Create Awesome LaTeX Table with knitr::kable and kableExtra

Hao Zhu

2017-09-04

Contents

Please see the package documentation site for how to use this package in HTML and more.

Overview

The goal of `kableExtra` is to help you build common complex tables and manipulate table styles. It imports the pipe `%>%` symbol from `magrittr` and verbalize all the functions, so basically you can add "layers" to a kable output in a way that is similar with `ggplot2` and `plotly`.

To learn how to generate complex tables in LaTeX, please visit http://haozhu233.github.io/kableExtra/awesome_table_in_html.html.

Installation

```r
install.packages("kableExtra")

# For dev version
# install.packages("devtools")
devtools::install_github("haozhu233/kableExtra")
```

Getting Started

Here we are using the first few columns and rows from dataset `mtcars`

```r
library(knitr)
library(kableExtra)
dt <- mtcars[1:5, 1:6]
```

When you are using `kable()`, if you don't specify `format`, by default it will generate a markdown table and let pandoc handle the conversion from markdown to HTML/PDF. This is the most favorable approach to render most simple tables as it is format independent. If you switch from HTML to pdf, you basically don't need to change anything in your code. However, markdown doesn't support complex table. For example, if you want to have a double-row header table, markdown just cannot provide you the functionality you need. As a result, when you have such a need, you should **define `format` in `kable()`** as either "html" or "latex". *You can also define a global option at the beginning using `options(knitr.table.format = "latex")` so you don't repeat the step everytime.* **In this tutorial, I'll still put `format="latex"` in the function in case users just want to quickly replicate the results.**

```r
options(knitr.table.format = "latex")
## If you don't define format here, you'll need put `format = "latex"`
## in every kable function.
```

LaTeX packages used in this package

If you are using a recent version of rmarkdown, you are recommended to load this package entirely via `library(kableExtra)` or `require(kableExtra)` because this package will load all necessary LaTeX packages, such as `booktabs` or `multirow`, for you automatically. Note that, if you are calling functions from `kableExtra` via `kableExtra::kable_styling()` or if you put `library(kableExtra)` in a separate R file that is **sourced** by the rmarkdown document, these packages won't be loaded. Furthermore, you can suppress this auto-loading behavior by setting a global option `kableExtra.latex.load_packages` to be `FALSE` before you load `kableExtra`.

```r
# Not evaluated. Ilustration purpose
options(kableExtra.latex.load_package = FALSE)
library(kableExtra)
```

If you are using R Sweave or some customized rmarkdown templates or you just want to load these LaTeX libraries by yourself, you can put the following meta data into the `yaml` section. If you are familar with LaTeX and you know what you are doing, feel free to remove unnecessary packages from the list.

```yaml
header-includes:
  - \usepackage{booktabs}
  - \usepackage{longtable}
  - \usepackage{array}
  - \usepackage{multirow}
```

```
- \usepackage[table]{xcolor}
- \usepackage{wrapfig}
- \usepackage{float}
- \usepackage{colortbl}
- \usepackage{pdflscape}
- \usepackage{tabu}
- \usepackage{threeparttable}
```

Plain LaTeX

Plain LaTeX table looks relatively ugly in 2017.

```
kable(dt)
```

	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

LaTeX table with booktabs

Similar with Bootstrap in HTML, in LaTeX, you can also use a trick to make your table look prettier as well. The different part is that, this time you don't need to pipe kable outputs to another function. Instead, you should call `booktabs = T` directly in `kable()`

```
kable(dt, format = "latex", booktabs = T)
```

	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

Table Styles

`kable_styling` in LaTeX uses the same syntax and structure as `kable_styling` in HTML. However, instead of `bootstrap_options`, you should specify `latex_options` instead.

LaTeX options

Similar with `bootstap_options`, `latex_options` is also a charter vector with a bunch of options including `striped`, `hold_position` and `scale_down`.

Striped

Even though in the LaTeX world, people usually call it `alternative row colors` but here I'm using its bootstrap name for consistency. Note that to make it happen, LaTeX package `xcolor` is required to be

loaded. In an environment like rmarkdown::pdf_document (rmarkdown 1.4.0 +), `kable_styling` will load it automatically if `striped` is enabled. However, in other cases, you probably need to import that package by yourself.

```
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling(latex_options = "striped")
```

	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

Hold position

If you provide a table caption in `kable()`, it will put your LaTeX tabular in a `table` environment, unless you are using `longtable`. A `table` environment will automatically find the best place (it thinks) to put your table. However, in many cases, you do want your table to appear in a position you want it to be. In this case, you can use this `hold_position` options here.

```
kable(dt, format = "latex", caption = "Demo table", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 1: Demo table

	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

Scale down

When you have a wide table that will normally go out of the page and you want to scale down the table to fit the page, you can use the `scale_down` option here. Note that, if your table is too small, it will also scale up your table. It was named in this way only because scaling up isn't very useful in most cases.

```
kable(cbind(dt, dt, dt), format = "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down"))
```

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

```
kable(cbind(dt), format = "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down"))
```

	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

Repeat header in longtable

In `kableExtra` 0.3.0 or above, a new option `repeat_header` was introduced into `kable_styling`. It will add header rows to longtables spanning multiple pages. For table captions on following pages, it will append *"continued"* to the caption to differentiate. If you need texts other than *"(continued)"* (for example, other languages), you can specify it using `kable_styling(..., repeat_header_text = "xxx")`. If you want to complete replace the table caption instead of appending, you can specify it in the option `repeat_header_method`.

```r
long_dt <- rbind(mtcars, mtcars)

kable(long_dt, format = "latex", longtable = T, booktabs = T, caption = "Longtable") %>%
  add_header_above(c(" ", "Group 1" = 5, "Group 2" = 6)) %>%
  kable_styling(latex_options = c("repeat_header"))
```

Table 2: Longtable

	Group 1					Group 2					
	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1
Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1

	Group 1					Group 2					
	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Toyota Corona	21.5	4	120.1	97	3.70	2.465	20.01	1	0	3	1
Dodge Challenger	15.5	8	318.0	150	2.76	3.520	16.87	0	0	3	2
AMC Javelin	15.2	8	304.0	150	3.15	3.435	17.30	0	0	3	2
Camaro Z28	13.3	8	350.0	245	3.73	3.840	15.41	0	0	3	4
Pontiac Firebird	19.2	8	400.0	175	3.08	3.845	17.05	0	0	3	2
Fiat X1-9	27.3	4	79.0	66	4.08	1.935	18.90	1	1	4	1
Porsche 914-2	26.0	4	120.3	91	4.43	2.140	16.70	0	1	5	2
Lotus Europa	30.4	4	95.1	113	3.77	1.513	16.90	1	1	5	2
Ford Pantera L	15.8	8	351.0	264	4.22	3.170	14.50	0	1	5	4
Ferrari Dino	19.7	6	145.0	175	3.62	2.770	15.50	0	1	5	6
Maserati Bora	15.0	8	301.0	335	3.54	3.570	14.60	0	1	5	8
Volvo 142E	21.4	4	121.0	109	4.11	2.780	18.60	1	1	4	2
Mazda RX41	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag1	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 7101	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive1	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout1	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant1	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 3601	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D1	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 2301	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
Merc 2801	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
Merc 280C1	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
Merc 450SE1	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
Merc 450SL1	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
Merc 450SLC1	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
Cadillac Fleetwood1	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
Lincoln Continental1	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
Chrysler Imperial1	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
Fiat 1281	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1
Honda Civic1	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
Toyota Corolla1	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1
Toyota Corona1	21.5	4	120.1	97	3.70	2.465	20.01	1	0	3	1
Dodge Challenger1	15.5	8	318.0	150	2.76	3.520	16.87	0	0	3	2
AMC Javelin1	15.2	8	304.0	150	3.15	3.435	17.30	0	0	3	2
Camaro Z281	13.3	8	350.0	245	3.73	3.840	15.41	0	0	3	4
Pontiac Firebird1	19.2	8	400.0	175	3.08	3.845	17.05	0	0	3	2
Fiat X1-91	27.3	4	79.0	66	4.08	1.935	18.90	1	1	4	1
Porsche 914-21	26.0	4	120.3	91	4.43	2.140	16.70	0	1	5	2
Lotus Europa1	30.4	4	95.1	113	3.77	1.513	16.90	1	1	5	2
Ford Pantera L1	15.8	8	351.0	264	4.22	3.170	14.50	0	1	5	4
Ferrari Dino1	19.7	6	145.0	175	3.62	2.770	15.50	0	1	5	6
Maserati Bora1	15.0	8	301.0	335	3.54	3.570	14.60	0	1	5	8
Volvo 142E1	21.4	4	121.0	109	4.11	2.780	18.60	1	1	4	2

Full width?

If you have a small table and you want it to spread wide on the page, you can try the `full_width` option. Unlike `scale_down`, it won't change your font size. Note that, if you use `full_width` in LaTeX, you will loss your in-cell text alignment settings and everything will be left-aligned.

```
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling(full_width = T)
```

	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

Position

Table Position only matters when the table doesn't have `full_width`. You can choose to align the table to `center` or `left` side of the page. The default value of position is `center`.

Note that even though you can select to `right` align your table but the table will actually be centered. Somehow it is very difficult to right align a table in LaTeX (since it's not very useful in the real world?). If you know how to do it, please send out an issue or PR and let me know.

```
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling(position = "center")
```

	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

Becides these three common options, you can also wrap text around the table using the `float-left` or `float-right` options. Note that, like `striped`, this feature will load another non-default LaTeX package `wrapfig` which requires rmarkdown 1.4.0 +. If you rmarkdown version < 1.4.0, you need to load the package through a customed LaTeX template file.

```
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling(position = "float_right")
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras sit amet mauris in ex ultricies elementum vel rutrum dolor. Phasellus tempor convallis dui, in hendrerit mauris placerat scelerisque. Maecenas a accumsan enim, a maximus velit. Pellentesque in risus eget est faucibus convallis nec at nulla. Phasellus nec

	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

lacinia justo. Morbi fermentum, orci id
varius accumsan, nibh neque porttitor
ipsum, consectetur luctus risus arcu ac ex. Aenean a luctus augue. Suspendisse et auctor nisl. Suspendisse
cursus ultrices quam non vulputate. Phasellus et pharetra neque, vel feugiat erat. Sed feugiat elit at mauris
commodo consequat. Sed congue lectus id mattis hendrerit. Mauris turpis nisl, congue eget velit sed,
imperdiet convallis magna. Nam accumsan urna risus, non feugiat odio vehicula eget.

Font Size

If one of your tables is huge and you want to use a smaller font size for that specific table, you can use the
`font_size` option.

```
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling(font_size = 7)
```

	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

Column / Row Specification

Column spec

When you have a table with lots of explanatory texts, you may want to specified the column width for
different column, since the auto adjust in HTML may not work in its best way while basic LaTeX table is
really bad at handling text wrapping. Also, sometimes, you may want to highlight a column (e.g. a "Total"
column) by making it bold. In these scenario, you can use `column_spec()`. You can find an example below.

```
text_tbl <- data.frame(
  Items = c("Item 1", "Item 2", "Item 3"),
  Features = c(
    "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin vehicula tempor ex. Morbi malesuada
    "In eu urna at magna luctus rhoncus quis in nisl. Fusce in velit varius, posuere risus et, cursus a
    "Vivamus venenatis egestas eros ut tempus. Vivamus id est nisi. Aliquam molestie erat et sollicitud
  )
)

kable(text_tbl, format = "latex", booktabs = T) %>%
  kable_styling(full_width = F) %>%
  column_spec(1, bold = T) %>%
  column_spec(2, width = "30em")
```

Row spec

Similar with `column_spec`, you can define specifications for rows. Currently, you can either bold or italiciz an
entire row. Note that, similar with other row-related functions in **kableExtra**, for the position of the target
row, you don't need to count in header rows or the group labelling rows.

Items	Features
Item 1	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin vehicula tempor ex. Morbi malesuada sagittis turpis, at venenatis nisl luctus a.
Item 2	In eu urna at magna luctus rhoncus quis in nisl. Fusce in velit varius, posuere risus et, cursus augue. Duis eleifend aliquam ante, a aliquet ex tincidunt in.
Item 3	Vivamus venenatis egestas eros ut tempus. Vivamus id est nisi. Aliquam molestie erat et sollicitudin venenatis. In ac lacus at velit scelerisque mattis.

```
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling("striped", full_width = F) %>%
  column_spec(7, bold = T) %>%
  row_spec(5, bold = T)
```

	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**

Grouped Columns / Rows

Add header rows to group columns

Tables with multi-row headers can be very useful to demonstrate grouped data. To do that, you can pipe your kable object into `add_header_above()`. The header variable is supposed to be a named character with the names as new column names and values as column span. For your convenience, if column span equals to 1, you can ignore the `=1` part so the function below can be written as 'add_header_above(c(" ","Group 1" = 2, "Group 2" = 2, "Group 3" = 2)).

```
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling() %>%
  add_header_above(c(" " = 1, "Group 1" = 2, "Group 2" = 2, "Group 3" = 2))
```

	Group 1		Group 2		Group 3	
	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

In fact, if you want to add another row of header on top, please feel free to do so. Also, in kableExtra 0.3.0, you can specify `bold` & `italic` as you do in `row_spec()`.

```r
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling(latex_options = "striped") %>%
  add_header_above(c(" ", "Group 1" = 2, "Group 2" = 2, "Group 3" = 2)) %>%
  add_header_above(c(" ", "Group 4" = 4, "Group 5" = 2)) %>%
  add_header_above(c(" ", "Group 6" = 6), bold = T, italic = T)
```

	Group 6					
	Group 4				Group 5	
	Group 1		Group 2		Group 3	
	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

Group rows via labeling

Sometimes we want a few rows of the table being grouped together. They might be items under the same topic (e.g., animals in one species) or just different data groups for a categorical variable (e.g., age < 40, age > 40). With the new function `group_rows()` in `kableExtra`, this kind of task can be completed in one line. Please see the example below. Note that when you count for the start/end rows of the group, you don't need to count for the header rows nor other group label rows. You only need to think about the row numbers in the "original R dataframe".

```r
kable(mtcars[1:10, 1:6], format = "latex", caption = "Group Rows", booktabs = T) %>%
  kable_styling() %>%
  group_rows("Group 1", 4, 7) %>%
  group_rows("Group 2", 8, 10)
```

In case some users need it, you can define your own gapping spaces between the group labeling row and previous rows. The default value is `0.5em`.

```r
kable(dt, format = "latex", booktabs = T) %>%
  group_rows("Group 1", 4, 5, latex_gap_space = "2em")
```

	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
Group 1						
Hornet 4 Drive	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440

Row indentation

Unlike `group_rows()`, which will insert a labeling row, sometimes we want to list a few sub groups under a total one. In that case, `add_indent()` is probably more apporiate. For advanced users, you can even define

Table 3: Group Rows

	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160.0	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875
Datsun 710	22.8	4	108.0	93	3.85	2.320
Group 1						
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440
Valiant	18.1	6	225.0	105	2.76	3.460
Duster 360	14.3	8	360.0	245	3.21	3.570
Group 2						
Merc 240D	24.4	4	146.7	62	3.69	3.190
Merc 230	22.8	4	140.8	95	3.92	3.150
Merc 280	19.2	6	167.6	123	3.92	3.440

your own css for the group labeling.

```
kable(dt, format = "latex", booktabs = T) %>%
  add_indent(c(1, 3, 5))
```

	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

Group rows via multi-row cell

Function `group_rows` is great for showing simple structural information on rows but sometimes people may need to show structural information with multiple layers. When it happens, you may consider to use `collapse_rows` instead, which will put repeating cells in columns into multi-row cells.

```
collapse_rows_dt <- data.frame(C1 = c(rep("a", 10), rep("b", 5)),
                 C2 = c(rep("c", 7), rep("d", 3), rep("c", 2), rep("d", 3)),
                 C3 = 1:15,
                 C4 = sample(c(0,1), 15, replace = TRUE))
kable(collapse_rows_dt, format = "latex", booktabs = T, align = "c") %>%
  column_spec(1, bold=T) %>%
  collapse_rows(columns = 1:2)
```

C1	C2	C3	C4
a	c	1	0
		2	0
		3	1
		4	0
		5	0
		6	1
		7	1
	d	8	0
		9	1
		10	0
b	c	11	1
		12	0
	d	13	1
		14	0
		15	1

```
kable(collapse_rows_dt, format = "latex", align = "c") %>%
  column_spec(1, bold = T, width = "5em") %>%
  collapse_rows(1:2)
```

C1	C2	C3	C4
a	c	1	0
		2	0
		3	1
		4	0
		5	0
		6	1
		7	1
	d	8	0
		9	1
		10	0
b	c	11	1
		12	0
	d	13	1
		14	0
		15	1

Table Footnote

Notation system

You can also use `add_footnote()` function from this package. You will need to supply a character vector with each element as one footnote. You may select from `number`, `alphabet` and `symbol` for different types of notations. Example are listed below.

Alphabet

```r
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling() %>%
  add_footnote(c("Footnote 1", "Have a good day."), notation = "alphabet")
```

	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

[a] Footnote 1
[b] Have a good day.

Number

```r
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling() %>%
  add_footnote(c("Footnote 1", "Have a good day."), notation = "number")
```

	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

[1] Footnote 1
[2] Have a good day.

Symbol

```r
kable(dt, format = "latex", booktabs = T) %>%
  kable_styling() %>%
  add_footnote(c("Footnote 1", "Footnote 2", "Footnote 3"), notation = "symbol")
```

	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

[*] Footnote 1
[†] Footnote 2
[‡] Footnote 3

In-table markers

By design, `add_footnote()` will transform any `[note]` to in-table footnote markers.

```r
kable(dt, format = "latex", caption = "Demo Table[note]", booktabs = T) %>%
  kable_styling(latex_options = "hold_position") %>%
  add_header_above(c(" ", "Group 1[note]" = 3, "Group 2[note]" = 3)) %>%
  add_footnote(c("This table is from mtcars",
                 "Group 1 contains mpg, cyl and disp",
                 "Group 2 contains hp, drat and wt"),
               notation = "symbol")
```

Table 4: Demo Table[*]

	Group 1[†]			Group 2[‡]		
	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

[*] This table is from mtcars
[†] Group 1 contains mpg, cyl and disp
[‡] Group 2 contains hp, drat and wt

LaTeX Only Features

Table on a Landscape Page

Sometimes when we have a wide table, we want it to sit on a designated landscape page. The new function `landscape()` can help you on that. Unlike other functions, this little function only serves LaTeX and doesn't have a HTML side.

```r
kable(dt, format = "latex", caption = "Demo Table (Landscape)[note]", booktabs = T) %>%
  kable_styling(latex_options = c("hold_position")) %>%
  add_header_above(c(" ", "Group 1[note]" = 3, "Group 2[note]" = 3)) %>%
  add_footnote(c("This table is from mtcars",
                 "Group 1 contains mpg, cyl and disp",
                 "Group 2 contains hp, drat and wt"),
               notation = "symbol") %>%
  group_rows("Group 1", 4, 5) %>%
  landscape()
```

Table 5: Demo Table (Landscape)[*]

	Group 1[†]			Group 2[‡]		
	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
Group 1						
Hornet 4 Drive	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440

[*] This table is from mtcars

[†] Group 1 contains mpg, cyl and disp

[‡] Group 2 contains hp, drat and wt