Getting Started with LaTeX:
A Biostatistics Brown Bag Seminar

Nick Seewald

6 November 2015

1 What is This?

Here are some terminological fun facts that go over great at parties.

- "TeX" is a markup language (like HTML) invented by Donald Knuth.
- A lot of hullabaloo has been made over the pronunciation of "TeX". From Knuth's *TeXbook*:

  > Insiders pronounce the X of TeX as a Greek chi, not as an x, so that TeX rhymes with the word blecchhh. Its the ch sound in Scottish words like loch or German words like ach; its a Spanish j and a Russian kh. When you say it correctly to your computer, the terminal may become slightly moist.

  My thoughts? Who cares. Just don't say "tecks." It makes you seem like you don't know what you're doing.
- LaTeX is more or less an offshoot of standard TeX that builds in a lot of nice features that make life easier.

2 Installation

1. The first thing you need is a LaTeX *distribution*. If you know Python, this is kind of analogous to Anaconda, Canopy, etc.
   - *If you use Windows:* download MiKTeX from `miktex.org`.
   - *If you use a Mac:* download MacTeX from `https://tug.org/mactex/`.
   - *If you use Linux:* download TeX Live from `https://www.tug.org/texlive/`. (TeX Live is also available for Mac and Windows, but behaves slightly differently than MiK/MacTeX)

2. You can write TeX in any standard text editor. In fact, a dedicated TeX editor usually comes with your LaTeX distribution. But we're better than that.

   I like to use a cross-platform editor called *TeXstudio*. Get it at `http://texstudio.sourceforge.net/`.

   For more (including some really nitty-gritty details), see `https://en.wikibooks.org/wiki/LaTeX/Installation`) [1]

3 Basic Syntax

In a traditional word processor, what you see is what you get. With LaTeX, however, you have to tell the program what you *mean*. As a consequence, it comes with a bit of a learning curve, but once you get the hang of it, you can easily produce beautiful documents with rich math formatting and much more.

3.1 Commands

LaTeXis built on *commands*. A command can do a number of things. It can describe formatting instructions, or include a character or symbol that's not on your keyboard (like epsilon, ϵ.) Commands start with \and are followed by some text. You can provide input to commands by putting it in between braces, {}. Let's look at some examples.

LaTeXCommand	Result
`\textit{Make text Italic}`	*Make text Italic*
`\section{}`	Make a new section
`$\sqrt{4} = 2$`	$\sqrt{4} = 2$
`$\bar{X}`	$\bar{X}$

You'll notice that some of those commands are between $'s. This is to indicate *math mode*. The majority of commands in LaTeXare meant to be used in math mode, which tells the compiler to making spacing, etc. behave nicely. Let's see an example.

```
\begin{quote}
    To find estimates for $\alpha$ and $\beta$ in simple linear regression,
    we have to minimize the \textit{least-squares equation},
     \[
            L(\alpha, \beta) = \sum_{i=1}^{n} \left(Y_{i} - \alpha -
            \beta X_{i}\right).
    \]
\end{quote}
```

> To find estimates for α and β in simple linear regression, we have to minimize the *least-squares equation*,
>
> $$L(\alpha, \beta) = \sum_{i=1}^{n} \left(Y_i - \alpha - \beta X_i\right).$$

Notice that `\alpha` and `\beta` must be called in math mode. We can create "displayed" math (that appears on its own line) by surrounding the math with brackets: `\[ \]`. We don't need to include $'s here; the brackets indicate math mode for us.

If you get stuck and don't know the command to produce a particular symbol, try using DeTeXify: `detexify.kirelabs.org`.

3.2 Environments

An *environment* behaves like a command, but applies its formatting rules to a larger part of the document. Open an environment using `\begin{environmentname}` and close it using `\end{environmentname}`. Options for how the environment works/looks can be provided in brackets following the `\begin` statement. Here's an example of the `figure` environment:

```
\begin{figure}[h]
    \includegraphics[width=.4\textwidth]{biostatryan.jpg}
    \caption{I forgive you, Ryan.}
    \label{fig: ryan gosling}
\end{figure}
```

Figure 1: I forgive you, Ryan.

There are a few key components included in the code to include Figure 1. First is the `[h]` option. `h` stands for `here`, which tells LaTeXto put the figure approximately where the code is on the page. It won't always do that, though, because LaTeXcan (and will!) make decisions about where to put stuff to make things as pretty as it can. You can override this by using the `float` package, and specifying the `[H]` option (which I think means `HERE!`).

4 Making Presentations in LaTeXwith `beamer`

- The most commonly-used way to make a presentation in LaTeXis to use a `documentclass` called `beamer`. The first line of your document should be

  ```
  \documentclass{beamer}
  ```

- In the preamble, specify theme, title, author, etc. You can choose both a `theme` and a `colortheme`. Think of `theme` as a layout, and `colortheme` as a color palette. See `http://www.hartwork.org/beamer-theme-matrix/` for a grid of what every default `theme` and `colortheme` look like together.
- Slides are called `frame`s, and are environments (like itemize, etc. – open with `begin` and close with `end`.) Inside a frame, you can do most anything you can do in normal LaTeX.
- More in-depth tutorials:
  - `http://www.math-linux.com/latex-26/article/how-to-make-a-presentation-with-late`
  - `https://en.wikibooks.org/wiki/LaTeX/Presentations`

References

[1] "LaTeX." Wikibooks, The Free Textbook Project. 23 Jan 2015, 02:30 UTC. 5 Nov 2015, 23:10 `https://en.wikibooks.org/w/index.php?title=LaTeX&oldid=2757849`.