

# L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>tutorial - A guide to better math writing

March 28, 2013

## Loading packages

First, load basic packages

```
\usepackage{amsmath, amsfonts, amssymb}  
\usepackage{mathtools}
```

Note that all packages to which you do not pass options can be loaded, separated by a comma.

Otherwise, options are passed to single package with

```
\usepackage[options]{package}, for example
```

```
\usepackage[utf8x]{inputenc}  
\usepackage[T1]{fontenc}
```

Note in passing that the two above packages allow for accents such as “é” to be inserted directly on keyboard instead of \’e.

## Inserting math

You all know this one, so quick review and motivation for better environments:

Inline maths can be dealt with using the `math` environment `$a^2+b^2$` signs (between `$` signs, or `\(a^2+b^2\)`). For both, the output is similar  $a^2 + b^2$ .

Basic equations can be obtained via the `displaymath \[ ... \]`, and if you are old school and used T<sub>E</sub>X, `$$ ... $$` command is also supported.

Why not use the above or

`\begin{equation} ... \end{equation}`?

## Inserting math

You all know this one, so quick review and motivation for better environments:

Inline maths can be dealt with using the `math` environment `$a^2+b^2$` signs (between `$` signs, or `\(a^2+b^2\)`). For both, the output is similar  $a^2 + b^2$ .

Basic equations can be obtained via the `displaymath \[ ... \]`, and if you are old school and used T<sub>E</sub>X, `$$ ... $$` command is also supported.

Why not use the above or

`\begin{equation} ... \end{equation}`?

**Because they only support one line input.**

## A digression. . .

Doing  $\text{\LaTeX}$  is easy, good  $\text{\LaTeX}$  is more rare. What do you think of the following?

- ▶ Inconsistent spacing
- ▶ Display style used for inline math
- ▶ Equations that are not aligned or go beyond the margins
- ▶ Parenthesis which do not match the size of the input  $\left(\frac{a}{b}\right)$
- ▶ Text in math environment that displays in italic:  $\textit{Prob}(X > a)$
- ▶ Inconsistent choice of font and symbols accross the document.

### Example

$\mathbf{x}^T$  or  $\mathbf{x}^T$  for transpose,  $\mathbb{R}$  and  $\mathbb{E}$  for respectively the reals and expectation.

First, a review of the mathematics environments for equations:

```
\begin{align} \end{align}
```

```
\begin{alignat}{#number} \end{alignat}
```

and the corresponding starred versions

```
\begin{align*} \end{align*}
```

`alignat` uses pair `lr` equations, with an argument indicating the number of pairs. To evacuate and have all left-aligned, use `&&`.

For example, with `align*`, the following equation gets no label

$$\int_{-\pi}^{\pi} \sum_{i=1}^n \frac{e^{inx}}{n} \times f(y) dx = 0 \iff \mathbf{CC}^T \varepsilon \geq 0$$

while

$$\underset{b}{\overset{a}{X}} \bigoplus \xrightarrow[X>0]{\text{as } n \rightarrow \infty} \varphi \rho \nabla \quad (1)$$

which uses `align` get numbered automatically. We will come back to this latter.

Using `align`, we can (as the name indicate) have equations that are aligned within **columns** . Include `&` in the code at the place where you want to align the equation and create new lines with `\\` command

### Example

$$\int_A f dm \leq \int_A s dm + \epsilon$$

By the last step,

$$\int_A s dm = \sum_{i=1}^{\infty} \int_{A_i} s dm$$

You can add as many columns as you wish, but if you have  $n \geq 3$  columns, then column  $n$  will be **right-aligned**.

## Example

Here with five ampersands, we get the following alignment

```
\begin{align*}
  a&\leq b& & = \text{while} & (\mathrm{by} \star)\backslash
  c&\geq \arcsin(x)& & \textbf{you get}& \mathsf{ the picture}
\end{align*}
```

gives

$$\begin{array}{lll} a \leq b & = \text{while} & (\text{by}\star) \\ c \geq \arcsin(x) & \textbf{you get} & \text{thepicture} \end{array}$$



# Adding space

## Vertical spacing commands

- ▶ **skip:** `\smallskip`, `\medskip`, `\bigskip`: for use between paragraphs
- ▶ **break:** `\smallbreak`, `\medbreak`, `\bigbreak`: remove previous space and adjust at length; terminate paragraph and “indicate” page break To allow breaks for equations, use `\allowdisplaybreaks` in the beginning of your document.
- ▶ `\addvspace{<skip>}`: merge with other spacing, output will be maximum of the two
- ▶ `\vspace{<fill>}`: white space, using `\vspace*{<skip>}` will do the same, but not ignore previous spacing. The spacing also disappears if there is a page break

## Horizontal spacing commands

- ▶ `\enskip`, `\quad`, `\qquad` give a horizontal space of respectively  $\frac{1}{2}$ , 1, 2 em (wide as letter *M* in font size).
- ▶ `\hspace{<skip>}` general spacing
- ▶ `\hfill` (or equivalently) `\hspace{\fill}` give space, while `\hspace*{<skip>}` does not disappear after line break.
- ▶ `\,`, `\:` and `\;` produce character size spaces of increasing size, `\!` produces negative space (for use with integrals or e.g. to create a new “statistical independence” symbol  $\perp\!\!\!\perp$  using `\perp` ( $\perp$ )).

## Spacing for paragraphs

The commands below allow to change the spacing

```
\setlength{\parindent}{1cm}
```

```
\setlength{\parskip}{1cm}
```

and you can manually indent or remove indentation with a paragraph, provided that indentation is positive using

```
\indent, \noindent
```

## Line spacing

With the package `\usepackage{setspace}`, you can adjust the spacing inline. Recommendation is to put `\setstretch {1.2}` to avoid the following annoying visual effect:

This is a paragraph which consists only of text which is completely irrelevant for this, but better than nothing  $\sum_{i=1}^n \int_a^b$ . Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore  $\lim_{n \rightarrow \infty} \frac{a}{b+a}$  et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamcol [...]

## Parenthesis

You can have parenthesis match the size of the content by using `\left( \right)`, for any set of parenthesis:

[ `\{ \}` `\|` `\vert` `\langle \rangle` `\lceil \rfloor` `\rceil \rangle` ] respectively

[ { || | < [ [ ] ] > || || } ].

(If you don't close the bracket, however, the code won't compile!

If you only need one delimiter, you can use `\left.` or `\right.` for the missing one, e.g.

$$\left. \frac{\partial}{\partial \theta} \mathcal{L}(X; \theta) \right|_{\theta=0}$$

`\left. \frac{\partial}{\partial \theta} \mathcal{L}(X; \theta) \right|_{\theta=0}`

# Parenthesis

Alternatively, you might want sometime to choose the size manually. In this case, the syntax is

`\big( \Big( \bigg( \Bigg(` which gives

$(((($

Cases for which you need to use parenthesis frequently can be dealt with building a new command, for example for  $P_{\theta} \left( X > \frac{1}{2} \right)$

## Superscripts and subscripts

To have subscripts or superscript, write `X^{\tilde{a}}_i` to get  $X_i^{\tilde{a}}$ . Inline math will place the symbols on the sign. You can use `\limits` for **operators** to have limits above and below

Some commands will have different behavior depending whether they are in an equation environment or inline. You can force the scripts to display like in equations using `\displaystyle` (**Avoid at all costs unless necessary.**)

Alternatively, you can use the command `\everymath{\displaystyle}` if you are truly an heretic. There are some already made commands like `\dfrac`, `\dcases` to use.

For example,

```
\sum_{i=1}^n \frac{a_i}{b_{i}},  
\bigcup_{\substack{j \geq k \\ j \neq 1}} \mathcal{O}_j,  
\lim_{n \to \infty}
```

will give a different output if used inline:

$$\sum_{i=1}^n \frac{a_i}{b_i} \quad \lim_{n \rightarrow \infty} \bigcup_{\substack{j \geq k \\ j \neq 1}} \mathcal{O}_j$$

or in an equation environment

$$\sum_{i=1}^n \frac{a_i}{b_i} \quad \lim_{n \rightarrow \infty} \bigcup_{\substack{j \geq k \\ j \neq 1}} \mathcal{O}_j$$

## Hyphenation and breaks

L<sup>A</sup>T<sub>E</sub>X sometimes need precision for words it does know how to hyphenate, like Bayesian; you need to help it by writing the word `Ba\ -ye\ -si\ -an`

To break, you can use `\paragraph{}` or use `\newline` or equivalently `\\`, but just after the text. This is also achieved by adding more than three enter, but `\newline` can always be used to get new lines.

**(Use only when necessary and you need to force break)**

Adding `~` will create a white space which is not allowed to break, useful later for references such as Theorem 7, or Theorem 7.

# Symbols

There are finitely many symbols one can use when appropriate, but notice some improve the look of your document. When you have  $\hat{\sigma}_{ij}^2$ , the rendering is not great as opposed to  $\widehat{\sigma}_i^2$  (the second looks better)

`\hat`, `\tilde` versus `\widehat`, `\widetilde`.

A comprehensive list of most symbols and size can be found [here](#)

I encourage you to use an IDE for L<sup>A</sup>T<sub>E</sub>X or to look up symbols. Use operators such as `\cos` (`\cos`) and **not** `cos` (`cos`). Also, sets like  $\{a \in \mathbb{R} \mid a^2 < 1\}$  have better spacing using `\mid` than just the vertical bar.

## amsmath commands

There are many useful commands provided by the AMS package, namely

- ▶ cases with `\begin{cases}`, variants of which (for text) are present in the `mathtools` package

$$p(x, x') = \begin{cases} q(x, x')\alpha(x, x') & \text{if } x' \neq x \\ 1 - \sum_{x' \neq x} q(x, x')\alpha(x, x') & \text{if } x' = x \end{cases}$$

- ▶ The commands `\intertext` (or `\shortintertext`) are used for short text insertion between equations, while keeping the alignment of multi-lines equations.
- ▶ `\boxed{}` for boxed equations  $\boxed{a^2 + b^2}$

The commands `overset` or `underset` or `stackrel` can be used to add subscript where needed

$$A \overset{!}{=} B; \quad A \underset{!}{=} B; \quad A \stackrel{!}{=} B$$

`A \overset{!}{=} B`; `A \underset{!}{=}B`; `A \stackrel{!}{=} B`

# Dots

The usual commands for dots are

`\ldots`, `\cdots`, `\ddots`, `\vdots` where the latter are for use in matrices and the former for ellipsis or between operators in math mode. AMS recommends to use the following instead

- ▶  $a, \dots, b$ : (`\dotsc`) for “dots with commas”
- ▶  $a + \cdots - c$ : (`\dotssb`) for “dots with binary operators”
- ▶  $a \times b \cdots c$ : (`\dotssm`) for “multiplication dots”
- ▶  $\int \cdots \int$ : (`\dotssi`) for “dots with integrals”
- ▶  $\dots$ : (`\dotssi`) for “other dots” (none of the above)

## mathtools package

This package should always be loaded, as it correct certain inconsistency in `amsmath`. It also provides additional options, some of which are detailed below.

If you want to use `align` everywhere and not care about starred versions, use the following

```
\mathtoolsset{
showonlyrefs,
mathic = true
}
```

## mathtools fixes

The `adjustlimits` command allows to get delimiters of similar height and better rendering

```
\[  
\text{a)} \adjustlimits\lim_{n\to\infty} \max_{p\geq n} \quad \quad  
\text{b)} \adjustlimits\lim_{n\to\infty} \max_{p^2\geq n} \quad \quad  
\text{c)} \adjustlimits\lim_{n\to\infty} \sup_{p^2\geq nK} \quad \quad  
\text{d)} \adjustlimits\limsup_{n\to\infty} \max_{p\geq n}  
\]
```

$$\text{a) } \lim_{n \rightarrow \infty} \max_{p \geq n} \quad \text{b) } \lim_{n \rightarrow \infty} \max_{p^2 \geq n} \quad \text{c) } \lim_{n \rightarrow \infty} \sup_{p^2 \geq nK} \quad \text{d) } \limsup_{n \rightarrow \infty} \max_{p \geq n}$$

$$\text{a) } \lim_{n \rightarrow \infty} \max_{p \geq n} \quad \text{b) } \lim_{n \rightarrow \infty} \max_{p^2 \geq n} \quad \text{c) } \lim_{n \rightarrow \infty} \sup_{p^2 \geq nK} \quad \text{d) } \limsup_{n \rightarrow \infty} \max_{p \geq n}$$

## mathclap and smashoperator

Use `\smashoperator[<pos>]{<operator with limits>}` or alternatively `<operator>_{\mathclap{<subscript>}}`

Sometimes, the subscript under a sum is too large and will cause bad display. The `smashoperator` will force the subscript or superscript to collapse around the rest of the output

```
X+ \smashoperator{\sum_{i, j \in \{1, \dots, n\}}} Y_i X_j  
X+ \mathclap{\sum_{i, j \in \{1, \dots, n\}}} Y_i X_j
```

$$X + \sum_{i, j \in \{1, \dots, n\}} Y_i X_j$$

$$X + \sum_{i, j \in \{1, \dots, n\}} Y_i X_j$$

$$X + \sum_{i, j \in \{1, \dots, n\}} Y_i X_j$$

# Arrows

There is a version of arrows that allow for superscript and subscripts to be added using

```
\xrightarrow[<subscript>]{<superscript>}
```

$$\widehat{F}_n \xrightarrow[\mathcal{D}]{n \rightarrow \infty} F$$

The line extend as the content increase, as opposed to adding a subscript to an arrow `\Longleftrightarrow` as  $n \rightarrow \infty$

The `MoveEqLeft` command allows to align with `&` for equations that are too long to display after first (in)equality.

```
\begin{align*}
\MoveEqLeft \framebox[10cm][c]{Long first line}\\
& = \framebox[6cm][c]{\hphantom{g} 2nd line}\\
& \leq \dots
\end{align*}
```

$$\begin{aligned} & \text{Long first line} \\ & = \text{2nd line} \\ & \leq \dots \end{aligned}$$

## New commands

Use the `\newcommand{<cmd>}{<def>}` and `\renewcommand...` to create shortcuts. To get operators with matching parenthesis, you can define things such as

```
\newcommand{\E}[2] [] {{\mathsf E}_{\#1}\left(\#2\right)}
```

for expectation. The syntax indicates that there are two arguments, the empty bracket next to it that the first one is **optional**, and the arguments will be placed in the corresponding positions. To use the command, simply put e.g. `\E[\theta]{X}`

```

\newcommand{\createHeader}[4]{
  \framebox{
\begin{minipage}{0.5\textwidth}
\begin{flushleft}
Pr. #1 \\\ \textsf{#2} - MATH #3
\end{flushleft}
\end{minipage}
\begin{minipage}{0.49\textwidth}
\begin{flushright}
<Name>, <course_code>\\
Assignment #4
\end{flushright}
\end{minipage}
}\linebreak}

```

## geometry package

```
\usepackage[left=2cm,top=1cm,right=3cm,nohead,nofoot]{geometry}
```

to control the margins. Note that the optimal length is already the one you find by default for reading purposes.

Do not use `fullpage` package, as setting margins `margin=1.5in` does the same thing and will override the commands that are passed to `geometry`.

# Header and footers

You have access in the class book to those, however you can also use the package `fancyhdr` to get more control

```
\usepackage{fancyhdr}
\pagestyle{fancy}
\lhead[lh-even]{lh-odd} \lfoot[lf-even]{lf-odd}
\chead[ch-even]{ch-odd} \cfoot[cf-even]{cf-odd}
\rhead[rh-even]{rh-odd} \rfoot[rf-even]{rf-odd}
```

and to set in article header and footers.

# Itemize

To get a list, use the environments

```
\begin{enumerate} \item \end{enumerate}
\begin{itemize} \item \end{itemize}
```

to get list with numbering or tag. Nested enumerate will have different counters,

## 1. First item

### 1.1 Subitem

#### 1.1.1 Yet further subitem

or letters (this is a beamer, so the output is not what it should be in article, per say.

## enumerate package

Using `\usepackage{enumerate}` package allows to control the tag for individual items or to change the complete list. To change the `itemize` symbol, use

```
\renewcommand{\labelitemi}{<new_symbol>}
```

```
\begin{enumerate}
```

```
\item[a]
```

```
\item
```

```
\end{enumerate}
```

respectively

a)

1.

```
\begin{enumerate}[(a)]
```

```
\item
```

```
\item
```

```
\end{enumerate}
```

(a)

(b)

# minipage

In the last slide, I used the `minipage` to get two columns (see Assignment template)

```
\begin{minipage}{0.49\textwidth}  
  Some text  
\end{minipage}
```

To change the behavior, spacing and indentation of enumerate you can add the following to your template, particularly for assignments

```
\usepackage[shortlabels]{enumitem}  
\setlist{nolistsep,leftmargin=0.1cm,itemindent=0.4cm}
```

## Verbatim and comments

To control some code that you would like to display, you have to often get rid of characters that are not supported in the “input” that you selected. To preserve spacing and formatting, use the `verbatim` package with `\begin{verbatim}` `\end{verbatim}` or `\verb=` = for inline.

Note that all characters after `\end{verbatim}` such as `}` are ignored by the compiler. Also note that `\verb` must finish on the same line it started and that `=` indicates any character (but `*`). The first occurrence closes the verbatim environment, so be careful.

## Verbatim and comments

You can use `\texttt{}` for text output in typewriter style Also note that the `\begin{comment}\end{comment}` can be accessed once `\usepackage{verbatim}` is loaded.

# Referencing

In this section, we consider references and cross-references for the document, including

- ▶ title
- ▶ footnote
- ▶ sections, chapters and formatting (including table of content)
- ▶ theorems and counters
- ▶ equations references and page references (including tags)
- ▶ hyperlinks
- ▶ bibliography and index

# Sectioning

It is easier to understand your work if it is **structured** in a comprehensive manner. To this effect, there is a set of commands that will allow you to build a **hierarchy** in your document.

`\chapter`, `\section`, `\subsection`, `\subsubsection` and the corresponding starred versions (which get no number and are not present in the toc). If you add `\tableofcontents` in the beginning of the document, you will get a ...table of content, with by default the first three levels.

N. B. `chapter` is the first part and is defined only in books and related classes (e.g. `thesis`)

# Changing format

Use the package

`\usepackage[calwidth, sf, big,compact]{titlesec}` with options (e.g. the ones on top) to change the look of the sections. For individual changes, here is an example of code.

```
\titleformat{\section}[block]{\sffamily}
{\color{JungleGreen} \centering \Large Section \thesection}
{0pt}
{\linebreak \color{JungleGreen} \Large\centering}
```

When using the `hyperref`, you will get links and a table of content with your pdf document. However, if you write `\section{\sigma-algebra}`, the output in Adobe Reader per say will be the string and not the symbol. To compensate this, you can use the command

```
{\texorpdfstring{<tex>}{<string>}}
```

to get a better result. Unfortunately, there is not way to get  $\sigma$ -algebra in the incorporated document, but you could have Sigma-algebra rather than  $\sigma$ -algebra.

## title

Before the `\begin{document}` statement, you can add fields such as `\title{}`, `\author{}`, `\date{\today}` and then have a title printed using `\maketitle`

You can redefine and change the format of the titlepage, to get either a full page or a short block of text using `\makeatletter ... \makeatother` commands. Ask me for examples or check it online.

## Footnotes

To add footnotes, simply add `\footnote{the footnote of interest}` to get a inline reference with an exponent corresponding to the number of the footnote. Try as a rule of thumb to put those references after punctuation signs (and not math, since otherwise the output is confusing).

The output may span multiple pages if needed, and is in `\footnotesize` (likewise, there are many different size for text, which you can change using e.g. `{\tiny <output>}`).

## Theorem environment

AMS has a default theorem environment, but it is restrictive and you might want to redefine the counter, the name of the actual theorems (such as `example`) and the look.

`\begin{theorem}[<name>]\end{theorem}` will create a “section” that gets referenced and formatted. You can have end symbols like  $\square$  for proofs like environments.

# Redefining theorems

Use the package

`\usepackage[amsmath, thref, thmmarks,hyperref]{ntheorem}` and change the content with some commands, e.g.

```
\theoremstyle{break}
```

```
\theoremheaderfont{\mdseries\sffamily\color{Green}}
```

```
\theorembodyfont{\normalfont}
```

```
\newtheorem{theorem}{Theorem}[section]
```

```
\newtheorem*{lemma}[theorem]{Lemma}
```

```
\newtheorem{exer}{Exercise}[section]
```

```
\renewcommand{\Box}{\rule{1.5ex}{1.5ex}} % end of proof
```

```
\newenvironment*{proof}{\par\noindent{\normalfont{\bf{Proof}}}  
\hfill \small$\blacksquare$\\[2mm]}
```

## Equations references

At the end of a unstarred equation, such as `align`, add `\label{key:number}` to get a reference.

You can now add `\ref{key:number}` or `\eqref{key:number}` to get respectively the number (or tag) and the version, but with parenthesis. Upon running your code twice, you will see the number printed. If the reference is undefined, you will notice familiar `??` symbols.

`\label` can take placed in delimiters such as sections and theorems, and will refer to the current one. Note that labels inside nested enumerate will not work properly.

## Reference numbering

For the counters of the equations, you can set it by adding something like

```
\renewcommand{\theequation}{\thesection.\arabic{equation}}  
\numberwithin{theorem}{section}
```

to have the counter take into consideration the sections and formatting desired or for the latter to reset any counter of your choice. If you want subreferences for equations, nest the corresponding between

```
\begin{subequations} \begin{align}... You can also remove  
the reference from a line with \nonumber command.
```

# Tags

Tags work just as for references, but putting tag next to the label allows to change the display (*i.e.* replace the number by something more meaningful, like † or ☆).

```
\begin{quote}
\renewcommand*{\rmdefault}{ppl}\normalfont\itshape
\begin{equation*}
a=b \label{eq:example}\tag*{Q\&A}
\end{equation*}
See \ref{eq:example} or is it better with \refeq{eq:example}?
\end{quote}
```

$$a = b$$

Q&A

See ?? or is it better with ???

## Other references

`\tag*` is always set in upright shape, while `\tag` is in parenthesis.  
`mathtools` provides `\refeq` for upright referencing.

$$a^2 = b^2 \tag{*}$$
$$a^2 = b^2 \tag{†}$$

```
\begin{align}
a^2&=b^2 \label{1}\tag{*$}$}
a^2&=b^2 \label{eq:2} \tag*{${\dagger}$}
\end{align}
```

You can also change the display, for e.g. to square brackets with `\usetagform{brackets}`. See `mathtools` package.

A particularly useful feature is the `\showonlyrefs`, but it clashes with `ntheorem` package. To fix this, use

```
\usepackage[overload,ntheorem]{empheq} before loading ntheorem. Then use  
\mathtoolsset{showonlyrefs,showmanualtags} to only have the  
equations that are referenced be numbered.
```

Finally, you can use `\pageref` to get a hook for a particular page to refer to.

## hyperref package

Load the package `hyperref` after `xcolor` and `tikz`. Use

```
\usepackage[pdftex, colorlinks=true]{hyperref}
\hypersetup{urlcolor=, linkcolor=, citecolor=}
% \usepackage{cleveref}
% \usepackage{autonum}
```

to change the colors of your links and references. You can either get colored box for the link (which does not get printed, (`colorlinks=false`) or color the string `colorlinks=true`, which is equivalent to adding `\textcolor{color}{<text>}` or `{\color{color} <text>}`. This package adds PDF support for the table of content and let you navigate to URLs.

## Bibliography (standard)

Two choices when it comes to bibliography; either using an external support (BibTeX) or the AMS default one.

```
\begin{thebibliography}{2} \setlength{\parindent}{0pt}
\bibitem{key1} Rohatgi, V.K., A.K.M.E. Saleh,
  \textit{An introduction to Probability and Statistics}, Wiley, 2001, 716p.
\bibitem{key2} Johanna Ne\v{s}lehov\{a}, \textit{MATH 356: Honours Probability}
\end{thebibliography}
```

The number of items in braces  $\{\}$  must be set as a function of the number of citations: e.g. you could have 9 for less than 10 references and 99 for anything above.

## natbib package

The default command is `\cite{key1}`. The natbib package allows to get finer referencing via

`\citep[<text_before>][<text_after>]{Holland:1986}`  
which adds parenthesis and corresponding remarks.

You can also access `\citeauthor`, `\citeyear`, etc. with reference to the bibliography. With BibTeX, you can change the format

```
\bibliographystyle{apalike}  
\bibliography{<filename.bib>}
```

# Formatting for BibTeX

There are many different formatting, depending whether you are citing a book or an article. Here is an example, see the related documentation for more

```
@Article{Kaplan:2012,  
title = "{A Two-Step Bayesian Approach for Propensity Score Analysis:  
Simulations and Case Study}",  
author = {Kaplan, David and Chen, Jianshen},  
year = {2012},  
journal = {Psychometrika},  
volume = {77},  
number = {3},  
pages = {581-609},  
keywords = {propensity score analysis; Bayesian inference},  
url = {http://EconPapers.repec.org/RePEc:spr:psycho:v:77:y:2012:i:3:p:581-609}  
}
```



# Index

To get an index, use the package `\usepackage{makeidx}` and the command `\makeindex` before the document. To print the index, use (duh) `\printindex`.

References are added using `\index{}`. For nesting, one has `\index{ref1@ref2}` which display `ref2` , while `\index{ref1!ref1.1}` creates nesting. For example, `\index{Sigma-field@$\sigma$-field}` to get math display, but have lexicographical order of S and `\index{Distributions!Normal}` for nesting.

To reference with a link some theorem environment (including the name rather than the sole number one can use the following, where the \* is used to avoid nested hyperlinks

```
\hyperref[mainlemma]{lemma \ref*{mainlemma} }
```

Other options include `\url{<url>}` which gets displayed in monofont, and `\href{<url>}{<description>}` whose text formatting is different. You can email someone using `\href{mailto:address}{<text>}`.

# Mathematical Operators

You can define new operators for your needs with the commands

```
\DeclareMathOperator{<name>}{<command>}
```

for e.g. `\DeclareMathOperator*{\argmax}{arg\,max}` The star indicates the limits are set on top and bottom as opposed to the right.

# Math Alphabet

```
\usepackage[mathscr]{eucal}  
\DeclareMathAlphabet{\mathpzc}{OT1}{pzc}{1}{n}
```

There are many alphabets available to you, either

- ▶ `\mathbf`: **ABCDEFgh** $\alpha\beta\gamma$
- ▶ `\mathrm`: *ABCDEFgh* $\alpha\beta\gamma$
- ▶ `\mathcal`: *ABCDE* $\{\}$  $\langle\alpha\beta\gamma$
- ▶ `\mathbb`: **ABCDE**  $\mathbb{U}$   $\mathbb{d}$   $\mathbb{z}$   $\alpha\beta\gamma$
- ▶ `\mathscr`: *ABCDE* $\{\}$  $\alpha\beta\gamma$
- ▶ `\mathpzc`: *ABCDEFgh* $\alpha\beta\gamma$
- ▶ `\mathtt`: **ABCDEFgh** $\alpha\beta\gamma$
- ▶ `\mathsf`: **ABCDEFgh** $\alpha\beta\gamma$

## Bold symbols

Note in the previous slide that we do not have upright greek letters, which you can get through `upgreek` package.

For bold symbols in general in italic ***and*** for bold greek letters, use `\boldsymbol{<char>}`. I strongly suggest that you create some shortcut for this one.

## Consistent notation

Try to be consistent. If you use  $\mathbb{R}^n$  for sets, don't take it for operators. Similarly, try to use  $X$  capital letter for random variables,  $x$  for non-stochastic realizations,  $\mathbf{X}$  or  $\mathbf{x}$  for vector quantities, upright bold  $\mathbf{C}$  for matrices, etc. Keep `mathcal` for particular uses, like distributions.

## Misc. packages for figures

```
\usepackage{color}
\usepackage[natural, dvipsnames, pdftex]{xcolor}
\usepackage{graphicx}
\usepackage{tikz}
```

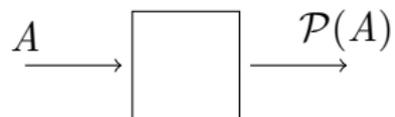
The `xcolor` package clashes with many other; it provides additional colors than the `color` package

To add a figure or picture, you will want the `graphicx` package loaded (for PDF, JPG, ... rather than EPS files). You can also add figures in text `\usepackage{wrapfig}` then use `\begin{wrapfigure}[lineheight]{position}{width}` or with `subfigure`

```
\begin{figure*}[h!]
\centering
\includegraphics[width=0.7\textwidth]{Q1-plot.pdf}
\end{figure*}
```

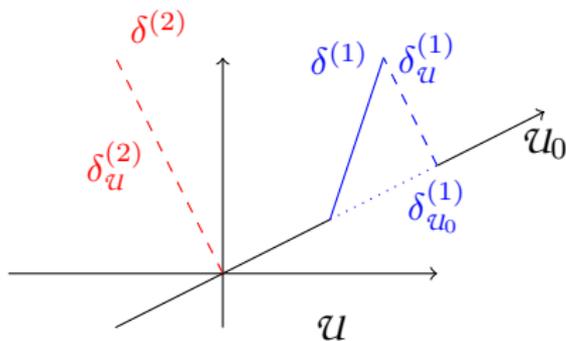
# Tikz

```
\begin{tikzpicture}
\node[above] at (-2,0) {$A$};
\draw[->] (0.1,0)--(1,0) node[above] {$\mathcal{P}(A)$};
\draw[->] (-2,0)--(-1.1,0);
\draw (-1,-0.5)--(0,-0.5)--(0,0.5)--(-1,0.5)--cycle;
\end{tikzpicture}
```



Many more cool examples online.

Figure: Geometric interpretation of UMVUE,  $\delta^{(2)} \perp \mathcal{U}_0$



# Table

```
\begin{table}[ht]
\begin{center}
\begin{tabular}{rllll}
\hline & Exact & MC & Biased IS & \\\ \hline
Value & 0.02883 & 0.027551 & 6.071\$\times 10^{-11}\$ & \\\
Bias & & 0.0013 & 0.0288 & \\\
Rel. Bias & & 0.0444 & 1.0000 & \\\ \hline
\end{tabular} \end{center} \end{table}
```

	Exact	MC	Biased IS
Value	0.02883	0.027551	$6.071 \times 10^{-11}$
Bias		0.0013	0.0288
Rel. Bias		0.0444	1.0000

## verbments package

The verbments package allow to have code highlighting using Python coloring script.

```
\begin{pyglist}[language=r]
#IS with truncated support
x2<-runif(10^3,-8,-1)
hIS<-function(x){
  return((exp(-(x-3)^2)+exp(-(x-6)^2/2))*dnorm(x)/7)
}
samp2<-hIS(x2)
val2<-mean(samp2)

\end{pyglist}
```

```
#IS with truncated support
x2<-runif(10^3,-8,-1)
hIS<-function(x){
  return((exp(-(x-3)^2)+exp(-(x-6)^2/2))*dnorm(x)/7)
}
samp2<-hIS(x2)
val2<-mean(samp2)
```

## That all. That's enough, isn't it?

- ▶ I will upload the slides on sums website [sums.math.mcgill.ca](http://sums.math.mcgill.ca).
- ▶ There is nice integration of `tikZ` with the `knitr` package in R; you can exports graphics in `tikZ` Apparent drawback of this method is for large graphs, when the number of lines is above 20 millions. If you have Linux, use this script to remove duplicate lines

```
awk ' !x[$0]++' plot.tex > newplot.tex
```

- ▶ I am around if you have further questions in the future.