

Basics of LaTeX for scientific papers and reports¹

Snežana Stanimirović, November 21, 2016

1. Introduction

Your next homework assignment is to write a short paper about our $l - V_r$ diagram project, essentially explaining the project goals, observations and data processing, and reporting on our preliminary results.

We will try to mimic the whole paper-writing process astro scientists typically go through when writing a paper say for the *Astrophysical Journal*. Most scientific papers follow a well-defined structure (e.g. start with an introduction, go through observations and data processing, and conclude with Conclusions). In addition, most manuscripts are submitted to the journal as LaTeX files. LaTeX is a text-editing package best known for its simple and easy dealing with equations and various symbols. LaTeX has been used for writing scientific papers for many years.

We will start by briefly talking about the basic structure of a scientific paper, and then covering the most essential LaTeX commands. “LaTeX user’s guide and reference manual” by Leslie Lamport is the most commonly used reference book (we have it in the library) in case you are looking for some specific things.

2. A typical outline for a scientific paper

The global outline of a typical scientific paper looks like the following.

1. **Title.** Should encapsulate the contents and meaning.
2. **Authors and affiliations.** Listed based on the level of contribution, with the leading author doing most of the work. If the level of involvement with the paper is similar it is common to use the alphabetical order. Check previous class notes on allocating properly credit.

¹This document was modified from Tristan Lewis’ LaTeX introduction at the UC Berkeley Undergraduate Lab.

3. **Abstract.** A short summary of the paper including the most important results. Purpose is to tell a reader whether it is worth spending more time on the article. Same structure and concerns when writing a poster abstract apply here as well. Your poster abstract will serve as a good draft paper abstract, you may want to make small tweaks when you finish your paper.

4. **Introduction.** Sets the context and provides a summary of the current state of knowledge. What did we hope to achieve, and why were we doing this project?

There is a database of all published astronomy papers:

ADS (http://adsabs.harvard.edu/abstract_service.html). You can use this to search for published research papers on the topic of interest. Remember, science is a cumulative enterprise, it is important to summarize previous work and provide appropriate references.

In our case, the main things to focus on are: what is the $l - V_r$ and what does it show, also why is it useful to plot data in this way.

5. **Observations or Data.** What you observed, who did it, when you did it, any particularities regarding observations and data processing (e.g. observing setup, bandwidth, velocity resolution, any technical problems during observations). Also, include details about the data reduction process (temperature calibration, polynomial fitting).

I suggest having two sub-sections here, one on Observations, and another one on Data Processing. You should show your intermediate results (spectrum before and after correction for the filter gain function), and also discuss how do your spectra compare with the spectra from LDS.

You probably want to re-visit your figures, use larger fonts etc. to make nice, publication-quality graphs.

6. **Data analysis.** How you did your analysis, what did you measure, etc. Sometimes papers can have several sections with specific titles explaining different analysis aspects performed.

In our case, this is the place to show the final $l - V_r$ diagram put together by using observations from the whole class.

7. **Discussion or Interpretation.** What the results mean in terms of previous state of knowledge. How do our results compare with previous studies? Any concerns in terms of interpretation of results? Any suggestions on how to improve the experiment in the future?

8. **Conclusions or Summary.** What have we learned? A summary of what has been done and important results. Prospects for future work.
9. **Acknowledgments.** Thank various people (e.g. classmates) who have contributed in scientific discussions or with data processing. Thank funding sources and various public databases/telescopes you used.
10. **References.** Provide information for all references you used in the paper.

PS. A common mistake: “data” is plural. Therefore, data are used, not data is used!

3. The most essential steps in writing the report

Please copy file called `sample.latex.tex` from the class main directory into your working directory. You probably want to place this file in your Reports directory. Rename the file to something meaningful, this will be your project report file.

Open your `.tex` file using ‘xemacs’ and follow the basic document structure to write your project report. To compile your file and produce a pdf version use:

```
/home/username% latex myfile.tex
/home/username% dvi2pdf myfile myfile.pdf
/home/username% acroread myfile.pdf &
```

4. Starting Your Document in L^AT_EX

Every document in L^AT_EX needs to start off (and end with) several basic commands. There are also a few optional commands that can make your life easier by adding titles and dates to your paper. For example, the document you are now reading starts and ends with the following lines:

```
\documentclass[12pt,preprint]{aastex}
\begin{document}
\title{Sample Latex file}
\author{Sne\v{z}ana Stanimirovi\'}{c}, \today}
[body of document here]
```

:

```
\end{document}
```

The `\documentclass` command sets up what kind of document this will be; in this case, we are going to use `aastex` the LaTeX style file required by the Astrophysical Journal. There is also an optional parameter in square brackets to make the body text size 12pt. The next three commands, `\title`, `\date`, and `\author`, define the title, date, and author, which will go into making the title. Then there is the all-important `\begin{document}` call, which every document needs. This lets the compiler know that your document has actually begun, and it should be entered before any of the document's text makes its way into the file. Finally, `\end{document}` finishes off the environment that was started all the way at the beginning with the `\begin{document}` and tells the compiler what should be obvious from the command: that the document has ended. Remember that any environment you enter with a `\begin{}` command must be closed off by a subsequent `\end{}` command.

To comment out a line of text simply start with `%` and that line will be ignored by the compiler.

5. Compiling LaTeX files

In order to actually go about compiling a tex file and producing a postscript file (.ps) or a pdf file for printing, there are a few simple steps to follow (all of which are typed at the Unix prompt).

```
/home/username% latex samplefile.tex  
/home/username% dvi2pdf samplefile samplefile.pdf  
/home/username% acroread samplefile.pdf &
```

Or:

```
/home/username% dvips samplefile -o samplefile.ps  
/home/username% /usr/local/bin/gv samplefile.ps &  
/home/username% lp -dfloor3 samplefile.ps
```

The first step in the process compiles the tex file you have written. Often times, there will be error messages, which will (usually) point out the exact line where the problem occurs. Other times, warnings are printed. These occur most often when referencing a figure for the

first time or when adding a new figure, and indicate that the internal labeling which LaTeX does might not be right. To solve the problem, simply repeat the first step a second time.

When you are just working on your paper, and want to compile and see how your paper looks, you do not need to worry about converting to a pdf file. Instead, you can simply view the dvi file in xdvi, which will instantly log changes after compilation of the tex file.

```
/home/username% latex filename.tex  
/home/username% xdvi filename.dvi &
```

6. Sections and subsections

Organization is key in any paper, and a scientific paper is no exception. Splitting your document into smaller, cohesive elements which focus on specific topics is beneficial to both the reader and the author. L^AT_EX makes it easy to do this with sections. To begin a section, you simply need to use the `\section{}` command, where the argument to be placed in the curly brackets is the name of the section. If you need further levels of detail within a single section, you can also make use of subsections, which you call predictably enough with the `\subsection{}` command. Sections and subsections are numbered automatically by the compiler for you².

7. Math Mode, Math expressions, and Equations

Let's admit it: easy and professional-looking typesetting of mathematical expressions is one of the biggest reasons that L^AT_EX is useful to people like mathematicians and astronomers. This great capability can't be taken advantage of in normal paragraph mode though—you must first enter something called *math mode*. The most basic way you can do this is by placing such expressions *in-line* in a normal paragraph. To do this, you need to simply surround the mathematical expression with a dollar sign, \$, on either side. So, placing `β` in my otherwise normal sentence will yield a β , and so on. We can also place more complicated mathematical expressions in as well, such as: $\bar{x} = \frac{1}{N} \sum_i^N x_i$. However, this example illustrates that when adding some expressions in-line, especially ones containing fractions or integral signs and summations with specified limits, it comes out a little

²If you want to disable the automatic numbering for any counter, place an asteric directly after the counter's type when you declare it (eg: `\section*{Foreword}`).

squished together and can even affect the line spacing of your document. To avoid this you can utilize math formulas.

In this mode you surround the expression with `\begin{equation}` and `\end{equation}`. The expression then receives its own space on the page, which really allows it to look as professional and elegant as it possibly can. Also, summation or integration limits are placed on the top and bottom of the summation or integral sign where they belong. Your equation is also given an numerical identity in your document which you can later use to reference it. First, let's see the code:

```
\begin{equation} \label{meaneq}
  \bar{x} = \frac{1}{N}\sum_{i=1}^N x_i
\end{equation}
```

And now for the result:

$$\bar{x} = \frac{1}{N} \sum_i^N x_i \tag{1}$$

I included a new command, `\label{}` to give this equation a name that I can refer to later in the document with another command, `\ref{}`. Hence now that I've labeled this equation as `meaneq`, I can later type `\ref{meaneq}` in my document to render whatever equation number it happens to be (in this case, 1). The best part of this is that if I end up going back and putting another equation before this one, `LATEX` will do all the work in renumbering the equations and update all instances of `meaneq` with its new equation number.

8. Figures

Figures break up text, add eye-candy, and (hopefully) convey useful information that would be impossible to get across with the use of words alone. The best way to see how to add figures is simply to look at a sample of code, such as the following:

```
\begin{figure}[htb]
\epsscale{0.6}
\plotone{galaxy.ps}
\caption{\label{f:myfig} HI emission spectrum of SPITZER192404+145632.}
\end{figure}
```

First, we see that like other environments, this figure environment is being delimited with `\begin{figure}` and `\end{figure}`. The argument to `\begin{figure}` in square

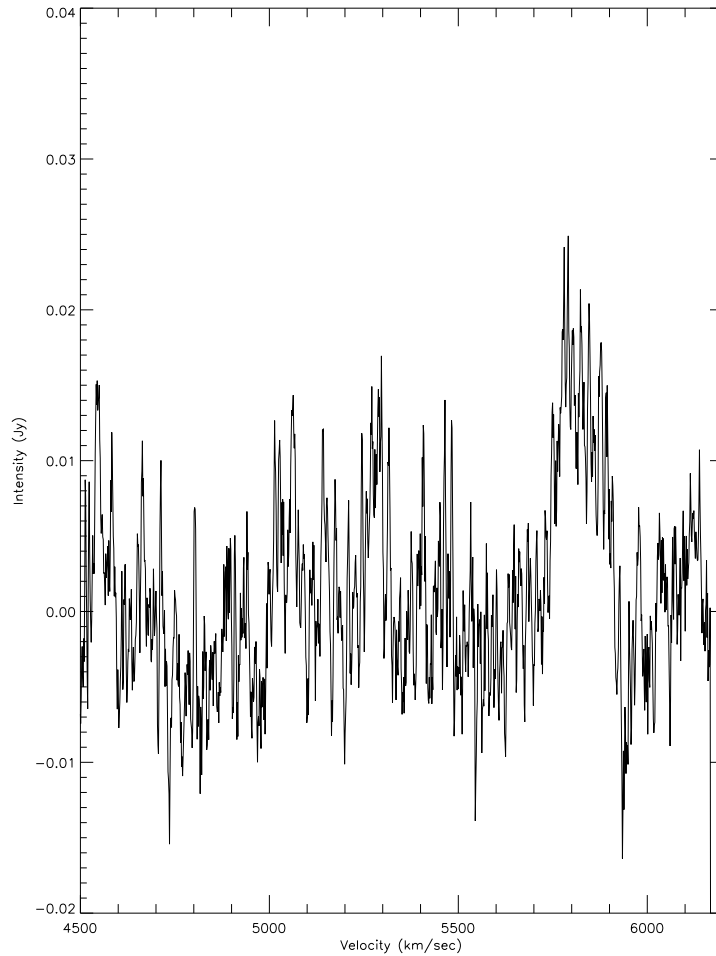


Fig. 1.— HI emission spectrum of SPITZER192404+145632.

brackets has to do with the figure placement. \LaTeX likes to have its own say as to exactly where your figures get placed, but you can give it some priority by adding a list of locations in square brackets, where **h** means **here**, **t** means **top** of page, and **b** means **bottom** of page. Furthermore, you can stress your preference by adding an exclamation mark, **!**, in the first spot. So, if you really want \LaTeX to put your graphic exactly where it appears in the `.tex` code, you can use `[!h]` and hope it listens to you.

Next, comes the actual call to `\plotone{}` which places the graphic (postscript file called `galaxy.ps`) in the output file. I have also scaled the figure size to 60% of the actual size using `\epsscale{0.6}`. Make sure your figure is in your working directory so \LaTeX can find it.

Two more commands you can throw in before you end your figure environment are `\caption{}` and `\label{}`. `\caption{}` places a caption underneath the figure, and labeling the figure works just as it did with equations, where you can later refer back to the figure’s label name (in this case, `linplot`), with the `\ref{}` command. The last two commands simply end the environments that were started here in the reverse order that they were started. Figure 1 shows how our figure came out.

9. Tables and Tabular

Tables can be an effective way to present data (especially numerical data) in a condensed, easy-to-read fashion. I suggest you present the Galactocentric radius and orbital velocity for all our measurements in a table.

Let’s say I wanted to compare the theoretical values for a quantity, $\Gamma(\alpha)$, with the measured values at different values of α ranging from 0-10. Here’s what a table illustrating these values might look like:

Table 1: Observed pulsars.

α	Γ_{meas}	Γ_{theor}
0	3.4	3.2
2	5.6	5.3
4	10.5	10.8
6	24.6	22.5
8	30.1	28.9
10	37.8	34.0

First let’s examine this table’s code:


```

\begin{table}[htb]
\caption{Observed pulsars. }
\centering
\label{t:observed_psr}
\begin{tabular}{rcc}
\noalign{\smallskip} \hline \hline \noalign{\smallskip}
 $\alpha$  &  $\Gamma_{meas}$  &  $\Gamma_{theor}$  \\
\hline
0 & 3.4 & 3.2 \\
2 & 5.6 & 5.3 \\
4 & 10.5 & 10.8 \\
6 & 24.6 & 22.5 \\
8 & 30.1 & 28.9 \\
10 & 37.8 & 34.0 \\
\hline \hline \noalign{\smallskip}
\end{tabular}
\end{table}

```

`\begin{table}` start the table environment, and we have specified the location of the table on the page. Now we start tabular environment and define the column format. Each letter represents a single column. The most common letters to choose from for defining columns are r, l, and c, which determine whether the column data will be right justified, left justified, or centered.

After specifying how the columns behave, the table content can then be entered. We do this row-by-row, using the ampersand (&) to switch over to the next column for each row. In this way I specify my first row of values, α , Γ_{meas} , and Γ_{theor} , which act as my column headers. With that row's three columns specified, it's time to tell the compiler that I want to move down to the next row, which is accomplished with double backslashes, `\\`. For this table, I've chosen to add two horizontal lines with `\hline` after the column headings. Then I proceed with the subsequent rows' entries, making sure to include a `\\` after each one until I reach the end. Now we end tabular and table environments.

10. Fonts and Shapes

Also present in L^AT_EX are font series and shapes. Essentially, these allow you to make your text **bold-faced**, *emphasized*, *italicized*, or *slanted*. These can be useful if you wish to

add the occasional stylistic touch to your words. The `\textbf{}`, `\emph{}`, `\textit{}`, and `\textsc{}` commands, respectively, let you use these features.

11. Special Symbols

There are oodles of special symbols available for use in L^AT_EX—so many that it would be futile for me to try to outline them all here. Luckily, there are some good sources on the internet for you to reference when you’re just aching to stick that proportionality sign or real numbers symbol into one of your equations. It’s probably the easiest just to Google “latex symbols” for all the information you’d ever need.