Tint Is Not Tufte

JJ Allaire, Yihui Xie, Dirk Eddelbuettel

2018-06-11

Introduction

In this paper, I selected to approach the problem from a different angle that I usually do. In place of my usual approach of attempting to solve the problem using the training from this course and previous ones, I decided to investigate what published journal or white papers have done previously, especially trying to pick those papers which have reported higher scores in the competition. This approach allowed me to learn new techniques used by more experienced practitioners as well as extend my deep learning material I learnt in predict 453 few quarters ago.

This paper is organized as follows.

Secion I deliniates the overview of the methodologies used, the papers referenced and the challenges faced at a high level. It also explains some technical challenges faced.

Section II explains some of the exploratory work done.

Section III explains the data preparation activities.

Section IV outlines details of the Method Of Analogues model.

Section V outlines details of the deep learning models.

Section VI talks about the Bayesian Regression model.

Section VII wraps up the paper.

Section I - Overview of Methodologies Used

I quickly read a plethora of published papers, white papers and class notes on this problem set. The difficulty of the problem revealed itself since almost everyone had used a different approach to solving the problem. Folks have attempted to solve this using everything from ensembled linear models to non-linear deep learning approaches to heuristic computational methods. I chose two papers to try and replicate. Both papers used methods not taught in the northwestern courses, but built upon techniques already taught in the courses so far. While I realized that trying to replicate a paper an entire paper created by a professor with his 3 PhD students within a span of a few weeks is not easily possible, I was determined to try. If nothing, I would learn new methods which I can apply at work.

The first paper [1] is an ensemble model of three sub-models: 10s of Method Of Analogue (MOA) models, 1000s of Additive Holt Winters models and naive models, with a novel median-voting based weight scheme. The MOA [3] is a method invented in 1969 for prediction of weather. It is widely used in metero-logical model building, and has been used for influenza prediction as well.

Since there is no pre-written package in R for this method, it required me to chase down the mathematically nitty gritties [4] in a few papers and implement my own version of the model. There are many versions of MOA depending on the search algorithm, or the analogue selection algorithm. I studied a few of them, and decided to implement the simplest version. I could not implement the paper as is, with the main constraints being computational time required to solve these search based models iterative models on such a large forecast horizon.

The second paper I read relied on an ensemble of linear regression, weighted linear regression, and Bayesian regression models Out of these, I decided to learn a bit about the Bayesian model.

The third model I decided to investigate is a Recurrent Neural Network (RNN) model, specifically the Gated Recurrent Unit (GRU) and the Long Short Term Memory (LSTM). These models were ones I was looking into at the end of the Predict 490 (Deep Learning) course. This was my first foray into these recurrent models.

Section II - EDA

1. Univariate studies

Time series plots were run for all the variables to get an idea of the underlying structure. While some signals don't show strong seasonal patterns like in figure 1. Others show very strong seasonality, like in figure 2. Depending on the chosen solution, this is useful information. The response variable `total_cases` shows the peaks and available information for the two cities. Note teh different time scales on the x-axis.

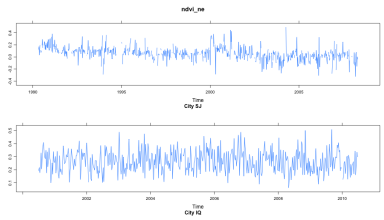


Figure 1: Lack of seasonality in NDVI NE

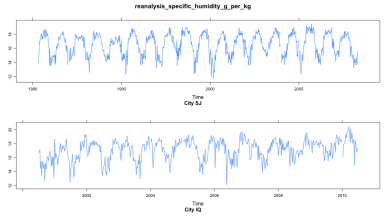


Figure 2: Seasonality in Spec Humidity

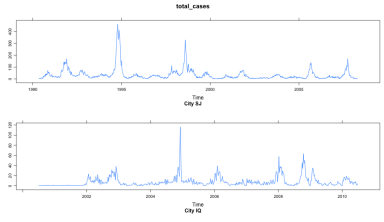


Figure 3: Total Cases Response Var

1. Multivariate studies (Linear Correlations)

Linear correlation study between the Xs and Y for the two cities show remarkable difference between the cities, along with some key insights into the underlying structure of the data. Some key highlights:

- `total_cases` is very weakly correlated (if at all) with any of the Xs. Doesn't let itself to a simple way of predicting the values. It's weakly correlated with the `weekofyear` variable, which makes sense. When it's hotter, and wetter, there is a higher chance of dengue.
- SJ's corrplot shows us that almost all the correlations are positive, it at all. As expected, all the vegetation indices are correlated positively. As are all the temperature related variables. Further investigation using PCA showed me that for these variable groups, at max 2 PCs were needed to achieve ~97%+ of explanatory power for the variation in each group.
- IQ's corrplot has a few strong negative correlations, especially with the `tdtr` variable, which explains the daily temperature fluctuation. When it's

very hot, or very humid, there is less temperature variable over the day. Given IQ's geographic location, perhaps this makes sense from a weather dynamics standpoint.

-->
-->

Before We Get Started...

tint is straightforward mix of the (html and pdf parts of the) excellent tufte package by JJ and Yihui, mixed with the Roboto Condensed font use and color scheme proposed by envisioned css plus minor style changes such as removal of *italics*—but otherwise true to the wonderful tufte package for R—all baked together into a small package providing another template.

We support *italic* aka *em* and **strong** annotations for text, as well as `code` snippets.

The package name is a standard package naming recursion: *tint is not tufte*.

The remainder of the tufte skeleton document follows as is, with only marginal changes to refer to this package for code.

Introduction

The Tufte handout style is a style that 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 style has been implemented in LaTeX and HTML/CSS[1], respectively. We have ported both implementations into the **tufte** package. If you want LaTeX/PDF output, you may use the `tufte_handout` format for handouts, and `tufte_book` for books. For HTML output, use `tufte_html`. These formats can be either specified in the YAML metadata at the beginning of an R Markdown document (see an example below), or passed to the `rmarkdown::render()` function. See Allaire et al. [2018] for more information about **rmarkdown**.

```
---
title: "An Example Using the Tufte Style"
author: "John Smith"
output:
  tufte::tufte_handout: default
  tufte::tufte_html: default
---
```

There are two goals of this package:

1. To produce both PDF and HTML output with similar styles from the same R Markdown document;

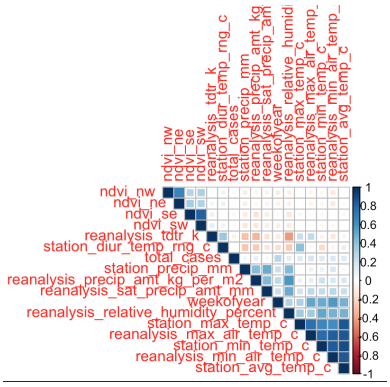


Figure 4: Cor Plot SJ

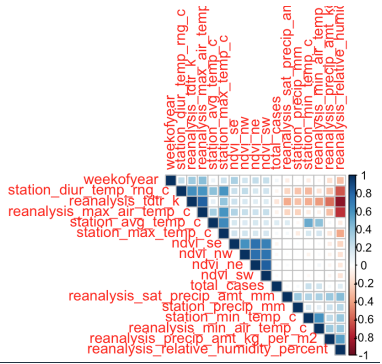
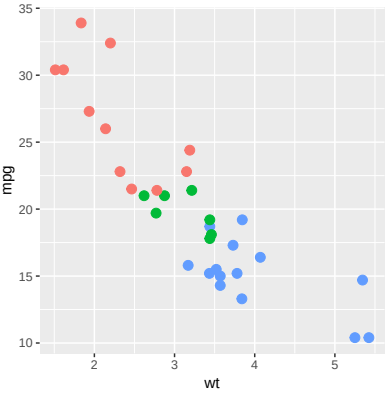


Figure 5: Cor Plot IQ



[1] See Github repositories tufte-latex and tufte-css

2. To provide simple syntax to write elements of the Tufte style such as side notes and margin figures, e.g. when you want a margin figure, all you need to do is the chunk option `fig.margin = TRUE`, and we will take care of the deails for you, so you never need to think about `\begin{marginfigure} \end{marginfigure}` or `<span class="marginfigure"> </span>`; the LaTeX and HTML code under the hood may be complicated, but you never need to learn or write such code.

If you have any feature requests or find bugs in **tufte**, please do not hesitate to file them to https://github.com/rstudio/tufte/issues. For general questions, you may ask them on StackOverflow: http://stackoverflow.com/tags/rmarkdown.

Headings

This style provides first and second-level headings (that is, # and ##), demonstrated in the next section. You may get unexpected output if you try to use ### and smaller headings.

In his later books[2], 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, call the `newthought()` function in **tufte** in an *inline R expression* `` `r ` `` as demonstrated at the beginning of this paragraph.[3]

[2] Beautiful Evidence

[3] Note you should not assume **tufte** has been attached to your R session. You should either `library(tufte)` in your R Markdown document before you call `newthought()`, or use `tint::newthought()`.

Figures

Margin Figures

Images and graphics play an integral role in Tufte's work. To place figures in the margin you can use the **knitr** chunk option `fig.margin = TRUE`. For example:

```
library(ggplot2)
mtcars2 <- mtcars
mtcars2$am <- factor(
  mtcars$am, labels = c('automatic', 'manual')
)
ggplot(mtcars2, aes(hp, mpg, color = am)) +
  geom_point() + geom_smooth() +
  theme(legend.position = 'bottom')
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

Note the use of the `fig.cap` chunk option to provide a figure caption. You can adjust the proportions of figures using the `fig.width` and `fig.height` chunk options. These are specified in inches, and will be automatically scaled down to fit within the handout margin.

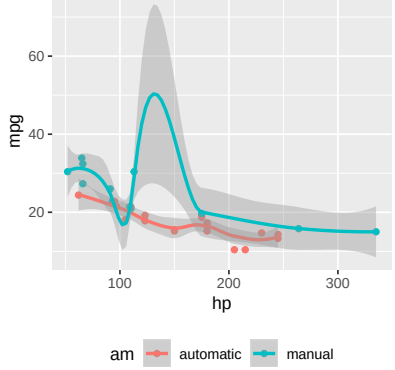


Figure 6: MPG vs horsepower, colored by transmission.

Arbitrary Margin Content

In fact, you can include anything in the margin using the **knitr** engine named `marginfigure`. Unlike R code chunks ```` ```{r} ````, you write a chunk starting with ```` ```{marginfigure} ```` instead, then put the content in the chunk. See an example on the right about the first fundamental theorem of calculus.

For the sake of portability between LaTeX and HTML, you should keep the margin content as simple as possible (syntax-wise) in the `marginefigure` blocks. You may use simple Markdown syntax like `**bold**` and `_italic_` text, but please refrain from using footnotes, citations, or block-level elements (e.g. blockquotes and lists) there.

We know from *the first fundamental theorem of calculus* that for x in $[a, b]$:

$$\frac{d}{dx}\left(\int_a^x f(u)\,du\right) = f(x).$$

Full Width Figures

You can arrange for figures to span across the entire page by using the chunk option `fig.fullwidth = TRUE`.

```
ggplot(diamonds, aes(carat, price)) + geom_smooth() +
  facet_grid(~ cut)
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

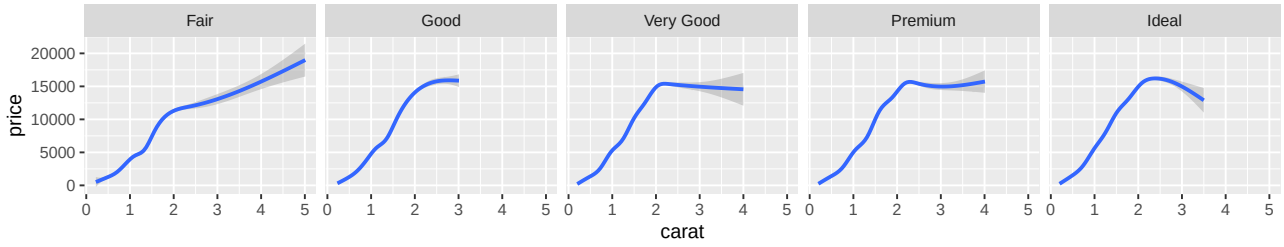


Figure 7: A full width figure.

Other chunk options related to figures can still be used, such as `fig.width`, `fig.cap`, `out.width`, and so on. For full width figures, usually `fig.width` is large and `fig.height` is small. In the above example, the plot size is 10×2.

Main Column Figures

Besides margin and full width figures, you can of course also include figures constrained to the main column. This is the default type of figures in the LaTeX/HTML output.

```
ggplot(diamonds, aes(cut, price)) + geom_boxplot()
```

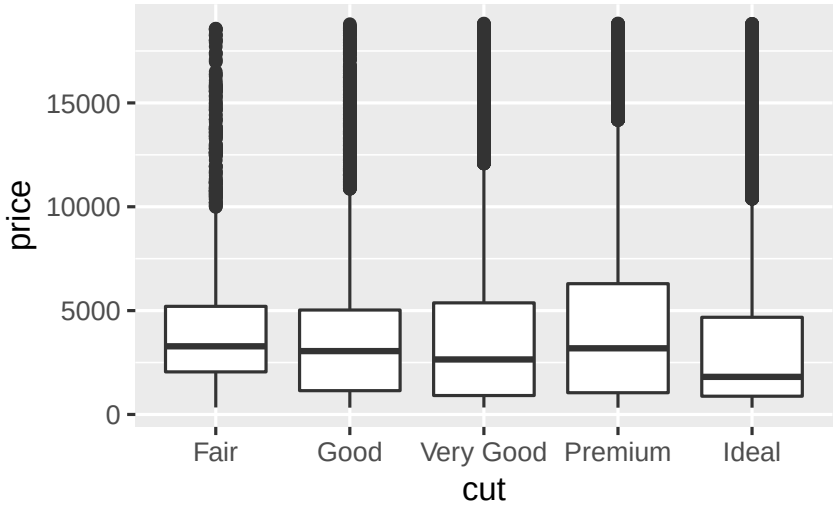


Figure 8: A figure in the main column.

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 use of a footnote will automatically be converted to a sidenote. [4]

If you'd like to place ancillary information in the margin without the sidenote mark (the superscript number), you can use the `margin_note()` function from **tufte** in an inline R expression. This function does not process the text with Pandoc, so Markdown syntax will not work here. If you need to write anything in Markdown syntax, please use the `marginfigure` block described previously.

[4] This is a sidenote that was entered using a footnote.

This is a margin note. Notice that there is no number preceding the note.

References

References can be displayed as margin notes for HTML output. For example, we can cite R here [R Core Team, 2017]. To enable this feature, you must set `link-citations: yes` in the YAML metadata, and the version of `pandoc-citeproc` should be at least 0.7.2. You can always install your own version of Pandoc from `http://pandoc.org/installing.html` if the version is not sufficient. To check the version of `pandoc-citeproc` in your system, you may run this in R:

```
system2('pandoc-citeproc', '--version')
```

If your version of `pandoc-citeproc` is too low, or you did not set `link-citations: yes` in YAML, references in the HTML output will be placed at the end of the output document.

Tables

You can use the `kable()` function from the **knitr** package to format tables that integrate well with the rest of the Tufte handout style. The table captions are placed in the margin like figures in the HTML output.

```
knitr::kable(
  mtcars[1:6, 1:6], caption = 'A subset of mtcars.'
)
```

Table 1: A subset of mtcars.

	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875
Datsun 710	22.8	4	108	93	3.85	2.320
Hornet 4 Drive	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440
Valiant	18.1	6	225	105	2.76	3.460

Block Quotes

We know from the Markdown syntax that paragraphs that start with > are converted to block quotes. If you want to add a right-aligned footer for the quote, you may use the function `quote_footer()` from **tufte** in an inline R expression. Here is an example:

> "If it weren't for my lawyer, I'd still be in prison. It went a lot faster with two people digging."
>
> — Joe Martin

Without using `quote_footer()`, it looks like this (the second line is just a normal paragraph):

> "Great people talk about ideas, average people talk about things, and small people talk about wine."
>
> — Fran Lebowitz

Responsiveness

The HTML page is responsive in the sense that when the page width is smaller than 760px, sidenotes and margin notes will be hidden by default. For sidenotes, you can click their numbers (the superscripts) to toggle their visibility. For margin notes, you may click the circled plus signs to toggle visibility.

More Examples

The rest of this document consists of a few test cases to make sure everything still works well in slightly more complicated scenarios. First we generate two plots in one figure environment with the chunk option `fig.show = 'hold'`:

```
p <- ggplot(mtcars2, aes(hp, mpg, color = am)) +
  geom_point()
p
p + geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

Then two plots in separate figure environments (the code is identical to the previous code chunk, but the chunk option is the default `fig.show = 'asis'` now):

```
p <- ggplot(mtcars2, aes(hp, mpg, color = am)) +
  geom_point()
p
```

```
p + geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

You may have noticed that the two figures have different captions, and that is because we used a character vector of length 2 for the chunk option `fig.cap` (something like `fig.cap = c('first plot', 'second plot')`).

Next we show multiple plots in margin figures. Similarly, two plots in the same figure environment in the margin:

```
p
p + geom_smooth(method = 'lm')
```

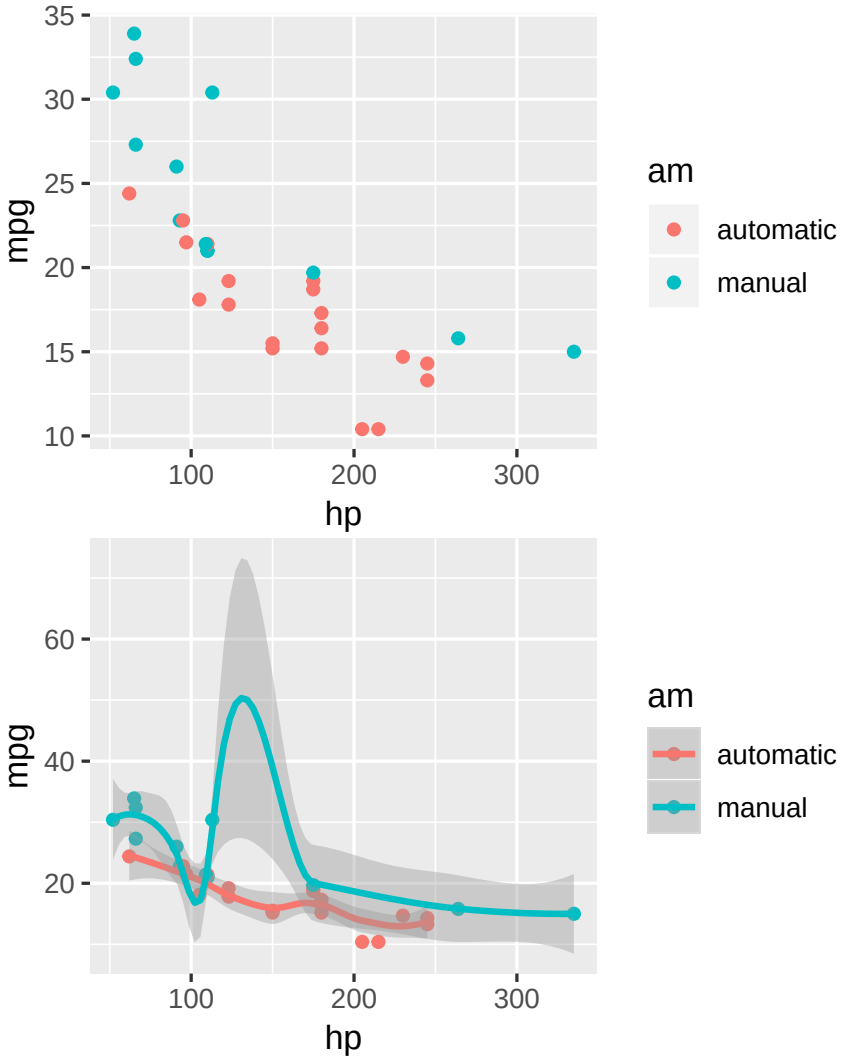


Figure 9: Two plots in one figure environment.

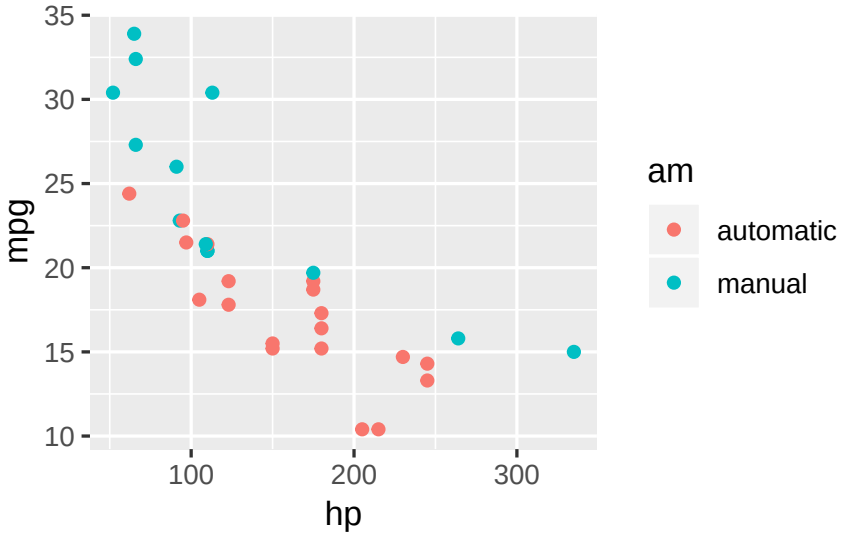


Figure 10: Two plots in separate figure environments (the first plot).

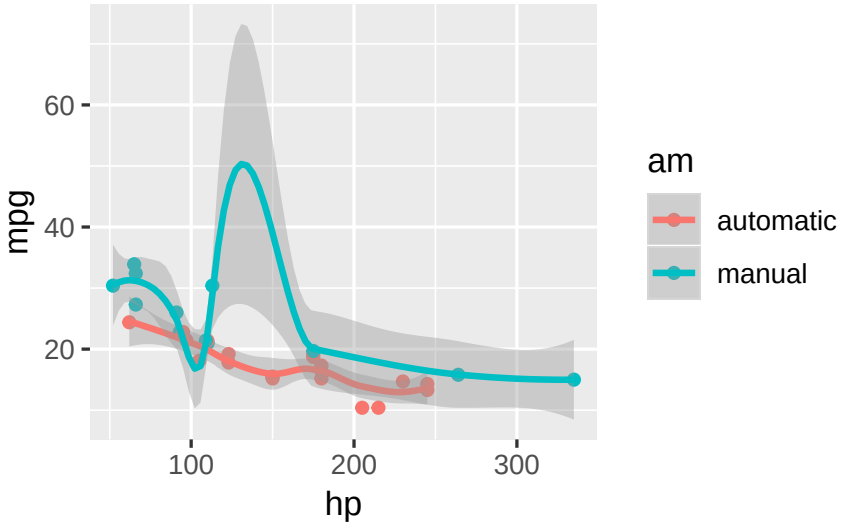


Figure 11: Two plots in separate figure environments (the second plot).

Then two plots from the same code chunk placed in different figure environments:

```
knitr::kable(head(iris, 15))
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa
5.8	4.0	1.2	0.2	setosa

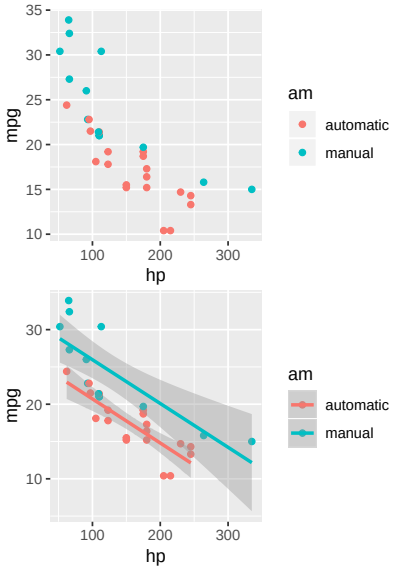


Figure 12: Two plots in one figure environment in the margin.

```
p
```

```
knitr::kable(head(iris, 12))
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa

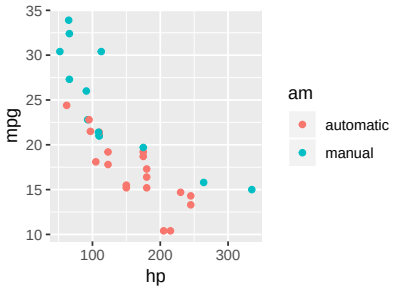


Figure 13: Two plots in separate figure environments in the margin (the first plot).

```
p + geom_smooth(method = 'lm')
```

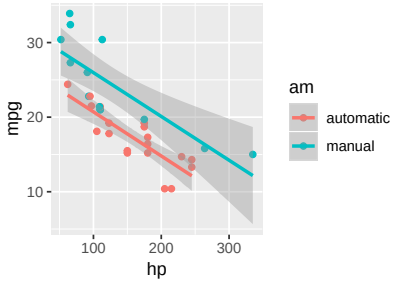


Figure 14: Two plots in separate figure environments in the margin (the second plot).

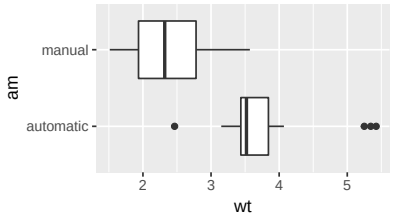
```
knitr::kable(head(iris, 5))
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa

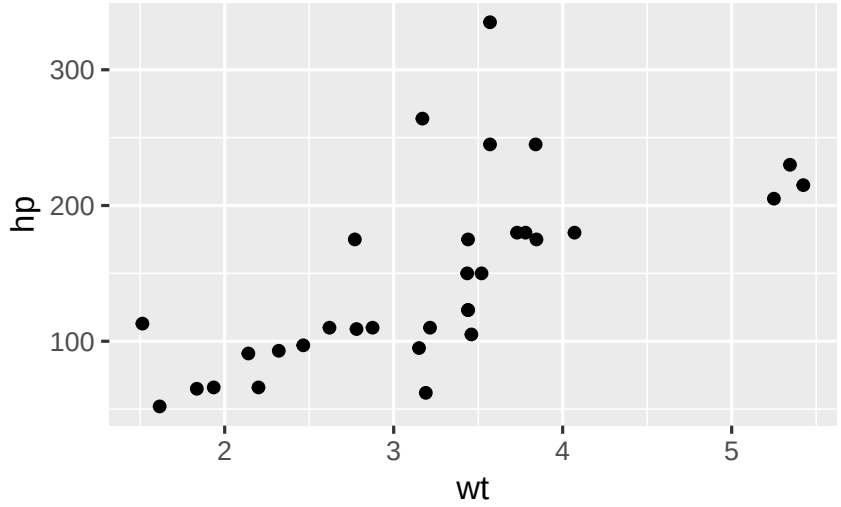
We blended some tables in the above code chunk only as *placeholders* to make sure there is enough vertical space among the margin figures, otherwise they will be stacked tightly together. For a practical document, you should not insert too many margin figures consecutively and make the margin crowded.

You do not have to assign captions to figures. We show three figures with no captions below in the margin, in the main column, and in full width, respectively.

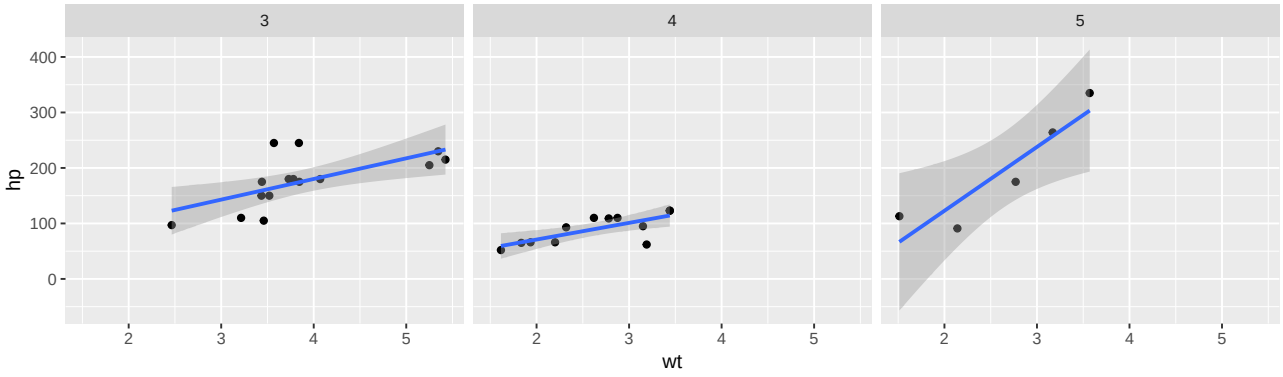
```
# a boxplot of weight vs transmission; this figure
# will be placed in the margin
ggplot(mtcars2, aes(am, wt)) + geom_boxplot() +
  coord_flip()
```




```
# a figure in the main column
p <- ggplot(mtcars, aes(wt, hp)) + geom_point()
p
```



```
# a fullwidth figure
p + geom_smooth(method = 'lm') + facet_grid(~ gear)
```



Some Notes on Tufte CSS

There are a few other things in Tufte CSS that we have not mentioned so far. If you prefer sans-serif fonts, use the function `sans_serif()` in **tufte**. For epigraphs, you may use a pair of underscores to make the paragraph italic in a block quote, e.g.

> *I can win an argument on any topic, against any opponent. People know this, and steer clear of me at parties. Often, as a sign of their great respect, they don't even invite me.*
>
> — Dave Barry

We hope you will enjoy the simplicity of R Markdown and this R package, and we sincerely thank the authors of the Tufte-CSS and Tufte-LaTeX projects for developing the beautiful CSS and LaTeX classes. Our **tufte** package would not have been possible without their heavy lifting.

To see the R Markdown source of this example document, you may follow this link to Github, use the wizard in RStudio IDE (`File -> New File -> R Markdown -> From Template`), or open the Rmd file in the package:

```
file.edit(
  tint:::template_resources(
    'tint', '..', 'skeleton', 'skeleton.Rmd'
  )
)
```

References

JJ Allaire, Yihui Xie, Jonathan McPherson, Javier Luraschi, Kevin Ushey, Aron Atkins, Hadley Wickham, Joe Cheng, and Winston Chang. *rmarkdown: Dynamic Documents for R*, 2018. URL https://CRAN.R-project.org/package=rmarkdown. R package version 1.9.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2017. URL https://www.R-project.org/.