

# PDF and Web (from Latex) and Miscellaneous

Latex Smart  
6356 Agricultural Road  
University of British Columbia  
Vancouver BC  
V6T 1Z2

April 4, 2003

# Outline

- The Web, its documents, and  $\text{\LaTeX}$
- Conversion between formats (latex, dvi, pdf, ps, html, Word docs)
- Revisit  $\text{\LaTeX}$  macro `\def`, `\newcommand` and `\renewcommand`
- Editors (e.g. gvim, emacs and wineedit)
- rejoinder

## Ways to publish L<sup>A</sup>T<sub>E</sub>X document into Web

- Generate POSTSCRIPT or DVI files and add hyperlinks to these files.

**pros** easy to do and convenient for authors.

**cons**

- ★ may be inconvenient for readers. Readers need to download these files to local disk and need viewers (e.g. ghostview, xdvi) to view the files.
- ★ cannot show hyperlinks for ps files<sup>1</sup>.

---

<sup>1</sup>xdvi supports the loading of a Web browser when a URL link is encountered, but it cannot be integrated within a Web browser.

## Ways to publish L<sup>A</sup>T<sub>E</sub>X document into Web (Ct'd)

- Generate PDF file.

### pros

- ★ easy to do and convenient for authors
- ★ can easily navigate through the document and exploit hypertext information that might be present in the L<sup>A</sup>T<sub>E</sub>X source.
- ★ don't need to download files to local disk. Web browsers can be configured to load Acrobat as a helper. Acrobat can also pass URL links to the browser with a seamless integration.
- ★ has the ability to render pages with layout essentially exactly as in an original (e.g. L<sup>A</sup>T<sub>E</sub>X) document.

### cons

- ★ “A helper or plug-in has to be installed on the viewing system”.
- ★ “It restricts searchability because the document is not text”.
- ★ “It requires a verbose and bulky file format that includes all the non-standard fonts that are used”[1].

## Ways to publish $\text{\LaTeX}$ document into Web (Ct'd)

- Translate  $\text{\LaTeX}$  into HTML.
- Use Java and browser plug-ins to display a  $\text{\LaTeX}$  source directly inside a browser.

## What are difficulties to do the translation?

- $\text{T}_{\text{E}}\text{X}$  is capable of extremely detailed page layout, specifying precisely where on the page symbols go. HTML is not, because HTML is a functional mark-up language (specifying primarily document structure) not a page layout language.
- HTML's exact rendering is not specified by the document that is published but is, to some degree, left to the discretion of the browser.
- $\text{T}_{\text{E}}\text{X}$ 's excellent mathematical capabilities are absent from HTML and browsers.

## Two main choices for representing equations in HTML

- using bit-mapped images

**pros** it uses capabilities that are essentially universal to every graphical browser.

**cons**

- ★ it requires a separate graphical file for every equation, which becomes very cumbersome and slow to download.
- ★ also the alignment and sizing of the graphical equations is uncertain with respect to the rest of the text.

Software: `LaTeX2HTML`, `TeX4ht`.

- using browser fonts and tables for layout

**pros** one HTML document contains all the information, giving portability and speed of download.

**cons** it depends on having the symbol font accessible on the browser, and that the equation layout is not as compact or elegant as  $\text{T}_{\text{E}}\text{X}$ 's.

Software: `TtH`.

## Direct display of $\text{\LaTeX}$ on the web

- Using browser plug-in, such as `techexplorer`.

### pros

- ★ understands a large subset of the  $\text{\LaTeX}$ .
- ★ fast

### cons

- ★ the plug-in must be downloaded and installed on all browsers you want to use.
- ★ a separate version of the plug-in is needed for each computer platform.



## Direct display of $\text{\LaTeX}$ on the web (Ct'd)

- Using Java applets, such as `WebEQ`.

### pros

- ★ produce good quality display
- ★ can adapt to resizing of the browser window size and fonts.

### cons

- ★ slow
- ★ does not strictly process  $\text{\LaTeX}$ , but rather a variant called `WebTeX`.

## The `hyperref` package

- it extends the functionality of all the  $\text{\LaTeX}$  cross-referencing commands (including the table of contents, bibliographies, and so on).
- it also provides new commands to allow the user to write ad hoc hypertext links, including those to external documents and URLs.
- it should be the **last** of the loaded package in case its commands being overwritten.

## Configuring hyperref

The syntax

```
\hypersetup{keyvalue pairs}
```

Example:

```
\hypersetup{  
  pdftoolbar=true,  
  pdfmenubar=true,  
  pdffitwindow=true,  
  pdfpagelayout=TwoColumnLeft  
}
```

TwoColumnLeft means displaying the pages in two columns ,with odd-numbered pages on the left.

## Commonly used options of hyperref

<i>Option</i>	<i>Value</i>	<i>Default</i>	<i>Description</i>
breaklines	boolean	false	Allows link text to break across lines.
implicit	boolean	true	Allows to produce hyperlink for all cross-references.
backref	boolean	false	Adds backlink text to the end of each item in the bibliography as a list of section numbers.
pagebackref	boolean	false	Adds backlink text to the end of each item in the bibliography as a list of page numbers.
colorlink	boolean	false	Colors the text of links and anchors.
linkcolor	color	red	Color for simple internal links (cross-references).
anchorcolor	color	black	Color for anchor text.
citecolor	color	green	Color for bibliographical citations in text.
pagecolor	color	red	Color for links to other pages.
urlcolor	color	cyan	Color for linked network URLs.
bookmarks	boolean	false	Write a set of Acrobat bookmarks, in a manner similar to the table of contents.
bookmarksopen	boolean	false	If Acrobat bookmarks are requested, show them with all the subtrees expanded.
bookmarksnumbered	boolean	false	If Acrobat bookmarks are requested, include the section numbers.

## Commonly used options of hyperref (Ct'd)

<i>Option</i>	<i>Value</i>	<i>Default</i>	<i>Description</i>
pdfpagemode	name	None	Determine how the file is opened in Acrobat; the possibilities are None, UseThumbs, UseOutlines and FullScreen
pdfpagelayout	name	SinglePage	Possible values: OneColumn, TwoColumnLeft and TwoColumnRight.
pdftitle	text		Sets the document information Title field.
pdfauthor	text		Sets the document information Author field.
pdfsubject	text		Sets the document information Subject field.
pdfkeywords	text		Sets the document information Keywords field.
pdfpagetransition	name		The effect used when going to a new page;

## Acrobat page transition options

Table 1: Acrobat page transition options

<i>Option</i>	<i>Key(s)</i>	<i>Description</i>
Split	/Dm, /M	Two lines sweep across the screen to show the new page.
Blinds	/Dm	Multiple lines synchronously sweep in the same direction.
Box	/M	A box sweeps from the center out or from the edges in.
Wipe	/Di	A single line sweeps across the screen from one edge to the other.
Dissolve		The page image dissolves in a piecemeal fashion.

- It seems that any word other than the above 6 options will produce normal display.
- **Example:** `\hypersetup{pdfpagetransition={Blinds /Dm /V}}.`

## Acrobat page transition options (Ct'd)

**/Di** (Direction) The direction of movement, in degrees (counterclockwise). Values are generally in  $90^\circ$  steps.

**/Dm** (Dimension) If a choice between horizontal or vertical is allowed, value is /H (horizontal) or /V (vertical).

**/M** (Motion) If an effect can be from the center out or from the edges in, value is /I (in) or /O (out).

# Hyperlinks

```
\href{url}{text}
```

make a hyperlink to the webpage with address `url`.

Example:

```
\href{http://hajek.stat.ubc.ca/~webmaste/howto/  
editor/latex.html#doublespace}{Howto set  
double space effect in LaTeX}
```

Howto set double space effect in LaTeX

In `url`, the special characters `#` and `~` do **not** need to be escaped in any way.

Note: The effect of this page is by

```
\hypersetup{pdfpagetransition={Split /Dm /V}}
```



## Hyperlinks (Ct'd)

```
\hyperlink{name}{text1}
```

make a hyperlink to a hypertext object defined somewhere by `\hypertarget`.

Example:

```
\hyperlink{target}{Definition of \texttt{hypertarget}}
```

Definition of `hypertarget`

Note: The effect of this page is by

```
\hypersetup{pdfpagetransition={Blinds /Dm /H}}
```

## Hyperlinks (Ct'd)

`\hypertarget{name}{text2}` defines a hypertext object. The `text2` is not necessarily the same as the `text1` in `\hyperlink{name}{text1}`. However, the `name` in `\hypertarget{name}{text2}` should be the same as that in `\hyperlink{name}{text1}`.

Example:

```
\hypertarget{target}{} 
```

Note: The effect of this page is by

```
\hypersetup{pdfpagetransition={Box /M /I}}
```

## Hyperlinks (Ct'd)

```
\hyperref{label}{text}
```

make a hyperlink to a point established with a normal  $\text{\LaTeX}$  `\label` command with the symbolic name `label`.

Example:

```
\hyperref[eq1]{This links to Equation (\ref*{eq1}).}
```

This links to Equation (1).

Note:

- `\ref*{eq1}` generates the right number but not to form a link.
- The effect of this page is by

```
\hypersetup{pdfpagetransition={Wipe /Di 90}}
```

## Hyperlinks (Ct'd)

This is the equation referred by the hyperlink defined in the previous page

`\hyperref[eq1]{This links to Equation (\ref*{eq1}).}`:

$$\int_a^b x dx = \frac{b^2 - a^2}{2} \quad (1)$$

The effect of this page is by

`\hypersetup{pdfpagetransition={Dissolve}}`

## pagebackref

1. Type `\usepackage[pagebackref]{hyperref}` in the preamble of the  $\text{\LaTeX}$  file.

2. Type just before `\end{document}`

```
\bibliographystyle{plain}  
\bibliography{workshop}
```

(Suppose the file `workshop.bib` is in the current directory.)

3. Use the command `\cite` to cite reference defined in `workshop.bib`. Here we cite [1] (`\cite{GoossensEtAl:1999}`).

## pagebackref (Ct'd)

- If there are more than two places referring to the same reference, use option `pagebackref` instead of `backref` since `backref` can generate only one back hyperlink.
- If there are some pages with `\thispagestyle{empty}`, then `pagebackref` will go back to wrong page.
- The effect of this page is by

```
%\hypersetup{pdfpagetransition={Dissolve}}
```

## Acrobatmenu

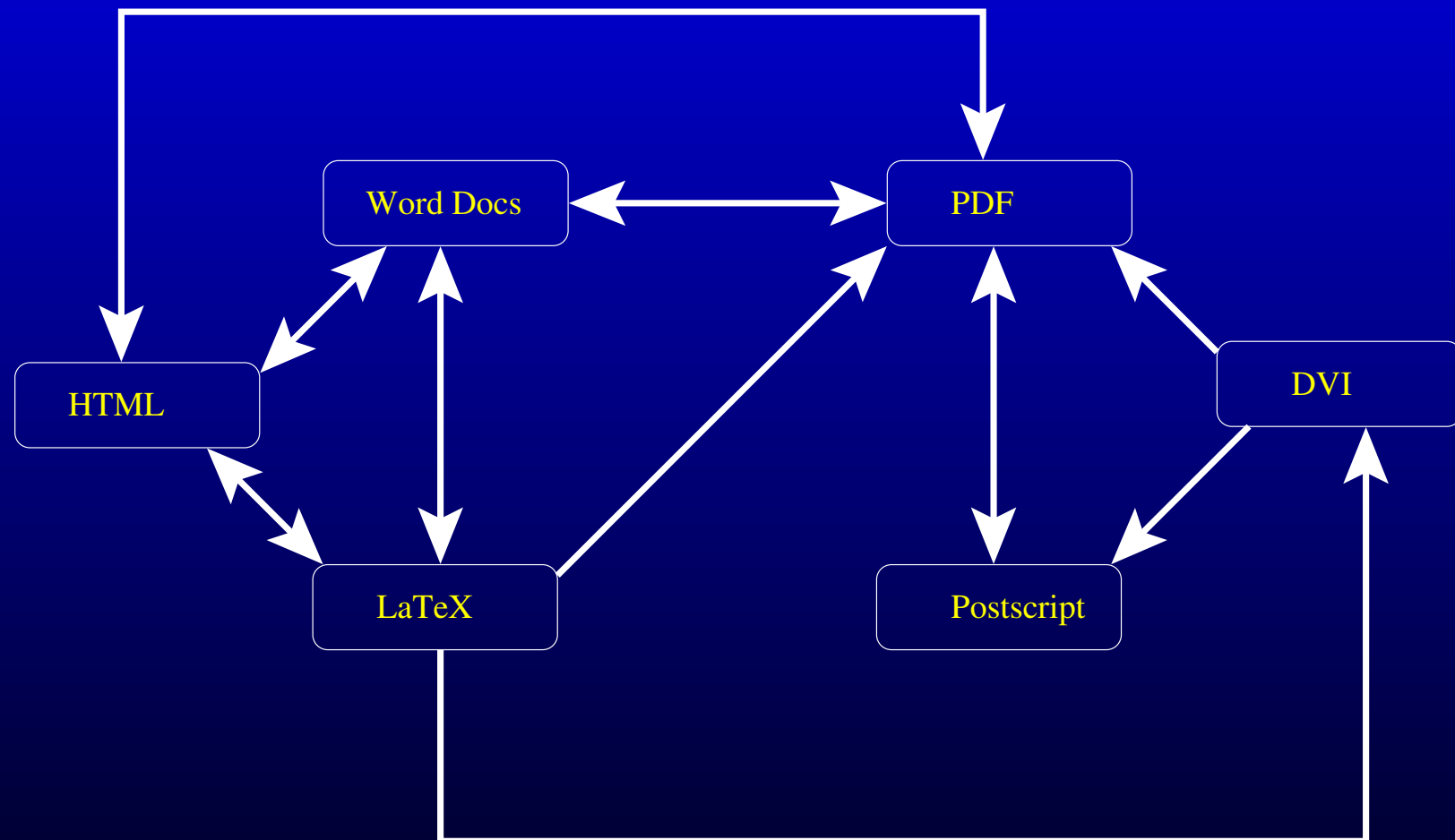
The navigation hyperlinks are constructed by adding the following in the preamble:

```
\usepackage{pifont}
\rightfooter{\NavigationBar}
\newcommand{\NavigationBar}{\Large
  \Acrobatmenu{PrevPage}{\reflectbox{\ding{227}}}
  \Acrobatmenu{NextPage}{\ding{227}}
  \Acrobatmenu{FirstPage}{\reflectbox{\ding{224}}}
  \Acrobatmenu{LastPage}{\ding{224}}
  \Acrobatmenu{GoBack}{\reflectbox{\ding{249}}}
  \Acrobatmenu{Quit}{\ding{54}}}%
}
```

If you have graphics for the navigation bars, you can replace Dingbat with graphic buttons, e.g.

```
\Acrobatmenu{NextPage}{\includegraphics{nextpic}}.
```

# Conversion between formats





## Conversion between formats (Ct'd)

Conversion	Software	free of charge?	available from dept?
$\text{\LaTeX} \Rightarrow \text{DVI}$	latex	free	Yes
$\text{\LaTeX} \Rightarrow \text{PDF}$	pdflatex	free	Yes
$\text{\LaTeX} \Rightarrow \text{HTML}$	latex2html	free	Yes
$\text{\LaTeX} \Rightarrow \text{WORD DOCS}$	Word2Tex	\$\$	No
$\text{WORD DOCS} \Rightarrow \text{\LaTeX}$	Tex2Word	\$\$	No
$\text{HTML} \Rightarrow \text{\LaTeX}$	html2latex	free	No
$\text{DVI} \Rightarrow \text{PS}$	dvips	free	Yes
$\text{DVI} \Rightarrow \text{PDF}$	dvipdfm	free	Yes
$\text{PS} \Rightarrow \text{PDF}$	ps2pdf	free	Yes
$\text{PDF} \Rightarrow \text{PS}$	pdf2ps	free	Yes
$\text{PDF} \Rightarrow \text{WORD DOCS}$	Drake Plugin Software	\$\$	No
$\text{WORD DOCS} \Rightarrow \text{PDF}$	BCL easyPDF 3.0 OR	\$\$	No
	Adobe Acrobat	\$\$	Yes
$\text{HTML} \Rightarrow \text{PDF}$	Adobe Acrobat	\$\$	Yes
$\text{PDF} \Rightarrow \text{HTML}$	IntraPDF . com	\$\$	No
various conversions to and from PDF	PDF995	free	No

## **def, newcommand and renewcommand**

`\def`, `\newcommand`, and `\renewcommand` are used to define or redefine  $\text{\LaTeX}$  command. The usages are:

```
\def{cmd}{def}  
\newcommand{cmd}[args][opt]{def}  
\renewcommand{cmd}[args][opt]{def}
```

**cmd** The name of the new or redefined command. The first character of `cmd` must be `\`.

**args** An integer from 1 to 9 denoting the number of arguments of the command being defined.

**opt** Default value for the first argument.

**def** The definition of the command. Use  $\#n$  to denote the  $n$ -th argument.

## def, newcommand and renewcommand (Ct'd)

Example:

```
\usepackage{amsmath, amssymb, amsfonts}
\newcommand{\gb}[1]{\ensuremath{\boldsymbol{\#1}}}
\newcommand{\ve}[2][\ensuremath{x}]
  {\ensuremath{(\#1_1, \ldots, \#1_{\#2})}}
```

The command `$\gb{\alpha} \ve{n}+\ve[y]{n}$` produces

$$\boldsymbol{\alpha}(x_1, \dots, x_n) + (y_1, \dots, y_n).$$

## **def, newcommand and renewcommand (Ct'd)**

- For `\newcommand`, `cmd` must **not** be already defined.
- For `\renewcommand`, `cmd` must already be defined.
- `\ensuremath` makes sure its argument be always processed in math mode.

## Editors

The choice of editor for tex/latex makes a big difference in time saving (for both inputs and making changes).

Recommended editors:

- `gVim` with special menus.
- `Emacs` with AUCTEX, LIGHTNING COMPLETION and ULTRA-TEX.
- `WinEdit` [Windows shareware only]

## vi (Visual editor) key mapping

Key mapping is used to change the meaning of typed keys.

Syntax:

`:map lhs rhs`

For example:

```
map ,l :!latex %  
map ,b :!bibtex %<  
map ,s :!dvips -o %<.ps %<.dvi  
map ,x :!xdvi %<.dvi &  
map ,p :!pdflatex % &  
map ,f :!xpdf %<.pdf &  
map ,a :!acroread %<.pdf &  
map ,g :!ghostview %<.ps &
```

## vi (Visual editor) key mapping (Ct'd)

- “`map ,l :!latex %`” means that map command “`:!latex %`” to “`,l`”. When you type “`,l`”, gVim will replace it to the command “`:!latex %`”.
- “`:!latex %`” means executing unix command `latex filename` where `filename` is the latex file “`filename.tex`” you are editing by using gVim.
- You can define the mapping in the file `.vimrc` in your home directory.
- To delete mapping, use `unmap`. The syntax is `unmap lhs`. For example `unmap ,l`.

## Abbreviations in gVim

You can define abbreviations in gVim. When you type the abbreviations, gVim will replace the abbreviation with the complete phrase it stands for.

Syntax:

```
:ab lhs rhs
```

For example:

```
:ab iid independent and identically distributed  
:ab wlog without loss of generality
```

When you type `iid` and space key, gVim will automatically replace `iid` with `independent and identically distributed`.

To delete abbreviations, use `una`. The syntax is `una lhs`. For example,  
`:una iid`.



## Transform a table to LaTeX format by gVim

You want to transform:

0.41	-1.00	0.27	0.01	-0.99	0.64
-2.90	0.24	-0.33	0.02	1.37	0.47

into:

```
\begin{center}
\begin{tabular}{|cccccc|} \hline
0.41 & -1.00 & 0.27 & 0.01 & -0.99 & 0.64 \\
-2.90 & 0.24 & -0.33 & 0.02 & 1.37 & 0.47 \\
\end{tabular}
\end{center}
```

To do this, type (while in command mode) “:1,2!table2tex.pl” (suppose that the data are in 1-st and 2-nd rows respectively).

## Transform a table to LaTeX format by gVim (Ct'd)

- You must have the Perl program `table2tex.pl` in your current directory or some directory covered in your `PATH` variable.
- You can obtain the source code of the file `table2tex.pl` at Dr. Harry Joe's webpage <http://hajek.stat.ubc.ca/~harry/local/table2tex.html>

## Transform a table to HTML format by gVim

You want to transform:

0.41	-1.00	0.27	0.01	-0.99	0.64
-2.90	0.24	-0.33	0.02	1.37	0.47

into:

```
<TABLE border=5 cellpadding=5 cellspacing=3>
<TR>
<TD> 0.41</TD><TD> -1.00</TD><TD> 0.27</TD>
<TD> 0.01</TD><TD> -0.99</TD><TD> 0.64</TD>
</TR>
<TR>
<TD> -2.90</TD><TD> 0.24</TD><TD> -0.33</TD>
<TD> 0.02</TD><TD> 1.37</TD><TD> 0.47</TD>
</TR>
</TABLE>
```

To do this, type (while in command mode) “:1,2!table2html.pl” (suppose that the data are in 1-st and 2-nd rows respectively).

## Transform a table to HTML format by gVim (Ct'd)

- You must have the Perl program `table2html.pl` in your current directory or some directory covered in your `PATH` variable.
- You can obtain the source code of the file `table2html.pl` at webpage <http://hajek.stat.ubc.ca/~webmaste/howto/editor/gvim.html>.

## Format (Splus) Output by gVim

Say you have output from S-plus which you want to format a little better ... say you want to get rid of the first and third and last column and make the rest of the columns have 1,2,3,4 decimals points respectively

First, above the output, type something like “0 8.1 0 8.2 8.3 8.4 0” (the 0’s signify getting rid of that column, the a.b format signifies a width of a with b decimal points) Set it up like the following:

```
0 8.1 0 8.2 8.3 8.4 0
[1] -1.35861443  0.30258917 -0.57190688  0.27911763  0.41829340 -0.33448098
[7]  0.14098338  0.26156014 -0.67817408  0.27750322  0.70046690  1.63321662
```

Call “:5,7!format.pl” and it should change the above to:

```
-1.4      -0.57      0.279      0.4183
0.1       -0.68      0.278      0.7005
```

## Format (Splus) Output by gVim (Ct'd)

- You must have the Perl program `format.pl` in your current directory or some directory covered in your `PATH` variable.
- You can obtain the source code of the file `format.pl` at webpage <http://hajek.stat.ubc.ca/~webmaste/howto/editor/gvim.html>.

## Emacs (Editing MACroS)

Many packages are developed for Emacs to facilitate the writing and formatting the  $\text{\LaTeX}$  documents.

- AUCTeX
- Lightning Completion
- Ultra-TeX

## Emacs (Editing MACroS) (Ct'd)

Examples of Ultra-TeX:

- Hit \$ once, Emacs inserts a pair of dollar signs and places the cursor between them.
- Hit \$ twice, Emacs inserts

$\backslash[$

$\backslash]$

and puts the cursor on the blank line.



## Emacs (Editing MACroS) (Ct'd)

- Hit \$ three times, Emacs inserts

```
\begin{equation}\label{}
```

```
\end{equation}
```

and puts the cursor in the argument `\label`.

- Hit ``a`, Emacs inserts `\alpha`.
- Hit ``b`, Emacs inserts `\beta`.

## Rejoinder

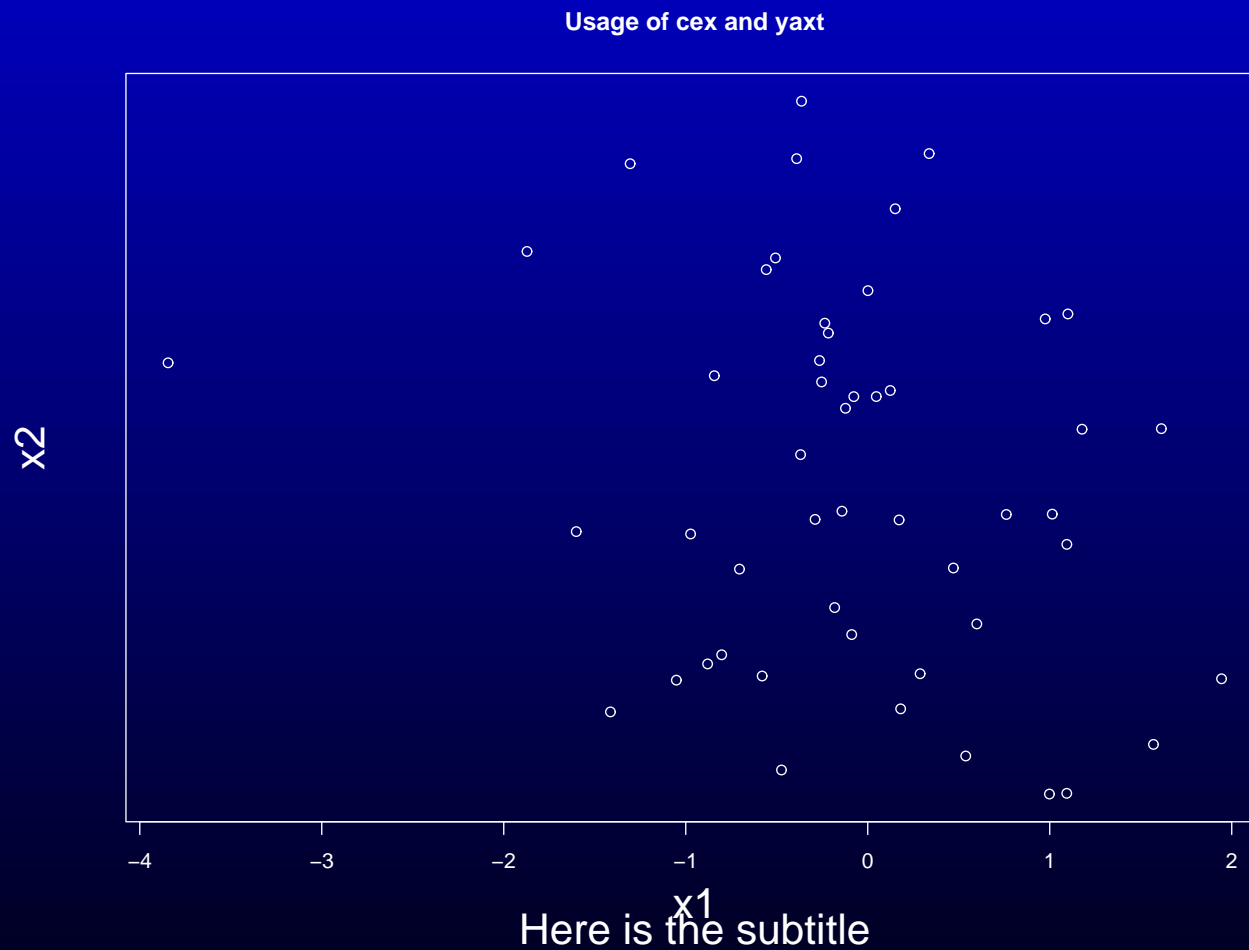
- How to change the text fonts in the title and in the x- and y-labels in R?  
Use `cex.main`, `cex.sub` (for title) and `cex.lab` (for labels).
- How to suppress plotting the numbers along the axes?  
Use `xaxt="n"` for suppressing numbers in the x-axis. Use `yaxt="n"` similarly.

## Rejoinder (Ct'd)

### Example

```
x1<-rnorm(50)
x2<-rnorm(50, mean=5)
postscript("cex.ps",horizontal=T,paper="letter",
  bg="transparent")
par(mfrow=c(1,1), mar=c(6.5, 5.7, 4.1, 2.1), las=0)
plot(x1,x2,cex.lab=2.0, cex.sub=2.0,
  main="Usage of cex and yaxt",
  sub="Here is the subtitle",
  yaxt="n")
dev.off()
system("ps2pdf cex.ps")
```

## Rejoinder (Ct'd)



## Rejoinder (Ct'd)

- How to wrapping text around figures?

You can use either `floatflt` package or `wrapfig` package. The difference between `floatingfigure` and `wrapfigure` is that the text typed after `\end{floatingfigure}` will be forced to be a new paragraph.

Example:

```
\begin{wrapfigure}{r}{0.4\textwidth}
\renewcommand{\captionfont}{\small \bfseries \sffamily}
\centering
\includegraphics[width=0.4\textwidth]{lake1}
\caption{Text wrap around figure}
\noindent \hrulefill
\label{test}
\end{wrapfigure}
```

Wrapfig.sty provides the environments “wrapfigure” and “wraptable” for typesetting a narrow float at the edge of the text, and making the text wrap around it. The “wrapfigure” and “wraptable” environments interact properly with the `\caption` command to produce proper numbering, but they are not regular floats like “figure” and “table”, so (beware!) they may be printed out of sequence with the regular floats. There are four parameters for `\begin{wrapfigure}`, two optional and two required, plus the text of the figure, with a caption perhaps. You must not specify a wrapfigure in any type of list environment or immediately before or immediately after one. It is OK to follow a list if there is a blank line (`\par`) in between. If you put a wrapfigure in a parbox or a minipage, or any other type of grouping, the text wrapping should end before the group does. It does work in two-column format, but are your figures that small? It may be out of sequence with regular floats. The hlines that may be printed above and below floats are ignored; you must insert them manually if desired. `\linewidth` is not adjusted within the wrapped text (because it can only be set for whole paragraphs at a time). It is set within the figure.



**Figure 1: Text wrap around figure**

## References

- [1] Goossens, M., Rahtz, S. with Gurari, E., M., Moore, R., and Sutor, R. S. *The L<sup>A</sup>T<sub>E</sub>X Web Companion*. Addison-Wesley, 1999. 3, 20
- [2] Hutchinson, I. Should i translate tex to html or not? <http://hutchinson.belmont.ma.us/tth/shouldi.html>, March 2003. 5, 6