Fourier', 'Gauss' or 'Gaussian', 'Sun', 'Moon', 'Orion', etc.
- Check spelling! From the UNIX prompt, type

  ```
  ispell -t mylab.tex
  ```

  which runs an interactive spell checker. The `-t` means "ignore TEX-related commands". Spell checking isn't a panacea because a typo can produce a properly spelled word that isn't appropriate. Example: "These data are like ship."

4. Plotting/IDL Issues

Some plotting- or IDL- related issues:

- Axis labels and annotations on plots need to be large enough to be legible. Also, you really ought to use nice fonts: don't underestimate the value of good looks! And you usually want thicker lines. To accomplish all this in postscript the cumbersome way is to specify a lot of stuff in `psopen`; the easy way is to use our canned wrapper procedure `ps_ch`:

  ```
  ps_ch, 'outputfile.ps', /defaults, xsize=5, ysize=6, /inch
  plot, x, y, xtitle='This is X', ytitle='This is Y', title="Carl's plot"
  ps_ch, /close
  ```

  You can look inside this procedure to see how to change font size and line thicknesses. See Figure 1.

- When plotting datapoints, it's usually a good idea to plot the points themselves, which you'd do with the keyword `psym=4`, for example [1]. Sometimes you also want to connect the datapoints with lines, which you do with `psym=-4`. Or—especially when you do least squares fits—you want to overplot the datapoints with a fitted curve; to do this, plot the datapoints and then use `oplot` to overplot the curve.

[1] Or some other number—the number gives the plot symbol shape.

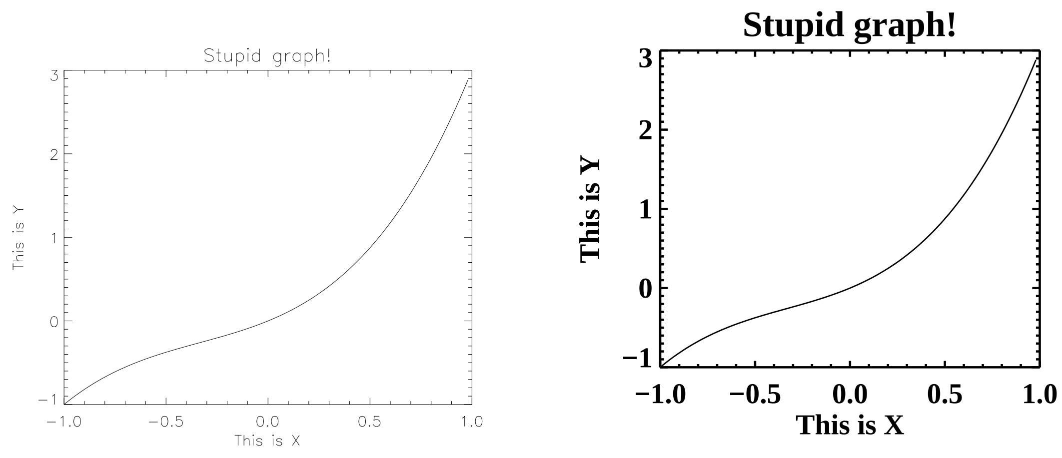

Fig. 1.— Left: Titles are too small, lines too thin, font doesn't look good. Right: Nicer! (But it could be even nicer!!) See §4.

- If you want to evaluate fancy mathematical functions you don't need to use Mathematica. IDL has all those functions too. For example, from the IDL prompt, type `?gamma` to get info on the Gamma function. A software language is just that—a *language*—and the more proficiently you speak it, the better off you are. It's a really bad idea to use different software for different jobs when you don't have to.

5. TEX hints

Some TEX hints:

1. Mathematical convention says: usually, write $(R^2 - x^2)^{1/2}$ instead of $\sqrt{R^2 - x^2}$. In TEX, the scripts are `(R^2-x^2)^{1/2}` and `\sqrt{R^2-x^2}`.

2. When you're doing complicated parenthetical expressions, it's nice to use embedded sizing. TEX does this automatically for you: instead of the not-very-elegant

$$x = \cos[2\pi({B_y \over \lambda} cos(\delta))sin(h)]$$

you can write

$$x = \cos \left[2\pi \left({B_y \over \lambda} cos(\delta) \right) sin(h) \right]$$

The scripts for these are

```
$$ x = \cos [2\pi({B_y \over \lambda} cos(\delta))sin(h)] $$
```

and

```
$$ x = \cos \left[ 2\pi \left( {B_y \over \lambda} cos(\delta) \right) sin(h) \right] $
```

Note the (double) use of `\left[` and `\right]` . Also, note the Roman letters for the trig function, i.e. convention prefers $\cos(ha)$ instead of $cos(ha)$; we accomplish this in TEX by writing `\cos(ha)` (note backwards slash in `\cos`) instead of `cos(ha)`.

3. You can print a Table of Contents by writing `\tableofcontents` in your TEX document (usually at the beginning, but you can do it anywhere). This is very helpful when organizing your lab report into sections and subsections.

4. You can get the proper looking quotes, either 'single' or "double", by writing `‘single’` or `‘‘double’’` .

5. You can get a proper "times" sign, as in 2×3, using `$2 \times 3$`.

6. You can get equations numbered 1a, 2b, and 3c instead of 4, 5, and 6 by using the `mathletters` environment like this:

$$x = \sin(y) \tag{1a}$$

and then you can insert as much text as you want and...

$$z = \tan(y) \tag{1b}$$

$$u = y^{1/2} \tag{1c}$$

by typing the following:

```
\begin{mathletters}
\begin{equation}
x = \sin (y)
\end{equation}

\noindent and then you can insert as much text as you want and\dots

\begin{equation}
z = \tan (y)
\end{equation}
\begin{equation}
u =  y^{1/2}
\end{equation}
\end{mathletters}
```

7. And finally, you can insert things verbatim into TEX, without the TEX translations, by using `\verb$verbatim into TEX$` (all must be on one line) or, for multiple lines, get into the `verbatim` environment by typing

```
\begin{verbatim}
Now we are in the verbatim environment
Here is a multiple line situation

\end{verbatim}
```