

A Short Manual of L^AT_EX

Haohan Chen *

August 13, 2015

In this short manual, I will summarize all L^AT_EX commands that we practiced in our workshop. Hope it serves as a quick reference when you start working on it.

1 To Write an Article: Everything you need to get started...

The following code should cover most of the functions we need from L^AT_EX. Feel free to copy from the following template before you are fully familiar with it.

<pre>\documentclass[11pt, letter]{article}</pre>	Define document class as 'article' and set the size of the text and type of the paper
<pre>\usepackage{graphicx}</pre>	You need it if you include figures.
<pre>\usepackage{natbib}</pre>	This package creates nice citations and bibliography for you
<pre>\usepackage{amsmath}</pre>	You need it if you type maths.
<pre>\usepackage{hyperref}</pre>	With this package, your table of contents, list of figures, list of tables, citations, reference to tables (using <code>\ref</code>) will have hyperlinks directing readers to those contents respectively.
<pre>\usepackage[margin=2.5cm]{geometry}</pre>	You may set your page margin here. If you want to further customize, type, for instance, <code>[left=2cm, right=2cm, top=2.5cm, bottom=2.5cm]</code> in place of <code>[margin = 2.5cm]</code>
<pre>\usepackage{times}</pre>	Use <i>Times</i> font.
<pre>\linespread{1.5}</pre>	Spacing between lines
<pre>\author{Your Name \thanks{Duke} }</pre>	Print author. Anything after <code>\thanks</code> appear as footnote on the page where you print your title.
<pre>\title{Your Title}</pre>	The title of your article
<pre>\date{}</pre>	If you do NOT want date printed, put nothing between <code>{}</code> . If you would like to print the date of today, put <code>\today</code> between <code>{}</code> , or simply do not include this command.

*haohan.chen@duke.edu

<code>\begin{document}</code>	This marks the start of your document. Anything before this sets up the document, while everything between this and <code>\end{document}</code> will appear in the final product.
<code>\maketitle</code>	Tell L ^A T _E X to print the title, author and date here.
<code>\newpage</code>	Start the following in a new page. You can put it anywhere you wish to break the page in your document.
<code>\tableofcontents</code>	Make a table of content
<code>\listoffigures</code>	Make a list of figures
<code>\listoftables</code>	Make a list of tables
<code>\section{...}</code>	Start a new section. Use <code>\section*{}</code> if you do not want the section to be numbered. The sign '*' serves the same function for many others command in L ^A T _E X.
<code>\subsection{...}</code>	
<code>\paragraph{}</code>	Start a paragraph. I use this when I need to add a little title for a paragraph. Otherwise I will simply start typing the content right away. Note that a blank line starts a new paragraph.
<code>\begin{itemize}</code> <code>\item YOUR TEXT \\\</code> <code>\item YOUR TEXT \\\</code> <code>...</code> <code>\end{itemize}</code>	Write bullet points The first item... Make sure to leave a space between <code>\item</code> and your text so that L ^A T _E X can recognize the start of each point. Also don't forget the end-of-paragraph sign <code>\\</code> before you start your next point. Note: If you want your bullet points number, replace <code>itemize</code> with <code>enumerate</code> , at the beginning and the end of the command. If you want a nested bullet point, you can put <code>\begin{itemize}...\end{itemize}</code> inside a bullet pint environment.
<code>\begin{figure}[hptp!]</code>	FIGURE. Adding a figure into your document. What's inside [...] tells L ^A T _E X where to place the figure. h : here; p : in a separate page with other figures/ tables; t : top; b : bottom; ! : tells L ^A T _E X to try HARD to over the default setting (i.e. <code>tbp</code>) with your preferred setting indicated here. Try not to use description such as 'the figure above' because L ^A T _E X is very likely to rearrange the placement of your figures as you edit the document.
<code>\centering</code>	Put the figure in the center
<code>\caption{Title}</code>	The title of your figure

```
\label{fig1}
```

```
\includegraphics[width=4in]{"name"}
```

```
\end{figure}
```

Create a reference point so that you can refer to the figure in your text using `\ref{}`. Here I set the reference name as `fig1`. You may use whatever name you want, as long as one figure has a unique reference name.

Set up the figure. You can set the width and height of the figure: `[width=4in, height=4in]`. If you include only one of them (as I do on the left), the figure will be re-sized proportionally. You use `{"name"}` to tell L^AT_EX the name of the image file, which you should put in the same folder as your `.tex` file. Caution: Use simple names for the source files of figures. Otherwise there might be strange errors.

```
\begin{table}[hptb!]
  \centering
  \caption{Your Title}
  \label{tab1}
  \begin{tabular}{|l|c|r|}
    \hline \hline
    11 & 12 & 13 \\
    21 & 22 & 23 \\
    31 & 32 & 33 \\
    \hline
    \multicolumn{2}{|l|}{41-2} & 43 \\
    \hline
    41 & \multicolumn{2}{|r|}{42-3} \\
    \hline
    \multicolumn{3}{|c|}{51-3} \\
    \hline \hline
  \end{tabular}
\end{table}
```

The code on the left gives you a table that appears as the following:

Table 1: Your Title

11	12	13
21	22	23
31	32	33
41-2		43
41	42-3	
51-3		

Some highlight:

- At the first line, `'|'` Tells L^AT_EX whether to add border between two columns or not. You can see border between the first and the second column but no border between the second and the third column here. `l,c,r` tells L^AT_EX to align all cell of a column to the left, center or right respectively.
- `'&'` moves from one cell from another cell within one row
- `'\\'` moves from one row to next row
- To add border between two rows, add `\hline` in between (after `\\` of the previous row)
- To merge columns, use `\multicolumn`. For instance `\multicolumn{3}{c}{51-3}` means: merging 3 cells, aligning to the center in this merged cell, the text in the cell being "51-3". Again, `'|'` adds borders

Play with the settings and you will see how it works!

...see Figure `\ref{fig1}`

...see Table `\ref{tab1}`

...we have the equation `$a+b=c$`...

```
\begin{equation}
(a + b) (c + d) = e
\end{equation}
```

```
\begin{align}
y = x^2 &+ 3x + 4 \\
&= a + b \\
&= 10
\end{align}
```

```
\begin{gather}
(x + 3)(y - 1) = 0 \\
xy - x + 3y - 3 = 0 \\
xy - x + 3y = 3
\end{gather}
```

Referring to tables or figures in your text. You may refer to the figure that you label. Note that the label of the figure does not have to be defined before you call it with `\ref`. Personally I think using the reference system (i.e. `\label` and `\ref`) instead of directly typing in the number of the table/ figures help managing your work more efficiently.

Same as above.

MATHS. When you type math in your main text, DON'T FORGET to use '\$' to start AND end your equations/ formulas! Remember to check that when you get error reports.

You need this when you enter a single equation. Using `{equation*}` get rid of the numbers of after equations.

You need this when you enter a group of equations and want them aligned to certain place. Here they align to the equal sign.

You need this when you enter a group of equations and want them aligned to the center.

CITATION AND BIBLIOGRAPHY. Before you start to cite in your document, please do the following:

1. Create a NEW file in L^AT_EX to store your bibliography information. Save it into the same folder as your `.tex` file. You may give it whatever name as you wish, but the extension should be `.bib`. For instance, in my example, The name is `ref.bib`.
2. Add bibliography information to the `.bib` file you have created and saved. I strongly recommend using Google Scholar to collect such information (but make sure to check for occasional mistakes). How to do it? Under your search result, click "cite". Next, at the bottom left of the pop-up window, click "Import to BibTeX". Then, select-all and copy the information to your `.bib` file. Last, you should save it, before which it will not appear in the document that cites it.
3. Cite in the document using the command on the left. Note that what you put in `{. .}` should be the *reference name* of the entry. Here is an example:

```
@article{acemoglu2006facto,
  title={De facto political power and institutional...},
  author={Acemoglu, Daron and Robinson, James A},
  journal={The American economic review},
  pages={325--330},
  year={2006},
  publisher={JSTOR}
}
```

Here the reference name of this work is 'acemoglu2006facto'. That is, to cite this work, you should enter, for instance, `\citet{acemoglu2006facto}`.

```
\citet{...}
\citeauthor{...}
\citep{...}
\citep[postfix] [prefix]{keylist}
```

```
\bibliographystyle{chicago}
```

```
\bibliography{ref}
```

```
\end{document}
```

At the end of your document, you may produce your bibliography. This line first sets the reference style as 'Chicago'. This command link your document with a bibliography file named 'ref.bib', which, as is explained above, you have created in the same folder as your `.tex` file.

End the document. Without this at the end, L^AT_EX will not compile. Sometimes you may accidentally enter stuff after `\end{document}`, in which case no error will be reported but you lose all information after this point. Be careful!

2 To Write a Presentation

```

\documentclass{beamer}
\usetheme{default}

\title{Title of the Presentation}
\subtitle{Subtitle}
\author{Your Name}
\institute{Duke University}
\date{\today}

\begin{document}

\maketitle

\section{Name of 1st Section}

\begin{frame}
  \frametitle{Name of 1st frame}
  \framesubtitle{Subtitle of 1st frame}
  \begin{itemize}
    \item First argument
    \pause
    \item Second argument
    \pause
    \item Third argument
    \pause
  \end{itemize}

  \begin{block}{This is a block}
    Additional information
    Additional information
  \end{block}
\end{frame}

\section{Name of 2nd Section}

\begin{frame}
  \begin{figure}
    \caption{...}
    \label{fig}
    \includegraphics[width = 4in]{...}
  \end{figure}
\end{frame}

\end{document}

```

If you pay close attention to the previous part, the code on the left is very likely to be easily understandable. Below are several highlights:

- `beamer` is the document class that is most commonly used for presentations. If you use `\pause` in your pages and you would like to generate handouts, simply substitute `\documentclass{beamer}` by `\documentclass[handout]{beamer}`.
- Available themes include (to be put after `\usetheme`, which is `default` in my example): *default*, *Antibes*, *Bergen*, *Berkeley*, *Berlin*, *Copenhagen*, *Darmstadt*, *Dresden*, *Frankfurt*, *Goettingen*, *Hannover*, *Ilmenau*, *JuanLesPins*, *Luebeck*, *Madrid*, *Malmoe*, *Marburg*, *Montpellier*, *PaloAlto*, *Pittsburgh*, *Rochester*, *Singapore*, *Szeged*, *Warsaw*, *boxes*. You can of course search and download other themes from external sources.
- Names of sections defined by `\section{}` will appear in the navigation bar (only applicable when using themes that show navigation bar, for instance *Warsaw*).
- Things between `\begin{frame}` and `\end{frame}` makes one slide.
- `\pause` tells L^AT_EX to put the content afterwards to a following page.
- You may insert figures or tables as what you do in an article (as I have demonstrated in the second frame of this example).

3 Knitr

Motivation Knitr is a handy tool for reproducible research. It allows you to mix your R code and output with your report or paper written in L^AT_EX. If you are following the recent LaCour (2014) scandal in our field, you may want to revisit this famous report that uncovers the fraud (Brookman et al. 2015, accessible from: http://stanford.edu/~dbroock/broockman_kalla_aronow_lg_irregularities.pdf). It is quite obvious that the report is generated by Knitr, which allows the authors demonstrate to the readers how they detect the fraud from the data. For our prospective study, a considerable number of methods courses require writing homework with L^AT_EX.

Editors RStudio is a good editor for knitr. However, its disadvantage compared to TeXstudio is that it does not automatically fill your T_EX command.

Setup I: Install the “knitr” package You may type in the console `install.packages(“knitr”)` and press “ENTER”. Or you may use the menu “Tools” → “install packages” and type “knitr” in the pop-up window.

Setup II: Change setting Enter the Preference Setting of RStudio (“Global Options” for Windows users, “Preferences” for Mac users). Go to the “Sweave” tab at the left panel. Choose “Weave Rnw files using knitr”.

Create a knitr doc To create, press “New File” → “R Sweave”. You will get a new windows with a file with the extension `.Rnw`.

Adding R code chunks In a knitr document, you indicate the start of a R code chunk with “`<<>=`” and its end with “`@`”. A sample knitr document including simple R code that print the value of a variable and plot a scatter plot is shown as below:

```
\documentclass{article}

\begin{document}
Below I calculate and print the value of  $a = 1 + 1$ 

<<>=
a = 1 + 1
print(a)
@

Below I draw a scatter plot of five points  $(1, 1), (2, 2), (3, 3), (4, 4)$ 

<<>=
plot(x = 1:4, y = 1:4)
@

\end{document}
```

Options You may specify some options for your R chunks: name of the chunk, show/ hide the code (`echo=`), how to output the result (`results=`), whether to run the code (`eval=`), the height and width of a graph (for instance, `fig.height = 4`, `fig.width= 4`). Examine how the output of the following chunks differ when I change the options.

```
<<>>=
a = 1 + 1
print(a)
plot(x = 1:4, y = 1:4)
@

<<eval=FALSE>>=
a = 1 + 1
print(a)
plot(x = 1:4, y = 1:4)
@

<<echo=FALSE>>=
a = 1 + 1
print(a)
plot(x = 1:4, y = 1:4)
@

<<fig.height=3, fig.width=4>>=
a = 1 + 1
print(a)
plot(x = 1:4, y = 1:4)
@
```

Above is a demonstration of frequently-used options, for more, visit the website of the package author: http://yihui.name/knitr/options/#chunk_options.

4 LyX

LyX is a user-friendly T_EXword processing software. If you would like to write on L^AT_EX without having to worry too much about its syntax, this is what you may try. LyX is downloadable from <http://www.lyx.org/> and a considerable number of tutorials are online.

5 Resources

You may find the following resources useful when you work on L^AT_EX:

- To have a better understanding on the 'big picture' of L^AT_EX, you may use one of the following textbooks:
 - *The No So Short Introduction to L^AT_EX 2_ε* by Tobias Oetiker et al. Downloadable from <http://tobi.oetiker.ch/lshort/lshort.pdf>
 - *The Art of LaTeX* by Helin Gai. Downloadable from http://www.math.ecnu.edu.cn/~latex/docs/Eng_doc/LaTeX_Manual_8_6.pdf
 - If you would like to know more about how to type maths in L^AT_EX, *The L^AT_EX Mathematics Companion* by Helin Gai is a good place to start. Downloadable from <http://hungrydummy.com/static/pdf/MathCompanion.pdf>
- You may heavily draw on the following online resources as well:
 - The L^AT_EX page on Wikibook: <http://en.wikibooks.org/wiki/LaTeX>. Especially, its 'mathematics' page covers nearly all we need in our first-year methods class: <http://en.wikibooks.org/wiki/LaTeX/Mathematics>
 - Answers to specific questions on the forum [stackoverflow](http://stackoverflow.com/) (<http://stackoverflow.com/>) and [StackExchange](http://tex.stackexchange.com/) (<http://tex.stackexchange.com/>) are usually reliable. Try the answers with the highest vote.
- Drawing Graphs with package TikZ
 - To draw scientific graphs, check out [Pgfplot](https://www.sharelatex.com/learn/Pgfplots_package). Tutorials and examples are available on https://www.sharelatex.com/learn/Pgfplots_package.
 - The package is a handy tool for game trees as well (you may use it for your Game Theory assignments). Here is a tutorial with good examples: http://www.sfu.ca/~haiyunc/notes/Game_Trees_with_TikZ.pdf
 - More generally, see the following page for examples of graphs that TikZ can draw: <http://www.texample.net/tikz/examples/tag/graphs/>

Enjoy! Thank you!