



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

Copyright © 2007—2019 by Kevin Godby, Bil Kleb, and Bill Wood.  
Copyright © 2025—2025 by Daniel Aaron Salwerowicz.

PUBLISHED BY MJ PUBLISHING HOUSE

<https://salwerowi.cz/modernized-tufte-latex>

Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*First printing, March 2025*

# *Contents*

<i>The Design of Tufte's Books</i>	15
<i>On the Use of the tufte-book Document Class</i>	21
<i>Customizing Tufte-LaTeX</i>	31
<i>New Features in Tufte-LaTeX Classes</i>	35
<i>Installing and Compiling Tufte-LaTeX</i>	39
<i>Compatibility Issues</i>	43
<i>Troubleshooting and Support</i>	45
<i>Index</i>	51



## *List of Figures*

1	This is an example of a margin figure	23
2	Sine graph showcasing full width figure environment	24
3	Hilbert curves of various degrees $n$	24
4	Comparison between modern and old section styles	37





## *List of Tables*

1	Overview of pages in front matter of Tufte’s books.	15
2	Comparison of full title page design features in Tufte’s books.	16
3	A list of L <sup>A</sup> T <sub>E</sub> X font sizes as defined by the Tufte-L <sup>A</sup> T <sub>E</sub> X document classes.	19
4	Heading styles used in <i>Beautiful Evidence</i> .	19
5	Environment styles used in <i>Beautiful Evidence</i> .	20
6	Dimensions of the margins in tufte-handout	25
7	Heading levels used with the secnumdepth counter.	32



*Dedicated to those who appreciate  $\text{\LaTeX}$   
and the work of Edward R. Tufte and Donald E. Knuth.*



## *Introduction*

This sample book discusses the design of Edward Tufte's books<sup>1</sup> and the use of the `tufte-book` and `tufte-handout` document classes.

<sup>1</sup> Tufte 1990, 1997, 2001, 2006.

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.



# 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 corresponding 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).

Page content	Books			
	<i>VDQI</i>	<i>EI</i>	<i>VE</i>	<i>BE</i>
Blank half title page	(1)	(1)	(1)	(1)
Frontispiece <sup>2</sup>	(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.

<sup>2</sup> 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, xiv, ...). 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. This book follows the style of *BE*, with folios on dedication and introduction pages. If that’s not desired, use the “`\thispagestyle{empty}`” command on the dedication and introduction pages.

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 is 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	<i>VDQI</i>	<i>EI</i>	<i>VE</i>	<i>BE</i>
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 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.

The following pages show examples of all four books’ title pages and tables of contents. As you can see this sample document does its best to mimic the design of *Beautiful Evidence*, but it’s not perfect. The main difference is caused by the fact that the fonts used in it are not freely available, and Tufte- $\LaTeX$  uses substitutes.



*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

PART I GRAPHICAL PRACTICE

1	<i>Graphical Excellence</i>	13
2	<i>Graphical Integrity</i>	53
3	<i>Sources of Graphical Integrity and Sophistication</i>	79

PART II THEORY OF DATA GRAPHICS

4	<i>Data-Ink and Graphical Roleplay</i>	94
5	<i>Chartjunk: Visions, Grids, and Dudes</i>	107
6	<i>Data-Ink Maximization and Graphical Design</i>	123
7	<i>Multifunctioning Graphical Elements</i>	139
8	<i>Data Density and Small Multiples</i>	166
9	<i>Aesthetics and Technique in Data Graphical Design</i>	177
	<i>Epilogue: Designs for the Display of Information</i>	194

Contents

ESCAPING FLATLAND	12
MICRO/MACRO READINGS	37
LAYERING AND SEPARATION	53
SMALL MULTIPLES	67
COLOR AND INFORMATION	81
NARRATIVES OF SPACE AND TIME	97
EPILOGUE	121

Contents

<i>Images and Quantities</i>	13
<i>Visual and Statistical Thinking: Displays of Evidence for Making Decisions</i>	27
<i>Explaining Magic: Pictorial Instructions and Disinformation Design</i>	55
<i>The Smallest Effective Difference</i>	73
<i>Parallelism: Repetition and Change, Comparison and Surprise</i>	79
<i>Multiples of Space and Time</i>	105
<i>Visual Confections: Juxtapositions from the Ocean of the Streams of Story</i>	121

Contents

<i>Mapped Pictures: Images as Evidence and Explanation</i>	12
<i>Sparklines: Intense, Simple, Word-Sized Graphics</i>	46
<i>Links and Causal Arrows: Ambiguity in Action</i>	64
<i>Words, Numbers, Images — Together</i>	82
<i>The Fundamental Principles of Analytical Design</i>	122
<i>Corruption in Evidence Presentations: Effects Without Causes, Cherry Picking, Overreaching, Chartjunk, and the Rage to Conclude</i>	140
<i>The Cognitive Style of PowerPoint: Pitching Out Corrupts Within</i>	156
<i>Sculptural Pedestals: Meaning, Practice, Depedestalization</i>	186
<i>Landscape Sculptures</i>	196

## 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. This version of Tufte- $\LaTeX$  uses `ETbb` for a Bembo-like font, `mathpazo` for Palatino math font, `gillius` for a Gill Sans-like font, and `FiraMono` as a monospaced font.

However this sample book overrides the default `FiraMono` it with `RecursiveMono` font. It uses the provided `custom-tufte-common.tex` file hook to override the default font. I have done that because I like the `Recursive Mono` font better. Sadly `Recursive` font is not available on CTAN, therefore I have used `FiraMono` as a default font. If you wish to use `Recursive` font, you can use font files provided in this package in the `fonts` directory.

While Tufte- $\LaTeX$  class tries to make documents compiled with  $\XeLaTeX$ ,  $\LuaLaTeX$ , and  $\pdfLaTeX$  look as similar as possible, it’s impossible to make them look identical. There seem to be small differences between how the engines typeset text, encode fonts, and while  $\pdfLaTeX$  uses `fontspec`, the other use `fontenc`.

$\LaTeX$ size	Font size	Leading	Used for
<code>\tiny</code>	5	6	sidenote numbers
<code>\scriptsize</code>	7	8	—
<code>\footnotesize</code>	8	10	sidenotes, captions
<code>\small</code>	9	12	quote, quotation, and verse environments
<code>\normalsize</code>	10	14	body text
<code>\large</code>	11	15	B-heads
<code>\Large</code>	12	16	A-heads, TOC entries, author, date
<code>\LARGE</code>	14	18	handout title
<code>\huge</code>	20	30	chapter heads
<code>\Huge</code>	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`,<sup>3</sup> `\sections`, `\subsections`, and `\paragraphs`. By default `\subsubsection` and `\subparagraph` headings are not defined in the Tufte- $\LaTeX$  classes.<sup>4</sup>

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

<sup>3</sup> Parts and chapters are defined for the `tufte-book` class only.

<sup>4</sup> For more information on this topic, see (Brighurst 2005), section 4.2.2

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

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

*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- $\text{\LaTeX}$  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- $\text{\LaTeX}$  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*”,<sup>5</sup> you should “use as many levels of headings as you need: no more, and no fewer”.

<sup>5</sup> Bringhurst 2005.

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

IN HIS LATER BOOKS,<sup>6</sup> 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 style, use the `\newthought` command

<sup>6</sup> 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. The wide margin on the right side provides ample room for sidenotes and small figures. Any `\footnote` will automatically be converted into a `\sidenote`.<sup>7</sup> If you’d like to place ancillary information in the margin without the sidenote mark (the superscript number), you can use the `\marginnote` command.

<sup>7</sup> This is a sidenote that was entered using the `\footnote` command.

In his books Tufte places margin on the right side of the page, regardless whether it’s an even or odd page. If you prefer to alternate the placement of margins, so they fall on outer edge you can use the `symmetric` class option.

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.

ON A PERSONAL NOTE, placing of sidenotes—be it footnote, citation, or other—should follow the following rules:

- If the sidenote applies to the whole sentence, it should be placed after the period or other punctuation mark.
- If the sidenote applies to a specific word, it should be placed immediately after that word, even if the word is in the middle of the sentence, or followed by a punctuation mark.
- If a sidenote is a complete sentence, or a citation, it should end with a period.

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 only change the number of the current sidenote, 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.<sup>8</sup>

You will need to specify a bibliography resource file in the preamble of your document using `\addbibresource{bibliography-file.bib}` command. The complete list of references may be printed automatically by using the `\printbibliography` command. See the end of this document

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

for an example, and the Bib $\text{\LaTeX}$  documentation for more information. Bibliography can be turned off with the help of `nobib` class option.

To enter multiple citations at one location,<sup>9</sup> you can provide a list of keys separated by commas:

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

<sup>9</sup> Tufte 1990, 2006.

IN THE NEW VERSION OF TUFTE- $\text{\LaTeX}$ , it's impossible to offset citation's position the same way sidenotes can be moved up or down the margin. This is caused by the change from `natbib` package and Bib $\text{\LaTeX}$  tool to `biblatex` package and biber tool. The `biblatex` provides it own optional arguments for the `\cite` commands, which are kept unchanged to avoid confusion. I see the possible breakage of old Tufte- $\text{\LaTeX}$  documents as a fair tradeoff for the new features and flexibility that `biblatex` provides. It is worth noting that the `natbib` is mostly kept on life support, so it's better to switch now and make Tufte- $\text{\LaTeX}$  more maintainable in the future. This is one of the reasons why this version of Tufte- $\text{\LaTeX}$  classes has a new major version number.

## 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 `marginfigure` environments as follows (see figure 1):

```
\begin{marginfigure}
  \includegraphics{margin-figure}
  \caption{Margin figure caption}%
  \label{fig:margin-figure-label}
\end{marginfigure}
```

The `marginfigure` and `marginfigure` environments accept an optional parameter `\langle offset \rangle` that adjusts the vertical position of the figure or table. See the “Sidenotes” section above for examples of how to use offsets. The specifications are:

```
\begin{marginfigure}[\langle offset \rangle]
  ...
\end{marginfigure}

\begin{marginfigure}[\langle offset \rangle]
  ...
\end{marginfigure}
```

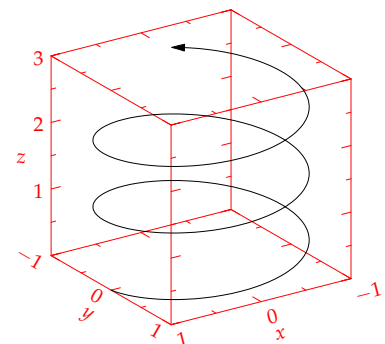


Figure 1: This is a margin figure. The helix is defined by  $x = \cos(2\pi z)$ ,  $y = \sin(2\pi z)$ , and  $z = [0, 2.7]$ . The figure was drawn using [Asymptote](http://asymptote.sf.net/) (<http://asymptote.sf.net/>).

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

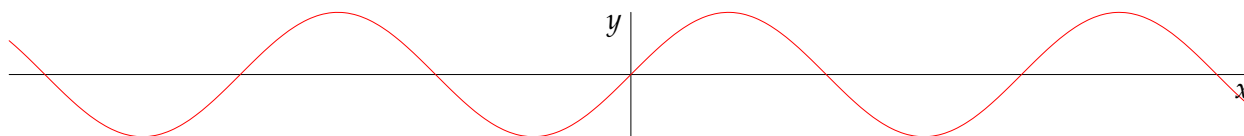


Figure 2: This graph shows  $y = \sin x$  from about  $x = [-10, 10]$ . Notice that this figure takes up the full page width.

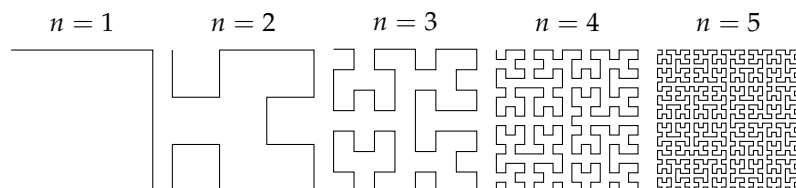


Figure 3: Hilbert curves of various degrees  $n$ . Notice that this figure only takes up the main textblock width.

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  $\langle offset \rangle$ . You may specify the caption in any one of the following forms:

```
\caption{long caption}
\caption[short caption]{long caption}
\caption[][ $\langle offset \rangle$ ]{long caption}
\caption[short caption][ $\langle offset \rangle$ ]{long caption}
```

A positive  $\langle offset \rangle$  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  $\langle short\ caption \rangle$  and  $\langle offset \rangle$  are both optional, they must be provided in order. Thus, to specify an  $\langle offset \rangle$  without specifying a  $\langle short\ caption \rangle$ , 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...}
```

CAPTION OFFSET IS UNAVAILABLE for `marginfigure` and `marginable` environments. In these cases you need to offset the whole figure or table. Captions in `marginfigure` and `marginable` still support short captions.

TUFTE STYLE TABLES ARE SIMPLE and should be styled with the `booktabs` package. 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, `colortbl` can be used 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 $\frac{1}{2}$ inches
Paper height	11 inches
Textblock width	6 $\frac{1}{2}$ inches
Textblock/sidenote gutter	$\frac{3}{8}$ 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:

```
You may only call \morefloats twice
See the Tufte-LaTeX documentation for alternatives
```

This is because allocating more floats may lead  $\LaTeX$  to run 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. Keep in mind that each sidenote actually uses *two* float 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- $\text{\LaTeX}$  document classes provide commands to override the caption position.

*Vertical alignment* In cases where the caption is too high or too low on the page, you can adjust its vertical position. To override the caption's vertical alignment, use the provided `\setfloatalignment` command inside the float environment. For example:

```
\begin{figure}
  \includegraphics{vertical-figure}
  \caption{vertical-figure-caption}%
  \label{fig:vertical-figure-label}
  \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}
  \includegraphics{horizontal-figure}
  \caption{horizontal-figure-caption}%
  \label{fig:horizontal-figure-label}
  \forceversofloat % forces caption 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, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur 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  $\text{Xe}\text{\LaTeX}$  or  $\text{Lua}\text{\LaTeX}$ , the Tufte- $\text{\LaTeX}$  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- $\text{\LaTeX}$  classes will use the ET-Bembo font from the `ETbb` package, as the main typeface. If it's unavailable, the  $\text{\TeX}$  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  $\text{\TeX}$  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  $\text{\TeX}$  Gyre Cursor font from the `tex-gyre-cursor` package will be used. However the provided `custom-tufte-common.tex` file hook overrides the default monospaced font with `RecursiveMono` font. This file shows how you can override the default fonts, and how the file hooks can be used.

The  $\text{\TeX}$  Gyre faces are usually included with  $\text{\TeX}$  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.

WHEN USING THE  $\text{pdf}\text{\LaTeX}$  engine, the Tufte- $\text{\LaTeX}$  classes will try to use the same default fonts, but will fall back to the default Computer Modern fonts if they are unavailable. The `fontspec` package is not available under  $\text{pdf}\text{\LaTeX}$ , so it uses the `fontenc` package to set the font encoding. This package doesn't make it easy to use non-standard fonts, so it's recommended to use  $\text{Xe}\text{\LaTeX}$  or  $\text{Lua}\text{\LaTeX}$  for the best results. Alternatively install and use font packages that are compatible with  $\text{pdf}\text{\LaTeX}$ .

In cases where `nofonts` option is used, the Tufte- $\text{\LaTeX}$  classes will not load any fonts. In  $\text{pdf}\text{\LaTeX}$  will not load the `fontenc` package. In other engines `no!s` option must also be used to disable loading of `fontspec` package. More info in following subsection.

### *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` commands, the letterspacing—that is, the spacing between the letters—should be increased slightly.<sup>10</sup> The `\allcaps` command was modified with proper letterspacing for strings of `FULL CAPITAL LETTERS`, and the `\smallcaps` command was modified with spacing for `SMALL CAPITAL LETTERS`. These commands will also automatically convert the case of the text to upper- or lowercase, respectively. You can see that in the source code of this document.

<sup>10</sup> Bringhurst 2005.

The `\textsc` command has also been redefined to include proper 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 `ℒTEX` `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. Tufte-`ℒTEX` offers a few additional options that are specific to the `tufte-book` and `tufte-handout` document classes. Besides the `nomoderntitles` options, which is only applicable to the `tufte-handout` class, all other options are available for both classes.

### *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-`ℒTEX` classes.

The `twoside` option will modify the running heads so that the page number is printed on the outside edge. In other words it will be placed on the right side of the odd pages, and on the left side of the even pages. When it comes to books, the head on the left side will also contain book title, and right side will contain chapter title. While in case of the handouts the left side head will use the author name, and right side will use the handout title. By default the Tufte-`ℒTEX` classes use the `twoside` option, as Tufte's *BE* book has done.<sup>11</sup> If you wish to disable it you can use the `oneside` option on a case by case basis.

<sup>11</sup> Tufte 2006.

The `symmetric` option typesets the sidenotes on the outside edge of the page, same way the `twoside` option does for the heads. This is the way books are traditionally printed, but Tufte's book design places the sidenotes on the right side of every page. This option implicitly sets the `twoside` option.

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

### *Font and Text Options*

The `sftitle` option will set the title page or block in a sans serif typeface. The `nosftitle` option will set the title page or block in a serif typeface. In case of `tufte-handout` these options also affect the abstract while in `tufte-book` they affect the epigraphs. By default the `tufte-book` class uses `sftitle` and the `tufte-handout` class uses `nosftitle`.

The `sfmarginals` option makes all marginal material use sans serif typeface instead of the default serif typeface.

The `justified` option fully justifies the main text (flush left and right). By default the text is ragged right, just as the body text of Tufte’s books is 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 pdf $\LaTeX$ . If you’re using X $\LaTeX$  or Lua $\LaTeX$ , you can use `fontspec` to set fonts, so this option is not necessary, but is available if you wish to use it. If you aren’t using the `no!s` option, the `fontspec` package will still be loaded as it is required for letterspacing.

The `no!s` option inhibits loading the code that modifies the letterspacing. The Tufte- $\LaTeX$  classes try to load the appropriate letterspacing package to adjust spacing of letters in all-caps environments. It uses `letterspace` or the `soul` under pdf $\LaTeX$ . In case of X $\LaTeX$  and Lua $\LaTeX$  it uses `fontspec`.

The `bid!s` option loads the `bid!s` package which is used with X $\LaTeX$  to typeset bi-directional text. Since the `bid!s` package needs to be loaded before the `sidenotes` and `cite` commands are defined, it can’t be loaded in the document preamble. Hence this option exists to load it in the class file.

### *Title Page Options*

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

### *Toggle Options*

The `nobib` option inhibits loading of the `natbib` and `bibtex` packages and modifying 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 `nonotes` option inhibits loading packages used for `ShadedNote` and `FramedNote` environments. By default it loads the `amsthm`, `cleveref`, and `thmtools` packages. Using those it defines environments for notes that are either shaded or framed.

The `nomodern!titles` 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, subsection, and paragraph titles. The default is to color the titles and add a colored box to the left with section

numbers inside it.

### *Marginal Options*

In the Tufte- $\text{\LaTeX}$  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** Sets the text to be justified (sets it flush left and right).

**raggedleft** Sets the text to be ragged left.

**raggedright** Sets the text to be ragged right.

**raggedouter** Sets the text to be ragged left on the left (verso) page, and ragged right on the right (recto) page. This is useful in conjunction with the `symmetric` document class option.

**auto** Justified the text if `justified` class option is on, otherwise used default ragged right text. This is the default justification option for marginal material.

Additionally, the `marginals` option can be used to set the justification settings for all marginal materials. See the [Customizing Marginal Material](#) section for more information on marginal material.

### *Debugging Options*

The `debug` option causes the Tufte- $\text{\LaTeX}$  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- $\text{\LaTeX}$  settings section. Additionally the `tufte-handout` will print out a dedicated Tufte- $\text{\LaTeX}$  Handout 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 common 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.

This project comes with a `custom-tufte-common.tex` file hook that demonstrates how to use the file hooks. It shows how to change the monospaced font to `RecursiveMono`. You can use it as a starting point for your own customizations.

## *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`. Note that this makes it impossible to use the `cleveref` package’s `\cref` command with sections and subsections. 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 <code>tufte-book</code> )	<code>-1</code>
Part (in <code>tufte-handout</code> )	<code>0</code>
Chapter (only in <code>tufte-book</code> )	<code>0</code>
Section	<code>1</code>
Subsection	<code>2</code>
Subsubsection	<code>3</code>
Paragraph	<code>4</code>
Subparagraph	<code>5</code>

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

## Changing the Paper Size

The Tufte- $\text{\LaTeX}$  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.<sup>12</sup>

<sup>12</sup> 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`. 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** Sets the text to be justified (sets it flush left and right).

**raggedleft** Sets the text to be ragged left, regardless of which page it falls on.

**raggedright** Sets the text to be ragged right, regardless of which page it falls on.



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

**auto** If the `justified` document class option was specified, then the marginal text will also be 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:

```
\setsidenoteont{font commands}
\setcaptionfont{font commands}
\setmarginnoteont{font commands}
\setcitationfont{font commands}
```

The `\setsidenoteont` sets the font and style for sidenotes, the `\setcaptionfont` for captions, the `\setmarginnoteont` 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:




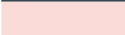



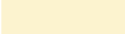

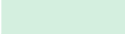

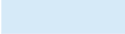


```
\setcaptionfont{\itshape\sffamily}
```



# New Features in Tufte- $\LaTeX$ Classes

## Custom Colors

The new Tufte- $\LaTeX$  document classes define a number of custom colors. They use these colors for things like links, citations, links, etc. In case of `tufte-handout` class it also uses them for the section titles. The common class uses the `xcolor` package to define the colors. You can choose to use these colors in your own documents as you see fit, redefine them, override them, or not use them at all. Here are the colors available in the Tufte- $\LaTeX$  document classes:

	tufte-black		tufte-grey
	tufte-red		tufte-pastel-red
	tufte-orange		tufte-pastel-orange
	tufte-yellow		tufte-pastel-yellow
	tufte-green		tufte-pastel-green
	tufte-blue		tufte-pastel-blue
	tufte-purple		tufte-pastel-purple

The colors are defined with following values:

```
\definecolor{tufte-black}{HTML}{282828}  
\definecolor{tufte-grey}{HTML}{F6F6F6}  
\definecolor{tufte-white}{HTML}{FFFFFF}  
\definecolor{tufte-red}{HTML}{E74C3C}  
\definecolor{tufte-pastel-red}{HTML}{FADBBD8}  
\definecolor{tufte-orange}{HTML}{E67E22}  
\definecolor{tufte-pastel-orange}{HTML}{FAE5D3}  
\definecolor{tufte-yellow}{HTML}{F1C40F}  
\definecolor{tufte-pastel-yellow}{HTML}{FCF3CF}  
\definecolor{tufte-green}{HTML}{27AE60}  
\definecolor{tufte-pastel-green}{HTML}{D4EFD8}  
\definecolor{tufte-blue}{HTML}{3498DB}  
\definecolor{tufte-pastel-blue}{HTML}{D6EAF8}  
\definecolor{tufte-purple}{HTML}{9B59B6}  
\definecolor{tufte-pastel-purple}{HTML}{EBDEF0}
```

## Note Environments

Another feature common to the new Tufte- $\LaTeX$  classes are two environments for notes. These notes can be used to highlight important information, provide references, or to simply make a note. It's useful for

making important informations stand out, without risk of being lost in the margins.

Both note environments provide a title, a label, and a continuation option. The title and label are optional, the label is used for referencing the note or for continuing a note later in the text. The `ShadedNote` environment creates a note with a shaded background. While the `FramedNote` environment places a frame to the left of the note. Both environments use the same counter as they are similar enough so it doesn't make sense to separate them.

**Note 1.** *This is an example of the `ShadedNote` environment. It provides a shaded background for the note text. The note text can be long or short, although they should be short and to the point. Notes should be used to crucial information, not be a substitute for a paragraph*

#### **Note 2 (Note Title)**

*This is an example of the `FramedNote` environment. Frames the note text with to the left of note and is more muted than the `ShadedNote`. In both cases the note text is italicized.*

If you label an note, you can reference it using the `\cref` command. For example, [Note 2](#) showcases the `FramedNote` environment. You can also use the `<continues>` option to continue an note:

#### **Note 2 ([continuing from p. 36](#))**

*This is a continuation of the previous note. It will be displayed in a new frame, but will have the same label, title, and number. Useful if you want to refer to or expand on previous note without having to repeat the information.*

The way to use the note environments is as follows:

```
\begin{ShadedNote}[
  title={Optional title},
  label={Optional label},
  continues={Optional label}
]
  Note text here
\end{ShadedNote}
```

### *Modified Section Headings*

The `tufte-handout` class provides a new section heading style. Every section, subsection and paragraph will have their names colored, in addition the the usual styling. Additionally section and subsection titles will have a colored box placed to the left of the title. Section number will be placed inside the box. This is done 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 the default styling, you can pass the `nomoderntitles` option to the `tufte-handout` class.

The following figure shows the difference between the default and the new section heading style.



Figure 4: Comparison between modern and old style section headings. Notice that the spacing between the sections is a little bit different. This was motivated by wanting to equalize the spacing between colored boxes.



# Installing and Compiling Tufte- $\LaTeX$

## Local Installation

To install the Tufte- $\LaTeX$  classes locally, download the source files from the GitHub repository<sup>13</sup>. Then you can place the following class files in the same directory as your  $\TeX$  document:

- `tufte-common.def`
- `tufte-handout.cls`
- `tufte-book.cls`
- `custom-tufte-common.tex` (*optional*)

You can then either compile the document using `latexmk` or use the provided `makefile` to compile the document. See the [Compiling Your Document](#) section for more information on compiling the document.

<sup>13</sup> Repository can be found at <https://github.com/MormonJesus69420/Modernized-Tufte-LaTeX>

## Global Installation

To install the Tufte- $\LaTeX$  classes globally, you can place the class files in your  $\TeX$  distribution's `tex/latex/tufte-latex` directory. When doing it by hand use the `kpsewhich -var-value TEXMFHOME` to find the correct directory, and copy the files there. After copying the files, you will need to update the  $\TeX$  distribution's file database. To do that run the `texhash` command in the terminal.

You can also use the `install` target from provided `makefile`. This will do all these steps for you, installing additional documentation files and more.

## Installing from CTAN

**TODO!** Provide installation guide from CTAN when package is published there

## Compiling Your Document

The Tufte- $\LaTeX$  classes can be compiled using `pdf $\LaTeX$` , `X $\LaTeX$` , or `Lua $\LaTeX$` . I personally recommend using `Lua $\LaTeX$` , as it provides the best support for Tufte- $\LaTeX$  functionality without and quirks that `X $\LaTeX$`  has.

The Tufte- $\text{\LaTeX}$  classes are designed to work with the `biblatex` package for bibliographies. As such you will need to have `biber` installed on your system to compile the bibliography files.

### *Compiling with makefile*

As mentioned before you can use the provided makefile to compile the sample documents with either `make compile-lua`, `make compile-xe`, or `make compile-pdf`. It is up to you which engine you prefer to use, but from my experience Lua $\text{\LaTeX}$  provides the best results and is the most reliable of them all. In case of single files use the `make compile-only FILE=<file>.tex` target. It is set up to use Lua $\text{\LaTeX}$  by default, but you can change it to use Xe $\text{\LaTeX}$  or pdf $\text{\LaTeX}$  by changing the `-lualatex` option to `-xelatex` or `-pdflatex`.

In general I would recommend changing the `-outdir=samples` in the `LMKFLAGS` variable to a directory of your choice, for example: `-outdir=out`.

If you wish to clean up the auxiliary files, use the `make clean` target. And the remove both auxiliary files and compiled PDFs use the `make clean-all` target.

IN MOST USECASES YOU CAN USE THIS simplified makefile, with only the necessary targets for most use-cases. You may use it to compile files using different  $\text{\LaTeX}$  classes too. As before you can change Lua $\text{\LaTeX}$  for other engines.

```
LMKFLAGS = -outdir=out -auxdir=aux -bibtex
```

```
compile-all:
```

```
  latexmk $(LMKFLAGS) -lualatex -jobname=%A-lualatex \
  $(shell find . -maxdepth 1 -name '*.tex' -not -path '*/custom-tufte-*')
```

```
compile-only:
```

```
  if [ -z "$(FILE)" ]; then \
    echo "You need to provide a FILE parameter when using this target"; \
    echo "make compile-only FILE=\"<file>.tex\""; \
  else \
    latexmk $(LMKFLAGS) -lualatex -jobname=%A "$(FILE)"; \
  fi
```

```
clean:
```

```
  rm -rfv aux
```

```
clean-all:
```

```
  rm -rfv aux out
```

### *Compiling manually*

You may also compile the documents with `latexmk` directly using the following command: Just substitute “`<engine>`” with your preferred engine, and “`<source.tex>`” with the name of your source file.

```
latexmk -outdir=out -auxdir=aux -bibtex -<<engine>> -jobname=%A-lualatex <<source.tex>>
```

You may also do things fully manually. To compile `tufte-handout`,



you should do something like this:

```
lualatex <<source.tex>> # Replace <<source.tex>> with the name of your source file
biber <<source>> # Note that the file extension is omitted
lualatex <<source.tex>>
lualatex <<source.tex>>
```

To compile tufte-handout, you should do something like this:

```
lualatex <<source.tex>> # Replace <<source.tex>> with the name of your source file
biber <<source>> # Note that the file extension is omitted
makeindex <<source.idx>> # If you are using an index
lualatex <<source.tex>>
lualatex <<source.tex>>
lualatex <<source.tex>>
```



# *Compatibility Issues*

When switching an existing document from one document class to a Tufte- $\text{\LaTeX}$  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- $\text{\LaTeX}$ Website*

The original website for the Tufte- $\text{\LaTeX}$  packages is located at <https://tufte-latex.github.io/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- $\text{\LaTeX}$  classes at my GitHub repository.

<https://github.com/MormonJesus69420/Modernized-Tufte-LaTeX>

I have also set up a custom website for the project which is hosted under <https://salwerowi.cz/Modernized-Tufte-LaTeX>.

If for some reason this address does not work it should also be available under <https://mormonjesus69420.github.io/Modernized-Tufte-LaTeX>.

## *Tufte- $\text{\LaTeX}$ Mailing Lists*

There is only one surviving mailing list for the Tufte- $\text{\LaTeX}$  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- $\text{\LaTeX}$  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- $\text{\LaTeX}$  classes is to follow the GitHub repository. You can use the *Discussions* feature to ask questions, suggest improvements, and interact with other users. In case you notice and bugs with the classes, or notice any issues with the documentation, you can open an issue on the GitHub repository. In case you want to contribute to the project, you can open a pull request on the GitHub repository.

<https://github.com/MormonJesus69420/Modernized-Tufte-LaTeX/discussions>

<https://github.com/MormonJesus69420/Modernized-Tufte-LaTeX/issues>  
<https://github.com/MormonJesus69420/Modernized-Tufte-LaTeX/pulls>

I DON'T PLAN ON USING MAILING LISTS in my project, as I never got used to them, and feel that GitHub provides a more user friendly way to interact with the project.

## 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 or start a discussion on the GitHub repository.

To help with troubleshooting the problem more quickly, please try to compile your document using the debug class option. When asking for help include the generated `.log` file in the issue, along with a brief description of the problem, and if possible a minimal working example that reproduces the issue. You can also upload your whole document if you’re not sure what’s causing the issue, and you are comfortable with sharing the document.

## Errors, Warnings, and Informational Messages

The following is a list of all of the known 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 alternatives

$\LaTeX$  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 25 for more information.

Warning: Option ‘ $\langle class option \rangle$ ’ is not supported -- ignoring option.

This warning appears when you’ve tried to use  $\langle class option \rangle$  with a Tufte- $\LaTeX$  document class, but  $\langle class option \rangle$  isn’t supported by the Tufte- $\LaTeX$  document class. In this situation,  $\langle class option \rangle$  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 28 for more information on the symmetric class option.

### Package Dependencies

The following is a list of packages that the Tufte- $\text{\LaTeX}$  document classes rely upon. Packages marked with an asterisk are either optional or only loaded for some class options.

- `ams{math,symb,thm,xtra}` \* *for Note environments*
- `biblatex` \* *only if nobib is off, requires biber backend*
- `bidi` \* *only if using bidi option*
- `changepage`
- `chnpage` \* *only if changepage is not available*
- `cleveref` \* *for Note environments*
- `ETbb` \* *if available, and nofonts is off*
- `fancyhdr`
- `FiraMono` \* *if available, and nofonts is off*
- `fontenc` \* *only with pdf $\text{\LaTeX}$ , and nofonts is off*
- `fontspec` \* *only with Xe $\text{\LaTeX}$  or Lua $\text{\LaTeX}$ , and nofonts is off*
- `geometry`
- `gillius2` \* *if available, and nofonts is off*
- `hardwrap`
- `hyperref` \* *only if nohyper is off*
- `iftex` \* *if not it assumes pdf $\text{\LaTeX}$*
- `letterspace` \* *only if no $\text{\textit{ls}}$  is off*
- `mathpazo` \* *if available, and nofonts is off*
- `multicol`
- `optparams`
- `paralist`
- `placeins`
- `ragged2e`
- `sectsty`
- `setspace`
- `soul` \* *only with pdf $\text{\LaTeX}$*
- `textcase`
- `textcomp` \* *only with pdf $\text{\LaTeX}$ , and nofonts is off*
- `thmtools` \* *for Note environments*
- `titlesec`
- `titletoc`
- `transparent`
- `xcolor`
- `xifthen`
- `xkeyval`





# 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, 23).
- (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).
  - (2006). *Beautiful Evidence*. First. Cheshire, Connecticut: Graphics Press, LLC. ISBN: 0-9613921-7-7 (cit. on pp. 13, 21, 23, 28).
- Umeki, Hideo (Dec. 2008). *The geometry package*. <http://ctan.org/pkg/geometry> (cit. on p. 32).



# Index

- 'symmetric' class option, 47
- 'twoside' class option, 47
- 10pt class option, 29, 43
- 11pt class option, 29, 43
- 12pt class option, 29, 43
  
- a4paper class option, 28
- a5paper class option, 28, 43
- \adbbibresource, 22
- \allcaps, 27
- amsthm package, 29
- \author, 29
- auto class option, 30, 33
- \autocite, 22
  
- b5paper class option, 28, 43
- Bib $\LaTeX$ , 23
- biblatex package, 23, 40
- bibtex package, 29
- bidi class option, 29
- bidi package, 29
- booktabs package, 24
  
- caption class option, 30, 32
- \caption, 24
- \chapters, 19
- citation class option, 30, 32
- \cite, 22, 23, 29
- class options, 28–30
  - 'symmetric', 47
  - 'twoside', 47
  - 10pt, 29, 43
  - 11pt, 29, 43
  - 12pt, 29, 43
  - a4paper, 28
  - a5paper, 28, 43
  - auto, 30, 33
  - b5paper, 28, 43
  - bidi, 29
  - caption, 30, 32
  - citation, 30, 32
  - debug, 30, 46
  - executivepaper, 28, 43
  - justified, 29, 30, 32, 33
  - landscape, 28, 43
  - legalpaper, 28, 43
  - marginals, 30, 32
  - marginnote, 30, 32
  - nobib, 23, 29
  - nofonts, 27, 29
  - nohyper, 29
  - nols, 27, 29
  - nomoderntitles, 28, 29, 32, 36
  - nonotes, 29
  - nosftitle, 28
  - notitlepage, 29
  - notoc, 29
  - onecolumn, 28, 43
  - oneside, 28
  - raggedleft, 30, 32
  - raggedouter, 30, 33
  - raggedright, 30, 32
  - sfmarginals, 28
  - sftitle, 28
  - sidenote, 30, 32
  - symmetric, 21, 26, 28, 30, 33, 47
  - titlepage, 29
  - twocolumn, 28
  - twoside, 28, 47
- \clearpage, 25
- cleveref package, 29, 32, 36
- colortbl package, 24
- \cref (in cleveref package), 32, 36
- custom-tufte-book.tex, 31
- custom-tufte-common.tex, 19, 27, 31
- custom-tufte-handout.tex, 31
  
- debug class option, 30, 46
- debug messages, 46
  
- environments
  - figure, 23, 24
  - figure\*, 23, 24
  - FramedNote, 29, 36
  - fullwidth, 26
  - marginfigure, 23, 24
  - marginfigure environment, 23, 24
  - marginnote, 23, 24
  - ShadedNote, 29, 36
  - table\*, 23
  - tabular, 23
  
- error messages, 46
- ETbb package, 19, 27
- executivepaper class option, 28, 43
  
- figure environment, 23, 24
- figure\* environment, 23, 24
- file hooks, 31
  - book, 31
  - common, 19, 27, 31
  - handout, 31
- FiraMono package, 19, 27
- \FloatBarrier, 25
- fontenc package, 19, 27
- fonts, *see* typefaces, *see* typefaces
- fontspec package, 19, 27, 29
- \footnote, 21
- \forcerectofloat, 26
- \forceversoofloat, 26
- FramedNote environment, 29, 36
- fullwidth environment, 26
  
- \geometry, 32
- \geometry (in geometry package), 30, 32
- geometry package, 30, 32
- gillius package, 19, 27
  
- headings, 19, 21
  - numbered, 31
- hyperref package, 29
  
- justified class option, 29, 30, 32, 33
  
- landscape class option, 28, 43
- legalpaper class option, 28, 43
- letterspace package, 29
- license, 4
- Lua $\LaTeX$ , 19, 27, 29, 39, 40, 47
  
- \maketitle, 29
  - marginals class option, 30, 32
  - marginfigure environment, 23, 24
  - marginnote class option, 30, 32
- \marginnote, 21, 22
- marginfigure environment, 23, 24
- mathpazo package, 19, 27

- `\morefloats`, 25, 46
- natbib package, 23, 29
- `\newthought`, 21
- nobib class option, 23, 29
- nofonts class option, 27, 29
- nohyper class option, 29
- nols class option, 27, 29
- nomoderntitles class option, 28, 29, 32, 36
- nonotes class option, 29
- nosftitle class option, 28
- notitlepage class option, 29
- notoc class option, 29
- onecolumn class option, 28, 43
- oneside class option, 28
- options, *see* class options
- packages
  - amsthm, 29
  - biblatex, 23, 40
  - bibtex, 29
  - bidi, 29
  - booktabs, 24
  - cleveref, 29, 32, 36
  - colortbl, 24
  - ETbb, 19, 27
  - FiraMono, 19, 27
  - fontenc, 19, 27
  - fontspec, 19, 27, 29
  - geometry, 30, 32
  - gillius, 19, 27
  - hyperref, 29
  - letterspace, 29
  - mathpazo, 19, 27
  - natbib, 23, 29
  - soul, 29
  - tex-gyre-cursor, 27
  - tex-gyre-heros, 27
  - tex-gyre-pagella, 27
  - thmtools, 29
  - xcolor, 35
- `\paragraph`, 20
- `\paragraphs`, 19
- `\parts`, 19
- pdf $\LaTeX$ , 19, 27, 29, 39, 40, 47
- `\printbibliography`, 22
- raggedleft class option, 30, 32
- raggedouter class option, 30, 33
- raggedright class option, 30, 32
- secnumdepth counter, 31, 32
- `\section`, 21
- `\sections`, 19
- `\setcaptionfont`, 33
- `\setcitationfont`, 33
- `\setcounter`, 32
- `\setfloatalignment`, 26
- `\setmarginnotefont`, 33
- `\setsidenotefont`, 33
- sfmarginals class option, 28
- sftitle class option, 28
- ShadedNote environment, 29, 36
- sidenote class option, 30, 32
- `\sidenote`, 21, 22
- `\smallcaps`, 27
- soul package, 29
- `\subparagraph`, 19, 21, 43, 46
- `\subsection`, 21
- `\subsections`, 19
- `\subsubsection`, 19, 21, 43, 46
- symmetric class option, 21, 26, 28, 30, 33, 47
- table of contents, 18
- table\* environment, 23
- tabular environment, 23
- tex-gyre-cursor package, 27
- tex-gyre-heros package, 27
- tex-gyre-pagella package, 27
- `\textsc`, 28
- `\thispagestyle`, 16
- thmtools package, 29
- `\title`, 29
- titlepage class option, 29
- twocolumn class option, 28
- twoside class option, 28, 47
- typefaces, 19, 27
  - sizes, 19
- warning messages, 46
- xcolor package, 35
- X $\LaTeX$ , 19, 27, 29, 39, 40, 47