THE PUBLIC IS MORE FAMILIAR WITH BAD DESIGN THAN GOOD DESIGN. IT IS, IN EFFECT, CONDITIONED TO PREFER BAD DESIGN, BECAUSE THAT IS WHAT IT LIVES WITH. THE NEW BECOMES THREATENING, THE OLD REASSURING.

PAUL RAND, *"DESIGN, FORM, AND CHAOS"*

IN ANYTHING AT ALL, PERFECTION IS FINALLY ATTAINED NOT WHEN THERE IS NO LONGER ANYTHING TO ADD, BUT WHEN THERE IS NO LONGER ANYTHING TO TAKE AWAY, WHEN A BODY HAS BEEN STRIPPED DOWN TO ITS NAKEDNESS.

ANTOINE DE SAINT-EXUPÉRY, *"TERRE DES HOMMES"*

. . . THE DESIGNER OF A NEW SYSTEM MUST NOT ONLY BE THE IMPLEMENTOR AND THE FIRST LARGE-SCALE USER; THE DESIGNER SHOULD ALSO WRITE THE FIRST USER MANUAL. . . . IF I HAD NOT PARTICIPATED FULLY IN ALL THESE ACTIVITIES, LITERALLY HUNDREDS OF IMPROVEMENTS WOULD NEVER HAVE BEEN MADE, BECAUSE I WOULD NEVER HAVE THOUGHT OF THEM OR PERCEIVED WHY THEY WERE IMPORTANT.

DONALD E. KNUTH, *"THE ERRORS OF TEX"*

THE TUFTE-LATEX DEVELOPERS

A TUFTE-STYLE BOOK

MJ PUBLISHING HOUSE

PUBLISHED BY MJ PUBLISHING HOUSE

TODO! Update this URL

TUFTE-LATEX.GITHUB.IO/TUFTE-LATEX/

First printing, March 2025

Contents

List of Figures

List of Tables

Dedicated to those who appreciate LaTeX

and the work of Edward R. Tufte and Donald E. Knuth.

Introduction

This sample book discusses the design of Edward Tufte's books[1] and the use of the `tufte-book` and `tufte-handout` document classes.

Additionally, it discusses changes made to the original Tufte-LaTeX document classes in attempt to modernize them. It also shows how to use the new features of the Tufte-LaTeX document classes. I want to say up front that I am neither a typographer nor a designer, this is my amateur attempt at making this project more accessible. After years of last official update to the Tufte-LaTeX project, it has become a bit outdated and harder to use on modern systems.

I freely admit that some of the changes made here are not in the spirit of the original Tufte-LaTeX project. I have tried to keep the changes as minimal as possible, and provide a way to turn them off if desired. These changes were motivated by my personal needs to make it more accessible to me, and I hope they will be useful to others as well.

[1] Tufte 1990, 1997, 2001, 2006.

The Design of Tufte's Books

The pages of a book are usually divided into three major sections: the front matter (also called preliminary matter or prelim), the main matter (the core text of the book), and the back matter (or end matter).

The front matter of a book refers to all of the material that comes before the main text. The following table from shows a list of material that appears in the front matter of: *The Visual Display of Quantitative Information*, *Envisioning Information*, *Visual Explanations*, and *Beautiful Evidence* along with its page number. Page numbers that appear in parentheses refer to folios that do not have a printed page number (but they are still counted in the page number sequence).

	Books			
Page content	*VDQI*	*EI*	*VE*	*BE*
Blank half title page	(1)	(1)	(1)	(1)
Frontispiece[2]	(2)	(2)	(2)	(2)
Full title page	(3)	(3)	(3)	(3)
Copyright page	(4)	(4)	(4)	(4)
Contents	(5)	(5)	(5)	(5)
Blank page	–	(6)	(6)	(6)
Dedication	(6)	(7)	(7)	7
Blank page	–	(8)	–	(8)
Epigraph	–	–	(8)	–
Introduction	(7)	(9)	(9)	9

Table 1: Overview of pages in front matter of Tufte's books.Page numbers in parentheses refer to folios without printed page numbers.

[2] The contents of this page vary from book to book. In *VDQI* this page is blank; in *EI* and *VE* this page holds a frontispiece; and in *BE* this page contains three epigraphs.

The design of the front matter in Tufte's books varies slightly from the traditional design of front matter. First, the pages in front matter are traditionally numbered with lowercase roman numerals (e.g., i, iv, ix, …). Second, the front matter page numbering sequence is usually separate from the main matter page numbering. That is, the page numbers restart at 1 when the main matter begins. In contrast, Tufte has enumerated his pages with arabic numerals, and share the count sequence with the main matter.

There are also some variations in design across Tufte's four books. The page opposite the full title page (labeled "frontispiece" in table 1)

has different content in every book. In *The Visual Display of Quantitative Information*, this page is blank; in *Envisioning Information* and *Visual Explanations*, this page holds a frontispiece; and in *Beautiful Evidence*, this page contains three epigraphs.

The dedication appears on page 6 in *VDQI* (opposite the introduction), and is placed on its own spread in the other books. In *VE*, an epigraph shares the spread with the opening page of the introduction.

None of the page numbers (folios) of the front matter are expressed except in *BE*, where the folios start to appear on the dedication page.

THE FULL TITLE PAGE of each of the books varies slightly in design. In all the books, the author's name appears at the top of the page, the title it set just above the center line, and the publisher is printed along the bottom margin. Some of the differences are outlined in the following table.

Feature	*VDQI*	*EI*	*VE*	*BE*
Author				
Typeface	serif	serif	serif	sans serif
Style	italics	italics	italics	upright, caps
Size	24 pt	20 pt	20 pt	20 pt
Title				
Typeface	serif	serif	serif	sans serif
Style	upright	italics	upright	upright, caps
Size	36 pt	48 pt	48 pt	36 pt
Subtitle				
Typeface	—	—	serif	—
Style	—	—	upright	—
Size	—	—	20 pt	—
Edition				
Typeface	sans serif	—	—	—
Style	upright, caps	—	—	—
Size	14 pt	—	—	—
Publisher				
Typeface	serif	serif	serif	sans serif
Style	italics	italics	italics	upright, caps
Size	14 pt	14 pt	14 pt	14 pt

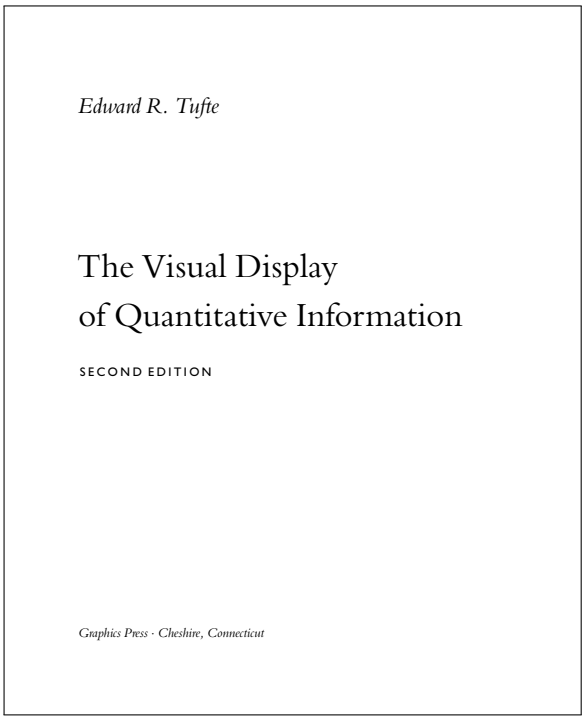
Table 2: Comparison of full title page design features in Tufte's books.

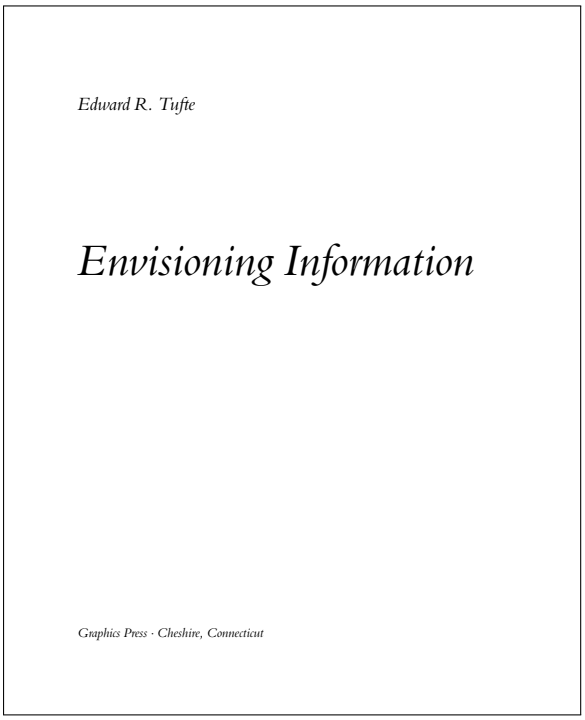
THE TABLES OF CONTENTS in Tufte's books give us our first glimpse of the structure of the main matter. *The Visual Display of Quantitative Information* is split into two parts, each containing some number of chapters. His other three books only contain chapters—they're not broken into parts.

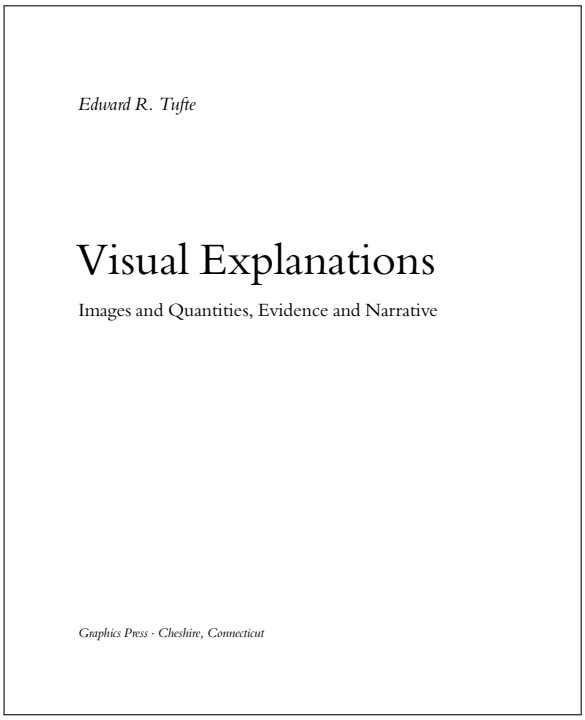
Typefaces

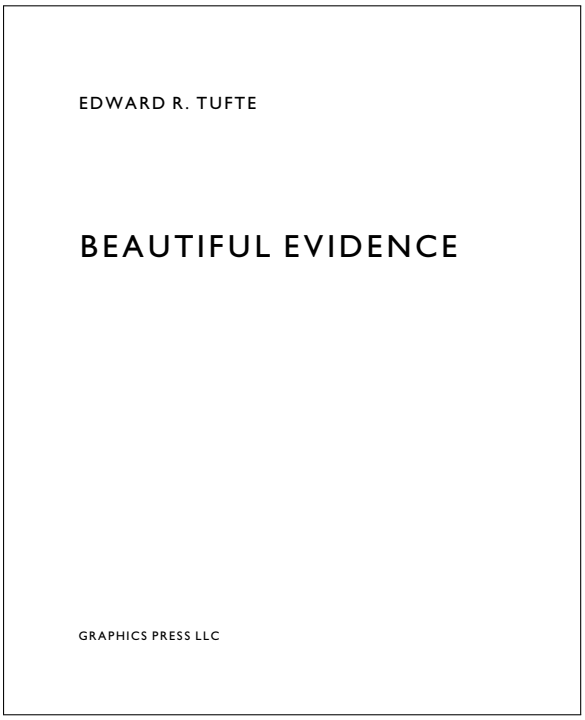
Tufte's books primarily use two typefaces: Bembo and Gill Sans. Bembo is used for the headings and body text, while Gill Sans is used for the title page and opening epigraphs in *Beautiful Evidence*.

Older versions of Tufte-LaTeX used Palatino, Helvetica, and Bera Mono fonts. That's because neither Bembo, Gill Sans nor any substi-


Edward R. Tufte
The Visual Display
of Quantitative Information
SECOND EDITION
Graphics Press · Cheshire, Connecticut


Edward R. Tufte
Envisioning Information
Graphics Press · Cheshire, Connecticut


Edward R. Tufte
Visual Explanations
Images and Quantities, Evidence and Narrative
Graphics Press · Cheshire, Connecticut


EDWARD R. TUFTE
BEAUTIFUL EVIDENCE
GRAPHICS PRESS LLC

Contents

Contents

Contents

Contents

tutes were freely available. Nowadays most LaTeXdistributions have them available. Hence this version of Tufte-LaTeX uses `ETbb` for a Bembo-like serif font, and `gillius` for a Gill-like sans serif font. By default it uses `FiraMono` for monospaced fonts, however provided `custom-tufte-common.tex` overrides it with `RecursiveMono` font. This shows how to use filehooks, override fonts, and shows off my favorite monospaced font.

It also attempts to make files compiled with XeLaTeX, LuaLaTeX, and pdfLaTeXlook as similar as possible. However, there seems to be small differences, which are probably caused by the differences between `fontspec` and `fontenc` packages, and how they handle font encoding/formatting.

LaTeX size	Font size	Leading	Used for
`\tiny`	5	6	sidenote numbers
`\scriptsize`	7	8	—
`\footnotesize`	8	10	sidenotes, captions
`\small`	9	12	quote, quotation, and verse environments
`\normalsize`	10	14	body text
`\large`	11	15	B-heads
`\Large`	12	16	A-heads, TOC entries, author, date
`\LARGE`	14	18	handout title
`\huge`	20	30	chapter heads
`\Huge`	24	36	part titles

Table 3: A list of LaTeX font sizes as defined by the Tufte-LaTeX document classes.

Headings

Tufte's books include the following heading levels: parts, chapters,[3] sections, subsections, and paragraphs. By default subsubsection and subparagraph headings are not defined in the Tufte-LaTeX classes.[4]

[3] Parts and chapters are defined for the `tufte-book` class only.

[4] For more information on this topic, see (Bringhurst 2005), section 4.2.2

Paragraph Paragraph headings (as shown here) are introduced by italicized text and separated from the main paragraph by a bit of space.

Heading	Style	Size
Part	roman	24/36×40 pc
Chapter	italic	20/30×40 pc
Section	italic	12/16×26 pc
Subsection	italic	11/15×26 pc
Paragraph	italic	10/14

Table 4: Heading styles used in *Beautiful Evidence*.

Environments

The following table lists characteristics defined for the various environments:

Environment	Font size	Notes
Body text	10/14×26 pc	
Block quote	9/12×24 pc	Block indent (left and right) by 1 pc
Sidenotes	8/10×12 pc	Sidenote number is set inline, followed by word space
captions	8/10×12 pc	

Table 5: Environment styles used in *Beautiful Evidence*.

On the Use of the `tufte-book` Document Class

The Tufte-LaTeX document classes define a style similar to the style Edward Tufte uses in his books and handouts. Tufte's style is known for its extensive use of sidenotes, tight integration of graphics with text, and well-set typography. This document aims to be at once a demonstration of the features of the Tufte-LaTeX document classes and a style guide to their use.

Page Layout

Headings

This style provides A- and B-heads (that is, `\section` and `\subsection`), demonstrated above.

If you need more than two levels of section headings, you'll have to define them yourself. This class does not provide pre-defined styles for `\subsubsection` or `subparagraph`. As Bringhurst points out in *"The Elements of Typographic Style"*,[5] you should "use as many levels of headings as you need: no more, and no fewer".

The Tufte-LaTeX classes will emit an error if you try to use `\subsubsection` or `\subparagraph`.

[5] Bringhurst 2005.

In his later books,[6] Tufte starts each section with a bit of vertical space, a non-indented paragraph,and sets the first few words of the sentence in small caps. To accomplish this using this style, use the `\newthought` command:

[6] Tufte 2006.

```
\newthought{In his later books}, Tufte starts...
```

Sidenotes

One of the most prominent and distinctive features of this style is the extensive use of sidenotes. There is a wide margin to provide ample room for sidenotes and small figures. Any `\footnotes` will automatically be converted to sidenotes.[7] If you'd like to place ancillary information in the margin without the sidenote mark (the superscript number), you can use the `\marginnote` command.

[7] This is a sidenote that was entered using the `footnote` command.

This is a margin note. Notice that there isn't a number preceding the note, and there is no number in the main text where this note was written.

The specification of the `\sidenote` command is:

`\sidenote[`⟨*number*⟩`][`⟨*offset*⟩`]{`*Sidenote text.*`}`

Both the ⟨*number*⟩ and ⟨*offset*⟩ arguments are optional. If you provide a ⟨*number*⟩ argument, then that number will be used as the sidenote

number. It will change the number of the current sidenote only and will not affect the numbering sequence of subsequent sidenotes.

Sometimes a sidenote may run over the top of other text or graphics in the margin space. If this happens, you can adjust the vertical position of the sidenote by providing a dimension in the ⟨*offset*⟩ argument. Some examples of valid dimensions are:

```
1.0in    2.54cm    254mm    6\baselineskip
```

If the dimension is positive, it will push the sidenote down the page; if the dimension is negative, it will pull the sidenote up the page.

While both the ⟨*number*⟩ and ⟨*offset*⟩ arguments are optional, they must be provided in order. To adjust the vertical position of the sidenote while leaving the sidenote number alone, use the following syntax:

```
\sidenote[][⟨offset⟩]{Sidenote text.}
```

The empty brackets tell the `\sidenote` command to use the default sidenote number.

If you *only* want to change the sidenote number, however, you may completely omit the ⟨*offset*⟩ argument:

```
\sidenote[⟨number⟩]{Sidenote text.}
```

The `\marginnote` command has a similar *offset* argument:

```
\marginnote[⟨offset⟩]{Margin note text.}
```

References

References are placed alongside their citations as sidenotes, as well. This can be accomplished using the normal `\cite` command or the `\autocite` command, which functions similarly.[8]

You will need to specify a bibliography resource file in the preamble of your document using `\addbibresource`. The complete list of references may also be printed automatically by using the `\printbibliography` command. See the end of this document for an example, and the BibLaTeX documentation for more information. Bibliography can be turned off with the help of `\nobibliography` command.

To enter multiple citations at one location,[9] you can provide a list of keys separated by commas: `\cite{Tufte2006,Tufte1990}`.

```
\cite{bibkey1,bibkey2,...}
```

In the new version of Tufte-LaTeX, it's impossible to offset citations the same way sidenotes can be offset. This is caused by change from `natbib` and `bibtex` to `biblatex` and `biber` packages. This switch is motivated by the fact that `natbib` is mostly kept on life support, and `biblatex` is more powerful and flexible. However it has its own optional arguments for `\cite` commands, which are kept unchanged to avoid confusion.

[8] If you use the `cite` command within a sidenote, it will render as an in-line parenthetical citation, as demonstrated here (Tufte 2001).

[9] Tufte 1990, 2006.

Figures and Tables

Images and graphics play an integral role in Tufte's work. In addition to the standard `figure` and `tabular` environments, this style provides special figure and table environments for full-width floats.

Full page–width figures and tables may be placed in `figure*` or `table*` environments. To place figures or tables in the margin, use the `marginfigure` or `margintable` environments as follows (see figure 1):

```
\begin{marginfigure}
  \includegraphics{helix}
  \caption{This is a margin figure.}
  \label{fig:marginfig}
\end{marginfigure}
```

The `marginfigure` and `margintable` environments accept an optional parameter ⟨*offset*⟩ that adjusts the vertical position of the figure or table. See the "Sidenotes" section above for examples. The specifications are:

```
\begin{marginfigure}[⟨offset⟩]
  ...
\end{marginfigure}

\begin{margintable}[⟨offset⟩]
  ...
\end{margintable}
```

Figure 2 is an example of the `figure*` environment and figure 3 is an example of the normal `figure` environment.

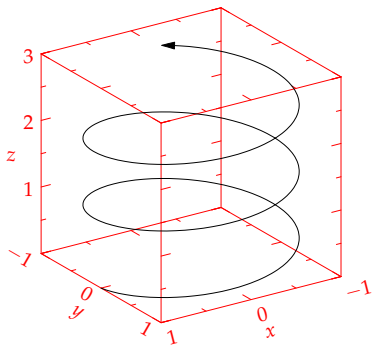


Figure 1: This is a margin figure. The helix is defined by $x = \cos(2\pi z)$, $y = \sin(2\pi z)$, and $z = [0, 2.7]$. The figure was drawn using Asymptote (`http://asymptote.sf.net/`).

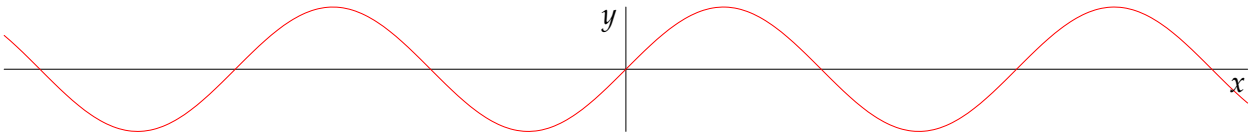


Figure 2: This graph shows $y = \sin x$ from about $x = [-10, 10]$. *Notice that this figure takes up the full page width.*

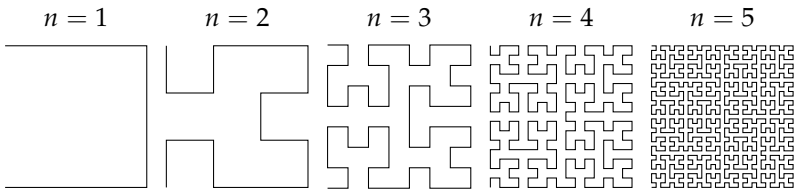


Figure 3: Hilbert curves of various degrees n. *Notice that this figure only takes up the main textblock width.*

As with sidenotes and marginnotes, a caption may require vertical adjustment. The `\caption` command can take a second optional argument which enables you to do this by providing a dimension ⟨*offset*⟩. You may specify the caption in any one of the following forms:

```
\caption{long caption}
\caption[short caption]{long caption}
\caption[][⟨offset⟩]{long caption}
\caption[short caption][⟨offset⟩]{long caption}
```

A positive ⟨*offset*⟩ will push the caption down the page. The short caption, if provided, is what appears in the list of figures/tables, otherwise the "long" caption appears there. Note that although the arguments ⟨*short caption*⟩ and ⟨*offset*⟩ are both optional, they must be provided in order. Thus, to specify an ⟨*offset*⟩ without specifying a ⟨*short caption*⟩, you must include the first set of empty brackets `[]`, which tell `\caption` to use the default "long" caption. As an example, the caption to figure 3 above was given in the form

```
\caption[Hilbert curves...][1em]{Hilbert curves...}
```

Note that caption offset is not available for `marginfigure` and `margintable` environments. In these cases you need to offset the whole figure or table. Captions in `marginfigure` and `margintable` still support short captions.

Table 6 shows table created with the `booktabs` package. Notice the lack of vertical rules—they serve only to clutter the table's data. Hence Tufte style tables use only horizontal rules. In cases where a table has many rows, one can use `colortbl` to make rows stand out visually from each other. Colors can be used to group related rows, highlight important data, or make one row stand out from the others.

Margin	Length
Paper width	8 1/2 inches
Paper height	11 inches
Textblock width	6 1/2 inches
Textblock/sidenote gutter	3/8 inches
Sidenote width	2 inches

Table 6: Here are the dimensions of the various margins used in the Tufte-handout class.

Too Many Floats

Occasionally LaTeX will generate an error message:

```
Error:  Too many unprocessed floats
```

LaTeX tries to place floats in the best position on the page. Until it's finished composing the page, however, it won't know where those positions are. If you have a lot of floats on a page (including sidenotes, margin notes, figures, tables, etc.), LaTeX may run out of "slots" to keep track of them and will generate the aforementioned error.

LaTeX initially allocates 18 slots for storing floats. To work around this limitation, the Tufte-LaTeX document classes provide a `\morefloats` command that will reserve more slots.

The first time `\morefloats` is called, it allocates an additional 34 slots. The second time `\morefloats` is called, it allocates another 26 slots.

The `\morefloats` command may only be used two times. Calling it a third time will generate an error message. (This is because allocating more floats may lead to LaTeX running out of memory.)

If, after using the `\morefloats` command twice, you continue to get the `Too many unprocessed floats` error, there are a couple things you can do.

The `\FloatBarrier` command will immediately process all the floats before typesetting more material. Since `\FloatBarrier` will start a new paragraph, you should place this command at the beginning or end of a paragraph.

The `\clearpage` command will also process the floats before continuing, but instead of starting a new paragraph, it will start a new page.

You can also try moving your floats around a bit: move a figure or table to the next page or reduce the number of sidenotes. (Each sidenote actually uses *two* slots.)

After placing the floats, LaTeX will mark those slots as unused so they are available for the next page to be composed.

Captions

You may notice that the captions are sometimes misaligned. Due to the way LaTeX's floats works, it's hard to know for sure where it decided to put the float. Therefore, the Tufte-LaTeX document classes provide commands to override the caption position.

Vertical alignment To override the vertical alignment, use the `\setfloatalignment` command inside the float environment. For example:

```
\begin{figure}[btp]
  \includegraphics{sine.pdf}
  \caption{This is an example of a sine wave.}%
  \label{fig:sinewave}
  \setfloatalignment{b} % forces caption to be bottom-aligned
\end{figure}
```

The syntax of the `\setfloatalignment` command is:

```
\setfloatalignment{⟨pos⟩}
```

where ⟨*pos*⟩ can be either `b` for bottom-aligned captions, or `t` for top-aligned captions.

Horizontal alignment To override the horizontal alignment, use either the `\forceversofloat` or the `\forcerectofloat` command inside of the float environment. Note that these commands only work when the `symmetric` option is enabled. For example:

```
\begin{figure}[btp]
  \includegraphics{sine.pdf}
  \caption{This is an example of a sine wave.}%
  \label{fig:sinewave}
  \forceversofloat % forces caption to be set to the left of the float
\end{figure}
```

The `\forceversofloat` command causes the algorithm to assume the float has been placed on a verso page—that is, a page on the left side of a two-page spread. Conversely, the `\forcerectofloat` command causes the algorithm to assume the float has been placed on a recto page—that is, a page on the right side of a two-page spread.

Full-width text blocks

In addition to the new float types, there is a `fullwidth` environment. This environment stretches across the main text block and the sidenotes area.

```
\begin{fullwidth}
  Lorem ipsum dolor sit amet...
\end{fullwidth}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Typography

Typefaces

When using XeLaTeX or LuaLaTeX, the Tufte-LaTeX classes will load the `fontspec` package. This package allows you to set the typeface to any installed font, any local font files, or to any font files you have installed in your `texmf` tree.

By default the Tufte-LaTeX classes will use the ET-Bembo font from the `ETbb` package, as the main typeface. If it's unavailable, the TeX Gyre Pagella from the `tex-gyre-pagella` package will be used as fallback serif font. For math fonts it tries to use the Palatino font from the `mathpazo` package. For sans serif text the Gillius No. 2 font from the `gillius` package will be used. If this one is unavailable, the TeX Gyre Heros font from the `tex-gyre-heros` package will be used. In case of monospaced text the `Fira Mono` font from the `FiraMono` package will be used. If it's not present, the TeX Gyre Cursor font from the `tex-gyre-cursor` package will be used. However the provided `custom-tufte-common.tex` filehook overrides the default monospaced font with `RecursiveMono` font. This file shows how you can override the default fonts, and how the filehooks can be used.

The TeX Gyre faces are usually included with TeX Live distributions, hence why they are used as fallback fonts. If any of the selected fonts don't suit you, you can easily change them using the `fontspec` package.

Under pdfLaTeX, the Tufte-LaTeX classes will try to use the same default fonts, but will fall back to the default Computer Modern fonts if they are not available. The `fontspec` package is not available under pdfLaTeX, so it uses the `fontenc` package to set the font encoding.

In cases where `nofonts` option is used, the Tufte-LaTeX classes will not load any fonts. It will not load `fontspec` or `fontenc` packages either. However if the `nols` was **not** used, `fontspec` will be loaded in LuaLaTeXor XeLaTeX engines, as they require it to set letterspacing.

Letterspacing

This document class includes two new commands and some improvements on existing commands for letterspacing.

When setting strings of ALL CAPS or SMALL CAPS, the letterspacing—that is, the spacing between the letters—should be increased slightly.[10] The `\allcaps` command has proper letterspacing for strings of FULL CAPITAL LETTERS, and the `\smallcaps` command has letterspacing for SMALL CAPITAL LETTERS. These commands will also automatically convert the case of the text to upper- or lowercase, respectively.

[10] Bringhurst 2005.

The `\textsc` command has also been redefined to include letterspacing. However, the case of the `\textsc` argument is left as is. This allows one to use both uppercase and lowercase letters: THE INITIAL LETTERS OF THE WORDS IN THIS SENTENCE ARE CAPITALIZED.

Document Class Options

The `tufte-book` class is based on the LaTeX `book` document class. Conversely the `tufte-handout` class is based on the `article` document class. Therefore, you can pass any of the typical book or article options to them. There are a few options that are specific to the `tufte-book` and `tufte-handout` document classes, however.

Paper Size and Layout Options

The `a4paper` option will set the paper size to A4 instead of the default US letter size.

The `b5paper` option will set the paper size to B5 instead of the default US letter size.

The `a5paper`, `executivepaper`, and `legalpaper` options are unavailable in the Tufte-LaTeX classes.

The `twoside` option will modify the running heads so that the page number is printed on the outside edge. By default Tufte-LaTeX classes always print the page number on the right-side edge.

The `symmetric` option typesets the sidenotes on the outside edge of the page. This is how books are traditionally printed, but is contrary to Tufte's book design which sets the sidenotes on the right side of the page. This option implicitly sets the `twoside` option.

The `landscape`, `onecolumn`, and `twocolumn` options are not available in the Tufte-LaTeX classes.

Font and Text Options

The `sftitle` option will set the title page and title block in a sans serif typeface. The `nosftitle` option will set the title page and title block in a serif typeface. In case of `tufte-handout` these options also have an effect on abstract. By default the `tufte-book` class uses `sftitle` and the `tufte-handout` class uses `nosftitle`.

The `sfmarginals` option makes all marginals use sans serif typeface

instead of the default roman typeface.

The `justified` option sets all the text fully justified (flush left and right). The default is to set the text ragged right. The body text of Tufte's books are set ragged right. This prevents needless hyphenation and makes it easier to read the text in the slightly narrower column.

The `10pt`, `11pt`, and `12pt` options are unavailable in the Tufte-LaTeX classes.

The `nofonts` option prevents the Tufte-LaTeX classes from automatically loading the Tufte typefaces You should use this option if you wish to load your own fonts in pdfLaTeX. If you're using XeLaTeX or LuaLaTeX, the will not be loaded, and you can use `fontspec` to set your own. If you aren't using the `nols` option, the `fontspec` package will be loaded as it is required for letterspacing.

The `nols` option inhibits the letterspacing code. The Tufte-LaTeX classes try to load the appropriate letterspacing package to adjust spacing of letters. It uses `letterspace` or the `soul` under pdfTeX. In case of XeLaTeX and LuaLaTeXit uses `fontspec`.

The `bidi` option loads the `bidi` package which is used with XeLaTeX to typeset bi-directional text. Since the `bidi` package needs to be loaded before the sidenotes and cite commands are defined, it can't be loaded in the document preamble.

Title Page Options

The `notitlepage` option causes `\maketitle` to generate a title block instead of a title page. By default the `tufte-book` class uses `titlepage` and the `tufte-handout` class uses `notitlepage`. There is an analogous `titlepage` option that forces `\maketitle` to generate a full title page instead of the title block.

Toggle Options

The `nobib` option inhibits loading of the `natbib` and `bibtex` packages and modification of the `\cite` command.

The `notoc` option suppresses Tufte-LaTeX's custom table of contents (TOC) design. The current TOC design only shows unnumbered chapter titles in books; it doesn't show sections or subsections. The `notoc` option will revert to LaTeX's TOC design.

The `nohyper` option prevents the `hyperref` package from being loaded. The default is to load the `hyperref` package and use the `\title` and `\author` contents as metadata for the generated PDF.

The `nomoderntitles` is a new option added in the latest version of Tufte-LaTeX. It only works in the `tufte-handout` class. It disables coloring and styling of the section and paragraph titles. The default is to color the titles and add a colored box to the left with section numbers.

Marginal Options

In the Tufte-LaTeX classes there are four types of marginal materials, which are: `sidenote`, `marginnote`, `caption`, and `citation`. Each of

those can have their justification set to one of the following options:

justified Fully justifies the text (sets it flush left and right).

raggedleft Sets the text ragged left.

raggedright Sets the text ragged right.

raggedouter Sets the text ragged left if on the left-hand (verso) page, otherwise ragged right. This is useful in conjunction with the `symmetric` document class option.

auto Fully justifies the text if `justified` class option was specified, otherwise the text is set ragged right. This is the default justification option for marginal material.

Additionally, the `marginals` option can be used to set the justification settings for all marginal texts. See the Customizing Marginal Material section for more information on marginal material.

Debugging Options

The `debug` option causes the Tufte-LaTeX classes to output debug information to the log file which is useful in troubleshooting bugs. It prints list of options and their values under the `Tufte-LaTeX settings` section. It will also cause the graphics to be replaced by outlines. When combined with `\geometry`⟨*showframe*⟩ command it will show margins for debugging page layout issues.

Customizing Tufte-LaTeX

The Tufte-LaTeX document classes are designed to closely emulate Tufte's book design by default. However, each document is different and you may encounter situations where the default settings are insufficient. This chapter explores many of the ways you can adjust the Tufte-LaTeX document classes to better fit your needs.

File Hooks

When creating many documents using the Tufte-LaTeX classes, it's easier to store customizations in one file. Otherwise they would need to be copied into the preamble of each document. The Tufte-LaTeX classes provide three file hooks: `custom-tufte-common.tex`, `custom-tufte-book.tex`, and `custom-tufte-handout.tex`.

`custom-tufte-common.tex` If this file exists, it will be loaded by all of the Tufte-LaTeX document classes, just prior to any class-specific code. If your customizations or code should be included in both the book and handout classes, use this file hook.

`custom-tufte-book.tex` If this file exists, it will be loaded after all of the common and book-specific code has been read. If your customizations apply only to the book class, use this file hook.

`custom-tufte-handout.tex` If this file exists, it will be loaded after all of the common and handout-specific code has been read. If your customizations apply only to the handout class, use this file hook.

Numbered Section Headings

While Tufte dispenses with numbered headings in his books, if you require them, they can be enabled by changing the value of the `secnumdepth`counter. From the table below, select the heading level at which numbering should stop and set the `secnumdepth` counter to that value. For example, if you want parts and chapters numbered, but don't want numbering for sections or subsections, use the command:

```
\setcounter{secnumdepth}{0}
```

The default value of `secnumdepth` for the `tufte-book` class is -1. This version of `tufte-handout` class sets the counter to 2 so sections and subsections are numbered. This change was made to make the sections

stand out more as I found it hard to distinguish them from the body text. If you wish to revert to no numbering, set the counter to −1. You can also pass the `nomoderntitles` option to the `tufte-handout` class to disable the coloring and styling of the section and paragraph titles.

Heading level	Value
Part (in `tufte-book`)	−1
Part (in `tufte-handout`)	0
Chapter (only in `tufte-book`)	0
Section	1
Subsection	2
Subsubsection	3
Paragraph	4
Subparagraph	5

Table 7: Heading levels used with the `secnumdepth` counter.

Changing the Paper Size

The Tufte-LaTeX classes currently only provide three paper sizes: A4, B5, and US letter. To specify a different paper size (and/or margins), use the `\geometry` command in the preamble of your document (or one of the file hooks). The full documentation of the `\geometry` command may be found in the `geometry` package documentation.[11]

[11] Umeki 2008.

Customizing Marginal Material

Marginal material includes sidenotes, citations, margin notes, and captions. Normally, the justification of the marginal material follows the justification of the body text. If you specify the `justified` document class option, all of the margin material will be fully justified as well. If you don't specify the `justified` option, then the marginal material will be set ragged right.

You can set the justification of the marginal material separately from the body text using the following document class options: `sidenote`, `marginnote`, `caption`, `citation`, and `marginals`. Each option refers to its obviously corresponding marginal material type. The `marginals` option simultaneously sets the justification on all four marginal material types.

Each of the document class options takes one of five justification types:

`justified` Fully justifies the text (sets it flush left and right).

`raggedleft` Sets the text ragged left, regardless of which page it falls on.

`raggedright` Sets the text ragged right, regardless of which page it falls on.

`raggedouter` Sets the text ragged left if it falls on the left-hand (verso) page of the spread and otherwise sets it ragged right. This is useful in conjunction with the `symmetric` document class option.

auto If the `justified` document class option was specified, then set the text fully justified; otherwise the text is set ragged right. This is the default justification option if one is not explicitly specified.

For example,

```
\documentclass[symmetric,justified,marginals=raggedouter]{tufte-book}
```

will set the body text of the document to be fully justified. All of the margin material (sidenotes, margin notes, captions, and citations) to be flush against the body text with ragged outer edges.

THE FONT AND STYLE of the marginal material may also be modified using the following commands:

```
\setsidenotefont{⟨font commands⟩}
\setcaptionfont{⟨font commands⟩}
\setmarginnotefont{⟨font commands⟩}
\setcitationfont{⟨font commands⟩}
```

The `\setsidenotefont` sets the font and style for sidenotes, the `\setcaptionfont` for captions, the `\setmarginnotefont` for margin notes, and the `\setcitationfont` for citations. The ⟨*font commands*⟩ can contain font size changes (e.g., `\footnotesize`, `\Huge`, etc.), font style changes (e.g., `\sffamily`, `\ttfamily`, `\itshape`, etc.), color changes (e.g., `\color{tufte-blue}`), and many other adjustments.

If, for example, you wanted the captions to be set in italic sans serif, you could use:

```
\setcaptionfont{\itshape\sffamily}
```

Compatibility Issues

When switching an existing document from one document class to a Tufte-LaTeX document class, a few changes to the document may have to be made.

Converting from `article` to `tufte-handout`

The following `article` class options are unsupported: `10pt`, `11pt`, `12pt`, `a5paper`, `b5paper`, `executivepaper`, `legalpaper`, `landscape`, `onecolumn`, and `twocolumn`.

The following headings are not supported: `\subsubsection` and `\subparagraph`.

Converting from `book` to `tufte-book`

The following `book` class options are unsupported: `10pt`, `11pt`, `12pt`, `a5paper`, `b5paper`, `executivepaper`, `legalpaper`, `landscape`, `onecolumn`, and `twocolumn`.

The following headings are not supported: `\subsubsection` and `\subparagraph`.

Troubleshooting and Support

Tufte-LaTeX Website

The website for the Tufte-LaTeX packages is located at `https://github.com/Tufte-LaTeX/tufte-latex`. On that website, you'll find links to the GIT repository, mailing lists, bug tracker, and documentation.

However as the project seems to be abandoned as of time of writing, the website may not be available in the future. Additionally some of the links there seem to have already been victim of link rot. You can find more help and information on the current development of the Tufte-LaTeX classes at the my GitHub repository. TODO! Link to the new git repository

Tufte-LaTeX Mailing Lists

There is only one surviving mailing list for the Tufte-LaTeX project:

Discussion list The `tufte-latex` discussion list is for asking questions, getting assistance with problems, and help with troubleshooting. Release announcements were also posted to this list. You can subscribe to the `tufte-latex` discussion list at `http://groups.google.com/group/tufte-latex`.

Commits list The `tufte-latex-commits` list used to exist as well as a read-only mailing list. Messages were sent to the list any time the Tufte-LaTeX code had been updated. This list was available at `http://groups.google.com/group/tufte-latex-commits`.

A more modern way to keep up with the development of the Tufte-LaTeX classes is to follow the GitHub repository. You can also open issues there if you encounter any problems or have suggestions for improvements. TODO! Link to the new git repository

Getting Help

If you've encountered a problem with one of the Tufte-LaTeX document classes, have a question, or would like to report a bug, please create an issue on the GitHub repository.

To help with troubleshooting the problem more quickly, please try to compile your document using the `debug` class option and include

the generated `.log` file in the issue, along with a brief description of the problem.

Errors, Warnings, and Informational Messages

The following is a list of all of the errors, warnings, and other messages generated by the Tufte-LaTeX classes and a brief description of their meanings.

```
Error:  \subparagraph is undefined by this class.
```

The `\subparagraph` command is not defined in the Tufte-LaTeX document classes. If you'd like to use the `\subparagraph` command, you'll need to redefine it yourself. See the Headings section on page 19 for a description of the heading styles available in the Tufte-LaTeX document classes.

```
Error:  \subsubsection is undefined by this class.
```

The `\subsubsection` command is not defined in the Tufte-LaTeX document classes. If you'd like to use the `\subsubsection` command, you'll need to redefine it yourself. See the Headings section on page 19 for a description of the heading styles available in the Tufte-LaTeX document classes.

```
Error:  You may only call \morefloats twice.  See the
        Tufte-LaTeX documentation for other workarounds.
```

LaTeX allocates 18 slots for storing floats. The first time `\morefloats` is called, it allocates an additional 34 slots. The second time `\morefloats` is called, it allocates another 26 slots.

The `\morefloats` command may only be called two times. Calling it a third time will generate this error message. See the Too Many Floats section on page 24 for more information.

```
Warning:  Option '⟨class option⟩' is not supported -- ignoring option.
```

This warning appears when you've tried to use ⟨*class option*⟩ with a Tufte-LaTeX document class, but ⟨*class option*⟩ isn't supported by the Tufte-LaTeX document class. In this situation, ⟨*class option*⟩ is ignored.

```
Info:  The ' symmetric' option implies ' twoside'
```

You specified the `symmetric` document class option. This option automatically forces the `twoside` option as well. See page 27 for more information on the `symmetric` class option.

Package Dependencies

The following is a list of packages that the Tufte-LaTeX document classes rely upon. Packages marked with an asterisk are optional.

- xifthen
- iftex*
- hyperref
- geometry
- ragged2e
- chngpage *or* changepage
- paralist
- textcase
- soul*
- letterspace*
- setspace
- biblatex *and* biber
- optparams
- placeins
- mathpazo*
- helvet*
- fontenc
- beramono*
- fancyhdr
- xcolor
- textcomp
- titlesec
- titletoc
- fontspec*
- etbb*

Bibliography

Bringhurst, Robert (2005). *The Elements of Typography*. 3.1. Vancouver, British Columbia: Hartley & Marks. ISBN: 0-88179-205-5 (cit. on pp. 19, 21, 27).

Tufte, Edward R. (1990). *Envisioning Information*. Cheshire, Connecticut: Graphics Press. ISBN: 0-9613921-1-8 (cit. on pp. 13, 22).

– (1997). *Visual Explanations*. Cheshire, Connecticut: Graphics Press. ISBN: 0-9613921-2-6 (cit. on p. 13).

– (2001). *The Visual Display of Quantitative Information*. Cheshire, Connecticut: Graphics Press. ISBN: 0-9613921-4-2 (cit. on pp. 13, 22).

– (May 2006). *Beautiful Evidence*. First. Cheshire, Connecticut: Graphics Press, LLC. ISBN: 0-9613921-7-7 (cit. on pp. 13, 21, 22).

Umeki, Hideo (Dec. 2008). *The `geometry` package*. `http://ctan.org/pkg/geometry` (cit. on p. 32).

Index