

Introduction to L^AT_EX*

Elizabeth Menninga
Odum Institute for Research in Social Science
University of North Carolina at Chapel Hill
menninga[at]email.unc.edu

February 7-8, 2013

1 Schedule

- Day 1
 - Getting started
 - Creating a basic document
 - Making the typical features of a full article
 - Adding figures and tables to your document
- Day 2
 - Mathematical expressions
 - Bibliographies
 - Example: UNC thesis
 - Creating a slideshow
 - Creating a poster
- Extras
 - Examples: Making a CV and cover letter

2 What is L^AT_EX?

L^AT_EX is a markup language and document preparation system.

Advantages: sharper presentation, more control over formatting and positioning of tables, mathematical symbols, equations, and graphs. It can also be used for a wide range of documents, from posters and presentation slides to articles and even books.

Disadvantages: higher learning curve.

3 Getting Started

Here are some resources to get L^AT_EX onto your computer and help for once it is up and running.

*These notes draw from short courses taught by Evan Parker-Stephen, Jamie Monogan, and Jeff Harden.

3.1 Essential components

You need two things to start using \LaTeX : the distribution—software that takes the code/text you write and makes it into a document—and a text editor for creating that code.

- Distributions

MikTeX (Windows): <http://www.miktex.org/>

MacTeX (Mac): <http://www.tug.org/mactex/2009/>

TeX Live (Linux): <http://tug.org/texlive/>

- Text editors

TeXnicCenter: <http://www.texniccenter.org/>

WinEDT: <http://www.winedt.com/>

Texmaker: <http://www.xmlmath.net/texmaker/index.html>

3.2 Resources

As questions emerge when you use \LaTeX , here are a few resources you may want to consider:

- The Comprehensive \TeX Archive Network: <http://www.ctan.org/>
- \LaTeX WikiBooks: <http://en.wikibooks.org/wiki/LaTeX>
- “The Not So Short Introduction to \LaTeX ”: <http://tobi.oetiker.ch/lshort/lshort.pdf>
- \LaTeX Community Message Boards: <http://www.latex-community.org/>
- Odum web tutorials: <http://www.odum.unc.edu/odum/contentSubpage.jsp?nodeid=671>

4 First Steps

To begin, open TeXnicCenter and save a new document. By typing in \LaTeX , you can control everything from the spacing of your document to the style of your tables to the (automatic) generation of your references. At a minimum, you will need to use the following typesetting commands to compile your document.

```
\documentclass[12 pt]{article}
\begin{document}
```

Start typing here

```
\end{document}
```

Notice the basic \LaTeX syntax of writing a command: `\command[<optional arguments>]{<text>}`.

- `\command` what you want \LaTeX to do
- `[]` square brackets for optional arguments
- `{ }` curly brackets for the main argument or text

4.1 The Preamble

In a real document, you will put all introductory commands that control the general layout of the paper together at the beginning of the document in a section referred to as the “preamble”. These commands are calls to packages that affect the document’s layout, styles, and so forth. This document, for example, was created using the preamble listed below. At this point these particulars are not nearly as important as the general lesson: you have control over the basic appearance of your documents and much of it is established by the packages you invoke in the preamble. Then T_EX’s typesetting algorithms take commands from those packages and those that you type to make the document.

```
\documentclass[12pt, letterpaper]{article}

%=====Packages & Commands=====
\usepackage{graphicx} % Graphics
\usepackage{indentfirst} % Tells LaTeX to indent every paragraph
\usepackage{setspace} % To set line spacing
\usepackage[longnamesfirst]{natbib} % For references
\usepackage{booktabs} % For tables
\usepackage{rotating} % For sideways tables/figures
\usepackage{amsmath} % Some math symbols
\usepackage[margin = .75in]{geometry}
\usepackage{subfig}
\usepackage{mdwlist}
\usepackage{url}
\usepackage{verbatim}
\urlstyle{same}
\usepackage{multirow}
\usepackage{multicol}
\usepackage[nolists]{endfloat} % Figures and tables at the end
\bibpunct{(}{)}{;}{a}{},{,} % Reference punctuation
\captionsetup[subfloat]{position = top, font = large} % For sub-figure captions
\usepackage{times}
\usepackage[compact]{titlesec}
\usepackage[colorlinks = TRUE, urlcolor = blue, linkcolor = black,

%=====Article Title, Authors=====
\title{Introduction to \ltx\thanks{These notes draw from short
courses taught by Evan Parker-Stephen, Jamie Monogan and Jeff Harden.}}
\author{Elizabeth Menninga \\\ Odum Institute for Research in Social Science \\\
University of North Carolina at Chapel Hill \\\ menninga[at]email.unc.edu}
\date{February 7-8, 2013}
\vspace{-24pt}

%=====Begin Manuscript=====
\begin{document}
\thispagestyle{empty}
\maketitle
\tableofcontents
```

5 Creating an Article

5.1 The preamble, title, and abstract

Let's make an article!

```
\documentclass[12pt, letterpaper]{article}

%=====Packages & Commands=====
\usepackage[indentfirst] % Tells LaTeX to indent every paragraph
\usepackage{setspace} % To set line spacing
\doublespacing % the setspace package allows you to use this for the whole document or any part
\newcommand{\ltxs}{\LaTeX\space}
```

Now our document needs a title, author, and date. Notice the use of double backslashes (\\) to end the line with a hard return.

```
\title{\ltxs Notes}
\author{Your Name Here \\ Odum Institute for Research in Social Science}
\date{February 7-8, 2013}
\vspace{24pt}
```

Now we need to actually start the document. This is done with the `\begin{document}` command and is followed by the `\maketitle` command, which tells TeX to put the title/author/date information that we typed above into our document. You should also type `\end{document}` a few lines down. After that, you can make an abstract:

```
\begin{document}
\maketitle

\begin{abstract}
Abstract goes here.
\end{abstract}
```

The rest of the paper goes here.

```
\end{document}
```

5.2 Section headings and paragraphs

There are three levels of section headings available: `\section{<text>}`, `\subsection{<text>}`, and `\subsubsection{<text>}`. To remove the numbering from any of these, type an asterisk after the word "section," as in `\subsection*{<text>}`.

5.3 Page numbers

To remove the page number from the title page, type `\thispagestyle{empty}` after your abstract.

To change where the page numbers are located, you need to add the following code to the preamble. The command `\rfoot{\thepage}` sets where the number goes. Change it to `l` for left and `head` for the top of the page.

```
\usepackage{fancyhdr} % Loads the package
\pagestyle{fancy} % Tells TeX you are going to make changes to the header/footer
\fancyhf{}
\rfoot{\thepage} % ''Put the page number on the right side of the footer.''
\renewcommand{\headrulewidth}{0pt} % Removes a line that appears from this change
```

For different types of page numbers, use the `\pagenumbering{}` command in your preamble (or where you want the numbering to be styled in that way).

```

\pagenumbering{Alph} % Uppercase letters
\pagenumbering{alph} % Lowercase letters
\pagenumbering{Roman} % Uppercase roman numerals
\pagenumbering{roman} % Lowercase roman numerals
\pagenumbering{arabic} % Arabic numerals (default)

```

5.4 Special characters

In \LaTeX , several characters have special meaning for programming purposes. Some examples are: `&` separates columns in tables, `$` starts a math environment, `%` starts a comment, and `\` precedes nearly any active command. If you want to include one of these characters in your text, \LaTeX needs a signal that this is how you want to use the character. The code to include these characters as well as other tricky characters, is listed below:

To get this...	... type this
#	<code>\#</code>
\$	<code>\\$</code>
%	<code>\%</code>
&	<code>\&</code>
-	<code>_</code>
{	<code>\{</code>
}	<code>\}</code>
^	<code>\^{}{}</code>
~	<code>\~{}{}</code>
\	<code>\$_backslash\$</code>
---	---
--	--
-	-
“	<code>\‘</code>
”	<code>\’</code>

5.5 \LaTeX Environments

\LaTeX offers several “environments,” each of which gives \TeX specific formatting instructions. We have used a few already. Every environment begins and ends in the same manner:

```

\begin{<environment name>}
...
\end{<environment name>}

```

5.5.1 Bulleted Lists

Creating a bulleted list requires the following environment:

```

\begin{itemize}
\item Item \#1
\item Item \#2
\item Item \#3
\end{itemize}

```

Which gives:

- Item #1
- Item #2

- Item #3

By including a character in square brackets after `\item`, then we can use any kind of bullet we like. See <http://www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf>. For example, `\item[\circ]` will place an open circle in place of the bullet. Text can also be a bullet, as in `\item[A] \#1`. If you want the list to be numbered, use the `enumerate` environment.

Also, you can nest lists up to four times. Just remember you have the same number of occurrences of `\begin{}` as `\end{}`. In these cases, it might help you to organize the list by indenting each nested list.

```
\begin{itemize} % Start first list
\item Item \#1
\begin{itemize} % Start second list
\item Item \#1.2
\begin{itemize} % Start third list
\item Item \#1.3
\end{itemize} % End third list
\end{itemize} % End second list
\item Item \#2
\item Item \#3
\end{itemize} % End first list
```

Which gives:

- Item #1
 - Item #1.2
 - * Item #1.3
- Item #2
- Item #3

5.6 Fonts

You can change several aspects of the font, including the family, type, size, and color.

5.6.1 Font family

Visit this link: <http://www.tug.dk/FontCatalogue/> to see the fonts available in \TeX . Clicking on each one will give preamble commands to use the font. Note that not all fonts available with \TeX are available with PDF, so some might cause errors.

5.6.2 Font type

Several font types are available:

Type	Command	Result
Italic	<code>\textit{This is the result.}</code>	<i>This is the result.</i>
Slanted	<code>\textsl{This is the result.}</code>	<i>This is the result.</i>
Small Capitals	<code>\textsc{This is the result.}</code>	THIS IS THE RESULT.
Bold	<code>\textbf{This is the result.}</code>	This is the result.
Sans-serif	<code>\textsf{This is the result.}</code>	This is the result.
Monospace	<code>\texttt{This is the result.}</code>	This is the result.

5.6.3 Font size

Font size commands depend on what the default font is for your document. You can set this in the first line of the `.tex` file: `\documentclass[12pt, letterpaper]{article}`. If you do not specify an option, 10 point is used. To increase or decrease font size, use one of the following commands:

- `\tiny`
- `\scriptsize`
- `\footnotesize`
- `\small`
- `\normalsize`
- `\large`
- `\Large`
- `\LARGE`
- `\huge`
- `\Huge`

These commands can be used as an environment or with curly braces (e.g., `\footnotesize{<Text>}`).

5.6.4 Font color

Font color is changed with the `color` package (add `\usepackage{color}` to your preamble). For more color options as well as the ability to change color within a table, you can also add `\usepackage[usenames, dvipsnames, svgnames, table]{xcolor}` to the preamble. The command `\textcolor{<name of color>}{<Text you want in that color.>}` will change the color of text. To see more details on available colors, see <http://en.wikibooks.org/wiki/LaTeX/Colors>.

5.7 Footnotes and endnotes

Footnotes are created by the `\footnote{}` command. For example, take a look at the bottom of this page.¹ If you would like to use endnotes, insert the following into your preamble:

```
\usepackage{endnotes}
\let\footnote = \endnote
```

Then, use the `\footnote{}` command as usual, but insert the command `\theendnotes` at the point where you would like the endnotes to start.

5.8 Spacing and block quotes

With the `setspace` package loaded, you can change spacing within your document. If you want a double-spaced document, type `\doublespacing` after `\maketitle` or the Abstract. If you would like to switch to single or one-half spacing at some point, use the commands `\singlespacing` and `\onehalfspacing`. You can also use these commands as environments, as in

```
\begin{onehalfspace}
<text>
\end{onehalfspace}
```

To make a block quote, use the `quote` environment.

¹Here is the footnote.

6 Floats

Floats are objects that go in your document but are not part of the main text, such as tables and figures. The main aspects to control are the content of your float, and its size and placement in the document.

6.1 Tables

You can create a table with two different environments: `tabular`, which handles the structure and content, and `table`, which handles the labeling, placement, and other descriptive aspects. Let's start with the following code using `tabular`:

```
\begin{tabular}{lcr} % Three columns, justified left, center, right.
1 & 2 & 3 \\ % Use & to separate columns, \\ to end a row.
4 & 5 & 6 \\
7 & 8 & 9 \\
\end{tabular}
```

This produces:

	1	2	3
	4	5	6
	7	8	9

Now let's center it, and make it a table using `table`.

```
\begin{table}
\begin{center}
\caption{An Important Table} % A caption
\toprule[1.2pt] % A top line (from the booktabs package)
\begin{tabular}{lcr} % Three columns, justified left, center, right.
1 & 2 & 3 \\ % Use & to separate columns, \\ to end a row.
4 & 5 & 6 \\
\midrule % A horizontal line (booktabs)
7 & 8 & 9 \\
\end{tabular}
\bottomrule[1.2pt] % Same as \toprule
\end{center}
\label{importanttable} % A label so we can reference it in text
\end{table}
```

Which produces:

Table 1: An Important Table

1	2	3
4	5	6
7	8	9

You can make text stretch across columns with `\multicolumn{<number of columns>{alignment}{Text}}`.

```
\begin{singlspacing}
\begin{table}[H]
\begin{center}
\caption{Table 2}
\label{Multicolumn}
\begin{tabular}{lrrrrrr}
\toprule[1.2pt]
```



```

& \multicolumn{3}{c}{OLS} & \multicolumn{3}{c}{MR} & \\
Variable & Coefficient & SE & RCSE & Coefficient & SE & KSE & \\
\midrule
\textit{Age} & 5 & 2 & 1 & 5 & 2 & 1 & \\
\textit{Gender} & -4 & 3 & 4 & -3 & 2 & 2 & \\
\bottomrule[1.2pt]
\end{tabular}
\newline
\parbox{5.5in}{\small{Note: These numbers are completely made up.}}
\end{center}
\end{table}
\end{singlspacing}

```

Which produces:

Table 2: Table 2

Variable	OLS			MR		
	Coefficient	SE	RCSE	Coefficient	SE	KSE
<i>Age</i>	5	2	1	5	2	1
<i>Gender</i>	-4	3	4	-3	2	2

Note: These numbers are completely made up.

6.2 Figures

With the `graphicx` package you can insert a wide variety of files to use as figures such as .pdf, .eps, .jpeg, and others. Similar to tables, two commands are used to create a figure. The `\includegraphics{}` command calls the file into your document and the `figure` environment is used for placement, captioning, etc.... Just make sure the file is in the same folder as your .tex file. Here is an example:

```

\begin{figure}
\begin{center}
\includegraphics[scale = .5]{example-figure1.pdf} % Note the scale command
\caption{Our First Figure}
\label{basic-figure}
\end{center}
\end{figure}

```

Which produces:

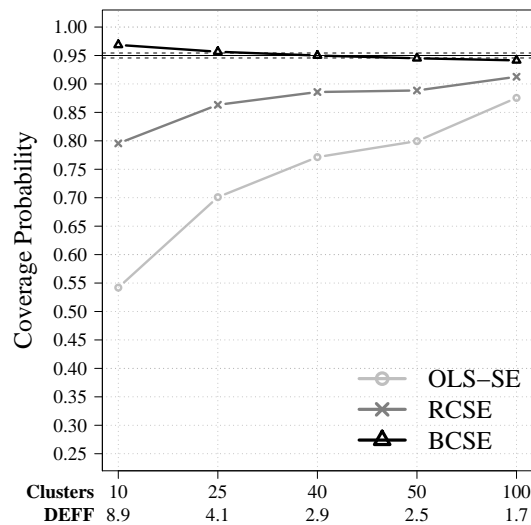


Figure 1: The First Figure

You can also put two (or more) graphic files together in one figure and have a caption for each with the `subfig` package:

```

\begin{figure}
\begin{center}
\subfloat[One Graphic]{\includegraphics[scale = .4]{example-figure1.pdf}}
\subfloat[The Other Graphic]{\includegraphics[scale = .4]{example-figure2.pdf}}
\caption{Two Graphics in One Figure}
\label{two-graphics}
\end{center}
\end{figure}

```

Which produces:

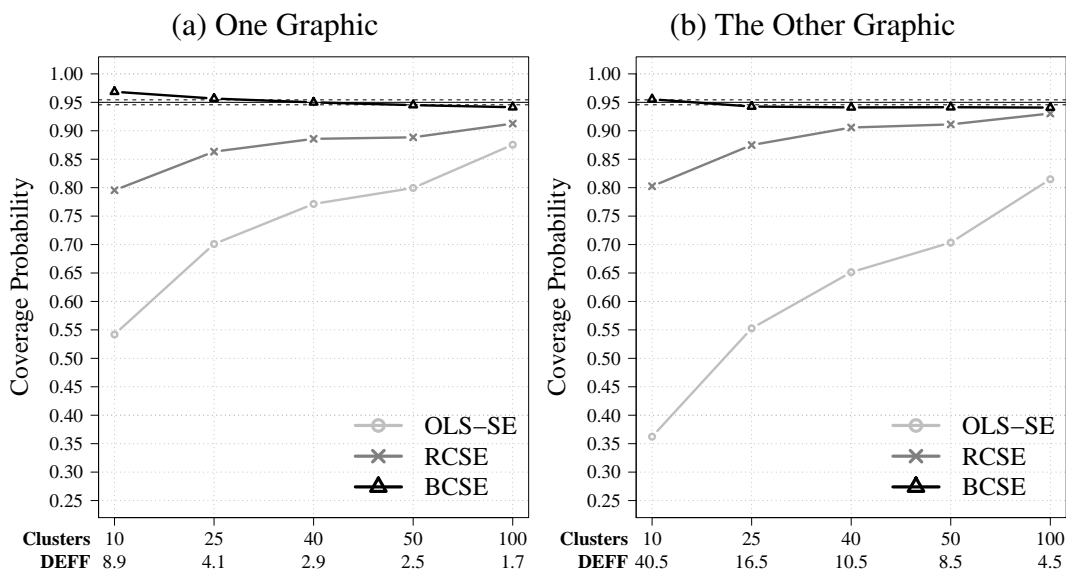


Figure 2: Two Graphics in One Figure

6.3 Float placement and rotation

Float placement is governed by an optional argument after you type `\begin{table}` or `\begin{figure}`. You type square brackets and a letter: `h` for exactly where the code is, `t` for top of the next page, `b` for bottom of this page, and `p` for a dedicated page of floats. For example, `\begin{table}[h]` will try to place your table where you insert the code.

However, this does not always work as well as it sounds. Sometimes the structure of the document makes it so that \TeX will put the table in odd places. One solution is to use the `float` package. This contains the argument `[H]`, which goes in the same place as `[h]`. This overrides \TeX 's figure defaults and puts the figure exactly where the code is located.

You can also rotate floats if they fit better sideways with the `rotating` package. This is done exactly as above, except you use the environments `sidewaystable` or `sidewaysfigure`. Here is an example:

```
\begin{sidewaysfigure}
\begin{center}
\subfloat[First Sideways Graphic]{\includegraphics[scale = .5]{example-figure1.pdf}}
\subfloat[Second Sideways Graphic]{\includegraphics[scale = .5]{example-figure2.pdf}}
\caption{Two Sideways Graphics in One Figure}
\label{two--sideways-graphics}
\end{center}
\end{sidewaysfigure}
```

Which produces the two figures on the next page (because of fit).

6.3.1 Placing floats at the end of the document

Many journals want submissions with figures and tables at the end, each on a separate page, and with some text indicator showing where each float goes. This can be accomplished easily with the `endfloat` package. With that package in your preamble, remove the `[h]` or `[H]` options and simply place your floats where they fall in the document. \TeX will then place “[Figure 1 about here.]” in that place, and the figure (or table) will go to the end. If you want to change this text, insert the following code in the preamble, making any changes where appropriate:

```
\renewcommand{\figureplace}{
\begin{center}
[Insert \figurename~\thepostfig\ here]
\end{center}}
\renewcommand{\tableplace}{
\begin{center}
[Insert \tablename~\theposttbl\ here]
\end{center}}
```

6.4 Referencing floats in text

One very helpful aspect of writing in \LaTeX is the ability to make reference to floats. Notice that we used `\label{}` to name each float. When we want to refer to that float in text, we use the `\ref{}` command. For instance, to refer to the table above—type the following:

As you can see in Table `\ref{Multicolumn}`, the results support my hypotheses.

This produces:

As you can see in Table 2, the results support my hypotheses.

You can label many things: sections/subsections/subsubsections, footnotes/endnotes, itemized lists, etc... I strongly recommend using this feature. It is helpful if you ever change the position of the labeled object because the reference will adjust accordingly. For instance, if you label a footnote and refer to it, then delete the footnote appearing before the labeled one, the counter will update to match.

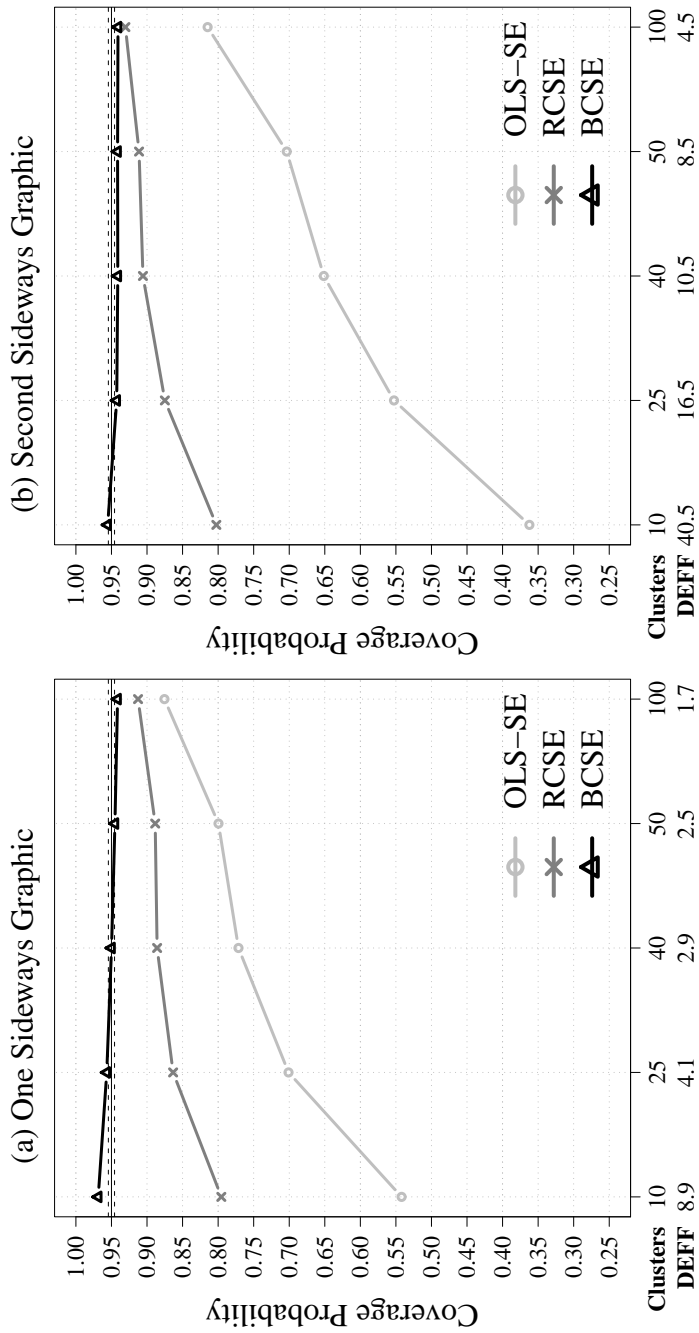


Figure 3: Two Sideways Graphics in One Figure

7 Mathematical Expressions

The math environments in \LaTeX expect different syntax than the regular text-processing environments. You can write in math in four ways: in-line text, the `equation` environment, the `eqnarray` environment, and the `array` environment. There are lots of different symbols you can make. Take a look at TeXnicCenter's drop down menus for examples.

The easiest call to the math environment, in-line text, starts and ends with a `$`. Hence, `$x^2$` returns x^2 , or `y_i` returns y_i . Normally, this environment will try to cram your equation into the normal space of a text line. For example, try the the summation command: `$\sum_{i=1}^n \{x^2\}$` . If you prefer, though, you can use the `\displaystyle` command: this makes the equation look sharper, but adds whitespace before and after the line to make room. See how the summation looks with this option: `$\displaystyle\sum_{i=1}^n \{x^2\}$` .

A second option that we have is to use the `equation` environment. Notice that equations can be labeled and referenced just like floats.

```
\begin{equation}
\frac{\partial}{\partial x} (z \times x^3) = 3z \times x^2 \label{derivative}
\end{equation}
```

$$\frac{\partial}{\partial x} (z \times x^3) = 3z \times x^2 \tag{1}$$

Typing `\begin{equation*}` will suppress the equation numbering, but requires the `amsmath` package to be loaded. A third option is to use the `eqnarray` environment. This syntax also illustrates how to write Greek letters and another way to suppress equation numbering.

```
\begin{eqnarray}
\label{normal}
\phi(x) & = & \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} \\
\Phi(x) & = & \int_{-\infty}^x \phi(t) dt \nonumber
\end{eqnarray}
```

Which produces:

$$\begin{aligned} \phi(x) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} \\ \Phi(x) &= \int_{-\infty}^x \phi(t) dt \end{aligned} \tag{2}$$

Notice that the `\phi` symbol formats the first column right-justified, the second centered, and the third left-justified. This makes it easier to line-up the equations.

As a final example, let's create a matrix using this code:

```
\[ \left[
\begin{array}{ccc}
a & & b & & c \\
d & & e & & f \\
g & & h & & i
\end{array}
\right] \]
```

Which produces the matrix shown below. The only optional parts of this code are `\left[` and `\right]`, which control what symbol will surround the matrix.

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

8 Automatic Citations

\LaTeX will create citations and a reference page for you. There are a few ways to do this, but I think the best is through the `natbib` package. To set it up, first add `\usepackage{natbib}` to your preamble. Next, you need a file with citations in it. We will use the file `sample.bib` for this purpose. Place that in the same folder as your `.tex` file, or include the complete path to the file in the `{ }`. I recommend using JabRef (available free at <http://jabref.sourceforge.net/>) to manage this file. Open `sample.bib` in JabRef to see how to add entries to your bibliography file. Notice that one line you need to define is the name for that citation that you will use in your `.tex` file. I generally follow the convention `<first author's last name><year>`.

Finally, you need a `.bst` file which tells \TeX the citation style you would like to use. In this case, we will use `aprs.bst` (*American Political Science Review*). Like with `sample.bib`, place these in the same folder as your `.tex` file, or include the complete path to the file in the `{ }`.

With these files in place, add the following code just before `\end{document}` in your `.tex` file:

```
\bibliographystyle{aprs}
\bibliography{sample}
```

The `natbib` package contains a number of ways to cite something from your `.bib` file. See <http://merkel.zoneo.net/Latex/natbib.php> for all the available options. Also, that reference sheet shows how to use the `\bibpunct{ }` command to set the punctuation you want with citations. Here are some examples. Hyperlinked citations come from using the `hyperref` package, though be careful because this package is notorious for producing errors. If you want to use `hyperref`, make sure it is the last package you load in the preamble. Also, you have to compile the document three times to get a new citation to appear.

- `\citep{Alden01}` → (Alden 2001)
- `\citet{Alden01}` → Alden (2001)
- `\citep{Bercovitch91}` → (Bercovitch, Anagnoson, Wille and Barro 1991)
- `\citep{Bercovitch91}` → (Bercovitch et al. 1991) [second time uses “et al.”]
- `\citep*{Bercovitch91}` → (Bercovitch, Anagnoson, Wille and Barro 1991) [back to the full author list with the *]
- `\citep[][]{Beardsley08, Alden01, Barro05}` → (Beardsley 2008; Alden 2001; Barro and McCleary 2005)
- `\citep[][104]{Beardsley08}` → (Beardsley 2008, 104)
- `\citep[see][]{Beardsley08}` → (see Beardsley 2008)
- `\citep[e.g.,][but see \citealt{Alden01}]{Beardsley08}` → (e.g., Beardsley 2008, but see Alden 2001)

Finally, there are many options for `natbib` in the preamble, which are called with square brackets. For instance, `\usepackage[longnamesfirst]{natbib}` will write out all of the authors in a citation with four or more authors the first time, then use “et al.” after that.

9 Creating a Slideshow

There are several packages available to make presentation slides in \LaTeX , with the most common being the `beamer` class. The first thing you need to do is specify the document class as `beamer`, then set the theme with the `usetheme{ }` command. We will use the `Warsaw` theme here, but there are many more (see <http://latex.simon04.net/>). Then you put in your title and name in the standard way. Also, the `\institute{ }` argument allows you to place your affiliation on the title slide.

```

\documentclass{beamer}
\usetheme{Warsaw}

\title{Example Presentation Created with \texttt{beamer}}
\author{Your Name}
\institute{UNC}
\date{\today}

```

Each slide is created with the `frame` environment. The first two frames you will want to make are the title slide and a slide with an outline of the presentation. This is done with the `\tableofcontents` command (which also works in articles). This command organizes your sections/subsections/subsubsections into a table of contents.

```

\begin{document}

\begin{frame}
\titlepage
\end{frame}

\section[Outline]{}
\begin{frame}
\tableofcontents
\end{frame}

```

To make substantive slides, use `\section{}`, `\subsection{}`, and `\subsubsection{}` like usual, and invoke the `frame` environment each time you want a new slide. To create a title for the slide, use `\frametitle{}`. Generally information is presented in bulleted lists. If you would like each bullet point to appear in sequence, use the `<#->` format:

```

\begin{frame}
\frametitle{Making an Itemized List}
\begin{itemize}
\item<1-> Item \#1
\item<2-> Item \#2
\item<3-> Item \#3
\item<4> Item \#4 (only stays for 4th click)
\item<5-> Item \#5
\end{itemize}
\end{frame}

```

You can also add figures as before:

```

\begin{frame}
\begin{figure}
\begin{center}
\includegraphics[scale = .4]{example-figure1.pdf} % Note the scale command
\caption{Our First Figure}
\label{basic-figure}
\end{center}
\end{figure}
\end{frame}

```

You can also make text stand out with the `block` environment:

```

\begin{frame}
\frametitle{Using Blocks}
\begin{block}{Citations}
\begin{itemize}
\item<1-> \citet{Beardsley08}
\item<2-> \citet{Barro05}
\end{itemize}
\end{block}
\end{frame}

```

Finally, to make references work, you need to do some lower-level \TeX programming (for one line of code). Make sure you have `\def\newblock{}` before the bibliography lines.

```

\begin{frame}
\begin{scriptsize}
\def\newblock{}
\bibliographystyle{apsr}
\bibliography{sample}
\end{scriptsize}
\end{frame}

```

10 Creating a Poster

One of the strongest (in my opinion) features of \LaTeX is its ability to make scientific posters relatively easily. This is done with the `sciposter` class. Below are some basic commands to get one started, but take a look at the package manual (<http://mirror.unl.edu/ctan/macros/latex/contrib/sciposter/scipostermanual.pdf>) to see all of the options. Here is a sample preamble:

```

\documentclass[landscape, plainboxedsections]{sciposter}
\usepackage{multicol}
\usepackage{sectionbox}
\usepackage{graphicx}
\usepackage{subfig}
\usepackage[numbers, longnamesfirst, sort]{natbib}

```

Next, you need to add title and author information. In addition, you can put a logo in the left (and/or right) corner:

```

\leftlogo[1.25]{unccentered.pdf}
\norightlogo

\title{Poster Title}
\author{Elizabeth Menninga}
\institute{University of North Carolina at Chapel Hill}
\conference{Name of Conference}

```

Then to break up the content into columns, use the `multicols{}` command. Once this is used, you add content in the normal way, including text, figures, tables, footnotes or endnotes, and references.

```

\begin{document}
\maketitle

\begin{multicols}{2}
\setlength{\columnseprule}{0pt} % Takes away the lines separating columns

```



```

\begin{abstract}
Abstract goes here.
\end{abstract}

\section{Section \#1}

\section{Section \#2}

\bibliographystyle{apsr}
\bibliography{sample}
\end{multicols}
\end{document}

```

11 Generating New Commands

Sometimes you will find that typing \LaTeX commands can be burdensome. For example, if you regularly referred to matrix \mathbf{X}_{ij} in the text, you would need to type `\mathbf{X}_{ij}` every time. `\newcommand{ }{ }` allows you to create short commands in the place of longer ones. For example, because I refer to \LaTeX a lot in this document, I created the new command `\newcommand{\ltxs}{\LaTeX\space}`. This way, I did not have to write out `\LaTeX` every time, but rather `\ltxs`, which is easier. Here are some more examples:

```

\newcommand{\be}{\begin{enumerate}}
\newcommand{\ee}{\end{enumerate}}
\newcommand{\bq}{\begin{quote}}
\newcommand{\eq}{\end{quote}}
\newcommand{\bd}{\begin{description}}
\newcommand{\ed}{\end{description}}
\newcommand{\bi}{\begin{itemize}}
\newcommand{\ei}{\end{itemize}}
\newcommand{\Mc}{ $M_i^{(c)}$ }
\newcommand{\Ma}{ $M_i^{(a)}$ }
\newcommand{\Wa}{ $\mathbf{W}^{(a)}_t$ }
\newcommand{\Wp}{ $\mathbf{W}^{(p)}_{it}$ }

```

12 Some More Commands

Here are a few more commands that can be useful.

- **To write out URLs:** call the package `url`, then type `\url{www.urlgoeshere}`.
- **To write in ASCII text:** use the `verbatim` environment or `"` for inline text.
- **To get a two-column document:** use the command `\twocolumn` in the preamble. If you want only a small part to be two column, call the `\multicol{ }` package. This allows you to use the `multicols` environment like when making a poster:

```

\begin{multicols}{2} % Two columns
<Text here>
\end{multicols}

```

13 Searching for Features: Some Examples

13.1 Making a CV

L^AT_EX can do a lot of things and the best way to explore its capabilities is online. I recommend starting any search with “latex,” then a description of what you want to find. Here is an example. Let’s say you want to find out how to make a CV in L^AT_EX. The Google search “latex create a cv” produces several results in the first few hits that have L^AT_EX code you can copy and paste into TeXnicCenter. Once you find one that you like, it is simple enough to fill in your own information.

13.2 Writing a cover letter

Now let’s say you want to write several cover letters in L^AT_EX. This can be a big task because each one goes to a different job, but you want to make sure it is done correctly. Googling “cover letters in latex” produces <http://www.zaphu.com/2007/12/07/automated-cover-letters-in-latex/>, which can help. The basic idea is that you create one document with the content of the letter, then a separate document that defines the information for one job, fills that information into the letter, then moves on to the next job.

References

- Alden, Chris. 2001. *Mozambique and the Construction of the New African State: From Negotiations to Nation Building*. Palgrave.
- Barro, Robert J. and Rachel M. McCleary. 2005. “Which Countries have State Religions?” *The Quarterly Journal of Economics* pp. 1331–1370.
- Beardsley, Kyle. 2008. “Agreement without Peace? International Mediation and Time Inconsistency Problems.” *American Journal of Political Science* 52(4):723–740.
- Bercovitch, Jacob, J. Theodore Anagnoson, Donnette L. Wille and Robert J. Barro. 1991. “Some Conceptual Issues and Empirical Trends in the Study of Successful Mediation in International Relations.” *Journal of Peace Research* 28(1):7–17.