Contents

1 Introduction to \LaTeX{} 2

2 Basic Layout of a \LaTeX{} Document 2

3 The Preamble of a \LaTeX{} Document 3
  3.1 \LaTeX{} Packages ................................................. 3
  3.2 Creating Your Own Commands ...................................... 3
  3.3 Formatting Your Document .......................................... 4

4 The Body of a \LaTeX{} Document 4
  4.1 The Text Environment ............................................ 4
  4.2 The Math Environment ............................................ 5
    4.2.1 Math Within Text ........................................... 5
    4.2.2 Unnumbered Math Equations ................................ 5
    4.2.3 Numbered Math Equations .................................... 5
    4.2.4 Equation Array ............................................... 6
    4.2.5 Math Functions ............................................... 6
  4.3 The Tabular Environment ......................................... 7
  4.4 The Figure Environment ......................................... 7

5 Compiling a \LaTeX{} Document 7

6 Bibliographies 8
1 Introduction to \LaTeX

\LaTeX\ is a document preparation system designed for producing scholarly articles with accurate typesetting and characters. \LaTeX\ is unlike the most commonly used document preparation programs (e.g. Microsoft Word) in that the user gives \LaTeX\ a series of commands and then, using a \LaTeX\ compiler, \LaTeX\ will put those commands into a professional looking pdf or ps document. \LaTeX\ is mandatory for submitting scholarly articles to journals. Using \LaTeX\ also provides an advantage to other document preparation programs in that writing complex mathematical equations is rather straightforward and the user can avoid using an equation editor.

This tutorial will walk through the basic steps to creating a document using \LaTeX. As the use of \LaTeX\ is vast, this document will only touch on the basics. You can learn more about other \LaTeX\ commands using a traditional web search engine.

A NOTE ON INSTALLATION

As this document is intended for incoming first year Ph.D. students at Duke University, this document will not cover how to install \LaTeX\ because \LaTeX\ is already installed on all department computers. Installation information for various operating systems can be found at http://www.latex-project.org/.

2 Basic Layout of a \LaTeX\ Document

While \LaTeX\ can be used to produce many types of documents, all \LaTeX\ documents have the same basic structure. The basic layout of a \LaTeX\ document looks like this:

\begin{verbatim}
\documentclass{class}
\author{authorname}
\title{document title}
\date{the date}

% Preamble

\begin{document}

% Document Body

\end{document}
\end{verbatim}

where the % sign is used as the comment symbol and the arguments in the braces need to be specified by the user. The commands \begin{document} and \end{document} will be the same in all \LaTeX\ documents.

First, the argument class in the command \documentclass\{\} is the type of document that is being prepared. For example, if the document is a report, then class in the above layout should be replaced by the word report. Or, if the document is intended to be a scholarly article, then article should be specified as the document class. There are many standard types of document classes including report, article, letter, book, beamer (for slide shows) as well as specific document classes such as thesis (which you will use for writing your Ph.D. dissertation). The most common document class is article. There are also special American Mathematical Society (AMS) document classes such as amsart which is similar to article except basic math functions are already loaded. No matter what document class you choose to use, leave the commands \begin{document} and \end{document} as shown because this is standard code for ALL \LaTeX\ document classes.
Second, the user should specify the title, date, and author of the document. By doing so, LATEX will NOT automatically generate a title (or title page) at the head of the document. To generate a title, place the \maketitle command in the document body section of the LATEX document (after the \begin{document} and before the \end{document} statement). This will tell LATEX to create a title for you using the information supplied by the author, title, and date commands.

Third, is the preamble of the LATEX document. This is where different LATEX packages are included and new commands are set. Anything put in this section will not appear in the final document. Rather, everything in this section are commands that will help format your final document after compiling. More on this section will be included in Section 3.

Finally, is the body of the document. Anything in this part of the document will be interpreted as text to be output to the final document. For example, if you type a sentence here it will be output to the final document after compiling. Different commands for the document body are discussed in Section 4.

3 The Preamble of a LATEX Document

The preamble section of a LATEX document specifies packages, formatting, commands, etc. for use within LATEX. Anything in this section will be treated as commands and will not be output to the final document.

3.1 LATEX Packages

LATEX has many packages freely available to help with formatting, writing math equations, etc. To use a package and the commands that come with a package, include the following command in the preamble:

\usepackage{packagename}

where packagename is replaced by the package you want to include. For example, the code

\usepackage{fullpage}

will load the package fullpage for use within your document. Here are a list of the most commonly used packages and what they do.

<table>
<thead>
<tr>
<th>Package Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>amsmath</td>
<td>Functions for math equations.</td>
</tr>
<tr>
<td>amsfonts</td>
<td>Functions to do math text.</td>
</tr>
<tr>
<td>amsthm</td>
<td>Functions for writing theorems and proofs.</td>
</tr>
<tr>
<td>graphicx</td>
<td>Functions for including figures</td>
</tr>
<tr>
<td>natbib</td>
<td>Function for automatically generating a bibliography.</td>
</tr>
<tr>
<td>fullpage</td>
<td>Formats your document to the standard 1 inch margins.</td>
</tr>
<tr>
<td>setspace</td>
<td>Functions to allow double or 1.5 spacing</td>
</tr>
<tr>
<td>hyperref</td>
<td>Functions which allow active hyperlinks.</td>
</tr>
<tr>
<td>fancyhdr</td>
<td>Functions for including headers, footers, etc.</td>
</tr>
</tbody>
</table>

Recall that if you are using an AMS document class such as amsart, all ams packages are automatically loaded. You can look up details of these packages on the Internet using a search engine.

3.2 Creating Your Own Commands

In the preamble, you can create your own commands that will perform a set of commands for you. For example, because the symbol “\” is used to indicate a command in LATEX (notice all the “\” symbols before the commands) to create the symbol “\” in LATEX, you would need to type the command $\backslash$ every time you wanted to insert a “\” symbol. Rather, you can create a new command \bs by including the statement,
in your preamble. Now, every time you put the command `\bs` in the body of your document, `\LaTeX` will insert the full commands to produce the “\” symbol.

The general syntax to create your own command is:

\texttt{newcommand\{\textit{newcommandname}\}\{\textit{commands}\}}

3.3 Formatting Your Document

In the preamble you can also specify margins. Messing with the formatting in `\LaTeX` is messy and you would be better off just using the fullpage package which will set all margins to 1 inch. If you want/need to specify your own margins, we refer you to the geometry package. However, it is recommended for scholarly journal articles you leave the formatting to `\LaTeX`.

4 The Body of a `\LaTeX` Document

Everything put in the body of the `\LaTeX` document will be considered as text that needs to be output to the file when you compile the `\LaTeX` file. Within the body of the `\LaTeX` document are several environments that let `\LaTeX` know what kind of text you want to output to the final document. For example, if you want to output a math equation, you need to tell `\LaTeX` that you are in the math environment and everything within that environment will be considered as math equations. As another example, if you want to include a figure, you need to tell `\LaTeX` that you are in the figure environment so `\LaTeX` knows to output a figure to your final document.

This section will cover a few of the basic environments that you will use most often: the text environment, the math environment, the tabular environment, and the figure environment. You are left to explore the use of other environments on your own.

4.1 The Text Environment

If you are just typing text, you need to be in the text environment. By default, `\LaTeX` assumes you are in the text environment. To add text to the final document, enter text into the document body as you would in any other word processor. To create a new paragraph in `\LaTeX`, separate the paragraph in the document body by hitting “return” twice.

The text environment also has several commands that you can use to output text to your final file. For example, the command `\section\{\textit{sectionname}\}` will create a new section of the document for you. The section will also be numbered. Here is a list of a few commands that are most commonly used within the text environment and what they do:
### 4.2 The Math Environment

Anytime you are writing math equations you need to be in the math environment. For the math environment to work properly, be sure to include the ams package in your preamble. There are 4 different types of math environments for you to choose from:

1. math within text (e.g. “Let $\mu$ represent ...”),
2. unnumbered math equation - math is on its own line but not numbered,
3. numbered math equation - math is on its own line and numbered.
4. equation array.

#### 4.2.1 Math Within Text

To enter the math within text mode use the $ symbol to indicate that you are entering math within text mode and then after the math equation type another $ to indicate that you are leaving the math mode. For example the code $\mu$ generates the output $\mu$. Notice that this is within the regular text of the document and not on it’s own line.

#### 4.2.2 Unnumbered Math Equations

The second type of math environment is the unnumbered math equation environment. To enter this math environment use either $$ or \[ and type $$ or \] to exit the unnumbered math environment back to text mode. For example the command, $$\sum x_i$$ generates the output: $\sum x_i$.

Notice that this math environment is on its own line but is not numbered. Some people will tell you to use \[ and \] instead of the double dollar sign because the double dollar sign had a different use in older versions of \LaTeX\.

#### 4.2.3 Numbered Math Equations

Oftentimes, you will want to create equations with numbers that you can reference in your document. To do this, you will want to use the numbered math environment. To enter the numbered math environment, type \begin{equation} \end{equation} followed by the math you want to write then \end{equation} to exit the environment. For example, the code:

\begin{equation}
\sum x_i.
\end{equation}
\begin{equation}
    p(y) = \frac{1}{\sqrt{2\pi}}e^{-y^2/2}
\end{equation}

\label{mymathequation}
\end{equation}

 generates the following output when compiled:

\[ p(y) = \frac{1}{\sqrt{2\pi}}e^{-y^2/2}. \] (1)

Note that this math equation is numbered for you to reference within your document. The \label command is optional but if you label your math equations, \LaTeX will reference them for you using the command \eqref{ }. For example, the command \eqref{mymathequation} generates the output (1). This is particularly useful because if your math equations get renumbered, \LaTeX will fill in the appropriate number for you and you don’t have to worry about changing the equation numbers manually.

4.2.4 Equation Array

Often you will need to display math equations on multiple lines. This type of math environment is called and equation array. You enter the equation array environment using the \begin{align} and \end{align} commands. This environment is particularly helpful for proofs and homework assignments where you need to use multiple lines to show your work. As an example of how to use the equation array environment, the following code:

\begin{align}
    p(\theta | y) &= \frac{p(y|\theta)p(\theta)}{p(y)} \\
    & \propto p(y|\theta)p(\theta)
\end{align}

 generates the following output:

\[ p(\theta | y) = \frac{p(y|\theta)p(\theta)}{p(y)} \propto p(y|\theta)p(\theta) \] (2)

In the above example, notice a few things about the \begin{align} and \end{align} environment. First, \begin{align} creates two columns of math text separated by the & sign and which are aligned (note the = and the \propto signs are aligned perfectly). Second, note that lines within the equation array environment are separated by the “\" command. Third, note that \LaTeX numbers each equation. You can suppress this numbering by using \begin{align*} and \end{align*} instead which will produce the output:

\begin{align*}
    p(\theta | y) &= \frac{p(y|\theta)p(\theta)}{p(y)} \\
    & \propto p(y|\theta)p(\theta)
\end{align*}

as sequential unnumbered math equations (notice the lack of numbering).

4.2.5 Math Functions

There are TONS of math functions that can be used with any math environment for producing Greek letters, fractions, superscripts, subscripts, etc. There are many websites which give nice summaries of lots of these functions. To find out more about how to produce math output (such as Greek letters), see http://www.maths.tcd.ie/~dwilkins/LaTeXPrimer/ for a nice (but not comprehensive) summary. A comprehensive list of \LaTeX symbols including math symbols is linked on the \LaTeX section of the computing resources webpage.
4.3 The Tabular Environment

\LaTeX uses the tabular environment to create tables. The basic syntax to create a table with \( n \) columns is:

\begin{tabular}{cols}
(1,1) & (1,2) & \ldots & (1,n) \\
\end{tabular}

To see more clearly how the tabular environment works, consider the following code:

\begin{tabular}{c|c|c|c}
(1,1) & (1,2) & (1,3) & (1,4) \\
\hline
(2,1) & (2,2) & (2,3) & (2,4)
\end{tabular}

which produces the following output:

\begin{tabular}{cccc}
(1,1) & (1,2) & (1,3) & (1,4) \\
(2,1) & (2,2) & (2,3) & (2,4)
\end{tabular}

In the above code, notice first that the argument \{c|c|c|c\} after \begin{tabular} specifies that the table will have 4 columns (there are 4 letters in the argument) and each column will be centered (the “c” is for center). The \{\} separating the letters indicate that each column will be separated by a vertical bar (as shown in the output). If \{c|c|c|c\} were changed to \{l|c|r|r\} then the table would now contain 4 columns with the first column having left justification, the second being centered, and the 3\textsuperscript{rd} and 4\textsuperscript{th} being right justified.

Second, notice that a new row of a table is created by putting \(\backslash\) at the end of the row. Once the columns are set, you can create as many rows of the table as you would like. The \hline command draws a horizontal line between the rows.

Tables in \LaTeX can be created automatically in R using the R package \texttt{xtable}.

4.4 The Figure Environment

The figure environment is used to insert figures into \LaTeX. All functions discussed in this section are included in the package \texttt{graphicx} which needs to be included in the preamble of the \LaTeX document. The basic syntax for including a figure in \LaTeX is:

\begin{verbatim}
\begin{figure}
\begin{center}
\includegraphics[options]{figname}
\end{center}
\caption{Insert Caption Here}
\end{figure}
\end{verbatim}

where \begin{center} will center the figure on the page and \caption{ } will insert a caption below the figure (to have a caption above a figure move the caption statement above the \includegraphics command). See \texttt{http://www.hep.manchester.ac.uk/u/jenny/jc/docs/latex/figures.html} for more details and options about the figure environment.

A few tips: it is best to use figures which have the .png, .jpg, or .pdf file extension. You can use the .ps or .eps file extension but postscript files can sometimes cause problems with compiling your \LaTeX document using pdflatex.

5 Compiling a \LaTeX Document

Before you compile your \LaTeX document, you need to have included at least everything listed in the basic layout of a \LaTeX document. Specifically, your document needs to have ended with the \end{document}
command. Once you have your document written (or partially written), you will use your Linux/Unix terminal window to compile the document. The best way to compile your document is within the Linux/Unix terminal window type the command:

\texttt{pdflatex mydocument.tex}

which will create a pdf document within the same folder called mydocument.pdf. If you have equations which are numbered and referenced you will need to run the \texttt{pdflatex} command twice - once to initialize the labels and once to reference the labels.

An alternative way of compiling your \LaTeX{} document is through the \texttt{latex} command in the terminal window. This will first create a device independent file (.dvi) which you will need to convert into a postscript and subsequently as pdf file using commands such as \texttt{dvipdf}. However, it is recommended that you use \texttt{pdflatex} instead.

6 Bibliographies

\LaTeX{} also has built in functions which will automatically generate a bibliography in a format of your choice. These functions are included in the \texttt{natbib} package of \LaTeX{}. To generate a bibliography, you will need to create a \texttt{.bib} file which contains all the information about the articles you are citing in your paper. An example \LaTeX{} file is provided on \LaTeX{} resources webpage which contains a \texttt{.bib} file for you to reference. Or, you can look at several example of \texttt{.bib} files using an Internet search engine. You specify the bibliography style in the document body of your \LaTeX{} file using the \texttt{\bibliographystyle{}} format and you tell \LaTeX{} to include a bibliography using the \texttt{\bibliography{mybib.bib}} command in the document body. See the example for more details.

To compile a \LaTeX{} document which includes a bibliography, in your Linux/Unix shell type:

\texttt{pdflatex mydocument}
\texttt{bibtex mydocument}
\texttt{pdflatex mydocument}
\texttt{pdflatex mydocument}

The final pdf document will then contain the necessary references in a bibliography style of your choice.