LaTeX Fundamentals: Part 1

Isabelle M. Cohen
DLab

Fall 2018

Contents

1 Welcome to LaTeX!

LaTeX is a document creation software which uses plain text with what are called "mark-up tagging conventions." We are in Section 1.

LaTeX has all kinds of useful features:

1. It right and left-justifies your text, in a way that looks much better than Microsoft Word.

   Doesn't it look good?

2. It's great for writing lengthy equations
3. It makes it easy to integrate your bibliography into your document, using what's called BibTex
4. It's easy to export statistical analysis from Stata or R directly into a convenient LaTeX format

Overall, while LaTeX might not be the easiest program to start using, it can help make your life easier, and your work look more professional.

The basic way that LaTeX works is that you write code and text together in a writing environment (like TexStudio, but it can also be simpler), then compile your code. The compiling reads the code into the text, and produces a formatted document, typically a .pdf (but we can also make a .ps or .dvi file). The text editor and the distribution (what does the compiling) are often bundled together, like in TexStudio.

In this workshop, we discuss basic set-up, lists, math mode, figures and other uses for LaTeX.

2 Basic set-up

A LaTeX document contains all kind of stuff you never see.

2.1 Preamble

In our preamble, we set our document type and call the packages we want! We can also add in custom commands or shortcuts, format page headers/footers and numbers, and do all kinds of other overall document formatting.

2.2 Title

We can also use the preamble to make our title. The default way is to specify your title, author and date information in the preamble, then use the maketitle command in the body of the document.

2.2.1 So cool

3 Lists

There are a lot of types of list in LaTeX! Lists are not only useful, but they'll also help introduce us to some of the principles of formatting text in LaTeX.

To make a list, we set what's called an *environment*. That means we need to begin the environment (which requires calling it by name), fill it with content, then end it (again by name). The list below is made with the enumerate environment; the one in Section 1 is made with the itemize environment.

A. We can do simple bullet points with the itemize environment

B. We can do numbers with the enumerate default environment

C. By using the enumitem package, we also get more exciting options, like the alphabet, which uses the phrase

```
[label=\alph*.]
```

after we begin the environment

D. We could even use roman numerals with the roman option.

   I That

   II looks

   III good!

E. We can choose whether or not to specify periods, parenthesis, etc.; for example, with

```
[label=(\roman*)]
```

   (i) Then

   (ii) we

   (iii) would

   (iv) get

   (v) this

- Lists
- can
- also

- look
- like
- this

4 Math Mode

Math mode is where LaTeX really shines. Math mode lets you add mathematical symbols and equations in your document. Let's say you want to write an equation about y, or y, and f(x), or $f(x)$.

We might want the equation on its own line. One way to do this is to use a slash, \, and a bracket, [, one at the beginning, and one at the end.

$$y = f(x)$$

Another way is to use the align environment, which gives our equation a number.

$$y = f(x) \tag{1}$$

We can also turn off the equation number by using align* instead.

$$y = f(x)$$

All three equations look almost exactly the same. No matter how you enter math mode, the equations look much the same. There are various other environments that also let you use math mode in its own paragraph, including $$ (not very popular), the displaymath environment, the equation environment, and the gather (or gather*) environment.

Notice how it isn't very easy to put a \ in your text. Here's what happens when we try to do it:

Nothing happens! Instead, we actually used math mode in the line, with the word backslash. This is the main way we write mathematical symbols, whether in or our of math mode. For example, to get the Greek alphabet:

$$\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \eta, \theta, \iota, \kappa, \lambda, \mu, \nu, \xi, \pi, \rho, \sigma, \tau, \upsilon, \phi, \chi, \psi, \omega$$

$$\lambda, \Lambda$$

which is just like writing alpha, beta, gamma, delta, epsilon, zeta, eta, theta, etc. with \ before them inside math mode.

LaTeX also allows us to write more complicated math, for example including square roots (sqrt) and superscripts (ˆ).

$$y = \sqrt{x+1} + x + x^2$$

We can also use subscripts, parenthesis, brackets, square brackets, and even fractions.

$$y = \frac{(x^3 + x^2)}{x + 4}$$

We can also write simple integrals:

$$\int_0^2 x^2 - 1dx$$

or derivatives of trigonometric functions:

$$\frac{d}{dx}(\sinh(u)) = \cosh(u)\frac{du}{dx}$$

Let's dig into subscripts and superscripts. We can write a squared variable as easily as x^2, or use a subscript the same way with x_2. For more complex subscripts, we would just place brackets around the content of the superscript (x^{2y}) or subscript (x_{ijk}). We can even include both: x^{2y}_{ijk}

It's also worth discussing parenthesis, brackets and curly brackets further.These may seem simple, but they can make a huge difference in the readability of your equation.

Without customization, you may find your parenthesis the wrong size:

$$(\frac{2x}{y})$$

This is pretty easy to fix by putting a \ left before the parenthesis and a \ right after:

$$\left(\frac{2x}{y}\right)$$

The left and right commands will automatically resize parenthesis or brackets no matter how large:

$$\left[\frac{\left(\frac{2x}{y}\right)}{z}\right]$$

To add in curly brackets, just use \ before the brackets, even without resizing.

$$\left[\frac{\left(\frac{2\{x+1\}}{y}\right)}{z}\right]$$

To make them scalable, we just use \ left \ bracket to start, and \ right \ bracket to close.

$$\left\{\frac{2x}{y}\right\}$$

If we want to write more complex sets of equations, we could use the align package. We can use \\ to mark the end of the line, and & to indicate where the equals signs should line up.

$$\begin{align}\frac{\partial x_i(\mathbf{p},w)}{\partial p_j} &= \frac{\partial h_i(\mathbf{p},u)}{\partial p_j} - \frac{\partial x_i(\mathbf{p},w)}{\partial w} x_j(\mathbf{p},w) \tag{2}\\ \epsilon_{p,ij} &= \epsilon^h_{p,ij} - \epsilon_{w,i} b_j \tag{3}\end{align}$$

Equation 2 is above!

Finally, what about matrices? For those, we like to use the array environment inside math mode, with scalable brackets:

$$\left[\begin{array}{ccc} 1 & 2 & \\ 3 & 4 & 5 \end{array}\right]$$

This array also gives us a little preview of Workshop 2; this structure closely resembles how LaTeXsets up tables.

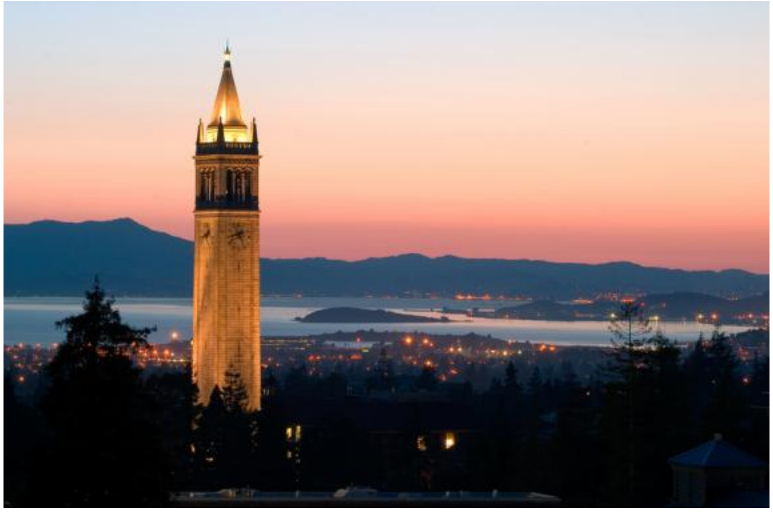
5 Figures

LaTeX has a lot of different ways to insert and customize figures. We'll focus on the graphicx package.

The most basic use is just the command includegraphics, where we specify the file path (from our current folder) and the image name after the command. For this to work, the campanile image should be in the same folder as our LaTeX document.

In the command, we specified a width for the image - specifically, textwidth, which is roughly the width of the text. If we wanted it to be half as wide as the text, we could specify .5 textwidth, as below.

Alternatively, we can fix the width or height to a certain size (1 cm, 2 cm, etc.) or adjust the scale of the image; instead of width, we type scale, and specify something relative to the original size of the image. We can also use the center environment again.



The equivalent to the table environment is the figure environment. This will let us include a caption, fix the location of the figure, etc., and to use the centering option. We do this in Figure 1 below.

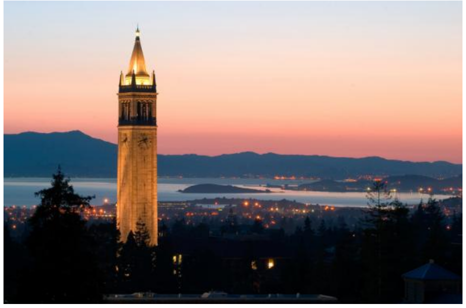


Figure 1: Berkeley at sunset

If we want to include two images next to each other, we just include them in the same figure (with the width slightly less than half the text width).

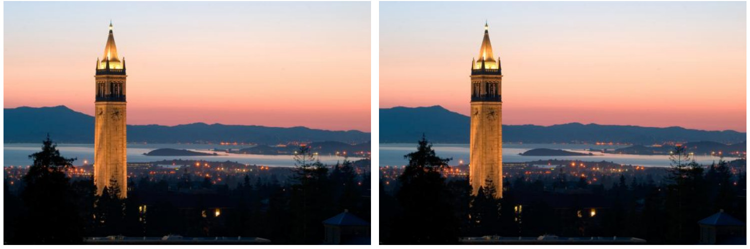
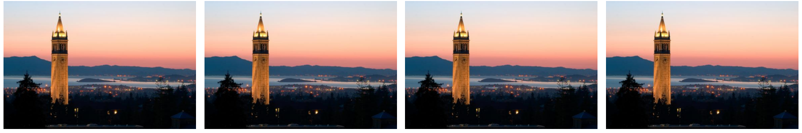


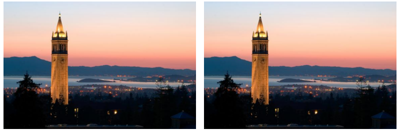
Figure 2: Berkeley at sunset - twice

Let's make things a little more complicated. Let's say we want to include two images in the same figure, but give them separate captions. There are a few ways to do that; one is below:



(a) Berkeley 1 (b) Berkeley 2 (c) Berkeley 1 (d) Berkeley 2

Figure 3: Berkeley at sunset - twice



(e) Berkeley 1 (f) Berkeley 2

In this figure, we have three separate references: one to the overall figure (Figure 3), one to the first panel (Figure 3c) and one to the second panel (Figure 3d). Notice also that instead of specifying the width relative to the text, we gave an explicit width in inches.

6 Other uses for LaTeX

LaTeX isn't actually the only use for its language. It's popularity has generated all kinds of other places it can be used, include a plug-in for gmail. You can even switch Word's input math mode into LaTeX math. In my word documents, when I type LaTeX math symbols into my equations, I actually get the symbols. This is a great, although it doesn't fix how slow Word gets when you include multiple equations.

In the next workshop, we'll talk about other features and uses of LaTeX including tables, footnotes and references/bibliographies, automating exporting from Stata/R to LaTeX and making presentation with LaTeX.

7 Resources for further study

There are a ton of "getting started in LaTeX" resources:

- Getting to Grips with LaTeX
- The Not So Short Introduction to LaTeX 2ϵ
- CTAN's Starting Out in LaTeX Guide

A Useful packages

There are a number of useful packages in LaTeX. Here are the packages we use in the Fundamentals workshop series:

- soul: highlighting
- appendix with

  $$toc, page$$

  : lets us make an appendix
- enumitem: this package lets us customize list types
- amsmath: this package will let us use the align environment, and generally make our equations prettier
- hyperref: lets you embed hyperlinks into your document
- booktabs: one of the best packages for table formatting
- array: lets us create fixed width cells
- graphicx: a great package for adding figueres
- caption: for making subcaptions in figures
- subcaption: for making subcaptions in figures
- dcolumn: required for stargazer
- natbib: for making our citations nice

We can't cover all of them today, so here are a few others that you might be interested in:

- tikzpicture: lets you draw lines and geometric shapes within a TeX document. See this link
- fancyhdr: lets you add custom headers and footers into your LaTeX document. See this link
- pdfpages: lets you include pdfs into your TeX document. See this link
- tabularx: lets you create tables that are the same width as the text. There are subtle differences between how tabular, tabular*, and tabularx handle spacing within tables; this stackexchange has some helpful examples.
- xspace: creates spaces after commands (like LaTeX); you can use xspace in creating commands to insert that extra space
- bibtex and biblatex: both are alternatives to natbib which contain other options for formatting. This page on bibliography management in LaTeX is a great resource.

- geometry: lets you customize header, footer and margin width in your document. See this link

B Useful symbols

All predefined mathematical symbols in LaTeX are available here. An incredibly comprehensive list of 2,590 symbols with the necessary packages and commands is available here. The second link also discuss amsmath extensively.

C Templates

Overleaf has a great database of LaTeX templates.