LaTeX tutorial

I. Holmes

Department of Bioengineering
University of California, Berkeley

November 9, 2016

Outline

Key

1. Exercises in white on red
2. `Shell commands in green type on black`
3. URLs in blue: http://tinyurl.com/texroll
4. Source code in boxes

```
This is some LaTeX source code
```

Try the URL now!

LaTeX advocacy

1. It's free, portable, open source & extensible
2. Source files are plain text, revision control easier
3. Typesetting is *much* better, especially math
4. Style changes are easier
5. Easy to integrate with programmatic workflows
6. **Separation of form and content**

LaTeX criticism

1. Possibly the worst programming language ever
2. Syntax is horrible
3. Compilation from source is almost impossible
4. Mostly trial and error, unless you're a guru
5. Some things you just can't do (unless, guru)
6. **Will mark you forever as a nerd pariah**

Text editor

Before you start, your life will be much easier with a text editor that has LaTeX syntax-coloring, such as `emacs` or `vim`.

You could also use a specialized LaTeX editor that gives you previews, such as TeXmaker, or even a WYSIWYG LaTeX editor such as LyX (free) or Texpad (OSX, $$$). But you need to understand the LaTeX underneath. So...

helloworld.tex


```
\documentclass{article}
\title{Marmosets Are Great}
\author{Ian Holmes}
\begin{document}
\maketitle
\abstract{A short treatise on marmosets.}
\section{Introduction}
Marmosets ({\em Callitrichidae})
are {\bf New World Monkeys}.
\end{document}
```

Compile with `pdflatex helloworld.tex`

Try this.

http://tinyurl.com/texhello

documentclass options

- `\documentclass[10pt]{article}`
- `\documentclass[twocolumn]{article}`
- `\documentclass[landscape]{article}`
- `\documentclass{letter}`
- `\documentclass{book}`
- `\documentclass{beamer}` — presentation

Further classes can be defined using a *class file*.
For example, the journal *Bioinformatics* provides a class file `bioinfo.cls` invoked with `\documentclass{bioinfo}`.

Section references

Can use \label and \ref as follows:

```
\section{Introduction}
\label{intro}
Marmosets are New World Monkeys.

\section{Geography}
Marmosets are found in the New World,
as mentioned in Section~\ref{intro}.
```

Note tilde ~ between Section and \ref: prevents linebreak.

Add a section or two, and recompile.

Makefiles

If you change the section numbers, you will have to re-run `pdflatex`. Consequently, it's common to run the program twice. Can do this with a `Makefile`:

```
helloworld.pdf: helloworld.tex
        pdflatex helloworld.tex
        pdflatex helloworld.tex
```

General form of Makefile stanza:

```
TARGET: DEPENDENCIES
  <TAB> COMMANDS
```

Make command-line usage

- General: `make helloworld.pdf`
- Force rebuild: `make -B helloworld.pdf`
- Dry run: `make -n helloworld.pdf`

By default, `make` just builds first target in Makefile.

Makefiles and replicability

Titus Brown's checklist for paper replicability:

- a link to the paper itself, in preprint form, stored at arXiv;
- a tutorial for running the software on a Linux machine hosted in the Amazon cloud;
- a git repository for the software itself (hosted on github);
- a git repository for the LaTeX paper and analysis scripts, including an ipython notebook for generating the figures;
- instructions on how to start up an EC2 cloud instance, install the software and paper pipeline, and build most of the analyses and all of the figures from scratch;
- the data necessary to run the pipeline;
- some of the output data discussed in the paper.

http://ivory.idyll.org/blog/replication-i.html

Pseudotargets, pattern rules and variables


```
MAIN = helloworld

all: $(MAIN).pdf

%.pdf: %.tex
        pdflatex $<
        pdflatex $<
        open $@

clean:
        rm *.toc *.log *.out *.pdf *.aux *~
```

Try this.

Use `make`, `make -n` and `make -B`.
If in doubt: `make clean`

Loading other files

LaTeX files can include other files via the `\input` command. This is particularly useful with Makefiles, because you can generate data-driven parts of your article automatically, and combine them with manually-written sections.

The `\include` command is like `\input` but does some extra book-keeping (such as adding a page break). Useful for e.g. separating a thesis into chapter files.

Comments, escapes, styling


```
% Comments
Actual percent sign: 100\%
Other escapes: \_, \&
Tilde escape: \~{}

``Pretty quotation marks''

Empty line signals new paragraph.

Space: \quad Explicit line \\ break
```

Can you get **bold**, *italic* & `typewriter` fonts?

Google these typefaces.

Typefaces

`{\bf Bold}`	**Bold**
`{\em Italic}`	*Italic*
`{\tt Typewriter}`	`Typewriter`

Page numbering

These go in the preamble:

- `\pagenumbering{arabic}` — default
- `\pagenumbering{roman}`
- `\pagenumbering{Roman}`
- `\pagenumbering{alph}`
- `\pagenumbering{Alph}`

To suppress page numbers altogether, use `\pagestyle{empty}`.

To add a table of contents: `\tableofcontents`

Lists


```
List of books about wizard school
\begin{itemize}
\item Earthsea
\item Harry Potter
\item The Magicians
\item The Once and Future King
\end{itemize}
```

Try this.

Also try `enumerate` instead of `itemize`, and try nesting lists inside other lists.

How many levels deep can you nest?

Tables

Right-justified	Centered	Left-justified
School vouchers	Science	Public education
Defense spending	Trade deals	Aid programs

```
\begin{tabular}{rcl}
Right-justified & Centered & Left-justified \\
\hline
School vouchers & Science & Public education \\
Defense spending & Trade deals & Aid programs
\end{tabular}
```

Time to add a table.

Make a table with some facts about marmosets. Or pick another vertebrate from hgdownload.cse.ucsc.edu and make a table about it.

Table captions and references


```
\begin{table}
\begin{tabular}
...
\end{tabular}
\caption{
 \label{MarmosetFacts}
 A table of marmoset facts.
}
\end{table}

For marmoset data, see Table~\ref{MarmosetFacts}.
```

Equations


```
Inline: $a = 3$, $b = 5$

Non-numbered:
\[
y = ax + b
\]
```

Inline: $a = 3$, $b = 5$

Non-numbered:

$$y = ax + b$$

Equations


```
Numbered (Equation~\ref{Gaussian}):
\begin{equation}
x \sim {\cal N}(\mu,\sigma):
\quad
P(x' \leq x < x' + dx') =
\frac{1}{\sqrt{2 \pi \sigma^2}}
e^{-\frac{(x'-\mu)^2}{2\sigma^2}} dx'
\label{Gaussian}
\end{equation}
```

Numbered (Equation 1):

$$x \sim \mathcal{N}(\mu, \sigma): \quad P(x' \leq x < x' + dx') = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x'-\mu)^2}{2\sigma^2}} dx' \tag{1}$$

Equations


```
Numbered (Equation~\ref{Gaussian}):
\begin{equation}
x \sim {\cal N}(\mu,\sigma):
\quad
P(x' \leq x < x' + dx') =
\frac{1}{\sqrt{2 \pi \sigma^2}}
e^{-\frac{(x'-\mu)^2}{2\sigma^2}} dx'
\label{Gaussian}
\end{equation}
```

Google "Latex math symbols".

Write out another distribution e.g. Poisson.

Brackets, arrays


```
\[
\left(
 \begin{array}{c}
  n \\
  k
 \end{array}
 \right)
= \frac{n \times (n-1) \ldots \times (n-k)}{k!}
= \frac{n!}{k!(n-k)!}
\]
```

$$\left(\begin{array}{c} n \\ k \end{array} \right) = \frac{n \times (n-1) \ldots \times (n-k)}{k!} = \frac{n!}{k!(n-k)!}$$

Macro commands


```
\newcommand\binomial[2]{
\left(
 \begin{array}{c}
  #1 \\
  #2
 \end{array}
 \right)
}
\[
 \binomial{5}{2} = (5 \times 4 \times 3) / 2 = 30
\]
```

$$\binom{5}{2} = (5 \times 4 \times 3)/2 = 30$$

Try this.

More arrays; text in math environments


```
\[
H(x) = \left\{
 \begin{array}{ll}
  0 & \mbox{for $x < 0$} \\
  1 & \mbox{for $x \geq 0$}
 \end{array}
\right.
\]
```

$$H(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$$

Equation arrays


```
\begin{eqnarray}
x_1 & = & 1 \\
x_2 & = & 1 \\
x_{n+2} & = & x_n + x_{n+1}
\end{eqnarray}
```

$$x_1 = 1 \tag{2}$$
$$x_2 = 1 \tag{3}$$
$$x_{n+2} = x_n + x_{n+1} \tag{4}$$

Equation arrays (cleaner numbering)


```
\begin{eqnarray}
x_1 & = & 1 \nonumber \\
x_2 & = & 1 \nonumber \\
x_{n+2} & = & x_n + x_{n+1}
\end{eqnarray}
```

$$\begin{aligned} x_1 &= 1 \\ x_2 &= 1 \\ x_{n+2} &= x_n + x_{n+1} \end{aligned} \tag{5}$$

Equation arrays (no numbering)


```
\begin{eqnarray*}
x_1 & = & 1 \\
x_2 & = & 1 \\
x_{n+2} & = & x_n + x_{n+1}
\end{eqnarray*}
```

$$
\begin{aligned}
x_1 &= 1 \\
x_2 &= 1 \\
x_{n+2} &= x_n + x_{n+1}
\end{aligned}
$$

Interlude

Do the following:

1. Download a set of predicted gene annotations from UCSC for your vertebrate of choice. (I used the Augustus gene predictions for marmoset.)
2. Also download the description of that table. Find out which column in the table has the number of exons for each gene.
3. Using `perl`, `python`, `sed`, `cut`, or another such tool, extract the number of exons as a column of numbers.
4. Plot the frequency distribution in R (or otherwise).
5. Export as a PDF file, `exonFreqs.pdf`

Example R script


```
require("ggplot2")
exons <- read.csv("numExons.txt")
plot <- qplot(exons,geom="histogram",binwidth=1,
       main="Marmoset exon count distribution") +
       scale_y_continuous() +
       scale_x_continuous(limits=c(0,100)) +
       xlab("Number of exons")
ggsave ("exonFreqs.pdf", plot)
```

Save as `plot.R`

Example Makefile


```
PREFIX := hgdownload.cse.ucsc.edu/goldenPath
SPECIES := calJac3
URL := http://$(PREFIX)/$(SPECIES)/database

augustusGene.sql augustusGene.txt.gz:
        curl -O $(URL)/$@

%.txt: %.txt.gz
        gunzip --keep $<

numExons.txt: augustusGene.txt
        cat $< | cut -f 9 >$@

exonFreqs.pdf: numExons.txt plot.R
        R -f plot.R
```

Figure


```
\includegraphics{numExons.pdf}
```

Of course, you can get more elaborate...

Figure, with caption


```
\begin{figure}
\includegraphics[width=\textwidth]{numExons.pdf}
\caption{
  \label{ExonDistribution}
  Distribution of exon frequencies in marmosets.
}
\end{figure}
```

BibTeX


```
Marmosets are highly social \cite{Marx2016}.
...
\bibliographystyle{natbib}  % or plain, unsrt, ...
\bibliography{references}
```

Implies the existence of a file `references.bib`

```
@Article{Marx2016,
 Author="Marx, V.",
 Title="{N}eurobiology: learning from marmosets",
 Journal="Nat. Methods",
 Year="2016",
 Volume="13",
 Number="11"
}
```

Running BibTeX

Typically you need to run `pdflatex`, then `bibtex`, then `pdflatex` again *twice* to ensure all numbering is correct.

In your `Makefile`:

```
%.pdf: %.tex references.bib
	pdflatex $<
	bibtex $<
	pdflatex $<
	pdflatex $<
```

Yes: this is really messed-up

TeXMed

http://www.bioinformatics.org/texmed/
BibTeX wrapper for PubMed.

Try adding a reference for your vertebrate of choice.

Commands

Command	Purpose
\hspace	Fill horizontal space
\fbox	Box with frame
\parbox	Box with line breaks
\newcounter	Create a new counter
\stepcounter	Increment counter
\color	Change text color
\colorbox	Change background color

See e.g. https://en.wikibooks.org/wiki/LaTeX

Packages

Loaded with `\usepackage`, e.g. `\usepackage{amsmath}`

Package	Purpose
`algorithm2e`	Writing out algorithms
`beamer`	Presentations (like this one)
`amsmath`	Better math formatting
`geometry`	Page formatting (e.g. margins)
`biblatex`	Better bibliographies
`chemfig`	Chemical structures

Try a few of these out...

CTAN (`ctan.org`): Comprehensive TeX Archive Network

These slides at `https://github.com/ihh/latex-tutorial`

Homework

Make a short report on a vertebrate in the UCSC genome database that is *not* a marmoset. Include:

1. Title, author, abstract
2. Two-column layout
3. Introduction, Results, References sections
4. A figure showing the distribution of exon lengths (or other data from UCSC)
5. A mathematical formula (e.g. a fit to the histogram)
6. At least one table
7. At least one reference