Typesetting papers

October 5, 2018

1 Introduction

Whether you like it or not, the appearance of your paper does contribute (possibly unconsciously) to a reader/reviewer's impression of the quality of your paper. This document highlights some common typesetting problems and how to fix them. Most of these problems apply to any typesetting system, but this document uses LaTeX and assumes that you already know LaTeX.

Incorrect examples are labeled according to their severity (admittedly, this is highly subjective):

*** These errors can lead to misunderstanding, or jump out at as obviously wrong. Either way, errors of this type can lead to a lower score.

** These errors are more minor but can make your paper appear less polished.

* Most people don't notice these and don't expect you to, either.

2 Text

2.1 Spacing

Most of the time, TeX knows how much horizontal or vertical space to insert. Although it has various commands for inserting positive or negative space, avoid them if you can. However, there are a few cases where TeX regularly needs your help.

Intersentence vs. interword spaces TeX by default attempts to put more space after sentence-final punctuation. The problem is that it isn't very good at figuring out what punctuation is sentence-final. Turn this behavior off by putting `\frenchspacing` in your preamble, and skip this part. If you like the extra space, or an editor (I mean a human editor) requires it, read on.

The heuristic for sentence-final punctuation is: a period (`.`), question mark (`?`), or exclamation point (`!`) that is *not* preceded by a capital letter. To force a period to be non-sentence-final, use `.\␣` instead.

*	Dr. Jekyll and Mr. Hyde	`Dr. Jekyll and Mr. Hyde`
	Dr. Jekyll and Mr. Hyde	`Dr.\ Jekyll and Mr.\ Hyde`

Italic correction The *italic correction* (`\/`) is used when a slanted letter is followed by a non-slanted letter.

*	*If* I may	`{\em If} I may`
	If I may	`{\em If}\/ I may`

The `\emph` and `\textit` macros insert the italic correction automatically, allowing you to forget about this rule in 90% of cases. (You can also suppress an automatic italic correction using `\nocorr`.)

2.2 Line breaking

TeX automatically breaks lines and hyphenates words for you, but, again, there are a few cases where it needs your help.

This might be a good time to remind you to check that your installation of TeX has the hyphenation patterns for English loaded. Otherwise you can get weird hyphenations like

*** Whereas recognition of the inheren-
t dignity and of the equal and inalienable

Tie If you don't want a line break to occur between two words, use a *tie* (~). For example, use the tie after abbreviations and before numbers or equations.

```
Dr.~Jekyll
Figure~1
greater than~$n$
There are $n!$~ways
```

Hyphenation exceptions To override TeX's idea of how to hyphenate a word, use the `\hyphenate` macro to indicate all the possible positions where a word can be broken:

```
\hyphenation{trans-mog-ri-fi-ca-tion}
\hyphenation{Chiang}
```

2.3 Punctuation

There are (infamously) three kinds of dashes in text:

hyphen	-	`-`
en-dash	–	`--`
em-dash	—	`---`

(In addition, there is the minus sign, which we'll deal with when we talk about math mode.)

Hyphen The hyphen is used for compound words. It is never preceded or followed by a space except inside coordination constructions (e.g., *left- or right-handed*).

En-dash The en-dash is used for ranges. It is not preceded or followed by a space.

**	pages 101-102	`pages 101-102`
	pages 101–102	`pages 101--102`

Em-dash The em-dash can be used—sparingly—for interrupting a sentence or for joining two sentences together with no conjunction. It is not preceded or followed by a space. You can also use an en-dash *with* a space before and after for the same purpose (but be consistent).

**	Go ahead - make my day.	`Go ahead - make my day.`
**	Go ahead–make my day .	`Go ahead--make my day.`
	Go ahead – make my day.	`Go ahead -- make my day.`
	Go ahead—make my day.	`Go ahead---make my day.`

Quotes Single and double quotes come in left, right, and straight versions:

	single		double	
left	‘	`` ` ``	“	``` `` ```
right	’	`'`	”	`''`
straight	`'`	`\texttt{'}`	`"`	`\texttt{"}`

The left and right versions are for opening and closing quotations. The straight version should be used for printing code or data.

**	”Every time”	`"Every time"`
**	”I fire a linguist,”	`''I fire a linguist,''`
	“my score goes up.”	``` ``my score goes up.'' ```

Remember that in American English, punctuation generally goes inside quotation marks.

If a quotation ends with a nested quotation, put a `\thinspace` in between:

“He said ‘bye’ ” and left.	``` ``He said `bye'\thinspace'' and left. ```

3 Citations

In-text Use `natbib` if you can (ACL style files now do). In `natbib`, the two main citation commands are `\citet` and `\citep`. This works for both author-date and numerical citations. Don’t use `\cite`.

If a citation can be deleted without affect the meaning of a sentence, use `\citep`. It should go before punctuation.

	The sky is blue (Smith, 1900).	`The sky is blue \citep{smith:1900}.`
**	The sky is blue. (Smith, 1900)	`The sky is blue. \citep{smith:1900}`
***	The sky is blue Smith (1900).	`The sky is blue \citet{smith:1900}.`

If a citation is used as a noun phrase, use `\citet` (again, before punctuation).

	According to Smith (1900), the sky is blue.	`According to \citet{smith:1900}, the sky is blue.`
*	According to (Smith, 1900), the sky is blue.	`According to \citep{smith:1900}, the sky is blue.`

Footnotes Footnote marks go after punctuation. They should never go after numbers or formulas.

***	We ran the algorithm for 10^2 iterations.
**	We ran the algorithm for 10 iterations[2].
**	We ran the algorithm for 10 iterations. [2]
	We ran the algorithm for 10 iterations.[2]

4 Math

Use the `amsmath` package for math and read the “Short Math Guide for LaTeX.” The `mathtools` package also has some handy extras. Below are just a few of the more common problems encountered when writing math.

4.1 Type

In math, upright (or roman), italic, and bold type are as different as uppercase and lowercase letters are. Don’t use them interchangeably.

upright	X	`\mathrm{X}`
italic	X	`X`
bold	$\mathbf{X}$	`\mathbf{X}`

Italics are used for variables.

**	$O(n^3)$, where n is	`$O(n^3)$, where n is`
	$O(n^3)$, where $n$ is	`$O(n^3)$, where $n$ is`

If you must use a multi-character variable name (e.g. in pseudocode), use `\mathit`.

**	$foo \leftarrow foo + 1$	`fi \leftarrow foo + 1`
	$\mathit{foo} \leftarrow \mathit{foo} + 1$	`\mathit{foo} \leftarrow \mathit{foo} + 1`

In the first line, `foo` is interpreted as “`f` times `o` times `o`” and is typeset as such (which looks funny). In the second line, it is interpreted as a single symbol.

Boldface is used for vectors and matrices. In our literature, this rule is often extended to strings. To get boldface (upright) Latin letters, use `\mathbf`. To get boldface (italic) Greek letters, use `\boldsymbol` (part of `amsmath`).

Upright letters are used for everything else. For example, names of standard functions:

***	$logL$	`log L`
	$\log L$	`\log L`

Note that `\log` and macros like it automatically insert a space. You can define your own macros that behave this way. In our field, the most common ones are:

```
\DeclareMathOperator*{\argmin}{arg\,min}
\DeclareMathOperator*{\argmax}{arg\,max}
```

Subscripts are often merely descriptors, not variables. In these cases, they should technically be upright, not italic:

	q_i	`q_i`
*	q_f	`q_f`
	q_{f}	`q_{\mathrm{f}}`

In q_i, if i is an index that takes on integer values, it should be in italics. If ‘f’ stands for ‘final’ and is not a variable, it should be upright, not italic.

If a formula contains an English word, use `\text` (part of `amsmath`).

**	$n_{upside-down}$	`n_{upside-down}`
*	$n_{\mathrm{upside-down}}$	`n_{\mathrm{upside-down}}`
	$n_{\text{upside-down}}$	`n_{\text{upside-down}}`

Sometimes a word that seems to be part of a formula really isn't, because it's actually two formulas separated by a word.

***	$\{ww \mid w \in \Sigma^\ast and w = w^R\}$	`$\{ww \mid w \in \Sigma^\ast and w = w^R\}$`
***	$\{ww \mid w \in \Sigma^\ast \text{and} w = w^R\}$	`$\{ww \mid w \in \Sigma^\ast \text{and} w = w^R\}$`
	$\{ww \mid \text{$w \in \Sigma^\ast$ and $w = w^R$}\}$	`$\{ww \mid \text{$w \in \Sigma^\ast$ and $w = w^R$}\}$`

4.2 Punctuation

Equations, even displayed equations, are part of sentences just like any other phrase, and therefore should have punctuation just like any other phrase. For displayed equations, the punctuation goes inside the equation. (For inline equations, it's supposed to go outside, but no one will notice the difference.)

	Since
	$$x > 0,$$
	we have
	$$y = 12.$$
		`Since`
		`\begin{equation*}`
		`x > 0,`
		`\end{equation*}`
		`we have`
		`\begin{equation*}`
		`y = 12.`
		`\end{equation*}`

*	Since
	$$x > 0$$
	we have
	$$y = 12$$
		`Since`
		`\begin{equation*}`
		`x > 0`
		`\end{equation*}`
		`we have`
		`\begin{equation*}`
		`y = 12`
		`\end{equation*}`

***	Since
	$$x > 0$$
	, we have
	$$y = 12$$
	.
		`Since`
		`\begin{equation*}`
		`x > 0`
		`\end{equation*}`
		`, we have`
		`\begin{equation*}`
		`y = 12`
		`\end{equation*}`
		`.`

4.3 Symbols

For the bar inside a set comprehension or conditional probability, use `\mid`, not `|`.

*	$\{a^nb^n|n \geq 0\}$
	$\{a^nb^n \mid n \geq 0\}$
*	$P(X, Y) = P(X|Y)P(Y)$
	$P(X, Y) = P(X \mid Y)P(Y)$

For angle brackets around tuples, use `\langle` and `\rangle`, not `<` and `>`, which are comparison operators.

**	$< x, y >$	`<x,y>`
	$\langle x, y \rangle$	`\langle x,y \rangle`

For ellipses, don’t use `...`. Use `\ldots` in a comma-separated list and `\cdots` otherwise.

*	for $i$ in $1, ..., n$	`for $i$ in $i, ..., n$`
	for $i$ in $1, \ldots, n$	`for $i$ in $i, \ldots, n$`
	$w = w_1 \cdots w_n$	`w = w_1 \cdots w_n`

Never use programming-language operators in math.

***	$x * y >= 0$	`x*y >= 0`
	$xy \geq 0$	`xy \geq 0`
	$x \cdot y \geq 0$	`x \cdot y \geq 0`

A slash (`/`) for division is fine if there isn’t enough room for a `\frac`.

Some characters display differently in normal and math mode.

**	between -1 and 1	`between -1 and 1`
	between $-1$ and $1$	`between $-1$ and $1$`
*	$n_{\mathrm{upside-down}}$	`n_{\mathrm{upside-down}}`
	$n_{\text{upside-down}}$	`n_{\text{upside-down}}`
**	Call it x’.	`Call it x'.`
	Call it $x'$	`Call it $x'$.`

Also,

**	$X \cap Y = \phi$	`X \cap Y = \phi`
	$X \cap Y = \emptyset$	`X \cap Y = \emptyset`

5 Fonts

By far the easiest way to typeset text in non-Latin scripts is XeLaTeX and `fontspec`.

```
\usepackage{fontspec}
\newfontfamily{\zh}[Mapping=tex-text]{STHeiti}
```

where `STHeiti` is a Chinese font that can be replaced with any other Chinese font. Then you can write

```
{\zh 中文}
```

You can use nearly any TrueType or OpenType font installed on your computer; unfortunately this probably means that you have to use different fonts under different operating systems.

This also breaks compatibility with the `times` package included by many conference style files. To get Times New Roman (the order matters):

```
\usepackage{newtxtext,newtxmath}
\usepackage[no-math]{fontspec}
\setmainfont[Mapping=tex-text]{Times New Roman}
```

Or (again the order matters):

```
\usepackage[no-math]{fontspec}
\setmainfont[Mapping=tex-text]{Times New Roman}
\usepackage[italic]{mathastext}
```

The former looks better, but the latter works with any font, which is especially handy for posters or slides.

6 References

http://www.economics.utoronto.ca/osborne/latex/LTXERR.HTM

ftp://ftp.ams.org/pub/tex/doc/amsmath/short-math-guide.pdf